Skyline Identification in Multi-Armed Bandits

Albert Cheu∗ Ravi Sundaram† Jonathan Ullman‡

College of Computer and Information Science
Northeastern University

August 12, 2018

Abstract

We introduce a variant of the classical PAC multi-armed bandit problem. There is an ordered set of \( n \) arms \( A[1], \ldots, A[n] \), each with some stochastic reward drawn from some unknown bounded distribution. The goal is to identify the skyline of the set \( A \), consisting of all arms \( A[i] \) such that \( A[i] \) has larger expected reward than all lower-numbered arms \( A[1], \ldots, A[i-1] \). We define a natural notion of an \( \varepsilon \)-approximate skyline and prove matching upper and lower bounds for identifying an \( \varepsilon \)-skyline. Specifically, we show that in order to identify an \( \varepsilon \)-skyline from among \( n \) arms with probability \( 1 - \delta \),

\[
\Theta \left( \frac{n}{\varepsilon^2} \cdot \min \left\{ \log \left( \frac{1}{\varepsilon^2 \delta} \right), \log \left( \frac{n}{\delta} \right) \right\} \right)
\]

samples are necessary and sufficient. When \( \varepsilon \gg 1/n \), our results improve over the naïve algorithm, which draws enough samples to approximate the expected reward of every arm; the algorithm of (Auer et al., AISTATS’16) for Pareto-optimal arm identification is likewise superseded. Our results show that the sample complexity of the skyline problem lies strictly in between that of best arm identification (Even-Dar et al., COLT’02) and that of approximating the expected reward of every arm.

∗cheu.a@husky.neu.edu
†Supported in part by NSF grant CCF-1535929. koods@ccs.neu.edu
‡Supported in part by NSF grant CCF-1718088. jullman@ccs.neu.edu
Contents

1 Introduction 3
   1.1 Model and Results ................................................. 3
   1.2 Related Work ..................................................... 4

2 An Improved Algorithm for $\varepsilon$-Skyline Identification 5
   2.1 Overview of the Algorithm ........................................ 5
   2.2 Useful Algorithmic Subroutines ................................... 6
   2.3 The Algorithm ..................................................... 6
   2.4 Analysis ............................................................ 6
      2.4.1 Correctness of Algorithm 1 (Claim 8) ....................... 9
      2.4.2 Efficiency of Algorithm 1 (Claim 9) ........................ 11
      2.4.3 Analysis of Algorithm 2 ..................................... 14

3 A Sample Complexity Lower Bound for $\varepsilon$-Skyline Identification 15
1 Introduction

In (one formulation of) the classical multi-armed bandit framework, we are given a set of \( n \) arms to choose from, each of which results in some stochastic reward, and our goal is to find an arm whose expected reward is as large as possible, while drawing as few samples from the arms as possible. One representative example of this framework comes from drug testing, where there are many available drugs, each of which has some unknown probability of success, and the goal is to find the most effective drug. We introduce a variant of this problem where the arms are sorted, and the goal is to find all arms \( A[i] \) whose expected rewards is as large as possible among the first \( i \) arms. In our representative example, the drugs may have different costs, and we want to produce a list of all drugs that are the most effective for their given cost. In other words, we want to compute the set of arms \( A[i] \) such that there is no \( i' \leq i \) with higher expected reward. Borrowing terminology from the database community, we call this the skyline of the set of arms [BKS01]. In this work, we define a natural PAC (Probably Approximately Correct) version of this problem that we call \( \varepsilon \)-skyline identification, and prove worst-case sample complexity bounds that are optimal up to constant factors in terms of all of the relevant parameters.

We remark that the problem we study can be cast as a special case of a much more general version of the Pareto-optimal arm identification problem that was recently introduced by Auer et al. [ACOD16], which we discuss in more detail in Section 1.2. Our results yield optimal sample complexity bounds for this special case, and may be useful for obtaining optimal sample complexity for other variants of Auer et al.’s model, and, more generally, with obtaining optimal sample complexity for other variants of the best-arm identification problem.

1.1 Model and Results

Like all multi-armed bandit models, our problem begins with a set of \( n \) arms \( A = \{A[1], \ldots, A[n]\} \). Each arm \( A[i] \) represents an probability distribution over \([0, 1]\) with unknown mean \( \mu[i] = E[A[i]] \). We will sometimes abuse terminology and associate an arm \( A[i] \) simply with its index \( i \). We can obtain a sample from any arm \( A[i] \), which we also refer to as pulling arm \( A[i] \).

A general objective is to learn useful information about the set of arms using as few samples as possible. Our specific goal is to compute an \( \varepsilon \)-skyline, which we define as follows.

**Definition 1** (\( \varepsilon \)-skyline). A set \( S \subseteq [n] \) is an \( \varepsilon \)-skyline of \( A \) if the following conditions hold:

1. For any arm index \( t \notin S \), the largest \( s \in S \) such that \( s < t \) satisfies \( \mu[s] \geq \mu[t] - \varepsilon \).

2. For every \( s \in S \), and every \( t \leq s \), \( \mu[s] \geq \mu[t] - \varepsilon \).

Before we proceed, we discuss how this definition is a natural notion of an approximation skyline. An arm \( t \in [n] \) is in the exact skyline if and only if \( \mu[t] \geq \mu[s] \) for every \( s < t \). Suppose that an arm \( t \) is convincingly in the skyline, meaning \( \mu[t] \geq \mu[s] + \varepsilon \) for every \( s < t \). Then the first condition asserts that \( t \in S \). On the other hand, suppose that an arm \( t \) is convincingly not in the skyline, meaning there exists \( s < t \) such that \( \mu[t] \leq \mu[s] - \varepsilon \). Then the second condition asserts that \( t \notin S \). Although the formulation of Definition 1 is a slightly indirect way of capturing this intuitive notion, it will be more convenient to work with in our analysis.

Let’s consider a naïve approach to identifying an \( \varepsilon \)-skyline: draw samples from each arm to obtain an empirical estimate \( \hat{\mu}[i] \approx \mu[i] \) and compute the exact skyline of the estimates. If we want to draw enough samples to obtain an \( \varepsilon \)-skyline with probability \( 1 - \delta \), then by standard concentration and anti-concentration arguments it is necessary and sufficient to draw \( \Theta(\frac{n}{\varepsilon^2} \log \frac{1}{\delta}) \) samples in total. Our first result is an algorithm that improves on this naïve approach in the regime where \( \varepsilon \) is not too small compared to \( n \).
Theorem 2. For \( \varepsilon, \delta \in (0, 1) \), there is an algorithm that gets sample access to a set of \( n \) arms \( A = \{A[1], \ldots, A[n]\} \) with rewards in \([0, 1]\) and, with probability at least \( 1 - \delta \), draws \( O(\frac{n}{\varepsilon^2} \log \frac{1}{\varepsilon \delta}) \) samples to return an \( \varepsilon \)-skyline for \( A \) with size at most \( O(\frac{1}{\varepsilon}) \).

This result shows that finding an approximate skyline can be strictly easier than estimating the payoff of every arm. In particular, when \( \varepsilon, \delta \) are constants, the naïve algorithm requires \( O(n \log n) \) samples, whereas the algorithm in Theorem 2 requires just \( O(n) \) samples. We give an overview of our algorithm in Section 2.1.

Combining the naïve algorithm with Theorem 2 yields a sample complexity upper bound of \( O\left(\frac{n}{\varepsilon^2} \cdot \min \left\{ \log \left(\frac{1}{\varepsilon \delta}\right), \log \left(\frac{n}{\delta}\right) \right\} \right) \).

In order to show this is optimal let’s consider a naïve lower bound on the sample complexity of \( \varepsilon \)-skyline identification. The largest-numbered arm in any \( \varepsilon \)-skyline is always within \( \varepsilon \) of the maximum expected reward among all the arms, thus finding an \( \varepsilon \)-skyline can be no easier than identifying an arm with nearly maximal payoff. Combining the algorithm of Even-Dar, Mannor, and Mansou [EDMM06] with the lower bound of Mannor and Tsitsiklis [MT04], the sample complexity of this latter problem is \( \Theta(\frac{n}{\varepsilon^2} \log \frac{1}{\delta}) \). However, this does not yield a tight lower bound for \( \varepsilon \)-skyline identification. We observe that, when \( \varepsilon \geq \frac{1}{n} \), an algorithm for \( \varepsilon \)-skyline identification can be used to solve \( \Omega(1/\varepsilon) \) independent \( \varepsilon \)-best arm identification problems, each on \( \Omega(\varepsilon n) \) arms. Formalizing this intuition yields the following tight lower bound, and shows that approximate skyline identification is strictly harder than approximate best arm identification.

