Adaptive Double-Exploration Tradeoff for Outlier Detection

Xiaojin Zhang,[1] **Honglei Zhuang,**[2] **Shengyu Zhang,**[3] **Yuan Zhou**[2]

[1]The Chinese University of Hong Kong, [2]University of Illinois Urbana-Champaign, [3]Tencent

xjzhang@cse.cuhk.edu.hk, {hzhuang3, yuanz}@illinois.edu, shengyuzhang@gmail.com

Abstract

We study a variant of the thresholding bandit problem (TBP) in the context of outlier detection, where the objective is to identify the outliers whose rewards are above a threshold. Distinct from the traditional TBP, the threshold is defined as a function of the rewards of all the arms, which is motivated by the criterion for identifying outliers. The learner needs to explore the rewards of the arms as well as the threshold. We refer to this problem as "double exploration for outlier detection". We construct an adaptively updated confidence interval for the threshold, based on the estimated value of the threshold in the previous rounds. Furthermore, by automatically trading off exploring the individual arms and exploring the outlier threshold, we provide an efficient algorithm in terms of the sample complexity. Experimental results on both synthetic datasets and real-world datasets demonstrate the efficiency of our algorithm.

Introduction

Multi-armed bandit (MAB) problems model the tradeoff between *exploration* and *exploitation* inherent in a great amount of sequential decision problems. In the canonical multi-armed bandit problem, the learner is presented with a set of arms. The learner needs to explore distinct arms to discover the potential of various arms, and exploit the most rewarding arms based on the information that has been collected. The goal of the learner is to maximize the cumulative reward (Robbins 1985; Auer, Cesa-Bianchi, and Fischer 2002). In terms of the *pure exploration* problem, the learner is interested in optimizing the performance in some decision-making task by making full use of the limited budget (Bubeck, Munos, and Stoltz 2011; Gabillon et al. 2011). A flurry of work focuses on identifying the set of K arms with the largest expected rewards, called the top-k arm identification problem (Mnih, Szepesvári, and Audibert 2008; Kalyanakrishnan and Stone 2010; Kalyanakrishnan et al. 2012; Kaufmann and Kalyanakrishnan 2013; Zhou, Chen, and Li 2014; Chen et al. 2014). This line of work could be categorized into two distinct settings: *fixed confidence* and *fixed budget*. The former setting focuses on identifying the best action satisfying a fixed confidence level using the minimum number of samples, while the latter one aims at max-

imizing the probability of outputting the best action using a fixed number of samples.

In this paper, we focus on the outlier detection problem, a practically important problem which was firstly investigated in the framework of multi-armed bandit under the *fixed confidence* setting by Zhuang, Wang, and Wang (2017). They regard the outliers as the arms with rewards above certain threshold, which is defined based on a generalized statistical technique called k-sigma rule of thumb (Coolidge 2012). Specifically, an arm is referred to as an outlier, if and only if the reward of which lies above k standard deviations of the mean. The expected reward of each arm is unknown to the learner. Moreover, the threshold is a function of the rewards of all the arms and is also unknown, which is the distinguishing feature from the classical thresholding bandit problem and top-k arm identification problem. The learner could select an arm to pull at each step, and the reward drawn from the distribution of pulled arm is observed as the feedback information. In this process, the rewards of the arms and the threshold are gradually learned. The final goal of the learner is to output the correct outlier set with high probability, and at the same time minimizing the number of samples used.

The outlier detection problem in the framework of multi-armed bandit raises two main challenges to the design of algorithms. On the one hand, the learning strategies for outlier detection needs to address the *double-exploration* dilemma, i.e. the search for a balance between exploring the arms and exploring the threshold. If the learner plays exclusively on the arms that are closer to the threshold, he might fail to estimate the threshold efficiently. If the learner is persistent in sampling all the arms uniformly, he might spend too much efforts on the arms that have already been identified. On the other hand, the construction of the confidence interval for the threshold, which is related to the standard deviation of the rewards of all the arms, is not an easy task. The tighter the confidence interval is, the smaller sample complexity the resulting algorithm could obtain.

We note that the confidence radius for the threshold constructed by Zhuang, Wang, and Wang (2017) increases with the number of arms, which might make the learning strategy inefficient in terms of the sample complexity. This issue becomes more severe when identifying outliers among a larger set of arms. Our approach alleviates this problem by constructing a confidence radius for the threshold which is

independent of the number of arms and could be tuned adaptively. Besides, their sampling strategies are forced to continue sampling the arms that have already been identified, since the confidence radius for the threshold is related to the harmonic mean of the number of samples of all the arms. We devise distinct sampling strategies for estimating the expected rewards of the arms and the thresholds separately, and make it feasible to directly remove the arms that have been identified. We provide an algorithm that balances between these two sampling strategies in an adaptive manner, and provide theoretical guarantee in terms of both correctness and sampling complexity. We further apply our algorithm on both synthetic datasets and real-world datasets. Experimental results demonstrate that our algorithm achieves considerable improvement over the state-of-the-art algorithms for outlier detection in the MAB framework.

Related Work

Since the first introduction of the multi-armed bandit by Thompson (1933) in the scenario of medical trials, it has received a great amount of interest. The goal of the learner is to minimize the cumulative regret. In the pure exploration setting, the learner is assessed in terms of the simple regret instead of the cumulative regret. Locatelli, Gutzeit, and Carpentier (2016) investigates a specific pure exploration problem in the fixed budget setting, referred to as the thresholding bandit problem. This problem aims at finding the arms with rewards larger than a given threshold within a fixed time horizon.

