Thresholding Bandits with Augmented UCB

Author names withheld

Abstract

We propose the Augmented-UCB (AugUCB) algorithm for the thresholding bandit problem, which is an instance of the combinatorial fixed-budget pure-exploration stochastic multi-armed bandit setup. Our algorithm is based on arm elimination, employing both the mean and variance estimates. Theoretically, our algorithm provides a weaker guarantee (in terms of an upper bound on the expected loss) than APT [Locatelli *et al.*, 2016] and CSAR [Chen *et al.*, 2014]. However, through simulation experiments we establish that our algorithm, being variance-aware, performs better than APT and CSAR algorithms, particularly when a large number of arms are involved.

1 Introduction

In this paper we study a specific combinatorial pure-exploration problem, called the thresholding bandit problem (TBP), in the context of stochastic multi-armed bandit (MAB) setting. MAB problems are instances of the classic sequential decision-making scenario; specifically, a MAB problem comprises a learner and a collection of actions (or arms), denoted $\mathcal{A}$; subsequent plays (or pulls) of an arm $i \in \mathcal{A}$ yields independent and identically distributed (i.i.d.) reward samples from a distribution (corresponding to arm i), whose expectation is denoted by r_i. The learner's objective is to identify an arm corresponding to the maximum expected reward, denoted r^*. Thus, at each time-step the learner is faced with the *exploration vs. exploitation dilemma*, whereby it can pull an arm which has yielded the highest mean reward (denoted $\hat{r}_i$) thus far (*exploitation*) or continue to explore other arms with the prospect of finding a better arm whose performance is yet not observed sufficiently (*exploration*).

Pure exploration problems are unlike their traditional (exploration vs. exploitation) counterparts where the objective is to minimize the cumulative regret, which is the total loss incurred by the learner for not playing the optimal arm throughout the time horizon T. Instead, in the pure exploration setup the learning algorithm is provided with a threshold τ, and the objective, after exploring for T rounds, is to output all arms i whose r_i is above τ. Thus, the learning algorithm, until time T, can invest entirely on exploring the arms without being concerned about the loss incurred while exploring.

Formally, the problem we consider is the following. First, we define the set $S_\tau = \{i \in \mathcal{A} : r_i \geq \tau\}$. Note that, S_τ is the set of all arms whose reward mean is greater than τ. Let S_τ^c denote the complement of S_τ, i.e., $S_\tau^c = \{i \in \mathcal{A} : r_i < \tau\}$. Next, let $\hat{S}_\tau = \hat{S}_\tau(T) \subseteq \mathcal{A}$ denote the recommendation of the learning algorithm after T time units of exploration, while $\hat{S}_\tau^c$ denotes its complement. The performance of the learning agent is measured by the accuracy with which it can classify the arms into S_τ and S_τ^c after time horizon T. Equivalently, using $\mathbb{I}(E)$ to denote the indicator of an event E, the *loss* $\mathcal{L}(T)$ is defined as

$$\mathcal{L}(T) = \mathbb{I}\big(\{S_\tau \cap \hat{S}_\tau^c \neq \emptyset\} \cup \{\hat{S}_\tau \cap S_\tau^c \neq \emptyset\}\big).$$

Finally, the goal of the learning agent is to minimize the expected loss:

$$\mathbb{E}[\mathcal{L}(T)] = \mathbb{P}\big(\{S_\tau \cap \hat{S}_\tau^c \neq \emptyset\} \cup \{\hat{S}_\tau \cap S_\tau^c \neq \emptyset\}\big).$$

Note that the expected loss is simply the *probability of error*, that occurs either if a good arm is rejected or a bad arm is accepted as a good one.

The above TBP formulation has several applications, for instance, from areas ranging from anomaly detection and classification [Locatelli *et al.*, 2016] to industrial application. Particularly in industrial applications a learners objective is to choose (i.e., keep in operation) all machines whose productivity is above a threshold. Similarly, TBP finds applications in mobile communications [Audibert and Bubeck, 2010] where the users are to be allocated only those channels whose quality is above an acceptable threshold.

1.1 Related Work

Significant amount of literature is available on the stochastic MAB setting with respect to minimizing the cumulative regret. While the seminal work of [Robbins, 1952], [Thompson, 1933], and [Lai and Robbins, 1985] prove asymptotic lower bounds on the cumulative regret, the more recent work of [Auer *et al.*, 2002] propose the UCB1 algorithm that provides finite time-horizon guarantees. Subsequent work such as [Audibert and Bubeck, 2009] and [Auer and Ortner, 2010] have improved the upper bounds on the cumulative regret. The authors in [Auer and Ortner, 2010] have proposed a *round-based*[1] version of the UCB algorithm, referred to as UCB-Improved. Of special mention is the work of [Audibert *et al.*, 2009] where the authors have introduced a *variance-aware* UCB algorithm, referred to as UCB-V; it is shown that the algorithms that take into account variance estimation along with mean estimation tends to perform better than the algorithms that solely focuses on mean estimation, for instance, such as UCB1. For a more detail survey of literature on UCB algorithms, we refer the reader to [Bubeck and Cesa-Bianchi, 2012].