Theorem 3. Fix \( \varepsilon, \delta \) smaller than some absolute constants and \( \varepsilon \geq 1/n \). Suppose there is an algorithm such that for every set of \( n \) arms, with probability at least \( 1 - \delta \), the algorithm returns an \( \varepsilon \)-skyline. Then this algorithm must draw \( \Omega\left(\frac{n}{\varepsilon^2} \log \frac{1}{\varepsilon \delta}\right) \) samples.

Note that the condition \( \varepsilon \geq 1/n \) is necessary for the lower bound to hold, since the naïve algorithm gives an upper bound of \( O(\frac{n}{\varepsilon^2} \log \frac{n}{\delta}) \) samples for every \( \varepsilon > 0 \). Theorem 3 immediately implies that the naïve algorithm is optimal up to constant factors when \( \varepsilon < 1/n \). Thus, by combining the naïve algorithm with Theorems 2 and 3 we obtain the sample complexity lower bound of

\[
\Omega\left(\frac{n}{\varepsilon^2} \cdot \min \left\{ \log \left(\frac{1}{\varepsilon \delta}\right), \log \left(\frac{n}{\delta}\right) \right\} \right).
\]

for approximate skyline identification. This resolves the sample complexity of approximate skyline identification up to constant factors for the entire range of the parameters \( n, \varepsilon, \delta \).

1.2 Related Work

The literature on multi-armed bandit problems is far too large to survey in its entirety, so we focus only on results that are closely related to ours. The most similar work to ours is that of Auer et al. [ACOD16], which considers a more general problem called Pareto-optimal arm identification. In this problem there are \( n \) arms, each of which has a stochastic reward supported on \([0, 1]^d\) with mean \( \mu = (\mu_1, \ldots, \mu_d) \in [0, 1]^d \). Although their parameterization is somewhat more general, they essentially prove an upper bound of \( O\left(\frac{n}{\varepsilon^2} \log \frac{n d}{\varepsilon \delta}\right) \) to identify an \( \varepsilon \)-Pareto-optimal set of arms with \( 1 - \delta \). The exact definition of \( \varepsilon \)-Pareto optimal roughly corresponds to our definition of \( \varepsilon \)-skyline, but the details are not crucial to this discussion. Skyline identification can be cast as a special case of Pareto-optimal arm identification, but in this special case the sample complexity of their algorithm is no better than the naïve algorithm. Indeed, in Auer et al.’s analysis, one of the first steps is to
accurately estimate the payoff of all arms while paying a union bound in the sample complexity, and this union bound is precisely what our algorithm was designed to avoid.

There is also a related line of work on generalizations of the best-arm-identification problem in which the goal is to identify a set of $k$ arms with approximately maximal payoffs. The first such work was by Kalyanakrishnan and Stone [KS10], which introduced the $k$-best-arm-identification problem, and subsequent works [GGLB11, KTAS12, CGL16, CGL+17] have proven tight upper and lower bounds, and also consider variants where the algorithm is only allowed to choose certain types of sets of $k$ arms. Comparing our results to the upper and lower bounds of [KS10, KTAS12] reveals that the $\varepsilon$-skyline identification problem has the same sample complexity identifying an approximately optimal set of $\Theta(1/\varepsilon)$ arms. However, the two problems do not seem to be directly comparable, since an $\varepsilon$-skyline is not necessarily a set of arms with approximately maximum reward and vice versa.

The work by Kleinberg [Kle06] is also related—it develops fast-converging algorithms that find the best arm for every prefix of a countably infinite sequence of arms, under the regret minimization measure in contrast to the work in this paper which finds the best PAC arm in every prefix of a finite sequence.

2 An Improved Algorithm for $\varepsilon$-Skyline Identification

2.1 Overview of the Algorithm

Our new algorithm that achieves lower sample complexity in the case of $\varepsilon > 1/n$ can be viewed as a sample-efficient reduction from identifying an $\varepsilon$-skyline to identifying an $\varepsilon$-best arm. For any set of arms $B$, with probability at least $1 - \delta$, an algorithm of Even-Dar et al. [EDMM06] finds an arm whose expected reward is within $\varepsilon$ of the maximum among arms in $B$ using $O(\frac{n}{\varepsilon^2} \log \frac{1}{\delta})$ samples.

Our reduction is roughly as follows. Let $S$ be the set of arms our algorithm will identify, and initially set $S = \emptyset$. Initially all of the $n$ arms are considered active. Our algorithm proceeds in a series of rounds, and we start by describing the first round. Start by partitioning the arms into $t \approx \frac{20}{\varepsilon}$ blocks $B_1, \ldots, B_t$, each of $\approx \frac{en}{20}$ consecutive arms.\(^1\) We then find an $\frac{\varepsilon}{20}$-best arm $a_1, \ldots, a_t$ for all of these blocks. We then obtain an estimate $\hat{\mu}[a_1], \ldots, \hat{\mu}[a_t]$ of the expected reward of each of these arms. Using the algorithm of [EDMM06] and paying a union bound over the blocks, we can do all of this using just $O(\frac{n}{\varepsilon^2} \log \frac{1}{\delta})$ samples.

Now we scan the blocks from 1 to $t$ (lower numbered arms are considered first) looking for arms to add to the skyline. As we go through, each block $B_i$ will be either deactivated or will remain partially active, depending on our estimate $\hat{\mu}[a_i]$ of the expected reward of the best arm in that block. When we get to block $B_i$, suppose that $B_j$ is the last block that we kept active. Then if $\hat{\mu}[a_i] > \hat{\mu}[a_j] + \frac{\varepsilon}{5}$, we may need to include the arm $a_i$ in order to have an approximate skyline, so we add $a_i$ to the set $S$, and we leave block $B_i$ active. (Our actual algorithm deactivates some of the arms in $B_i$ but this detail is not crucial for this informal summary.) On the other hand, if $B_j$ is the last block that we left active and $\hat{\mu}[a_i] \leq \hat{\mu}[a_j] + \frac{\varepsilon}{5}$, then no arm in $B_i$ dominates the best arm in $B_j$, which is already included in the set $S$. Thus, we can safely deactivate all the arms in $B_i$.

At the end of this process, some blocks will be deactivated and some blocks will remain at least partially active. The key fact is that every time we keep a block $B_i$ active, its true expected reward must be at least $\frac{\varepsilon}{10}$ larger than the last block we kept active, and therefore only $\frac{10}{\varepsilon}$ blocks remain active, and these contain a total of at most $\frac{10}{\varepsilon} \cdot \frac{en}{20} = \frac{n}{2}$ arms. Thus, we have made significant progress in the sense that we have eliminated a constant fraction of the arms from consideration while still leaving enough arms to guarantee that we can form an approximate skyline. Now we can recurse.

\(^1\)The constants in this informal description are somewhat arbitrary, and we have not optimized them.
on this set remaining \( \frac{2}{3} \) arms by further dividing each of the active blocks into \( \approx 20 \) sub-blocks. We repeat essentially the same process on these sub-blocks. See Figure 1 for a diagram of how the algorithm progresses.

In the rest of this section we describe and analyze our algorithm, and thereby prove Theorem 2. Before describing the algorithm in detail, we will need to introduce a few useful subroutines, and some helpful terminology.

### 2.2 Useful Algorithmic Subroutines

The first primitive simply captures the fact that we can estimate the mean of any given arm with high confidence by drawing a sufficient number of samples. We will denote this procedure by the subroutine \text{EstMean}. The proof of Lemma 4 is a standard application of Hoeffding’s Inequality, and is omitted.