There has been a large body of work on the problem of best arm identification in both fixed confidence and fixed budget setting in the literature (Even-Dar, Mannor, and Mansour 2002; Bubeck, Munos, and Stoltz 2011; Gabillon, Ghavamzadeh, and Lazaric 2012; Jamieson et al. 2014; Kaufmann, Cappé, and Garivier 2016). A line of work uses the elimination-based approach, which successively removes the arms that are believed to be suboptimal with certain confidence (Paulson 1964; Maron and Moore 1994; Even-Dar, Mannor, and Mansour 2002). Mnih, Szepesvári, and Audibert (2008) proposed an algorithm taking advantage of the empirical variance of the arms. This problem is further extended to the top-k arm identification problem.

Our problem fits into the fixed confidence framework, but is distinct from the top-k arm identification problem. The number of outliers is not fixed, and depends on the rewards of all the arms, thereby could not be directly reduced to the top-k arm identification problem. Another line of work relaxed the classical optimal arm identification problem to ϵ-optimal arm identification, aiming at finding an arm which is at least ϵ-close to the optimal arm with probability at least $1 - \delta$ (Domingo, Gavaldà, and Watanabe 2002; Even-Dar, Mannor, and Mansour 2006). Kalyanakrishnan and Stone (2010) generalized the setting as identifying (ϵ, m)-optimal arm, the rewards of which is within ϵ of the m-th optimal arm. Kalyanakrishnan et al. (2012) further proposed the LUCB algorithm based on the upper and lower confidence bound for the generalized case. In this paper, we focus on the case when ϵ is 0. That is, the arms with rewards above the exact threshold are regarded as the outliers.

The problem of outlier detection has been widely investigated in the field of data mining (Chandola, Banerjee, and Kumar 2009). The approaches used for outlier detection include classification-based (De Stefano, Sansone, and Vento 2000; Roth 2006), clustering-based (Ester et al. 1996; Yu, Sheikholeslami, and Zhang 2002), nearest neighbor-based (Guttormsson et al. 1999; Kou, Lu, and Chen 2006), and statistical techniques (Torr and Murray 1993; Horn et al. 2001). Most existing approaches on outlier detection do not belong to the domain of online algorithm. The first work casting the outlier detection problem into the MAB framework belongs to a recent work proposed by Zhuang, Wang, and Wang (2017), which is the work mostly related to ours. They proposed two algorithms named RR and WRR, aiming at finding the correct outlier set with high probability. The rewards of the arms and the threshold are unknown but could be learned by sampling. RR algorithm simply samples each arm evenly in an iterative manner, while WRR algorithm allocates more samples to the arms that are not yet identified. The confidence radius for the threshold constructed in their work scales with the number of arms, which may greatly hinder the proposal of an efficient algorithm. Our work overcomes this deficiency by constructing a confidence radius for the threshold which is independent of the number of arms (if we ignore logarithmic factors), and provides a more flexible trade-off between exploring the arms and exploring the threshold.

Problem Formulation

In this section, we introduce the outlier detection problem in the framework of multi-armed bandits. The learner is presented with a set of n arms which are enumerated by $[n] = \{1, 2, \ldots, n\}$. The reward distribution associated with each arm i is bounded in $[a, b]$ with mean y_i. Without loss of generality, we assume that $a \geq 0$. Denote by $R = b - a$, and by $R' = b^2 - a^2$. Each pull of arm $i \in [n]$ generates a sample drawn from the distribution corresponding to arm i, which is independent from the historical pulls. The term "outlier" refers to the arm whose reward lies above k standard deviations of the mean. Consequently, the threshold for distinguishing the outliers from the normal arms is defined as:

$$\theta = \mu_y + k\sigma_y, \tag{1}$$

where $\mu_y = \dfrac{1}{n} \sum_{i=1}^{n} y_i$ is the mean of all the arms, $\sigma_y = \sqrt{\dfrac{1}{n} \sum_{i=1}^{n} (y_i - \mu_y)^2}$ is the standard deviation of all the arms, and k is a constant that could be set based on specific application scenario.

Denote by $\mathcal{O} = \{S : S \subset [n]\}$ the set of all subsets of $[n]$. Define $O^* = \{i \in [n] : y_i \geq \theta\}$ as the correct outlier set which contains the arms with expected reward above the threshold θ. The goal of the learner is to identify the correct outlier set O^* from $\mathcal{O}$ by sampling the arms in the following sequential manner.

Initially, the learner is presented with n arms, he is therefore aware of all the probable combinations of the arms $\mathcal{O}$, but the reward distributions of the arms are unknown to the learner. At each round, the learner selects an arm to pull according to the sampling strategy, and the reward drawn from the corresponding distribution is revealed to the learner. This process continues until the learner has identified a set O satisfying that $\mathbb{P}(O = O^*) \geq 1 - \delta$ for a given confidence parameter δ. The performance of the learner is measured by the sample complexity, which is the number of samples it requires with high probability.

Algorithms and Results

In this section, we illustrate an online learning algorithm for the outlier detection problem in the fixed confidence setting, and present the theoretical results including the correctness guarantee and sample complexity.

The Algorithm

The arms with expected reward deviating the mean k standard deviation are referred to as the outliers. In order to efficiently identify the outliers, the algorithm needs to balance well between exploration for the rewards of the arms and exploration for the threshold. Two distinct sampling strategies are designed to address the *double-exploration* dilemma. Specifically, we use sequential sampling approach to estimate the expected rewards of the arms, and random sampling approach to estimate the threshold separately.

- **Sequential Sample for The Arms** The learner pulls each arm i in the candidate set once, and observes the rewards drawn from the corresponding distribution. The observed rewards are used to estimate the expected rewards of the arms.

- **Random Sample for The Threshold** The learner samples an arm uniformly at random from the n arms, and pulls the sampled arm twice. The observed rewards are used to estimate the threshold.

