Thresholding Bandits with Augmented UCB

author names withheld

Editor: Under Review for COLT 2016

Abstract

To be written

1. Introduction

In this paper we study a specific combinatorial pure exploration problem called thresholding bandit problem in the stochastic multi-armed bandit setting. In the stochastic multi-armed bandit setting a learning agent is required to choose from a set of decisions or arms at every round. The agent is then presented with a reward for that round, which is an independent draw from a stationary distribution specific to the arm selected. The agent, however, does not know the mean of the distributions associated with each arm, denoted by r_i, including the optimal arm which will give it the best reward, denoted by r^*. The agent attempts to make arm choices that will maximize some performance measure by keeping track of the reward that has been gathered from previous selections of the arm, for each arm. This is called the estimated mean reward of an arm denoted by $\hat{r}_i$. The bandit problem can be conceptualized as a sequential decision making process where the agent is at each round presented with an *exploration-exploitation dilemma*. The agent could pull the arm which has the highest observed mean reward till now (exploitation) or to explore other arms, with the prospect of finding superior performance which was previously unobserved (exploration).

Formally, let r_i, $i = 1, \ldots, K$ denote the mean rewards of the K arms and $r^* = \max_i r_i$ the optimal mean reward. The objective in some of the stochastic bandit problem is to minimize the cumulative regret, which is defined as follows:

$$R_T = r^*T - \sum_{i \in A} r_i N_i(T),$$

where T is the number of rounds, $N_i(T) = \sum_{m=1}^{T} I(I_m = i)$ is the number of times the algorithm chose arm i up to round T. The expected regret of an algorithm after T rounds can be written as,

$$\mathbb{E}[R_T] = \sum_{i=1}^{K} \mathbb{E}[N_i(T)]\Delta_i,$$

where $\Delta_i = r^* - r_i$ denotes the gap between the means of the optimal arm and of the i-th arm.

In the pure exploration thresholding bandit setup the goal is different than minimizing the cumulative regret. Here the learning algorithm is provided with a threshold τ and it has to output all such arms i whose mean of reward distribution r_i is above τ after T rounds. This is a specific instance of combinatorial pure exploration where the learning algorithm can explore as much as possible given

© 2016 .

a fixed horizon T and not be concerned with the usual exploration-exploitation dilemma. Let A be the set of all arms. Formally we can define a set $S_\tau = \{i \in A : r_i \geq \tau\}$ and the complementary set $S_\tau^C = \{i \in A : r_i < \tau\}$. Also we define $\hat{S}_\tau = \hat{S}_\tau(T) \subset A$ and its complementary set $\hat{S}_\tau^C$ as the recommendation of the learning algorithm after T rounds. Given such sets exists, the performance of the learning agent is measured by how much accuracy it can discriminate between S_τ and S_τ^C after time horizon T. The loss $\mathcal{L}$ is defined as:-

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

The goal of the learning agent is to minimize $\mathcal{L}(T)$. So, the expected loss after T rounds is

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

which we can say is the probability of making mistake, that is whether the learning agent at the end of round T rejects arms from S_τ or accepts arms from S_τ^C in its final recommendation. Also, we are looking at an anytime algorithm, so the knowledge of T may not be known to the learner.

2. Motivation

The thresholding bandit problem has several relevant industrial applications. The variants of TopM problem (identifying the best M arms from K given arms) can be readily used in the thresholding problem.

1. *Product Selection:* A company wants to introduce a new product in market and there is a clear separation of the test phase from the commercialization phase. In this case the company tries to minimize the loss it might incur in the commercialization phase by testing as much as possible in the test phase. So from the several variants of the product that are in the test phase the learning agent must suggest the product variant(s) that are above a particular threshold τ at the end of the test phase that have the highest probability of minimizing loss in the commercialization phase. A similar problem has been discussed for single best product variant identification without threshold in Bubeck et al. (2011).