In this work we are particularly interested in *pure-exploration MABs*, where the focus in primarily on simple regret rather than the cumulative regret. The relationship between cumulative regret and simple regret is proved in [Bubeck *et al.*, 2011] where the authors prove that minimizing the simple regret necessarily results in maximizing the cumulative regret. The pure exploration problem has been explored mainly under the following two settings:

1. Fixed Budget setting: Here the learning algorithm has to suggest the best arm(s) within a fixed time-horizon T, that is usually given as an input. The objective is to maximize the probability of returning the best arm(s). This is the scenario we consider in our paper. In [Audibert and Bubeck, 2010] the authors propose the UCBE and the Successive Reject (SR) algorithm, and prove simple-regret guarantees for the problem of identifying the single best arm. In the combinatorial fixed budget setup [Gabillon *et al.*, 2011] propose the Gap-E and Gap-EV algorithms that suggests, with high probability, the best m arms at the end of the time budget. Similarly, [Bubeck *et al.*, 2013] introduce the Successive Accept Reject (SAR) algorithm, which is an extension of the SR algorithm; SAR is a round based algorithm whereby at the end of each round an arm is either accepted or rejected (based on certain confidence conditions) until the top m arms are suggested at the end of the budget with high probability. A similar combinatorial setup was explored in [Chen *et al.*, 2014] where the authors propose the Combinatorial Successive Accept Reject (CSAR) algorithm, which is similar in concept to SAR but with a more general setup.

2. Fixed Confidence setting: In this setting the learning algorithm has to suggest the best arm(s) with a fixed confidence (given as input) with as fewer number of attempts as possible. The single best arm identification has been studied in [Even-Dar *et al.*, 2006], while for the combinatorial setup [Kalyanakrishnan *et al.*, 2012] have proposed the LUCB algorithm which, on termination, returns m arms which are at least ϵ close to the true top-m arms with probability at least $1 - \delta$. For a detail survey of this setup we refer the reader to [Jamieson and Nowak, 2014].

Apart from these two settings some unified approach has also been suggested in [Gabillon *et al.*, 2012] which proposes the algorithms UGapEb and UGapEc which can work in both the above two settings. The thresholding bandit problem is a specific instance of the pure exploration setup of [Chen *et al.*, 2014]. In the latest work of [Locatelli *et al.*, 2016] Anytime Parameter-Free Thresholding (APT) algorithm comes up with an improved anytime guarantee than CSAR for the thresholding bandit problem.

[1] An algorithm is said to be *round-based* if it pulls all the arms equal number of times in each round, and then proceeds to eliminate one or more arms that it identifies to be sub-optimal.

1.2 Our Contribution

In this paper we propose the Algorithm AugUCB which is an action elimination algorithm suited for the TBP problem. It combines the approach of UCB-Improved, CCB ([Liu and Tsuruoka, 2016]) and APT algorithm. Our algorithm is a variance-aware algorithm which takes into account the empirical variance of the arms. We also address an open problem raised in [Auer and Ortner, 2010] of coming up with an algorithm that can eliminate arms based on variance. Both CSAR and APT are not variance-aware algorithms. The expected loss of various algorithms is shown in Table 1. The terms $H_1, H_2, H_{CSAR,2}, H_1^\sigma$ and H_2^σ signifies problem complexity and are defined in section 3. Theoretically, we can compare the first term (containing H_2) of our expected loss and see that for all $K \geq 4$, $H_2 \log(\frac{3}{16}K \log K) > (\log K)H_{CSAR,2} \geq H_1$ and hence our result is weaker than CSAR and APT. The term containing H_2^σ is comparable to the similar terms (containing H_1^σ) for the error probability of Gap-EV([Gabillon *et al.*, 2011] or UGapE-V([Gabillon *et al.*, 2012]) algorithm which we modify to perform in the TBP problem. The error probability of Gap-EV for single bandit multi-armed case is given by $6TK \exp(-\frac{1}{512}\frac{T-2K}{H_1^\sigma})$ where $\log(\frac{3}{16}K \log K)H_2^\sigma > H_1^\sigma$ and hence our algorithm is weaker with respect to Gap-EV for single multi-armed bandit scenario. But Gap-EV algorithm needs the complexity factor H_1^σ as input for optimal performance (which is not a realistic scenario) whereas AugUCB requires no such complexity factor as input.