Denote by $m_{i,t}$ the number of times arm i is sampled prior to round t in the process of sequential sample, and by $y_{i,1}, y_{i,2}, \ldots, y_{i,m_{i,t}}$ the sequence of the sampled rewards. The estimator for y_i at round t is the empirical mean of arm i after $m_{i,t}$ samples, represented as

$$\hat{y}_{i,t} = \frac{1}{m_{i,t}} \sum_{l=1}^{m_{i,t}} y_{i,l}. \tag{2}$$

Denote by $m_{\theta,t}$ the number of times the learner performs random sample for the threshold prior to round t, and by $(x_{1,1}, x_{1,2}), (x_{2,1}, x_{2,2}), \ldots, (x_{m_{\theta,t},1}, x_{m_{\theta,t},2})$ the sequence of associated rewards. The estimator for threshold θ at round t is

$$\hat{\theta}_t = \hat{\mu}_{y,t} + k\hat{\sigma}_{y,t}, \tag{3}$$

where $\hat{\mu}_{y,t} = \frac{1}{m_{\theta,t}} \sum_{l=1}^{m_{\theta,t}} x_{l,1}$, and $\hat{\sigma}_{y,t} =$

$$\sqrt{\left| \frac{1}{m_{\theta,t}} \sum_{l=1}^{m_{\theta,t}} x_{l,1} x_{l,2} - \frac{1}{m_{\theta,t}^2} \sum_{l=1}^{m_{\theta,t}} \sum_{h=1}^{m_{\theta,t}} x_{l,1} x_{h,2} \right|}.$$

The pseudo-code of our proposed algorithm Adaptive Double Exploration (ADE) is illustrated in Algorithm 1. We denote by $m_{a,t}$ the number of times the learner performs sequential sampling prior to round t. In the initialization phase, we perform two sampling strategies separately, ensuring that $m_{\theta,t} \geq 1$ and $m_{a,t} \geq 1$ after the initialization steps.

At each round t, the algorithm chooses to sample for either the arms or the threshold, whichever has larger confidence radius. As a result, the sampling for estimating the threshold and the rewards are balanced well in an adaptive manner. Then, it calculates the estimator $\hat{y}_{i,t}$ on the expected reward of each arm $i \in [n]$ and the estimator $\hat{\theta}_t$ on the threshold θ, which are defined based on Eq.(2) and Eq.(3). Besides, the confidence radius $r_{i,t}$ for each arm $i \in [n]$ and $r_{\theta,t}$ for the threshold θ are also maintained.

We separate the outliers from the normal arms based on the calculated confidence intervals in the following manner. If the lower confidence bound of arm i at round t (represented as $L_{i,t}$) is greater than or equal to the upper confidence bound of θ (represented as $U_{\theta,t}$), then arm i is identified as an outlier. If $L_{\theta,t} \geq U_{i,t}$, then arm i is regarded as a normal arm. Otherwise, arm i remains in the candidate set and the algorithm continues to sample this arm in the next round. The arms that have been identified could be directly removed in the process of sequential sampling. As a result, the arms that are farther from the threshold could be pulled with less number of times in the process of the sequential sampling. The algorithm is allowed to terminate until all the arms are determined as either an outlier or a normal arm. Figure 1 illustrates the confidence intervals of the arms and the threshold at the initial and termination rounds of the algorithm.

Remark: The samples obtained could be used to estimate both the threshold and the expected rewards of the arms. The sample complexity could be reduced by at most a half, which is still of the same order as the algorithm we presented. To ensure that the main idea of our algorithm could be presented in a concise and clear way, we decided to present this version of algorithm. The proofs of all the Lemmas and Theorems that are not presented are deferred to the full version of this paper.

One of the key issues of designing an efficient algorithm lies in the construction of the confidence intervals. Denote by $\delta_t = 3\delta/((n+4)\pi^2 t^2)$, the confidence intervals constructed for the threshold and the arms are illustrated in the following section.

Construction of Confidence Interval

Conditioned on $m_{\theta,t} = m$, the confidence interval of μ_y is illustrated in the following lemma.

Lemma 1. *With probability at least $1 - 2\delta_t$, we have*
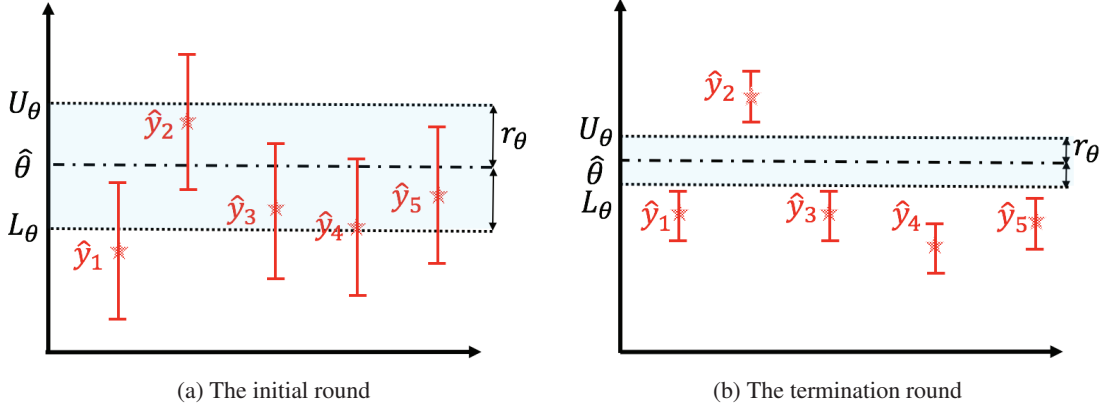
(a) The initial round   (b) The termination round

Figure 1: Illustration of outlier detection based on confidence interval

$|\hat{\mu}_{y,t} - \mu_y| \leq r_{\mu,t}$, where $r_{\mu,t} = R\sqrt{\dfrac{\log(1/\delta_t)}{2m}}$, and

$\delta_t = \dfrac{3\delta}{(n+4)\pi^2 t^2}$.