**Lemma 4 (EstMean).** There is an algorithm \text{EstMean} that takes a tuple \((A[i], \varepsilon, \delta)\) as input, draws \(O\left(\frac{1}{\varepsilon^2} \log \frac{1}{\delta}\right)\) samples from \(A[i]\) and outputs an estimate \(\hat{\mu}[i] \in [0, 1]\) such that, with probability at least \(1 - \delta\), \(|\mu[i] - \hat{\mu}[i]| \leq \varepsilon\).

The second subroutine is an algorithm of Even-Dar et al. [EDMM06] that identifies an arm with approximately maximum payoff among any set of arms \(A\), using sample complexity that is just linear in \(|A|\). First, we introduce some useful terminology for discussing multi-armed bandit problems.

**Definition 5 (\(\varepsilon\)-Better and \(\varepsilon\)-Best).** Arm \(A[i]\) is \(\varepsilon\)-better than arm \(A[j]\) if the expected payoff of \(A[i]\) is at least within \(\varepsilon\) of that of \(A[j]\). That is, \(\mu[i] \geq \mu[j] - \varepsilon\). For a set of arms \(A\), \(A[i] \in A\) is an \(\varepsilon\)-best arm for \(A\) if \(A[i]\) is \(\varepsilon\)-better than all arms \(A[j] \in A\). That is \(\forall A[j] \in A\; \mu[i] \geq \mu[j] - \varepsilon\).

**Theorem 6 (FindBest).** There is an algorithm \text{FindBest} that takes a tuple \((A', \varepsilon, \delta)\), where \(A' \subseteq A\) is a set of arms and \(\varepsilon, \delta\) are parameters, draws \(O\left(\frac{|A'|}{\varepsilon^2} \log \frac{1}{\delta}\right)\) samples from the arms in \(A'\), and outputs an arm \(A[i] \in A'\) such that, with probability at least \(1 - \delta\), \(A[i]\) is an \(\varepsilon\)-best arm for \(A'\).

### 2.3 The Algorithm

We are now ready to describe our algorithm. The bulk of the work is done in Algorithm 1, which identifies an approximately optimal set of arms. We also define an auxiliary procedure Algorithm 2 that parens down the set of arms to have size \(O(1/\varepsilon)\), which will establish the final guarantee of Theorem 2. We remark that, by Definition 1, the first arm is always part of an \(\varepsilon\)-skyline, thus it will be more notationally convenient in our algorithm if we consider a set of \(n+1\) arms labeled \(A[0], A[1], \ldots, A[n]\).

### 2.4 Analysis

To aid in the analysis, we recall some notation for the algorithm and its execution. The algorithm proceeds in levels and in each level the arms will be partitioned into blocks. We use \(\ell\) to index levels and \(m\) to index blocks, so that \(B_{\ell,m}\) is the \(m\)-th block in the \(\ell\)-th level (which is given the shorthand \((\ell, m)\) in Figure 1). We use \(n_{\ell,m}\) to denote the number of arms in block \(B_{\ell,m}\) and \(n_{\ell}\) to denote the number of arms in level \(\ell\). For each block, \(B_{\ell,m}\) the algorithm \text{FindBest} outputs an arm \(k_{\ell,m}\) that is purported to be an approximate best arm in \(B_{\ell,m}\), and then we call \text{EstMean} to obtain an estimate \(\hat{\mu}[k_{\ell,m}] \approx \mu[k_{\ell,m}]\). We let \(\mu^*_m\) denote the exact mean of the exact best arm in \(B_{\ell,m}\).

First, we will show that with high probability, all calls to \text{EstMean} and \text{FindBest} return “correct” outputs. This will allow us to proceed with all of our analysis as if these two algorithms succeed with probability 1. To do so, we first define key events.
Algorithm 1

**Input:** Sample access for arms \( A = \{A[0], A[1], \ldots, A[n]\} \) and error parameters \( \varepsilon, \delta \).

**Output:** An \( \varepsilon \)-skyline of \( S \subseteq A \).

1: \( S \leftarrow \{0\} \) //A set to hold the \( \varepsilon \)-skyline
2: \( \hat{\mu}[0] \leftarrow \text{EstMean}(A[0], \varepsilon/12, \delta/2) \)
3: \( M \leftarrow \{\hat{\mu}[0]\} \) //A set to hold measurements of arms in \( S \)
4: \( B_1 \leftarrow \{[1, 2, \ldots, n]\} \) //The set of blocks for the 1st level
5: **for** \( \ell = 1, 2, \ldots \) **do** //Outer loop over levels
6: \( b_\ell \leftarrow |B_\ell| \)
7: **if** \( b_\ell = 0 \) **then** //Stop when no blocks are left
8: **break**
9: \( \delta_\ell \leftarrow \frac{\delta}{2^{\ell+1}} \) //The set of blocks for the next level (initially empty)
10: \( B_{\ell+1} \leftarrow \emptyset \)
11: **for all** \( m := 1, \ldots, b_\ell \) **do** //Inner loop over blocks in this level
12: \( \delta_{\ell,m} \leftarrow \delta_\ell/b_\ell \)
13: \([i_\ell,m, \ldots, j_\ell,m]\) \( \leftarrow \) \( B_{\ell,m} \) //\( m^{th} \) block of \( \ell^{th} \) level
14: \( \text{prev} \leftarrow \) largest member of \( S \) to the left of \( i_\ell,m \)
15: \( L_{\ell,m} \leftarrow (\varepsilon/2) + \hat{\mu}[\text{prev}] \) //Retrieve \( \hat{\mu}[\text{prev}] \) from \( M \)
16: \( k_{\ell,m} \leftarrow \text{FindBest}(B_{\ell,m}, \varepsilon/12, \delta_{\ell,m}/2) \)
17: \( \hat{\mu}[k_{\ell,m}] \leftarrow \text{EstMean}(A[k_{\ell,m}], \varepsilon/12, \delta_{\ell,m}/2) \)
18: **if** \( L_{\ell,m} + \varepsilon/4 > \hat{\mu}[k_{\ell,m}] \) **then**
19: **continue** //Deactivate this block (red in Fig. 1)
20: \( S \leftarrow S \cup k_{\ell,m} \) //Otherwise, store index (blue in Fig. 1)
21: \( M \leftarrow M \cup \hat{\mu}[k_{\ell,m}] \) //Store measurement
22: \( U_{\ell,m} \leftarrow \hat{\mu}[k_{\ell,m}] + \varepsilon/6 \) //Upper bound on arm quality in block
23: \( b_{\ell,m} \leftarrow (4/\varepsilon)(U_{\ell,m} - L_{\ell,m}) \) //Target no. of blocks to produce
24: \( \text{Break} \) \( [i_{\ell,m}, \ldots, k_{\ell,m} - 1] \) into blocks of size \( \max(1, [(k_{\ell,m} - i_{\ell,m})/b_{\ell,m}]) \)
25: \( B_{\ell+1} \leftarrow B_{\ell+1} \cup \) those new blocks
26: **return** \( S \)

Algorithm 2

**Input:** The \( \varepsilon \)-skyline \( S \) output by Algorithm 1 and the set \( M \) from its execution

**Output:** An \( \varepsilon \)-skyline \( S \) of size \( O(1/\varepsilon) \)

1: \( s \leftarrow 0 \)
2: **while** \( s \neq \max S \) **do**
3: Let \( s' \) denote the next member of \( S \)
4: Fetch \( \hat{\mu}[s], \hat{\mu}[s'] \) from \( M \) //These estimates were already obtained by Algorithm 1
5: **if** \( \hat{\mu}[s] + 3\varepsilon/4 > \hat{\mu}[s'] \) **then**
6: **Remove** \( s' \) from \( S \)
7: **else**
8: \( s \leftarrow s' \)
9: **return** \( S \)
Consider round \( \ell, m \) in Algorithm 1. We shall say the \texttt{FindBest} call at that round succeeds if it identifies an arm \( A[k_{\ell,m}] \) that is close-to-best in the block:

\[
\mu^*_{\ell,m} - \mu[k_{\ell,m}] < \varepsilon/12
\]

and, similarly, \texttt{EstMean} succeeds if the measurement of \( A[k_{\ell,m}] \) has low error:

\[
|\mu[k_{\ell,m}] - \hat{\mu}[k_{\ell,m}]| < \varepsilon/12
\]

Let \( E_{\ell,m} \) be the event that both calls to \texttt{EstMean} and \texttt{FindBest} succeed in round \((\ell, m)\). Likewise, let \( E_\ell \) be the event that all calls to \texttt{EstMean} and \texttt{FindBest} succeed in level \( \ell \). Let \( E \) be the success event for the whole of Algorithm 1: \( E \) is the event that line 1’s \texttt{EstMean} succeeds and \( E_{\ell,m} \) occurs for all rounds of the nested loop. We prove that all calls are successful with high probability.