Table 1: Expected Loss for different bandit algorithms

Algorithm	Upper Bound on Expected Loss
APT	$\exp(-\frac{T}{64H_1} + 2\log((\log(T)+1)K))$
CSAR	$K^2 \exp(-\frac{T-K}{72\log(K)H_{CSAR,2}})$
AugUCB	$\exp\left(-\frac{T}{64H_2 a} + \log\left(K\left(\log_2\frac{T}{e}+1\right)\right)\right) +$ $\exp\left(-\frac{T}{4096H_2^\sigma a} + \log\left(2K\left(\log_2\frac{T}{e}+1\right)\right)\right)$ where $a = \log(\frac{3}{16}K\log K)$

Empirically we show that for a large number of arms when the variance of the arms lying above τ are high, our algorithm performs better than all other algorithms, except the algorithm UCBEV (modified from Gap-EV for TBP) which has access to the underlying problem complexity and also is a variance-aware algorithm. Irrespective of this case AugUCB also employs elimination of arms based on mean estimation only and is the first such algorithm which uses elimination by both mean and variance estimation simultaneously. AugUCB requires three input parameters and the exact choices for these parameters are derived in Theorem 3.1. Also, unlike SAR or CSAR, AugUCB does not have explicit accept or reject set rather the arm elimination conditions simply removes arm(s) if it is sufficiently sure that the mean of the arms are very high or very low about the threshold based on mean and variance estimation thereby re-allocating the remaining budget among the surviving arms. This although is a tactic similar to SAR or CSAR, but here at any round, an arbitrary number of arms can be accepted or rejected thereby improving upon SAR and CSAR which accepts/rejects one arm in every round. Also their round lengths are non-adaptive and they pull all the arms equal number of times in each round.

The remainder of the paper is organized as follows. In section 2 we present our AugUCB algorithm. Section 3 contains our main theorem on expected loss, while section 4 contains simulation experiments. We finally draw our conclusions in section 5.

2 Augmented-UCB Algorithm

Notations and assumptions: $\mathcal{A}$ denotes the set of arms, and $|\mathcal{A}| = K$ is the number of arms in $\mathcal{A}$. For arm $i \in \mathcal{A}$, we use r_i to denote the true mean of the distribution from which the rewards are sampled, while $\hat{r}_i(t)$ denotes the estimated mean at time t. Formally, using $n_i(t)$ to denote the number of times arm i has been pulled until time t, we have $\hat{r}_i(t) = \frac{1}{n_i(t)} \sum_{z=1}^{n_i(t)} X_{i,z}$, where $X_{i,z}$ is the reward sample received when arm i is pulled for the z-th time. Similarly, we use σ_i^2 to denote the true variance of the reward distribution corresponding to arm i, while $\hat{v}_i(t)$ is the estimated variance, i.e., $\hat{v}_i(t) = \frac{1}{n_i(t)} \sum_{z=1}^{n_i(t)} (X_{i,z} - \hat{r}_i)^2$. Whenever there is no ambiguity about the underlaying time index t, for simplicity we neglect t from the notations and simply use $\hat{r}_i, \hat{v}_i$, and n_i, to denote the respective quantities. Let $\Delta_i = |\tau - r_i|$ denote the distance of the true mean from the threshold τ.

Finally, we assume that all the reward distributions are 1-sub-Gaussian (note that, 1-sub-Gaussian includes Gaussian distributions with variance less than 1, distributions supported on an interval of length less than 2, etc). Further, the rewards are assumed to take values in the interval $[0, 1]$.

The Algorithm: The Augmented-UCB (AugUCB) algorithm is presented in Algorithm 1. AugUCB is essentially based on the arm elimination method of the UCB-Improved [Auer and Ortner, 2010], but adapted to the thresholding bandit setting proposed in [Locatelli *et al.*, 2016]. However, unlike the UCB improved (which is based on mean estimation) our algorithm employs *variance estimates* (as in [Audibert *et al.*, 2009]) for arm elimination; to the best of our knowledge this is the first variance-aware algorithm for the thresholding bandit problem. Further, we allow for arm-elimination at each time-step, which is in contrast to the earlier work (e.g., [Auer and Ortner, 2010; Chen *et al.*, 2014]) where the arm elimination task is deferred to the end of the respective exploration rounds. The finer details of the algorithm are presented below.

The active set B_0 is initialized with all the arms from $\mathcal{A}$. We divide the entire budget T into rounds/phases like in UCB-Improved, CCB, SAR and CSAR. At every time-step AugUCB checks for arm elimination conditions, while updating parameters at the end of each round. As suggested by [Liu and Tsuruoka, 2016] to make AugUCB to overcome too much early exploration, we no longer pull all the arms equal number of times in each round. Instead, we choose an arm in the active set B_m that minimizes $(|\hat{r}_i - \tau| - 2s_i)$ where

$$s_i = \sqrt{\frac{\rho\psi_m(\hat{v}_i + 1)\log(T\epsilon_m)}{4n_i}}$$

with ρ being the arm elimination parameter and ψ_m being the exploration regulatory factor. The above condition ensures that an arm closer to the threshold τ is pulled; parameter ρ can be used to fine tune the elimination interval. The choice of exploration factor, ψ_m, comes directly from [Audibert and Bubeck, 2010] and [Bubeck *et al.*, 2011] where it is stated that in pure exploration setup, the exploring factor must be linear in T (so that an exponentially small probability of error is achieved) rather than being logarithmic in T (which is more suited for minimizing cumulative regret).