Lemma 2. *With probability at least $1 - \delta_t$, we have that*

$$|\hat{\sigma}_{y,t}^2 - \sigma_y^2| \leq \epsilon_{\sigma,t}, \qquad (4)$$

where $\epsilon_{\sigma,t} = (R' + 2bR)\sqrt{\dfrac{\log(6/\delta_t)}{2m}}$.

Proof.

$$\mathbb{E}[\frac{1}{m}\sum_{l=1}^m x_{l,1}x_{l,2}] = \frac{1}{m}\sum_{l=1}^m \mathbb{E}[x_{l,1}x_{l,2}]$$

$$= \frac{1}{m}\sum_{l=1}^m \mathbb{E}[\mathbb{E}[x_{l,1}x_{l,2}|y]] = \frac{1}{n}\sum_{i=1}^n y_i^2.$$

Denote by $V_t = \dfrac{1}{m}\sum_{l=1}^m x_{l,1}x_{l,2} - \dfrac{1}{m^2}\sum_{l=1}^m\sum_{h=1}^m x_{l,1}x_{h,2}$.

$$\mathbb{P}(|\sigma_y^2 - V_t| \geq \epsilon)$$

$$\leq \mathbb{P}\left(|\frac{1}{m}\sum_{l=1}^m(\mathbb{E}[x_{l,1}x_{l,2}] - x_{l,1}x_{l,2})| \geq \epsilon_1\right)$$

$$+ \mathbb{P}\left(|\frac{1}{m^2}\sum_{l=1}^m x_{l,1}\sum_{l=1}^m x_{l,2} - \mu_y^2| \geq \epsilon_2\right), \qquad (5)$$

where $\epsilon = \epsilon_1 + \epsilon_2$.

If $|(\dfrac{1}{m}\sum_{l=1}^m x_{l,1})^2 - \mu_y^2| \leq \epsilon_2$ and $|(\dfrac{1}{m}\sum_{l=1}^m x_{t,2})^2 - \mu_y^2| \leq$

ϵ_2, then we have $|\dfrac{1}{m^2}\sum_{l=1}^m x_{l,1}\sum_{l=1}^m x_{l,2} - \mu_y^2| \leq \epsilon_2$, since

$\dfrac{1}{m}\sum_{l=1}^m x_{l,1} \geq a \geq 0$, and $\dfrac{1}{m}\sum_{l=1}^m x_{l,2} \geq a \geq 0$. Therefore,

the event $|\dfrac{1}{m^2}\sum_{l=1}^m x_{l,1}\sum_{l=1}^m x_{l,2} - \mu_y^2| \geq \epsilon_2$ implies that either

$|(\dfrac{1}{m}\sum_{l=1}^m x_{l,1})^2 - \mu_y^2| \geq \epsilon_2$ or $|(\dfrac{1}{m}\sum_{l=1}^m x_{l,2})^2 - \mu_y^2| \geq \epsilon_2$.

Thus, we have that

$$\mathbb{P}\left(|\frac{1}{m^2}\sum_{l=1}^m x_{l,1}\sum_{l=1}^m x_{l,2} - \mu_y^2| \geq \epsilon_2\right)$$

$$\leq \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,1})^2 - \mu_y^2| \geq \epsilon_2\right)$$

$$+ \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,2})^2 - \mu_y^2| \geq \epsilon_2\right). \qquad (6)$$

Combine Ineq.(5) and Ineq.(6), we have

$$\mathbb{P}\left(|\sigma_y^2 - V_t| \geq \epsilon\right)$$

$$\leq \mathbb{P}\left(|\frac{1}{m}\sum_{l=1}^m(\mathbb{E}[x_{l,1}x_{l,2}] - x_{l,1}x_{l,2})| \geq \epsilon_1\right)$$

$$+ \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,1})^2 - \mu_y^2| \geq \epsilon_2\right)$$

$$+ \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,2})^2 - \mu_y^2| \geq \epsilon_2\right).$$

According to Hoeffding's inequality, we have $\mathbb{P}(|\dfrac{1}{m}\sum_{l=1}^m(\mathbb{E}[x_{l,1}x_{l,2}] - x_{l,1}x_{l,2})| \geq \epsilon_1) \leq \delta_1$, where $\epsilon_1 = R'\sqrt{\log(2/\delta_1)/(2m)}$. Note that $\dfrac{1}{m}\sum_{l=1}^m x_{l,1} + \mathbb{E}[\dfrac{1}{m}\sum_{l=1}^m x_{l,1}] \leq 2b$. Therefore,

$$\mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,1})^2 - (\mathbb{E}[\frac{1}{m}\sum_{l=1}^m x_{l,1}])^2| \geq \epsilon_2\right)$$

$$\leq \mathbb{P}\left(|\frac{1}{m}\sum_{l=1}^m x_{l,1} - \mathbb{E}[\frac{1}{m}\sum_{l=1}^m x_{l,1}]| \geq \frac{\epsilon_2}{2b}\right)$$

$$\leq \delta_2,$$

Algorithm 1 Adaptive Double Exploration (ADE)