**Claim 7.** The probability that \( E \) occurs is at least \( 1 - \delta \).

**Proof.** In level \( \ell \), we make \( 2b_\ell \) calls to \texttt{FindBest} and \texttt{EstMean}, each with error-probability parameter \( \delta_{\ell,m}/2 = \delta_\ell/(2b_\ell) \). From Theorem 6 and Lemma 4, the probability that any one of the calls does not succeed is \( \Pr(\neg E_\ell) \leq \delta_\ell \). Since \( \delta_\ell = \frac{\delta}{2^{b_\ell}} \), the failure probability over all levels is \( \leq \delta/2 \). The probability that \texttt{EstMean} on line 1 fails is also \( \leq \delta/2 \). Having covered all executions of \texttt{EstMean} and \texttt{FindBest}, we conclude that the probability of failure is \( \Pr(\neg E) \leq \delta \). \( \square \)

We now make two claims that state that if \( E \) holds, then Algorithm 1 outputs an \( \varepsilon \)-skyline of \( A \) and has the desired sample complexity. The proofs of these claims are in Sections 2.4.1 and 2.4.2, respectively.

**Claim 8.** If \( E \) occurs, then Algorithm 1 outputs an \( \varepsilon \)-skyline of \( A \).

**Claim 9.** If \( E \) occurs, then Algorithm 1 takes \( O(\frac{n}{\varepsilon^2} \log \frac{1}{\delta}) \) samples.
2.4.1 Correctness of Algorithm 1 (Claim 8)

Recall from Definition 1 that $S$, the output of Algorithm 1, is an $\varepsilon$-skyline of $A$ if and only if the following two hold:

1. For any arm index $t \notin S$, the largest $s \in S$ such that $s < t$ satisfies $\mu[s] \geq \mu[t] - \varepsilon$
2. Any $s \in S$ is the index to an $\varepsilon$-best arm of $\{A[0], A[1], \ldots, A[s]\}$

We will prove each of these two points separately. Throughout this section we will condition on the event $E$, that is, we will assume throughout that (1) and (2) hold.

**Lemma 10.** If $E$ occurs, then for any $t \notin S$, the largest $s \in S$ such that $s < t$ satisfies $\mu[s] \geq \mu[t] - \varepsilon$

**Proof.** Because $t \notin S$, the arm $A[t]$ was deactivated at some level $\ell$ in Algorithm 1 when it was a member of $m$-th block of that level; let $B_{\ell,m}$ denote the block. Let $k_{\ell,m}$ denote the index identified by FindBest. There are two ways that $A[t]$ could have been deactivated.

**Case 1:** $A[t]$ was deactivated when $k_{\ell,m}$ was added to $S$. In this case, Algorithm 1 deactivates only arms to the right of $k_{\ell,m}$ in $B_{\ell,m}$ (see line 24). Because $A[t]$ was deactivated, this means $k_{\ell,m} < t$. We also know Algorithm 1 deactivates all arms to the right of $k_{\ell,m}$ in $B_{\ell,m}$: no indices between $k_{\ell,m}$ and $t$ will ever be added to $S$. Ergo, $k_{\ell,m}$ is the largest member of $S$ to the left of $t$. From Theorem 6 and $E$, $k_{\ell,m}$ is an $\varepsilon/12$ best arm of $B_{\ell,m}$. Thus $\mu[k_{\ell,m}] \geq \mu[t] - \varepsilon/12$. In sum, $k_{\ell,m}$ is the largest member of $S$ to the left of $t$ and $\mu[k_{\ell,m}] \geq \mu[t] - \varepsilon/12$. Lemma 10 holds in this case.

**Case 2:** $A[t]$ was deactivated along with all other members of $B_{\ell,m}$. In this case, we shall first argue (2a) that $\text{prev}$, defined in Algorithm 1 as the largest $s \in S$ such that $s < i_{\ell,m}$ at level $\ell$, is the largest $s \in S$ such that $s < t$ at the end of the algorithm. Then we shall argue (2b) that the inequality $\mu[\text{prev}] \geq \mu[t] - \varepsilon$ holds.

(2a). Because the entirety of $B_{\ell,m}$ is deactivated, no indices between $i_{\ell,m}$ and $t$ will ever be added to $S$. Ergo, the largest $s \in S$ such that $s < i_{\ell,m}$ (at the end of Algorithm 1) must be the same as the largest $s \in S$ such that $s < t$ (at the end of Algorithm 1).

By definition, no index between $\text{prev}$ and $i_{\ell,m}$ is in $S$ at level $\ell$. Observe that all arms between $A[\text{prev}]$ and $A[i_{\ell,m}]$ are deactivated by the end of level $\ell$: were it not so, some block would have contributed an arm index to $S$. Ergo, no indices between $\text{prev}$ and $i_{\ell,m}$ will ever be added to $S$.

(2b). Having shown that $\text{prev}$ is the largest member of $S$ to the left of $t$ when Algorithm 1 ends, now we show that $\mu[\text{prev}] \geq \mu[t] - \varepsilon$. Because $k_{\ell,m} \notin S$, we have

$$L_{\ell,m} + \varepsilon/4 = \hat{\mu}[\text{prev}] + 3\varepsilon/4 > \hat{\mu}[k_{\ell,m}]$$

otherwise we would have added the index by line 18. By assuming $E$, we can use the fact that measurements are accurate (2):

$$\mu[\text{prev}] > \mu[k_{\ell,m}] - 11\varepsilon/12$$

By assuming $E$, we can use (1), the fact that $A[k_{\ell,m}]$ has mean close to the best arm of $B_{\ell,m}$, which has mean $\mu_{\ell,m}^*:

$$\mu[\text{prev}] > \mu_{\ell,m}^* - \varepsilon$$

By definition, $\mu_{\ell,m}^*$ is the maximum mean in $B_{\ell,m}$, so $\mu[\text{prev}] > \mu_t - \varepsilon$. In sum, $\text{prev}$ is the largest member of $S$ to the left of $t$ and $\mu[\text{prev}] \geq \mu[t] - \varepsilon$. Lemma 10 holds in this case.

Combining the two cases completes the proof of Lemma 10.
Now we shall prove the second part of Definition 1—that the members of $S$ are approximate best arms for their prefixes.

**Lemma 11.** If $E$ occurs, then any $s \in S$ is the index to an $\varepsilon$-best arm of $\{A[0] \ldots A[s]\}$

**Proof.** Label the arms in $S$ by $S[1], S[2], \ldots$, sorted by their position in the original list of arms $A$, so that $S[1] < S[2] < \ldots$. Note that this ordering does not correspond to the order that the arms were added to $S$ in the execution of Algorithm Algorithm 1. In keeping with the figures, we will visualize the arms in $A$ and the elements of $S$ from left to right where the lower-numbered arms are left and higher-numbered arms are right. We prove the lemma by induction on the list of arms $S[1], S[2], \ldots$ from left to right.

The base case is $r = 1$, where $S[1] = 0$. Clearly, $A[0]$ is the best arm of the trivial prefix $\{0\}$.