Algorithm 1 AugUCB

Input: Time budget T; parameter ρ; threshold τ
Initialization: $B_0 = \mathcal{A}$; $m = 0$; $\epsilon_0 = 1$;

$$M = \left\lfloor \frac{1}{2}\log_2 \frac{T}{e} \right\rfloor; \quad \psi_0 = \frac{T\epsilon_0}{\left(\log(\frac{3}{16}K\log K)\right)^2};$$

$$\ell_0 = \left\lceil \frac{2\psi_0 \log(T\epsilon_0)}{\epsilon_0} \right\rceil; \quad N_0 = K\ell_0$$

for $t = K+1, .., T$ **do**
  Pull arm $j \in \arg\min_{i\in B_m} \left\{|\hat{r}_i - \tau| - 2s_i\right\}$
  $t \leftarrow t+1$
  for $i \in B_m$ **do**
    if $(\hat{r}_i + s_i < \tau - s_i)$ or $(\hat{r}_i - s_i > \tau + s_i)$ **then**
      $B_m \leftarrow B_m \backslash \{i\}$ (Arm deletion)
    end if
  end for
  if $t \geq N_m$ and $m \leq M$ **then**
    Reset Parameters
    $\epsilon_{m+1} \leftarrow \frac{\epsilon_m}{2}$
    $B_{m+1} \leftarrow B_m$
    $\psi_{m+1} \leftarrow \frac{T\epsilon_{m+1}}{(\log(\frac{3}{16}K\log K))^2}$
    $\ell_{m+1} \leftarrow \left\lceil \frac{2\psi_{m+1}\log(T\epsilon_{m+1})}{\epsilon_{m+1}} \right\rceil$
    $N_{m+1} \leftarrow t + |B_{m+1}|\ell_{m+1}$
    $m \leftarrow m+1$
  end if
end for
Output: $\hat{S}_\tau = \{i : \hat{r}_i \geq \tau\}$.

3 Theoretical Results

Let us begin by recalling the following definitions of the *problem complexity* as introduced in [Locatelli *et al.*, 2016]:

$$H_1 = \sum_{i=1}^{K} \frac{1}{\Delta_i^2} \quad \text{and} \quad H_2 = \min_{i\in\mathcal{A}} \frac{i}{\Delta_{(i)}^2}$$

where $(\Delta_{(i)} : i \in \mathcal{A})$ is obtained by arranging $(\Delta_i : i \in \mathcal{A})$ in an increasing order. Also, from [Chen *et al.*, 2014] we have

$$H_{CSAR,2} = \max_{i \in \mathcal{A}} \frac{i}{\Delta^2_{(i)}}.$$

$H_{CSAR,2}$ is the complexity term appearing in the bound for the CSAR algorithm. The relation between the above complexity terms are as follows (see [Locatelli *et al.*, 2016]):

$$H_2 \leq H_1 \leq \log(2K)H_2 \text{ and } H_1 \leq \log(K)H_{CSAR,2}$$

As ours is a variance-aware algorithm, we require H_1^σ (as defined in [Gabillon *et al.*, 2011]) that incorporates reward variances into its expression as given below:

$$H_1^\sigma = \sum_{i=1}^{K} \frac{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}{\Delta_i^2}.$$

Finally, analogous to H_2, in this paper we introduce the complexity term H_2^σ, which is given by

$$H_2^\sigma = \min_{i \in \mathcal{A}} \frac{i}{\tilde{\Delta}^2_{(i)}}$$

where $\tilde{\Delta}_i^2 = \frac{\Delta_i^2}{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}$, and $(\tilde{\Delta}_{(i)})$ is an increasing ordering of $(\tilde{\Delta}_i)$. Similar to the relation between H_1 and H_2, it can be shown that $H_2^\sigma \leq H_1^\sigma \leq \log(2K)H_2^\sigma$.

Our main result is summarized in the following theorem where we prove an upper bound on the expected loss.

Theorem 3.1. *For $K \geq 4$ and $\rho = 1/3$, the expected loss of the AugUCB algorithm is given by,*

$$\mathbb{E}[\mathcal{L}(T)] \leq 2K \left(\log_2 \frac{T}{e} + 1 \right) \exp\left(- \frac{T}{4096 H_2^\sigma a} \right)$$

where $a = \log(\frac{3}{16} K \log K)$.

Proof. The proof comprises three modules. In the first module we investigate the necessary conditions for arm elimination within a specified number of rounds, which is motivated by the technique in []. Bounds on the arm-elimination probability is then obtained; however, since we use variance estimates, we invoke the Bernstein inequality (as in []) rather that the Chernoff-Hoeffding bounds (which is appropriate for the UCB-Improved []). In the second module, as in [], we define a favourable event that will yield a bound for the expected loss. In the final module we conclude by combining the results for the first two modules. The details are as follows.