2. *Mobile Phone Channel Allocation:* Another similar problem as above concerns channel allocation for mobile phone communications (Audibert et al. (2009)). Here there is a clear separation between the allocation phase and communication phase whereby in the allocation phase a learning algorithm has to explore as many channels as possible to suggest the best possible set of channel(s) that are above a particular threshold τ. The threshold depends on the subscription level of the customer. With higher subscription the customer is allowed better channel(s) with the τ set high. Each evaluation of a channel is noisy and the learning algorithm must come up with the best possible suggestion within a very small number of attempts.

3. *Anomaly Detection and Classification:* Thresholding bandit can also be used for anomaly detection and classification where we define a cutoff level τ and for any samples above this cutoff gets classified as an anomaly. For further reading we point the reader to section 3 of Locatelli et al. (2016).

3. Contribution

To be written

4. Related Works and Previous Results

A significant amount of work has been done on the stochastic multi-armed bandit setting regarding minimizing cumulative regret with a single optimal arm. For a survey of such works we refer the reader to Bubeck and Cesa-Bianchi (2012). An early work involving a bandit setup is Thompson (1933), where the author deals with the problem of choosing between two treatments to administer on patients who come in sequentially. Following the seminal work of Robbins (1952), bandit algorithms have been extensively studied in a variety of applications. From a theoretical standpoint, an asymptotic lower bound for the regret was established in Lai and Robbins (1985). Several other works such as Auer et al. (2002a), Audibert and Bubeck (2009) and Auer and Ortner (2010) have shown results for minimizing cumulative regret in stochastic bandit setup whereas works such as Auer et al. (2002b) have concentrated on adversarial bandit setup.

There have been several algorithms with strong regret guarantees. The foremost among them is UCB1 by Auer et al. (2002a), which has a regret upper bound of $O\left(\dfrac{K \log T}{\Delta}\right)$, where $\Delta = \min_{i:\Delta_i>0} \Delta_i$. This result is asymptotically order-optimal for the class of distributions considered. However, the worst case gap independent regret bound of UCB1 can be as bad as $O\left(\sqrt{TK \log T}\right)$. In Audibert and Bubeck (2009), the authors propose the MOSS algorithm and establish that the worst case regret of MOSS is $O\left(\sqrt{TK}\right)$ which improves upon UCB1 by a factor of order $\sqrt{\log T}$. However, the gap-dependent regret of MOSS is $O\left(\dfrac{K^2 \log\left(T\Delta^2/K\right)}{\Delta}\right)$ and in certain regimes, this can be worse than even UCB1 (see Audibert and Bubeck (2009),Lattimore (2015)). The UCB-Improved algorithm, proposed in Auer and Ortner (2010), is a round-based algorithm[1] variant of UCB1 that has a gap-dependent regret bound of $O\left(\dfrac{K \log T\Delta^2}{\Delta}\right)$, which is better than that of UCB1. On the other hand, the worst case regret of UCB-Improved is $O\left(\sqrt{TK \log K}\right)$.

In the pure exploration setup, a significant amount of research has been done on finding the best arm(s) from a set of arms. The pure exploration setup has been explored in mainly two settings:-

1. Fixed Budget setting: In this setting the learning algorithm has to suggest the best arm(s) within a fixed number of attempts that is given as an input. The objective here is to maximize the probability of returning the best arm(s). One of the foremost papers to deal with single best arm identification is Audibert et al. (2009) where the authors come up with the algorithm UCBE and Successive Reject(SR) with simple regret guarantees. 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. In the combinatorial fixed budget setup Gabillon et al. (2011) come up with Gap-E and Gap-EV

[1] An algorithm is *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.

algorithm which suggests the best m (given as input) arms at the end of the budget with high probability. Similarly, Bubeck et al. (2013) comes up with the algorithm Successive Accept Reject(SAR) which is an extension of the SR algorithm. SAR is a round based algorithm whereby at the end of round an arm is either accepted or rejected based on certain conditions till the required top m arms are suggested at the end of the budget with high probability.