Now assume the inductive hypothesis that for all $r' \in [1, r - 1]$, $S[r']$ is an index of an $\varepsilon$-best arm of the prefix $\{0, 1, \ldots S[r']\}$. We will prove that $S[r]$ is an index to an $\varepsilon$-best arm of $\{0, 1, \ldots S[r]\}$; because every addition to $S$ is an output of FindBest, $S[r]$ is an $\varepsilon$-best arm $k_{\ell,m}$ from some $B_{\ell,m}$, and was added to $S$ from some $B_{\ell,m}$ because it did not satisfy the condition on line 18,

$$L_{\ell,m} + \varepsilon/4 = \hat{\mu}[\text{prev}] + 3\varepsilon/4 \leq \mu[k_{\ell,m}]$$

where prev is the nearest member of $S$ to the left of $B_{\ell,m}$ in round $(\ell, m)$. Because we assumed $E$, all calls to EstMean succeeded; using (2),

$$\mu[\text{prev}] + 7\varepsilon/12 \leq \mu[k_{\ell,m}] \tag{5}$$

We will use the above to show that $k_{\ell,m}$ indexes an $\varepsilon$-best arm of $A[0,1, \ldots k_{\ell,m}]$. To do so, we decompose the interval into subintervals and prove that $A[k_{\ell,m}]$ is $\varepsilon$-better than all arms in each. They are labeled as intervals $(a,b,c,d)$ in Figure 2.

Let $B_{\ell,m}$ be the block which contained $k_{\ell,m}$ when $k_{\ell,m}$ was added to $S$. It spans $i_{\ell,m}$ and $j_{\ell,m}$. Let $B_{\text{prev}}$ be the block which contained prev when prev was added to $S$. It spans $i_{\text{prev}}$ and $j_{\text{prev}}$. The subintervals we shall analyze are $[0, \text{prev}], [i_{\text{prev}}, j_{\text{prev}}], (j_{\text{prev}}, i_{\ell,m}), [i_{\ell,m}, k_{\ell,m}]$; in Figure 2, they are labeled a-d. Note that they are not all disjoint but this does not affect the conclusion. For each labeled interval, the corresponding entry in the list below proves that $A[k_{\ell,m}]$ is $\varepsilon$-better than all arms in the interval.

(a) By inductive assumption, $A[\text{prev}]$ is an $\varepsilon$-best arm of $[0, \text{prev}]$. By (4), the mean of $A[k_{\ell,m}]$ can only be larger.

(b) Because every member of $S$ is an $\varepsilon$-best arm of the block that last contained it, $A[\text{prev}]$ is an $\varepsilon$-best arm of $B_{\text{prev}} = [i_{\text{prev}}, j_{\text{prev}}]$. By (4), the mean of $A[k_{\ell,m}]$ can only be larger.

(c) Here we show that $A[k_{\ell,m}]$ is $\varepsilon$-better than all of $(j_{\text{prev}}, i_{\ell,m})$. When $k_{\ell,m}$ was added to $S$, no arm in that interval was in $S$ because prev is defined to be the closest member of $S$ to the left. These arms are deactivated; by Lemma 10, $A[\text{prev}]$ is $\varepsilon$-better than all of them. By (4), the mean of $A[k_{\ell,m}]$ can only be larger.

(d) $A[k_{\ell,m}]$ is an $\varepsilon$-best arm of $B_{\ell,m}$ which implies it is an $\varepsilon$-best arm of the subset $[i_{\ell,m}, k_{\ell,m}]$.

Because the union of the above intervals is $[0, k_{\ell,m}]$, $A[k_{\ell,m}]$ is an $\varepsilon$-best arm of that prefix. Thus, $S[r] = k_{\ell,m}$ indexes an $\varepsilon$-best arm of $[0, S[r]]$. The lemma now follows by induction. \qed
2.4.2 Efficiency of Algorithm 1 (Claim 9)

We now turn to showing that, if (1) and (2) hold, then Algorithm 1 draws $O\left(\frac{n}{\varepsilon^2} \log \frac{1}{\delta}\right)$ samples. First we will outline the proof while introducing some technical lemmas, then we will conclude by proving those lemmas.

**Proof of Claim 9.** Observe that Algorithm 1 only draws samples via the subroutines `EstMean` and `FindBest`. Measuring $A[0]$ on Line 1 of Algorithm 1 is a call to `EstMean` with error-parameters $(\varepsilon/12, \delta/2)$, so it requires just $O\left(\frac{1}{\varepsilon^2} \log \frac{1}{\delta}\right)$ samples by Lemma 4. It remains to show that the total number of samples taken by `EstMean` and `FindBest` in the main loop is bounded by $O\left(\frac{n}{\varepsilon^2} \log \frac{1}{\delta}\right)$.

Recall $n_{\ell,m}$ is the number of arms in block $B_{\ell,m}$ and $n_{\ell}$ is the number of arms in level $\ell$. Let $C_\ell$ denote the sample complexity at level $\ell$ and let $C_{\ell,m}$ denote the sample complexity incurred by block $B_{\ell,m}$ so that

$$C_\ell = \sum_{m=1}^{b_\ell} C_{\ell,m}$$

When considering block $(\ell, m)$, `EstMean` and `FindBest` are called with parameters $(\varepsilon/12, \delta_{\ell,m}/2)$. By Lemma 4 and Theorem 6, the subroutines together take $O((n_{\ell,m}/\varepsilon^2) \log(1/\delta_{\ell,m}))$ samples. Recalling that $\delta_{\ell,m} = \delta_\ell/b_\ell$ where $\delta_\ell = \delta/2^{\ell+1}$ and where $b_\ell$ is the number of blocks in the level, we can write
\[ C_\ell = O \left( \sum_{m=1}^{b_\ell} \frac{n_{\ell,m}}{\varepsilon^2} \log \frac{1}{\delta_{\ell,m}} \right) = O \left( \frac{1}{\varepsilon^2} \sum_{m=1}^{b_\ell} n_{\ell,m} \log \frac{1}{\delta_{\ell,m}} \right) = O \left( \frac{\log b_\ell}{\varepsilon^2} \sum_{m=1}^{b_\ell} n_{\ell,m} \right) = O \left( \frac{n_\ell}{\varepsilon^2} \left( \frac{\log b_\ell}{\delta} + \log 2^{\ell+1} \right) \right) = O \left( \frac{n_\ell}{\varepsilon^2} \left( \ell + \log \frac{b_\ell}{\delta} \right) \right) \]

Note that \( b_\ell \) is increasing with \( \ell \). To finish the analysis, we will show that \( n_\ell \) shrinks exponentially with \( \ell \) sufficient speed that the overall sample complexity at each level is also decaying exponentially.

First, we show that the number of blocks \( b_\ell \) can increase at most exponentially in \( \ell \).

**Lemma 12.** If \( E \) occurs, then at every level \( \ell \), there are \( b_\ell = O(\frac{10}{3^\ell} \cdot \frac{1}{\varepsilon}) \) blocks

The proof will be presented shortly. From this lemma, we have

\[ C_\ell = O \left( \frac{n_\ell}{\varepsilon^2} \left( \ell + \log \frac{1}{\varepsilon \delta} \right) \right) \quad (*) \]

Let \( \tau \) denote the number of levels \( \ell \) considered by the algorithm before terminating, let \( C \) denote the sample complexity of the main loop. Naturally,

\[ C = \sum_{\ell=1}^{\tau} C_\ell = O \left( \frac{1}{\varepsilon^2} \sum_{\ell=1}^{\tau} n_\ell \left( \ell + \log \frac{1}{\varepsilon \delta} \right) \right) = O \left( \frac{1}{\varepsilon^2} \sum_{\ell=1}^{\infty} n_\ell \left( \ell + \log \frac{1}{\varepsilon \delta} \right) \right) \quad (**) \]

where we use (*) to expand \( C_\ell \).

Finally, we show that the number of arms \( n_\ell \) decreases exponentially in \( \ell \)

**Lemma 13.** If \( E \) occurs, the number of active arms in level \( \ell \) is \( n_\ell \leq \frac{1}{2} \cdot n_{\ell-2} \) for all \( \ell > 2 \).

The proof will be presented shortly. From this lemma, we have

\[ C = (**) = O \left( \frac{n}{\varepsilon^2} \cdot \sum_{\ell=1}^{\infty} \left( \frac{1}{2} \right)^\ell \left( \ell + \log \frac{1}{\varepsilon \delta} \right) \right) = O \left( \frac{n}{\varepsilon^2} \log \frac{1}{\varepsilon \delta} \right) \]

Now that we have outlined the structure of the proof, we complete the analysis by proving Lemma 12 and Lemma 13.