Arm Elimination: Recall the notations used in the algorithm, Also, for each arm $i \in \mathcal{A}$, define $m_i = \min\left\{ m | \sqrt{\rho \epsilon_m} < \frac{\Delta_i}{2} \right\}$. In the m_i-th round, whenever $n_i = \ell_{m_i} \geq \frac{2\psi_{m_i} \log(T\epsilon_{m_i})}{\epsilon_{m_i}}$, we obtain (as $\hat{v}_i \in [0, 1]$)

$$s_i \leq \sqrt{\frac{\rho(\hat{v}_i + 1)\epsilon_{m_i}}{8}} \leq \frac{\sqrt{\rho \epsilon_{m_i}}}{2} < \frac{\Delta_i}{4}. \tag{1}$$

First, let us consider a bad arm $i \in \mathcal{A}$ (i.e., $r_i < \tau$). We note that, in the m_i-th round whenever $\hat{r}_i \leq r_i + 2s_i$, then arm i is eliminated as a bad arm. This is easy to verify as follows: using (1) we obtain,

$$\begin{aligned} \hat{r}_i &\leq r_i + 2s_i \\ &= r_i + 4s_i - 2s_i \\ &< r_i - \Delta_i - 2s_i \\ &= \tau - 2s_i \end{aligned}$$

which is precisely one of the elimination conditions in Algorithm 1. Thus, the probability that a bad arm is not eliminated correctly in the m_i-th round (or before) is given by

$$\mathbb{P}(\hat{r}_i > r_i + 2s_i) \leq \mathbb{P}(\hat{r}_i > r_i + 2\bar{s}_i) + \mathbb{P}\left(\hat{v}_i \geq \sigma_i^2 + \sqrt{\rho_v \epsilon_{g_i}}\right) \tag{2}$$

where

$$\bar{s}_i = \sqrt{\frac{\rho \psi_{m_i}(\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} + 1)\log(T\epsilon_{m_i})}{4n_i}}$$

Note that, substituting $n_i = \ell_{m_i} \geq \frac{2\psi_{m_i} \log(T\epsilon_{m_i})}{\epsilon_{m_i}}$, $\bar{s}_i$ can be simplified to obtain,

$$2\bar{s}_i \leq \frac{\sqrt{\rho \epsilon_{m_i}(\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} + 1)}}{2} \leq \sqrt{\rho \epsilon_{m_i}}. \tag{3}$$

The first term in the LHS of (2) can be bounded using the Bernstein inequality as below:

$$\begin{aligned} &\mathbb{P}(\hat{r}_i > r_i + 2\bar{s}_i) \\ &\leq \exp\left(- \frac{(2\bar{s}_i)^2 n_i}{2\sigma_i^2 + \frac{4}{3}\bar{s}_i} \right) \\ &\leq \exp\left(- \frac{\rho \psi_{m_i}(\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} + 1)\log(T\epsilon_{m_i})}{2\sigma_i^2 + \frac{2}{3}\sqrt{\rho \epsilon_{m_i}}} \right) \\ &\overset{(a)}{\leq} \exp\left(- \frac{3\rho T \epsilon_{m_i}}{256 a^2} \left(\frac{\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}}} \right) \log(T\epsilon_{m_i}) \right) \\ &:= \exp(-Z_i) \end{aligned} \tag{4}$$

where, for simplicity, we have used α_i to denoted the exponent in the inequality (a). Also, note that (a) is obtained by using $\psi_{m_i} = \frac{T\epsilon_{m_i}}{128a^2}$, where $a = (\log(\frac{3}{16} K \log K))$.

The second term in the LHS of (2) can be simplified as follows:

$$\begin{aligned} &\mathbb{P}\left\{ \hat{v}_i \geq \sigma_i^2 + \sqrt{\rho_v \epsilon_{g_i}} \right\} \\ &\leq \mathbb{P}\left\{ \frac{1}{n_i} \sum_{t=1}^{n_i} (X_{i,t} - r_i)^2 - (\hat{r}_i - r_i)^2 \geq \sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} \right\} \\ &\leq \mathbb{P}\left\{ \frac{\sum_{t=1}^{n_i} (X_{i,t} - r_i)^2}{n_i} \geq \sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} \right\} \\ &\overset{(a)}{\leq} \mathbb{P}\left\{ \frac{\sum_{t=1}^{n_i} (X_{i,t} - r_i)^2}{n_i} \geq \sigma_i^2 + 2\bar{s}_i \right\} \\ &\overset{(b)}{\leq} \exp\left(- \frac{3\rho \psi_{m_i}}{2} \left(\frac{\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho \epsilon_{m_i}}} \right) \log(T\epsilon_{m_i}) \right) \\ &= \exp(-Z_i) \end{aligned} \tag{5}$$

where inequality (a) is obtained using (3), while (b) follows from the Bernstein inequality.