2. Fixed Confidence setting: In this setting the the learning algorithm has to suggest the best arm(s) with a fixed (given as input) confidence with as less number of attempts as possible. The single best arm identification has been handled in Even-Dar et al. (2006) where they come up with an algorithm called Successive Elimination (SE) which comes up with an arm that is ϵ close to the optimal arm. In the combinatorial setup recently Kalyanakrishnan et al. (2012) have suggested the LUCB algorithm which on termination returns m arms which are atleast ϵ close to the true top m arms with $1-\delta$ probability.

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. A similar combinatorial setup was also explored in Chen et al. (2014) where the authors come up with more similarities and dissimilarities between these two settings in a more general setup. In their work, the learning algorithm, called Combinatorial Successive Accept Reject (CSAR) is similar to SAR with a more general setup. The thresholding bandit problem is a specific instance of the pure exploration setup of Chen et al. (2014). In the latest work in Locatelli et al. (2016) the algorithm Anytime Parameter-Free Thresholding (APT) algorithm comes up with a better anytime guarantee than CSAR for the thresholding bandit problem.

5. Notation Used and Assumptions

In this paper A is the set of all arms and $|A| = K$ denotes the number of arms in the set. Any arm is denoted by i. The average estimated payoff for any arm is denoted by $\hat{r}_i$ whereas the true mean of the distribution from which the rewards are sampled is denoted by r_i. The optimal arm is denoted by $*$. The '*' superscript is used to denote anything related to optimal arm. $\Delta_i^\tau = |\tau - r_i|$ and $\hat{\Delta}_i^\tau = |\tau - \hat{r}_i|$. Also we define $\Delta_i = r^* - r_i$ and $\hat{\Delta}_i = \hat{r}^* - \hat{r}_i$. In all cases $\min_{i \in A} \Delta_i$ is denoted by Δ. c_i denotes the confidence interval for arm i. ψ denotes the exploration regulatory factor and ρ as arm elimination parameter.

It is assumed that the distribution from which rewards are sampled are identical and independent sub-Gaussian distributions. Throughout the paper, we assume that the distributions v_i are sub-Gaussian including Gaussian distributions with variance less than 1 and distributions supported on an interval of length less than 2. We will also assume that all rewards are bounded in $[0, 1]$.

6. Augmented UCB

In algorithm 1, hence referred to as AugUCB, we have three input parameters, ρ which is the arm elimination parameter, ψ which is the exploration regulatory factor and the threshold τ. The salient features of the algorithm are listed below:-

- AugUCB combines the power of UCB-Improved (Auer and Ortner (2010)) , APT (Locatelli et al. (2016)) and SAR (Gabillon et al. (2011)). The main approach is based on UCB-

Algorithm 1 AugmentedUCB

```
Input: Time horizon $T$, exploration parameters $\rho$ and $\psi$, threshold $\tau$.
Initialization: Set $B_0 := A$, $M = \lfloor \frac{1}{2}\log_2 \frac{T}{e} \rfloor$, $m := 0$, $\epsilon_0 := 1$, $n_0 = \lceil \frac{2\log(\psi T \epsilon_0^2)}{\epsilon_0} \rceil$ and $N_0 = K * n_0$.
Pull each arm once
for $t = K+1, .., T$ do
    Pull arm $i$ in $B_m$ such that $\min_{i \in B_m} \left\{ |\hat{r}_i - \tau| - \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}} \right\}$, where $n_i$ is the number of times the arm $i$ has been pulled.
    $t := t + 1$
    Arm Elimination
        For each arm $i \in B_m$, remove arm $i$ from $B_m$ if
            $\hat{r}_i + \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}} < \tau - \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}}$
        For each arm $i \in B_m$, remove arm $i$ from $B_m$ if
            $\hat{r}_i - \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}} > \tau + \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}}$
    if $t \geq N_m$ and $m \leq M$ then
        Reset Parameters
            $\epsilon_{m+1} := \frac{\epsilon_m}{2}$
            $B_{m+1} := B_m$
            $n_m = \left\lceil \frac{2\log(\psi T \epsilon_m^2)}{\epsilon_m} \right\rceil$
            $N_{m+1} := t + |B_m| n_m$
            $m := m + 1$
    end
end
Output $\hat{S}_\tau = \{i : \hat{r}_i \geq \tau\}$.
```

Improved with modifications suited for the thresholding bandit problem. The active set B_0 is initialized with all the arms from A.