**Proof of Lemma 12.** Let \( P_\ell \) denote the set of blocks in level \( \ell \) that are “parents” of “child” blocks, by which we mean those blocks of \( \ell \) that are divided into subblocks for \( \ell + 1 \). Naturally, \( P_\ell \subseteq B_\ell \) and thus \( |P_\ell| \leq |B_\ell| = b_\ell \).
If \( B_{\ell,m} \in P_\ell \), then line 24 bounds the number of children by
\[
b_{\ell,m} := \frac{4}{\varepsilon} (U_{\ell,m} - L_{\ell,m}) = (\hat{\mu}[k_{\ell,m}] - \hat{\mu}[prev] - \varepsilon/3)
\]
where \( prev \) denotes the largest member of \( S \) smaller than \( k_{\ell,m} \) at round \((\ell, m)\). Thus the number of blocks in \( \ell + 1 \) is bounded by
\[
b_{\ell+1} \leq \sum_{B_{\ell,m} \in P_\ell} b_{\ell,m} = \frac{4}{\varepsilon} \left( -\frac{\varepsilon}{3} |P_\ell| + \sum_{B_{\ell,m} \in P_\ell} \hat{\mu}[k_{\ell,m}] - \hat{\mu}[prev_{\ell,m}] \right) \tag{***}
\]
where we insert subscript to \( prev \) to disambiguate between rounds.

We will upper bound the term in the summation by lower bounding \( \hat{\mu}[prev_{\ell,m}] \). Consider two members of \( P_\ell \), \( B_{\ell,m'} \) and \( B_{\ell,m} \), such that \( m' < m \) and there is no member of \( P_\ell \) between them. By the manner in which \( prev_{\ell,m} \) is defined, \( k_{\ell,m'} \leq prev_{\ell,m} \). Recall Lemma 11, which states that \( \forall s \in S \ A[s] \) is a \( \varepsilon \)-best arm of the prefix \([s]\). This means \( \hat{\mu}[prev_{\ell,m}] > \hat{\mu}[k_{\ell,m'}] - \varepsilon \). Recall that EstMean is given error parameter \( \varepsilon/12 \). Because we assumed \( E \), (2) gives \( \hat{\mu}[prev_{\ell,m}] > \hat{\mu}[k_{\ell,m}] - 7\varepsilon/6 \). Therefore 
\[
\hat{\mu}[k_{\ell,m}] - \hat{\mu}[prev_{\ell,m}] < \hat{\mu}[k_{\ell,m}] - \hat{\mu}[k_{\ell,m'}] + 7\varepsilon/6.
\]
This lets us upper bound (*** ) by
\[
b_{\ell+1} \leq \frac{4}{\varepsilon} \left( \frac{5\varepsilon}{6} |P_\ell| + \sum_{B_{\ell,m} \in P_\ell} \hat{\mu}[k_{\ell,m}] - \hat{\mu}[k_{\ell,m'}] \right)
\]
Measurements lie within \([0, 1]\) so the summation telescopes nicely
\[
< \frac{4}{\varepsilon} \left( \frac{5\varepsilon}{6} |P_\ell| + 1 \right) < \frac{10}{3} b_\ell + \frac{4}{\varepsilon}
\]
Since \( b_1 = 1 \), we can solve the recurrence to obtain
\[
b_\ell \leq \left( \frac{10}{3} \right)^\ell + \frac{4}{\varepsilon} \sum_{\ell' = 0}^{\ell-1} \left( \frac{10}{3} \right)^{\ell'} \leq \left( \frac{10}{3} \right)^\ell \left( 1 + \frac{4}{\varepsilon} \right) = O \left( \left( \frac{10}{3} \right)^\ell \cdot \frac{1}{\varepsilon} \right)
\]
\( \square \)

Finally, we prove Lemma 13, stating that the number of active arms in a level \( \ell \) decays exponentially with \( \ell \). To do so we need two intermediate lemmas concerning the children of a given block. Lemma 14 states that, for every block \( B_{\ell,m} \), there is a bound on the number of arms its children can “contribute” to \( S \). Lemma 15 relates Lemma 14 to \( b_{\ell,m} \), which is the number of children of \( B_{\ell,m} \). Because each child contributes at most one arm to \( S \) (specifically, an arm identified by FindBest), if the number of child blocks is twice the maximum number of contributions (which we ensure in the algorithm), at least half of the child blocks do not make a contribution and are deactivated.

Once we have this guarantee for every block in a level \( \ell \), if child blocks are of roughly equal size (which we ensure in the algorithm), it follows that at least half of the active arms in level \( \ell \) are deactivated before level \( \ell + 2 \).

**Lemma 14.** Assuming \( E \) occurs, if block \( B_{\ell,m} \) contributes \( k_{\ell,m} \) to \( S \), the number of contributions made by its children is bounded by
\[
\frac{2}{\varepsilon} (U_{\ell,m} - L_{\ell,m}) = \frac{2}{\varepsilon} (\hat{\mu}[k_{\ell,m}] - \hat{\mu}[prev] - \varepsilon/3).
\]
Proof of Lemma 14. We will show that the mean of any arm in $B_{\ell,m}$ resides within an interval $(L_{\ell,m},U_{\ell,m})$ and that the $(r+1)$-st-from-the-left contribution to $S$ by the children implies a “large step” from the $r$-th. It follows that there can only be so many steps.

First, observe that the mean of the leftmost child’s contribution is greater than $\hat{\mu}[\text{prev}] + \varepsilon/2 = L_{\ell,m}$. Specifically, consider $k_1$, the leftmost contribution to $S$ by a child block. For an addition to $S$ to occur, the inequality of line 18 is false: $\hat{\mu}[k_1]$ is larger than the the measured mean of the closest arm to the left. By assumption, this is $\text{prev}$. (4) tells us that $\mu[k_1] \geq \mu[\text{prev}] + 7\varepsilon/12$. From (2) we know $\mu[\text{prev}] \geq \hat{\mu}[\text{prev}] - \varepsilon/12$ so $\mu[k_1] \geq \hat{\mu}[\text{prev}] + \varepsilon/2$.

Second, observe that the mean of any child’s contribution is at most $\leq \hat{\mu}[k_{\ell,m}] + \varepsilon/6 = U_{\ell,m}$. Specifically, by assuming $E$, we have from (2) that $\mu[k_{\ell,m}] \leq \hat{\mu}[k_{\ell,m}] + \varepsilon/12$ and from (1) that $|\mu[k_{\ell,m}] - \mu^*_{\ell,m}| \leq \varepsilon/12$. Combining these inequalities, we have $\mu^*_{\ell,m} \leq \mu[k_{\ell,m}] + \varepsilon/12 \leq \hat{\mu}[k_{\ell,m}] + \varepsilon/6$.

From this, we can show that every contribution of level $\ell + 1$ has mean $\varepsilon/2$ more than the previous such contribution. Consider $B_{\ell+1,p}$, the child block of $B_{\ell,m}$ that makes the $r$th contribution in level $\ell + 1$. Let $B_{\ell+1,q}$ denote the block that makes the $(r+1)^{th}$ contribution. At the time of adding $k_{\ell+1,q}$ to $S$, $k_{\ell+1,p}$ is the closest member of $S$ to its left. Because $k_{\ell+1,q}$ was added to $S$, the inequality of line 18 is false: from (4), $\mu[k_{\ell+1,q}] > \mu[k_{\ell+1,p}] + \varepsilon/2$ where $k_{\ell+1,p}$ stands in for $\text{prev}$.

Combining these three facts immediately implies the lemma. \qed

Lemma 15. Assuming $E$ occurs, if $B_{\ell,m}$ contributes some $k_{\ell,m}$ to $S$ in level $\ell$, then at most half of the arms in $B_{\ell,m}$ are active in level $\ell + 2$.

Proof of Lemma 15. There are two cases, depending on the size of $B_{\ell,m}$ relative to $b_{\ell,m}$, an upper bound on the number of child blocks that will be made.