1: **Input:** n: the number of arms; k: threshold parameter
2: **Output:** the set of outliers
3: // Initialization
4: $S_0 \leftarrow [n], m_{\theta,1} \leftarrow 0, m_{a,1} \leftarrow 0, t \leftarrow 1$
5: Sample an arm uniformly at random from the set S_0, and pull this arm twice independently
6: $m_{\theta,t+1} \leftarrow m_{\theta,t} + 1, m_{a,t+1} \leftarrow m_{a,t}, t \leftarrow t + 1$
7: Pull each arm $i \in S_0$ once
8: $m_{\theta,t+1} \leftarrow m_{\theta,t}, m_{a,t+1} \leftarrow m_{a,t} + 1, t \leftarrow t + 1$
9: $S_t \leftarrow [n]$
10: Update $\hat{\theta}_t, r_{a,t}, r_{\theta,t}$ and $\hat{y}_{i,t}$ for $i \in S_t$
11: **while** $S_t \neq \emptyset$ **do**
12:     // Sample for either the threshold or the arm whichever has larger radius
13:     **if** $r_{a,t} \leq r_{\theta,t}$ **then**
14:         Sample an arm uniformly at random from the set S_0, and pull this arm twice independently
15:         $m_{\theta,t+1} \leftarrow m_{\theta,t} + 1, m_{a,t+1} \leftarrow m_{a,t}$
16:     **else**
17:         Sample each arm $i \in S_t$ and pull this arm once
18:         $m_{\theta,t+1} \leftarrow m_{\theta,t}, m_{a,t+1} \leftarrow m_{a,t} + 1$
19:     **end if**
20:     $t \leftarrow t + 1$
21:     Update $\hat{\theta}_t, r_{a,t}, r_{\theta,t}$ and $\hat{y}_{i,t}$ for $i \in S_{t-1}$
22:     // Distinguish the outliers from the normal arms
23:     **for** $i \in S_{t-1}$ **do**
24:         **if** $\hat{y}_{i,t} + r_{i,t} \leq \hat{\theta}_t - r_{\theta,t}$ **then**
25:             $N \leftarrow N \cup \{i\}$
26:             $S_t \leftarrow S_{t-1} \setminus \{i\}$
27:         **else if** $\hat{y}_{i,t} - r_{i,t} \geq \hat{\theta}_t + r_{\theta,t}$ **then**
28:             $O \leftarrow O \cup \{i\}$
29:             $S_t \leftarrow S_{t-1} \setminus \{i\}$
30:         **end if**
31:     **end for**
32: **end while**

where $\epsilon_2 = 2bR\sqrt{\log(2/\delta_2)/(2m)}$.

Similarly, we have $\mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,2})^2 - \mu_y^2| \geq \frac{\epsilon_2}{2b}\right) \leq \delta_2$.

Thus,

$$\mathbb{P}(|\sigma_y^2 - V_t| \geq \epsilon)$$
$$\leq \mathbb{P}\left(|\frac{1}{m}\sum_{l=1}^m (\mathbb{E}[x_{l,1}x_{l,2}] - x_{l,1}x_{l,2})| \geq \epsilon_1\right)$$
$$+ \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,1})^2 - \mu_y^2| \geq \epsilon_2\right)$$
$$+ \mathbb{P}\left(|(\frac{1}{m}\sum_{l=1}^m x_{l,2})^2 - \mu_y^2| \geq \epsilon_2\right)$$
$$\leq \delta_1 + 2\delta_2.$$

If we chose $\delta_1 = \delta_2 = \delta_t/3$, then $\epsilon_1 = R'\sqrt{\dfrac{\log(6/\delta_t)}{2m}}$

and $\epsilon_2 = 2bR\sqrt{\dfrac{\log(6/\delta_t)}{2m}}$. Thus, $|\sigma_y^2 - V_t| \leq \epsilon_{\sigma,t}$ with probability at least $1 - \delta_t$, where $\epsilon_{\sigma,t} = \epsilon_1 + \epsilon_2 = (R' + 2bR)\sqrt{\dfrac{\log(6/\delta_t)}{2m}}$.

Note that $\hat{\sigma}_{y,t} = \sqrt{|V_t|}$, then with probability at least $1 - \delta_t$,

$$|\sigma_y^2 - \hat{\sigma}_{y,t}^2| \leq |\sigma_y^2 - V_t| \leq \epsilon_{\sigma,t}.$$

$\square$

Lemma 2 only informs us the gap between σ_y^2 and $\hat{\sigma}_{y,t}^2$, while what we are really interested is the gap between σ_y and its estimator $\hat{\sigma}_{y,t}$. The confidence radius of σ_y is constructed adaptively using $\hat{\sigma}_{y,t}$, which is formally stated in the following lemma.

Lemma 3. *With probability at least $1 - \delta_t$, we have $|\hat{\sigma}_{y,t} - \sigma_y| \leq \sqrt{\dfrac{2}{U_{\sigma,t}}}\epsilon_{\sigma,t}$, where $U_{\sigma,t} = \min_{1 \leq \tau \leq t}\{\hat{\sigma}_{y,\tau}^2 + \epsilon_{\sigma,\tau}\}$, and $\epsilon_{\sigma,t} = (R' + 2bR)\sqrt{\dfrac{\log(6/\delta_t)}{2m}}$.*

Now we could derive the confidence bound of θ with the confidence bound of μ_y and σ_y.

Lemma 4. *Conditioned on $m_{\theta,t} = m$, we have $|\hat{\theta}_t - \theta| \leq r_{\theta,t}$ with probability at least $1 - 8\delta_t$, where $r_{\theta,t} = \left(R + \sqrt{2}k\dfrac{(R' + 2bR)}{\sqrt{U_{\sigma,t}}}\right)\sqrt{\dfrac{\log(1/\delta_t)}{2m}}, U_{\sigma,t} = \min_{1 \leq \tau \leq t}\{\hat{\sigma}_{y,\tau}^2 + \epsilon_{\sigma,\tau}\}, and \epsilon_{\sigma,t} = (R' + 2bR)\sqrt{\dfrac{\log(1/\delta_t)}{2m}}$.*