Thus, using (4) and (5) in (2) we obtain $\mathbb{P}(\hat{r}_i > r_i + 2s_i) \leq 2\exp(-Z_i)$. Proceeding similarly, for a good arm $i \in \mathcal{A}$, the probability that it is not correctly eliminated in the m_i-th round (or before) is also bounded by $\mathbb{P}(\hat{r}_i < r_i - 2s_i) \leq 2\exp(-Z_i)$. In general, for any $i \in \mathcal{A}$ we have

$$\mathbb{P}(|\hat{r}_i - r_i| > 2s_i) \leq 4\exp(-Z_i). \tag{6}$$

Favourable Event: Following the notation in [Locatelli *et al.*, 2016] we define the event

$$\xi = \left\{\forall i \in \mathcal{A}, \forall t = 1, 2, .., T : |\hat{r}_i - r_i| \leq 2s_i\right\}.$$

Note that, on ξ each arm $i \in \mathcal{A}$ is eliminated correctly in the m_i-th round (or before). Thus, it follows that $\mathbb{E}[\mathcal{L}(T)] \leq P(\xi^c)$. Since ξ^c can be expressed as an union of the events $(|\hat{r}_i - r_i| > 2s_i)$ for all $i \in \mathcal{A}$ and all $t = 1, 2, \cdots, T$, using union bound we can write

$$\begin{aligned}
&\mathbb{E}[\mathcal{L}(T)] \\
&\leq \sum_{i\in\mathcal{A}}\sum_{t=1}^{T}\mathbb{P}(|\hat{r}_i - r_i| > 2s_i) \\
&\leq \sum_{i\in\mathcal{A}}\sum_{t=1}^{T} 4\exp(-Z_i) \\
&\leq 4T\sum_{i\in\mathcal{A}}\exp\left(-\frac{3\rho T\epsilon_{m_i}}{256a^2}\left(\frac{\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}}\right)\log(T\epsilon_{m_i})\right) \\
&\overset{(a)}{\leq} 4T\sum_{i\in\mathcal{A}}\exp\left(-\frac{3T\Delta_i^2}{4096a^2}\left(\frac{4\sigma_i^2 + \Delta_i + 4}{12\sigma_i^2 + \Delta_i}\right)\log(\frac{3}{16}T\Delta_i^2)\right) \\
&\overset{(b)}{\leq} 4T\sum_{i\in\mathcal{A}}\exp\left(-\frac{12T\Delta_i^2}{(12\sigma_i + 12\Delta_i)}\frac{\log(\frac{3}{16}K\log K)}{4096a^2}\right) \\
&\overset{(c)}{\leq} 4T\sum_{i\in\mathcal{A}}\exp\left(-\frac{T\Delta_i^2\log(\frac{3}{16}K\log K)}{4096(\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i})a^2}\right) \\
&\overset{(d)}{\leq} 4T\sum_{i\in\mathcal{A}}\exp\left(-\frac{T\log(\frac{3}{16}K\log K)}{4096\max_i(i\tilde{\Delta}_{(i)}^{-2})(\log(\frac{3}{16}K\log K))^2}\right) \\
&\overset{(e)}{\leq} 4KT\exp\left(-\frac{T}{4096H_2^{\sigma}(\log(\frac{3}{16}K\log K))}\right).
\end{aligned}$$

The following are used to achieve the above simplification:

- (a) is obtained by noting that in round m_i we have $\frac{\Delta_i}{4} \leq \sqrt{\epsilon_{m_i}\rho} < \frac{\Delta_i}{2}$.
- For (b), we note that the function $x \mapsto x\exp(-Cx^2)$, where $x \in [0,1]$, is decreasing on $[1/\sqrt{2C}, 1]$ for any $C > 0$ (see [Bubeck *et al.*, 2011; Auer and Ortner, 2010]). Thus, using $C = \lfloor\sqrt{e/T}\rfloor$ and putting $\min_{i\in\mathcal{A}}\Delta_i = \Delta = \sqrt{\frac{K\log K}{T}} > \sqrt{\frac{e}{T}}, \forall i \in \mathcal{A}$ we obtain (b).
- To obtain (c) we have used the inequality $\Delta_i \leq \sqrt{\sigma_i^2 + (16/3)\Delta_i}$ (which holds because $\Delta_i \in [0,1]$).
- To obtain (d) and (e), recall that $\tilde{\Delta}_i = \frac{\Delta_i^2}{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}$ and $a = \log(\frac{3}{16}K\log K)$. Finally, note that

$$\tilde{\Delta}_i^{-2} \leq \max_{j\in\mathcal{A}}\Delta_{(j)}^{-2} \leq \max_{j\in\mathcal{A}} j\Delta_{(j)}^{-2} = H_2^{\sigma}.$$