Case 1: $k_{\ell,m} - i_{\ell,m} \geq b_{\ell,m}$. Because there are enough arms, the algorithm breaks $B_{\ell,m}$ into exactly $b_{\ell,m} = (4/\varepsilon)(U_{\ell,m} - L_{\ell,m})$ child blocks (on line 24). Lemma 14 bounds the number of contributions in level $\ell + 1$ by $(2/\varepsilon)(U_{\ell,m} - L_{\ell,m})$. Thus the number of contributions is at most half of $b_{\ell,m}$. If a block does not add an arm to $S$, all arms in that block are deactivated. Because at most half of the child blocks contribute to $S$ and the child blocks are of (approximately) equal size, at most half of the arms in $B_{\ell,m}$ are active after level $\ell + 1$.

Case 2: $k_{\ell,m} - i_{\ell,m} < b_{\ell,m}$. Line 24 breaks $B_{\ell,m}$ into single-arm blocks. If one of those blocks does not contribute to $S$, then the block is deactivated. If it does contribute to $S$, line 24 cannot form child blocks. Thus no arm in $B_{\ell,m}$ is active after level $\ell + 1$. \qed

Given Lemma 15, it is clear that the number of active arms of a level decays exponentially, which is the assertion of Lemma 13.

2.4.3 Analysis of Algorithm 2

Claim 16. Let $S$ be the output of Algorithm 1 on input $(A,\varepsilon,\delta)$. If $E$ occurs, the output of Algorithm 2 on $S$ is an $\varepsilon$-skyline of size $O(1/\varepsilon)$.

Note that Algorithm 2 is deterministic, and we only rely on the condition $E$ to establish that the input $S$ is an $\varepsilon$-skyline and that the estimates in $M$ are accurate.

Proof. From Claim 8, and the fact that $E$ occurs, we know that $S$ is an $\varepsilon$-skyline of $A$.

First, we shall see that the changes Algorithm 2 makes to $S$ do not invalidate either Lemma 10 or Lemma 11, which means Claim 8 still holds. Observe that when line 5 of Algorithm 2 is true, every removed index $s'$ is ensured that the $s \in S$ to its left indexes an $\varepsilon$-better arm. This follows from identical algebra that led to (3): substitute $\text{prev}$ with $s$ and $k_{\ell,m}$ with $s'$. Because every removed
index \( s' \notin S \) is ensured that the \( s \in S \) to its left indexes an \( \varepsilon \)-better arm, Lemma 10 still holds. Lemma 11 is unaffected as Algorithm 2 does not add any arms.

Finally, we will see that only \( O(1/\varepsilon) \) arms remain in \( S \) after running Algorithm 2. We know \( \hat{\mu}[S[i]] + 3\varepsilon/4 \leq \hat{\mu}[S[i+1]] \) holds for all \( i \in [1,|S|] \). From identical algebra that led to (4), every \( S[i+1] \) has mean \( \mu[S[i+1]] \) at least \( \frac{7}{12} \) more than \( \mu[S[i]] \). Because the mean of each arm is within \([0,1]\), there can be at most \( \frac{12}{7\varepsilon} = O(\frac{1}{\varepsilon}) \) indices in \( S \).

\[ \square \]

3 A Sample Complexity Lower Bound for \( \varepsilon \)-Skyline Identification

Notice that the bottleneck in the sample complexity of our algorithm is that, in the first stage, we have to solve each of these problems successfully with probability at least \( 1 - \frac{\varepsilon}{T} \), which (by a slight strengthening of [MT04]) requires \( O(\frac{n}{\varepsilon^2} \log \frac{1}{\varepsilon^2}) \) samples. We then show that, when \( \varepsilon > \frac{1}{n} \), any algorithm for solving the \( \varepsilon \)-Pareto-optimal-arm-identification problem must solve \( \Omega(\frac{1}{\varepsilon}) \) independent \( \varepsilon \)-best-arm identification problems, thereby obtaining our lower bound.

In order to obtain our lower bound, we will need to use some of the specific properties of the hard instances for best-arm identification that were used in the lower bound of [MT04]. Specifically, implicit in their proof is the following theorem:

**Theorem 17** (Implicit in [MT04]). Fix any \( p \in [\frac{1}{4},\frac{3}{4}] \). Let \( A \) be an algorithm, and define the following game \( \text{id}_{1,n,p,\varepsilon}(A) \):

1. Choose a random \( c \in [n] \)
2. Define \( A = \{A[1],\ldots,A[n]\} \) so that \( A[c] \sim \text{Ber}(p+2\varepsilon) \) and for \( i \neq c \), \( A[i] \sim \text{Ber}(p) \).
3. \( A \) draws samples from the arms \( A \)
4. \( A \) outputs a guess \( c^* \)

Then there exists a function \( S(n,\varepsilon,\delta) = \Omega(\frac{n}{\varepsilon^3} \log \frac{1}{\delta}) \) such that if \( A \) draws fewer than \( S(n,\varepsilon,\delta) \) samples in expectation,

\[
P_{\text{id}_{1,n,p,\varepsilon}(A)}[c^* = c] < 1 - \delta.
\]

Note that the unique \( \varepsilon \)-best-arm is \( c \), therefore any algorithm that has at least a \( 1 - \delta \) probability of identifying an \( \varepsilon \)-best-arm on this sort of instance must draw at least \( S(n,\varepsilon,\delta) \) samples in expectation.

The next lemma is a simple “direct-product lemma” for this game, which roughly asserts that any algorithm \( A \) that is asked to solve \( T \) independent copies of this game, and succeeds in all of these copies with probability at least \( 1 - \delta \), must draw nearly \( T \cdot S(n,\varepsilon,\delta/T) \) samples, which means that the strategy of solving each copy independently with probability \( 1 - \delta/T \) and suffering a union bound over the \( T \) copies is essentially optimal.

**Lemma 18.** Fix any \( T \in \mathbb{N} \) and any \( \vec{p} = (p_1,\ldots,p_T) \in [\frac{1}{4},\frac{3}{4}]^T \). Let \( A \) be an algorithm, and define the following game \( \text{id}_{T,n,\vec{p},\varepsilon,T}(A) \):

1. Choose independent random \( \vec{c} = (c_1,\ldots,c_T) \in [n]^T \)
2. For every \( t \in [T] \), define a set of arms \( A_t = \{A_t[1],\ldots,A_t[n]\} \) so that \( A[c_t] \sim \text{Ber}(p_t+2\varepsilon) \) and for \( i \neq c_t \), \( A[i] \sim \text{Ber}(p_t) \). Let \( A = A_1 \cup \cdots \cup A_T \).
3. \( A \) draws samples from the arms \( A \)
4. \( A \) outputs a guess \( \vec{c}^* \)

15
Then if $\mathcal{A}$ draws fewer than $\frac{T}{2} \cdot S(n, \varepsilon, \sqrt{32/\varepsilon^3} \sqrt{T}) = \Omega(\frac{T}{\varepsilon^2} \log \frac{T}{\delta})$ samples in expectation,

$$\Pr_{\text{idT}_{n, \varepsilon, T}(\mathcal{A})}[c^* \neq c] < 1 - \delta.$$  

As above, note that for every set of arms $A_t$, the unique best arm is $c_t$, and therefore the lower bound on the number of samples applies to any algorithm $\mathcal{A}$ that solves $\varepsilon$-best-arm identification simultaneously on all sets $A_1, \ldots, A_T$. We now give a high level sketch of why this direct product result holds.

**Proof.** Note that the lemma would be immediate if we could argue that the probability that $\mathcal{A}$ succeeds in identifying each value $c_t$ is independent across the $T$ copies, but this is unfortunately not true. However, suppose we let the random variable $V$ denote the entire sequence of samples and values obtained by $\mathcal{A}$ throughout the course of the algorithm. Then for every realization $v$, the random variables $(c_t | V = v)$ are mutually independent. This claim can be proven by induction on the number of samples made by $\mathcal{A}$—initially all values $c_t$ are chosen independently, and in each step the response received by $\mathcal{A}$ depends on only a single value $c_t$.