Remark: The estimator for the threshold constructed by Zhuang, Wang, and Wang (2017) is $\tilde{\theta}_t = \tilde{\mu}_{y,t} + k\sqrt{\dfrac{1}{n}\sum_{i=1}^n(\tilde{y}_{i,t} - \tilde{\mu}_{y,t})^2}$, where $\tilde{y}_{i,t} = \dfrac{1}{m_{i,t}}\sum_{j=1}^{m_{i,t}}x_i^{(j)}$, and $\tilde{\mu}_{y,t} = \dfrac{1}{n}\sum_{i=1}^n\tilde{y}_{i,t}$. They state that with probability at least $1 - 2\tilde{\delta}_t$, $|\tilde{\theta}_t - \theta| \leq R\sqrt{\dfrac{l(k)}{2h(m)}\log\left(\dfrac{1}{\tilde{\delta}_t}\right)}$, where $l(k) = \left[\sqrt{\dfrac{(1 + k\sqrt{n-1})^2}{n}} + \sqrt{\dfrac{k^2}{2\log(\pi^2 n^3/(6\tilde{\delta}_t))}}\right]^2$, $\tilde{\delta}_t = \dfrac{6\delta}{\pi^2(n+1)t^2}$, and $h(m)$ is the harmonic mean of $m_{i,t}$ over all the arms. The confidence radius of $\tilde{\theta}$ increases with n, which might make the learning strategy inefficient especially when identifying outliers for large-scale dataset. Our approach alleviates this problem by constructing a confidence radius for θ which is independent of the number of arms (if we ignore logarithmic factors).

Lemma 5. *Conditioned on $m_{i,t} = m$, we have $|y_i - \hat{y}_{i,t}| \leq r_{i,t}$ with probability at least $1 - 2\delta_t$, where $r_{i,t} = R\sqrt{\log(1/\delta_t)/(2m)}$.*

Applying the union bound over all arms and the possible number of iterations, we conclude that the confidence intervals hold for the threshold and any arm i at any round t with probability at least $1 - \delta$, which is formally illustrated in the following lemma.

Lemma 6. *Let $\mathcal{A} = \{|y_i - \hat{y}_{i,t}| \leq r_{i,t}, |\theta - \hat{\theta}_t| \leq r_{\theta,t}, \forall i, \forall t\}$ be the event that the expected rewards of the arms and the threshold lie in the confidence intervals around $\hat{y}_{i,t}$ and $\hat{\theta}_t$ for any arm i and any round t. Then event $\mathcal{A}$ occurs with probability at least $1 - \delta$.*

Therefore, with probability at least $1 - \delta$, for any round t and any arm $i \in [n]$, $L_{i,t} = \hat{y}_{i,t} - r_{i,t}$ and $L_{\theta,t} = \hat{\theta}_t - r_{\theta,t}$ could be regarded as the lower confidence bound of the expected reward y_i and the threshold θ separately. Similarly, $U_{i,t} = \hat{y}_{i,t} + r_{i,t}$ and $U_{\theta,t} = \hat{\theta}_t + r_{\theta,t}$ are respectively the upper confidence bound of y_i and θ.

Theoretical Results

If $\mathcal{A}$ is satisfied, then at any round t, the arms contained in N are normal arms, and the arms contained in O are outliers. The algorithm terminates when all the arms have been assigned to either the normal arm set N or the outlier set O. Therefore, the arm set returned by Algorithm 1 is the correct outlier set with probability at least $1 - \delta$.

Theorem 1 (Correctness). *For any $\delta > 0$, the algorithm returns the correct outlier set O^* with probability at least $1 - \delta$.*

For each arm $i \in [n]$, we define the gap between its expected reward and the threshold as $\Delta_i = |y_i - \theta|$, and denote by $\Delta_{\min} = \min_{i \in [n]} \Delta_i$ the minimum gap among all the arms. The following theorem illustrates a problem-dependent sample complexity bound of Algorithm 1.

Theorem 2 (Sample Complexity). *With probability at least $1 - \delta$, the total number of samples of Algorithm 1 could be bounded by*

$$O\Bigg(\sum_{i=1}^{n} \frac{1}{\Delta_i^2} \log \Big(\sqrt{\frac{n}{\delta}} \frac{1}{\Delta_i^2} \max\{1, (\frac{k}{\sigma_y})^2\} \Big)$$

$$+ \max\{1, (\frac{k}{\sigma_y})^2\} \frac{1}{\Delta_{\min}^2} \log \Big(\sqrt{\frac{n}{\delta}} \frac{1}{\Delta_{\min}^2} \max\{1, (\frac{k}{\sigma_y})^2\} \Big) \Bigg).$$

Comparison with RR and WRR: ADE attains the same correctness guarantee as the state-of-the-art algorithms RR and WRR, and at the same time it achieves considerable improvement in terms of the sample complexity. Sepcifically, the sample complexity of RR is bounded by $O(\mathbf{H}_{RR} \log(n\mathbf{H}_{RR}/\delta))$, and that of WRR is bounded by $O(\mathbf{H}_{WRR} \log(n\mathbf{H}_{WRR}/\delta))$, where $\mathbf{H}_{RR} = \mathbf{H}_1(1 + \sqrt{l(k)})^2$, $\mathbf{H}_{WRR} = (\mathbf{H}_1/\rho + \mathbf{H}_2(\rho - 1)/\rho)(1 + \sqrt{l(k)\rho})^2$, $l(k) = \left(\sqrt{(1 + k\sqrt{n-1})^2/n} + \sqrt{k^2/(2\log(\pi^2 n^3/(6\delta)))} \right)^2$, $\mathbf{H}_1 = n/\Delta_{\min}^2$, $\mathbf{H}_2 = \sum_{i=1}^{n} 1/\Delta_i^2$, and ρ is a constant. Compared with RR and WRR, ADE reduces the $n/\Delta_{\min}^2$ factor in the upper bound of the sample complexity to $1/\Delta_{\min}^2$.

Experiments

In this section, we conduct experiments on both synthetic datasets and real datasets to show the performance of distinct algorithms. The confidence parameter δ is set as 0.1 and the results are averaged across ten independent simulations. We compare the performance of our proposed ADE algorithm with the state-of-the-art algorithms RR and WRR proposed by Zhuang, Wang, and Wang (2017).

- **Round-Robin (RR)**: RR samples the arms in a round-robin way. Specifically, each arm is sampled in turn and in the following circular order, $1, 2, \ldots, n, 1, 2, \ldots$. Thereby, each arm is sampled with identical frequency.