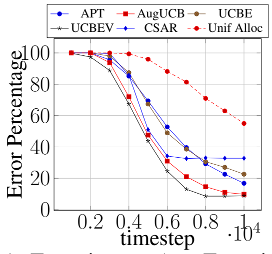
□

4 Numerical Experiments

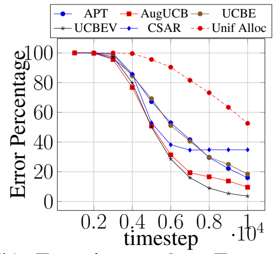
In this section we compare the empirical performance of AugUCB against APT, Uniform Allocation, CSAR, UCBE and UCBEV algorithm. The threshold τ is set at 0.5 for all experiments. Each algorithm is run independently 500 times for 10000 timesteps and the output set of arms suggested by the algorithms at every timestep is recorded. The output is considered erroneous if the correct set of arms is not $i = \{6, 7, 8, 9, 10\}$ (true for all the experiments). The error percentage over 500 runs is plotted against 10000 timesteps. For the uniform allocation algorithm, for each $t = 1, 2, .., T$ the arms are sampled uniformly. For UCBE algorithm ([Audibert *et al.*, 2009]) which was built for single best arm identification, we modify it according to [Locatelli *et al.*, 2016] to suit the goal of finding arms above the threshold τ. So the exploration parameter a in UCBE is set to $a = \frac{T-K}{H_1}$. Again, for UCBEV, following [Gabillon *et al.*, 2011], we modify it such that the exploration parameter $a = \frac{T-2K}{H_1^{\sigma}}$. Then for each timestep $t = 1, 2, .., T$ we pull the arm that minimizes $\{|\hat{r}_i - \tau| - \sqrt{\frac{a}{n_i}}\}$, where n_i is the number of times the arm i is pulled till $t-1$ timestep and a is set as mentioned above for UCBE and UCBEV respectively. Also, APT is run with $\epsilon = 0.05$, which denotes the precision with which the algorithm suggests the best set of arms and we modify CSAR as per [Locatelli *et al.*, 2016] such that it behaves as a Successive Reject algorithm whereby it rejects the arm farthest from τ after each phase. For AugUCB we take $\rho_\mu = \frac{1}{8}$ and $\rho_v = \frac{1}{3}$ as in Theorem 3.1.

The first experiment is conducted on a testbed of 100 arms involving Gaussian reward distribution with expected rewards of the arms $r_{1:4} = 0.2 + (0:3) * 0.05$, $r_5 = 0.45$, $r_6 = 0.55$, $r_{7:10} = 0.65 + (0:3) * 0.05$ and $r_{11:100}$=0.4. The means of first 10 arms are set as arithmetic progression. Variance is set as $\sigma_{1:5}^2 = 0.5$ and $\sigma_{6:10}^2 = 0.6$. Then $\sigma_{11:100}^2$ is chosen uniform randomly between $0.38-0.42$. The means in the testbed are chosen in such a way that any algorithm has to spend a significant amount of budget to explore all the arms and variance is chosen in such a way that the arms above τ have high variance whereas arms below τ have lower variance. The result is shown in Figure 1(a). In this experiment we see that UCBEV which has access to the problem complexity and is a variance-aware algorithm beats all other algorithm including UCBE which has access to the problem complexity but does not take into account the variance of the arms. AugUCB with the said parameters outperforms UCBE, APT and the other non variance-aware algorithms that we have considered.

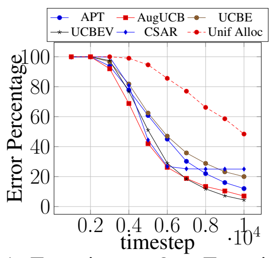
The second experiment is conducted on a testbed of 100 arms with the means of first 10 arms set as Geometric Progression. The testbed involves Gaussian reward distribution



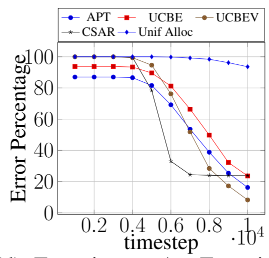
(a) Experiment 1: Experiment with Arithmetic Progression



(b) Experiment 2: Experiment with Geometric Progression



(c) Experiment 3: Experiment with three Group Setting



(d) Experiment 4: Experiment with Geometric Progression (Bernoulli)

Figure 1: Experiments with thresholding bandit

with expected rewards of the arms as $r_{1:4} = 0.4 - (0.2)^{1:4}$, $r_5 = 0.45$, $r_6 = 0.55$ and $r_{7:10} = 0.6 + (0.2)^{5-(1:4)}$. The variances of all the arms and $r_{11:100}$ are set in the same way as in experiment 1. AugUCB, APT, CSAR, Uniform Allocation, UCBE and UCBEV with the same settings as experiment 1 are run on this testbed. The result is shown in Figure 1(b). Here, again we see that AugUCB beats APT, UCBE and all the non-variance aware algorithms with only UCBEV beating AugUCB.

The third experiment is conducted on a testbed of 100 arms with the means of first 10 arms set in three groups. The testbed involves Gaussian reward distribution with expected rewards of the arms as $r_{1:3} = 0.1$, $r_{4:7} = \{0.35, 0.45, 0.55, 0.65\}$ and $r_{8:10} = 0.9$. The variances of all the arms and $r_{11:100}$ are set in the same way as in experiment 1. AugUCB, APT, CSAR, Uniform Allocation, UCBE and UCBEV with the same settings as experiment 1 are run on this testbed. The result is shown in Figure 1(c). Here, also we see that AugUCB beats APT, UCBE and all the non-variance aware algorithms with only UCBEV beating AugUCB.

The fourth experiment is the replication of the second experiment on a testbed of 100 arms involving Bernoulli reward distribution with expected rewards and variances of the arms set in same way as in Experiment 2. The result is shown in Figure 1(d). AugUCB is only beaten by UCBEV in this setup as well.