To simplify arithmetic, let $p := \sqrt{32/\varepsilon^3} \sqrt{T}$ denote the parameter from the theorem statement. Now, suppose we have an algorithm $\mathcal{A}$ that draws fewer than $\frac{T}{2} \cdot S(n, \varepsilon, p)$ samples in expectation. By linearity of expectation and Markov’s inequality, there is a set $U \subseteq [T]$ of size at least $T/2$ such that for every $t \in U$, the expected number of samples made by $\mathcal{A}$—initially all values $c_t$ are chosen independently, and in each step the response received by $\mathcal{A}$—depends on only a single value $c_t$.

From this, and two applications of Markov’s inequality we obtain the following: there exists a subset of views $V'$ such that $\Pr[V'] \geq \frac{p}{4}$ and a collection of subsets $\{U' \subseteq U\} \subset U$, each of size at least $|U| \cdot \frac{p}{8} = \frac{pT}{16}$ such that

$$\forall v \in V', \forall t \in U' \Pr_{\text{idT}_{n, \varepsilon, T}(\mathcal{A})}[c^*_t \neq c_t | v] > p/4$$

Using the mutual independence of the variables $(c_t | v)$, we obtain

$$\forall v \in V' \Pr_{\text{idT}_{n, \varepsilon, T}(\mathcal{A})}[\exists t \in U' c^*_t \neq c_t | v] > 1 - (1 - p/4)^{|U'|} \geq 1 - e^{-p^2T/32}$$

Since $\exists t \in U' c^*_t \neq c_t$ implies that $\bar{c} \neq c$, we have

$$\forall v \in V' \Pr_{\text{idT}_{n, \varepsilon, T}(\mathcal{A})}[\bar{c} \neq c | v] > 1 - e^{-p^2T/32}$$

Using the fact that $\Pr[V'] \geq \frac{p}{4}$, we have

$$\Pr_{\text{idT}_{n, \varepsilon, T}(\mathcal{A})}[\bar{c} \neq c] > \frac{p}{4} \left(1 - e^{-p^2T/32}\right) \geq \frac{p^3T}{256} = \delta$$

where the final inequality uses the fact that $p^3T$ is smaller than some absolute constant so that we can write $1 - e^{-p^2T/32} \geq p^2T/64$ and the final equality is by substituting our choice of $p$. This completes our proof sketch.

Finally, we can show that, for some $T = \Omega(1/\varepsilon)$, and some choice of $\bar{p} = (p_1, \ldots, p_T)$, and $\varepsilon$-skyline identification algorithm can be used to “win” the game $\text{idT}_{n, \bar{p}, \varepsilon, T}$, from which we immediately obtain a lower bound on the sample complexity of $\varepsilon$-skyline identification.
Theorem 19. Fix any $n \in \mathbb{N}$ and fix $\varepsilon, \delta > 0$ smaller than some absolute constants. Suppose there is an algorithm $A$ that, with probability at least $1 - \delta$, identifies an $\varepsilon$-skyline on $n$ arms. Then in expectation $A$ draws at least $\frac{1}{4\varepsilon} \cdot S(2n, \varepsilon, \sqrt[3]{128\varepsilon}) = \Omega(\frac{n}{\varepsilon^2} \log \frac{1}{\delta^2})$ samples.

Proof. Define $T = \frac{1}{2\varepsilon}$ and $m = 2\varepsilon n$ so that $n = Tm$. For notational simplicity, we assume $T$ and $m$ are integers. Define

$$\vec{p}^\prime := \left(\frac{1}{4}, \frac{1}{4} + 2\varepsilon, \frac{1}{4} + 4\varepsilon, \ldots, \frac{3}{4}\right)$$

By construction, $\vec{p}^\prime \in [\frac{1}{4}, \frac{3}{4}]^T$. By Lemma 18, if $A$ is an algorithm such that

$$\mathbb{P}_{\text{id}_T(m, \vec{p}^\prime, \varepsilon, T)}[\bar{c} = c^*] \geq 1 - \delta,$$

then $A$ draws $\frac{T}{2} \cdot S(m, \varepsilon, \sqrt[3]{256\varepsilon/3}) = \Omega(\frac{n}{\varepsilon^2} \log \frac{1}{\delta^2})$ samples in expectation.

Thus, all that remains is to show that, given any algorithm $A'$ that identifies an $\varepsilon$-skyline on $n$ arms with probability at least $1 - \delta$, we can construct an algorithm $A$ satisfying (6). We can construct such an algorithm as follows:

1. Generate a vector $\vec{c}$ and generate the sets $A_1, \ldots, A_T$ as in $\text{id}_T(m, \vec{p}, \varepsilon, T)$. Then generate a set of $n = Tm$ arms $A'$ by concatenating the sets $A_1, \ldots, A_T$ together in order. Specifically, for $t \in [T], i \in [m]$, let $A'[\lfloor (t - 1)T + i \rfloor] = A_t[i]$.
2. Obtain $S \subseteq [n]$ by running $A'$ on the set of arms $A'$
3. For every $t \in [T]$, let $A'_t = \{(t - 1)m + 1, \ldots, tm\}$ and let $c^*_t = \max S \cap A'_t$ be the largest index of any arm in the skyline that lies in $A'_t$. Note that $A[c^*_t]$ corresponds to the arm $A_t[j \mod m]$.
4. Output $\bar{c}^* = (c^*_1, \ldots, c^*_T)$

To complete the proof, we will show that if $S$ is an $\varepsilon$-skyline for $A'$ then $\bar{c}^* = \vec{c}$. First, by construction, $c_t$ denotes the unique $\varepsilon$-best arm in $A_t$, and by construction of $\vec{p}^\prime$, it is also better than all arms in $A_1, \ldots, A_{t-1}$. When we construct the arms $A', A_t$ becomes the set of arms $A'_t$ and similarly $c_t$ becomes the arm $(t - 1)m + c_t$. By construction of $\vec{p}^\prime$, arm $(t - 1)m + c_t$ in $A'$ is the best arm among the set $\{1, \ldots, tm\}$. By the first condition of Definition 1, any valid $\varepsilon$-skyline $S$ must contain each arm $(t - 1)n + c_t$. Similarly, by the second condition of Definition 1, arms in $A'$ that are in the set $\{(t - 1)m + c_t + 1, tm\}$ cannot appear in $S$ because they are dominated by arm $(t - 1)m + c_t$. Finally, because $(t - 1)m + c_t \in S \cap A'_t$, and no larger element of $A' \cap A'_t$ is in $S$, we have $c^*_t = c_t$ for every $t$. This completes the proof.

Acknowledgements

We are grateful to Jay Aslam, Maryam Aziz, Bobby Kleinberg, and Alex Slivkins for many helpful discussions related to this work.

References

[ACOD16] Peter Auer, Chao-Kai Chiang, Ronald Ortner, and Madalina Drugan. Pareto front identification from stochastic bandit feedback. In Artificial Intelligence and Statistics, 2016.

[BKS01] Stephan Borzsony, Donald Kossmann, and Konrad Stocker. The skyline operator. In International Conference on Data Engineering (ICDE). IEEE, 2001.
[CGL16] Lijie Chen, Anupam Gupta, and Jian Li. Pure exploration of multi-armed bandit under matroid constraints. In Conference on Learning Theory, 2016.

[CGL+17] Lijie Chen, Anupam Gupta, Jian Li, Mingda Qiao, and Ruosong Wang. Nearly optimal sampling algorithms for combinatorial pure exploration. arXiv preprint arXiv:1706.01081, 2017.

[EDMM06] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. Journal of Machine Learning Research, 2006.

[GGLB11] Victor Gabillon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Sébastien Bubeck. Multi-bandit best arm identification. In Advances in Neural Information Processing Systems, 2011.

[Kle06] Robert D. Kleinberg. Anytime algorithms for multi-armed bandit problems. In Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2006.

[KS10] Shivaram Kalyanakrishnan and Peter Stone. Efficient selection of multiple bandit arms: Theory and practice. In Proceedings of the 27th International Conference on Machine Learning (ICML), 2010.

[KTAS12] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In Proceedings of the 29th International Conference on Machine Learning (ICML), 2012.

[MT04] Shie Mannor and John N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. Journal of Machine Learning Research, 2004.