- **Weighted Round-Robin (WRR)**: WRR samples the arms in a weighted round-robin way. The arms that are not yet determined are sampled more frequently than the arms that have been determined.

Since RR and WRR might take a long period of time to terminate, we speed up these two algorithms by pulling an arm $1,000$ times at each round instead of pulling an arm once at each round. The termination condition of all these algorithms are identical. Specifically, each algorithm terminates when there exists no overlapping between the confidence interval of the threshold and that of any arm.

Experiments on the Synthetic Datasets

We construct the synthetic datasets with distinct settings of n and $\Delta_{\min}$. For each setting, we generate 10 test cases independently. The expected reward of each arm is generated uniformly at random in $[0, 1]$. The sampled reward corresponding to each pull is generated independently from the Bernoulli distribution. Accordingly, R is set as 1.

We compare the average number of samples used by distinct algorithms with regard to distinct number of arms. Specifically, we vary n as $\{100, 200, 400, 600, 800, 1000\}$, k is set as 2.5, the range of $\Delta_{\min}$ is $[0.1, 0.2]$. We illustrate the experimental results in Figure 2(a), based on which we have the following observations. (1) When n increases, the average number of samples used by each algorithm also tends to increase, which is consistent with the theoretical results. (2) When n is as small as 100, the number of samples used by RR and WRR are smaller than ADE, and WRR performs slightly better than RR. The constant embedded in our algorithm counteracts its advantage when the hardness of the identification problem is small. (3) For all the remaining settings of n, ADE has the best performance while WRR performs the worst, and the improvement of ADE over RR and WRR tends to be more significant with increased number of arms. Figure 2(b) shows the performance of distinct algorithms on synthetic datasets with various settings of $\Delta_{\min}$. We vary the range of $\Delta_{\min}$ as $\{[0.059, 0.061], [0.089, 0.091], [0.119, 0.121]\}$, n is fixed as 900, k is set as 2. The average number of samples required by each algorithm tends to decrease when $\Delta_{\min}$ becomes larger, which is a key component that captures the hardness of the problem. ADE performs better than both RR and WRR in these datasets.
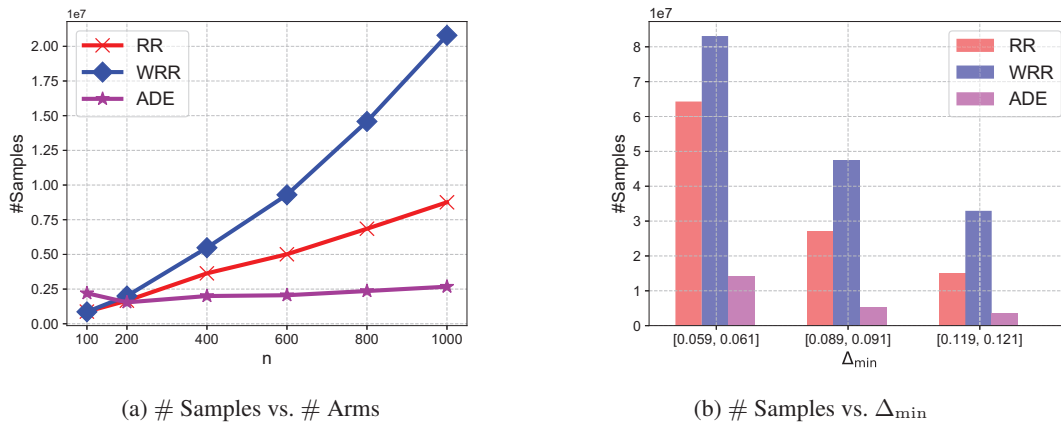
(a) # Samples vs. # Arms

(b) # Samples vs. $\Delta_{\min}$

Figure 2: The performance of distinct algorithms on synthetic datasets

(a) The number of samples, $k = 2$

(b) The number of samples, $k = 3$
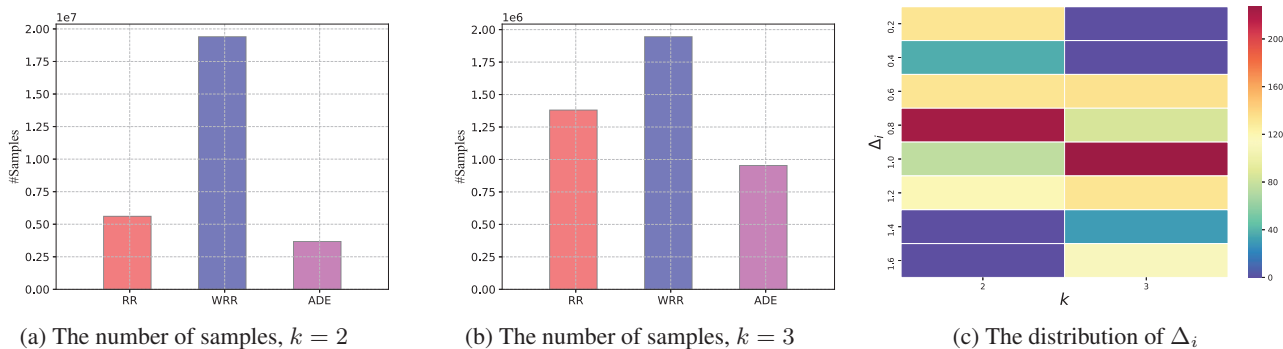

(c) The distribution of Δ_i

Figure 3: The performance of distinct algorithms on HyunCatherines dataset

Experiments on the Real Datasets

We characterize the performance of distinct algorithms on the real dataset HyunCatherines which is available at http://ir.ischool.utexas.edu/square/data.html. 722 workers are extracted from the HyunCatherines dataset. The workers with extremely high error rate are regarded as outliers, and should be excluded from the crowd in order to obtain labels with high quality. Each worker could be regarded as an arm, and the error rate of this worker is the expected reward corresponding to this worker. Figure 3(a) and 3(b) illustrate the average number of samples each algorithm uses when k is set as 2 and 3 respectively. It could be viewed that ADE achieves considerable improvement over the state-of-the-art algorithms. Note that when k is increased from 2 to 3, the number of samples required for these three algorithms all decreases. Although k exists in the upper bound of the sample complexity of all these algorithms, the hardness of the problem does not necessarily increase with k. It is worth noting that the distribution of Δ_i may vary as k changes. Figure 3(c) illustrates the distribution of Δ_i with distinct settings of k. For both datasets, the minimum gap between the arms and the threshold as well as the gaps of the bulk of arms tend to be larger when k is increased from 2 to 3. This comparison explains the smaller usage of samples when k is larger.