5 Conclusion and Future work

From a theoretical viewpoint, we conclude that considering only arm elimination by mean estimation, the expected loss AugUCB is more than APT and CSAR while considering arm elimination by both mean and variance estimation we see that the expected loss of AugUCB is more than only UCBEV (which has access to problem complexity). From the numerical experiments on settings with large number of arms with different mean and variance, we observed that AugUCB outperforms all the non-variance aware algorithms. It would be interesting future research to come up with an anytime version of AugUCB algorithm. This is also the first paper to apply elimination by variance estimation in the TBP problem by modifying UCB-Improved and CCB algrithms.

References

[Audibert and Bubeck, 2009] Jean-Yves Audibert and Sébastien Bubeck. Minimax policies for adversarial and stochastic bandits. In *COLT*, pages 217–226, 2009.

[Audibert and Bubeck, 2010] Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*, pages 13–p, 2010.

[Audibert *et al.*, 2009] Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

[Auer and Ortner, 2010] Peter Auer and Ronald Ortner. Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

[Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[Bubeck and Cesa-Bianchi, 2012] Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

[Bubeck *et al.*, 2011] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852, 2011.

[Bubeck *et al.*, 2013] Sébastien Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In *ICML (1)*, pages 258–265, 2013.

[Chen *et al.*, 2014] Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 379–387, 2014.

[Even-Dar *et al.*, 2006] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *The Journal of Machine Learning Research*, 7:1079–1105, 2006.

[Gabillon *et al.*, 2011] Victor Gabillon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Sébastien Bubeck. Multi-bandit best arm identification. In *Advances in Neural Information Processing Systems*, pages 2222–2230, 2011.

[Gabillon *et al.*, 2012] Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.

[Jamieson and Nowak, 2014] Kevin Jamieson and Robert Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*, pages 1–6. IEEE, 2014.

[Kalyanakrishnan *et al.*, 2012] 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-12)*, pages 655–662, 2012.

[Lai and Robbins, 1985] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[Liu and Tsuruoka, 2016] Yun-Ching Liu and Yoshimasa Tsuruoka. Modification of improved upper confidence bounds for regulating exploration in monte-carlo tree search. *Theoretical Computer Science*, 2016.

[Locatelli *et al.*, 2016] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. *arXiv preprint arXiv:1605.08671*, 2016.

[Robbins, 1952] Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1952.

[Thompson, 1933] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, pages 285–294, 1933.

6 Appendix(We will comment this out later)

$$H_1^\sigma = \sum_{i=1}^{K} \frac{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}{\Delta_i^2}$$

$$H_2^\sigma = \min_{i \in \mathcal{A}} i \tilde{\Delta}_{(i)}^{-2}, \text{ where } \tilde{\Delta}_i^{-2} = \frac{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}{\Delta_i^2}$$

We know that $\sigma_i \in [0,1], \forall i \in \mathcal{A}$ and $\Delta_i \in [0,1], \forall i \in \mathcal{A}$ and so $\sigma_i^2 \leq \sigma_i$ and $\sqrt{\Delta_i} \geq \Delta_i$.

$$\begin{aligned}
(3\Delta_i^2).\left(\frac{4\sigma_i^2 + \Delta_i + 4}{12\sigma_i^2 + \Delta_i}\right) &> \left(\frac{12\Delta_i^2}{12\sigma_i^2 + \Delta_i}\right) \\
&> \left(\frac{12\Delta_i^2}{12\sigma_i^2 + 12\Delta_i}\right) \\
&> \left(\frac{\Delta_i^2}{\sigma_i + \Delta_i}\right) \\
&> \left(\frac{\Delta_i^2}{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}\right) \\
&> \left(\frac{1}{\min_i i\tilde{\Delta}_i^2}\right)
\end{aligned}$$

Now, from [Audibert and Bubeck, 2010] we know that,

$$\begin{aligned}
\sum_{i=1}^{K} \tilde{\Delta}_i^{-2} = \tilde{\Delta}_{(2)}^{-2} + \sum_{i=2}^{K} \frac{1}{i} i \tilde{\Delta}_{(i)}^{-2} &\leq \bar{\log} K \min_i i \tilde{\Delta}_{(i)}^{-2} \\
&\leq \log(2K) H_2^\sigma, \text{ as } \bar{\log} K \leq \log(2K)
\end{aligned}$$

So, $H_2^\sigma \leq H_1^\sigma \leq \log(2K) H_2^\sigma$

Regarding union bound

$$\begin{aligned}
\mathbb{P}\{\xi_1 \cup \xi_2\} &= \mathbb{P}\{\xi_1\} + \mathbb{P}\{\xi_2\} - \mathbb{P}\{\xi_1 \cap \xi_2\} \\
&\leq \mathbb{P}\{\xi_1\} + \mathbb{P}\{\xi_2\}
\end{aligned}$$

So,

$$1 - \mathbb{P}\{\xi_1 \cup \xi_2\} \geq 1 - \mathbb{P}\{\xi_1\} + \mathbb{P}\{\xi_2\} \geq \mathbb{E}[\mathcal{L}(T)]$$