Conclusion

We studied the problem of outlier identification in the framework of multi-armed bandit. The learner is asked to identify the arms with rewards above a threshold, which is a function of the rewards of all the arms. We proposed two distinct sampling strategies to address this *double-exploration* dilemma, and constructed an adaptively adjusted confidence radius for the threshold which is independent of the number of arms. We put forward an algorithm that automatically balances between distinct sampling strategies. Theoretical analyses and experimental results illustrate the efficiency of our algorithm. An interesting direction for future work is to address this problem in the fixed budget setting. Another interesting question is whether it is possible to construct a better estimator for the threshold with tighter confidence interval. Besides, it remains open to derive a lower bound for this problem.

Acknowledgements

References

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47(2-3):235–256.

Bubeck, S.; Munos, R.; and Stoltz, G. 2011. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science* 412(19):1832–1852.

Chandola, V.; Banerjee, A.; and Kumar, V. 2009. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 41(3):15.

Chen, S.; Lin, T.; King, I.; Lyu, M. R.; and Chen, W. 2014. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, 379–387.

Coolidge, F. L. 2012. *Statistics: A gentle introduction*. Sage Publications.

De Stefano, C.; Sansone, C.; and Vento, M. 2000. To reject or not to reject: that is the question-an answer in case of neural classifiers. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 30(1):84–94.

Domingo, C.; Gavaldà, R.; and Watanabe, O. 2002. Adaptive sampling methods for scaling up knowledge discovery algorithms. *Data Mining and Knowledge Discovery* 6(2):131–152.

Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.; et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, 226–231.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2002. Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, 255–270. Springer.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2006. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research* 7(Jun):1079–1105.

Gabillon, V.; Ghavamzadeh, M.; Lazaric, A.; and Bubeck, S. 2011. Multi-bandit best arm identification. In *Advances in Neural Information Processing Systems*, 2222–2230.

Gabillon, V.; Ghavamzadeh, M.; and Lazaric, A. 2012. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, 3212–3220.

Guttormsson, S. E.; Marks, R.; El-Sharkawi, M.; and Kerszenbaum, I. 1999. Elliptical novelty grouping for on-line short-turn detection of excited running rotors. *IEEE Transactions on Energy Conversion* 14(1):16–22.

Horn, P. S.; Feng, L.; Li, Y.; and Pesce, A. J. 2001. Effect of outliers and nonhealthy individuals on reference interval estimation. *Clinical Chemistry* 47(12):2137–2145.

Jamieson, K.; Malloy, M.; Nowak, R.; and Bubeck, S. 2014. lil?ucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, 423–439.

Kalyanakrishnan, S., and Stone, P. 2010. Efficient selection of multiple bandit arms: Theory and practice. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 511–518.

Kalyanakrishnan, S.; Tewari, A.; Auer, P.; and Stone, P. 2012. Pac subset selection in stochastic multi-armed bandits. In *ICML*, volume 12, 655–662.

Kaufmann, E., and Kalyanakrishnan, S. 2013. Information complexity in bandit subset selection. In *Conference on Learning Theory*, 228–251.

Kaufmann, E.; Cappé, O.; and Garivier, A. 2016. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research* 17(1):1–42.

Kou, Y.; Lu, C.-T.; and Chen, D. 2006. Spatial weighted outlier detection. In *Proceedings of the 2006 SIAM international conference on data mining*, 614–618. SIAM.

Locatelli, A.; Gutzeit, M.; and Carpentier, A. 2016. An optimal algorithm for the thresholding bandit problem. *arXiv preprint arXiv:1605.08671*.

Maron, O., and Moore, A. W. 1994. Hoeffding races: Accelerating model selection search for classification and function approximation. In *Advances in neural information processing systems*, 59–66.

Mnih, V.; Szepesvári, C.; and Audibert, J.-Y. 2008. Empirical bernstein stopping. In *Proceedings of the 25th international conference on Machine learning*, 672–679. ACM.

Paulson, E. 1964. A sequential procedure for selecting the population with the largest mean from k normal populations. *The Annals of Mathematical Statistics* 174–180.

Robbins, H. 1985. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*. Springer. 169–177.

Roth, V. 2006. Kernel fisher discriminants for outlier detection. *Neural computation* 18(4):942–960.

Thompson, W. R. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25(3/4):285–294.

Torr, P. H., and Murray, D. W. 1993. Outlier detection and motion segmentation. In *Sensor Fusion VI*, volume 2059, 432–444. International Society for Optics and Photonics.

Yu, D.; Sheikholeslami, G.; and Zhang, A. 2002. Findout: finding outliers in very large datasets. *Knowledge and Information Systems* 4(4):387–412.

Zhou, Y.; Chen, X.; and Li, J. 2014. Optimal pac multiple arm identification with applications to crowdsourcing. In *International Conference on Machine Learning*, 217–225.

Zhuang, H.; Wang, C.; and Wang, Y. 2017. Identifying outlier arms in multi-armed bandit. In *Advances in Neural Information Processing Systems*, 5210–5219.