- We divide the entire budget T into rounds/phases as like UCB-Improved, SAR and CSAR. The choice of M comes from UCB-Improved which necessarily entails that the $\epsilon_m \geq \sqrt{\frac{e}{T}}$. So, M is the total number of rounds and is the same as UCB-Improved. After the end of each such round m we eliminate arm(s) from active set B_m and update parameters.

- As suggested by Liu and Tsuruoka (2016) to make AugUCB an anytime algorithm and to overcome too much early exploration, we no longer pull all the arms equal number of times in each round but pull the arm that minimizes

$$\left\{|\hat{r}_i - \tau| - \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}}\right\}$$

in the active set B_m.

- $\min_{i \in B_m} \left\{|\hat{r}_i - \tau| - \sqrt{\frac{\rho \log(\psi T \epsilon_m^2)}{2n_i}}\right\}$ condition actually makes it possible to pull the arms closer to the threshold τ. This is a strategy used by APT.

- This also gets rid of the excessive initial exploration employed by UCB-Improved and yet with suitable choice of ρ and ψ we can fine tune the exploration.

- The arm elimination condition simply removes arm(s) if the algorithm is sufficiently sure that the mean of the arms are very high or very low about the threshold. 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.

- At the end of the budget T the algorithm outputs all the arms whose estimated average payoff $\hat{r}_i$ is above the threshold τ thereby making this an anytime algorithm whereby we need not finish every round.

- The arm elimination condition(s) helps in re-allocating the remaining budget/pulls among the surviving arms. Those among the remaining arms are pulled which are closer to the threshold. Arms lying far to the either side of the threshold are eliminated from the active set B_m.

7. Main Results

7.1. Problem Complexity

We define problem complexity as,

$$H^\tau = \sum_{i=1}^{K} \frac{1}{(\Delta_i^\tau)^2}, \text{ where } \Delta_i^\tau = |r_i - \tau|$$

This is same as the problem complexity defined in Locatelli et al. (2016) for the thresholding bandit problem and is similar to the problem complexity defined in Audibert and Bubeck (2010) for single best arm identification.

7.2. Theorem 1

Proof

According to the algorithm, the number of rounds is $m = \{0, 1, 2, ..M\}$ where $M = \lfloor \frac{1}{2} \log_2 \frac{T}{e} \rfloor$. So, $\epsilon_m \geq 2^{-M} = \sqrt{\frac{e}{T}}$. Also each round m consists of $|B_m| n_m$ timesteps where $n_m = \frac{\log(\psi T \epsilon_m^2)}{\epsilon_m}$ and B_m is the set of all surviving arms.

Let $c_i = \sqrt{\dfrac{\rho \log{(\psi T \epsilon_m^2)}}{2 n_i}}$ denote the confidence interval, where n_i is the number of times an arm i is pulled. Let $A^{'} = \{i \in A | \Delta_i^\tau \geq b\}$, for $b \geq \sqrt{\frac{e}{T}}$. Let m_i be the minimum round that an arm i gets eliminated. So $m_i = min\{m | \epsilon_m < \frac{\Delta_i^\tau}{2}\}$.

At the end of any round m, for any arm i, two cases are possible.

7.2.1. *Case a: Some arm i is not eliminated on or before round m_i*

For any arm i, if it is eliminated from active set B_m then the below two events have to come true,

$$\hat{r}_i + c_i < \tau - c_i, \tag{1}$$
$$\hat{r}_i - c_i > \tau + c_i, \tag{2}$$

For 1 we can see that it eliminates arms that have performed poorly and removes them them from B_m. Similarly, 2 eliminates arms from B_m that have performed very well compared to threshold τ.

Each round consist of $|B_{m_i}| n_m$ timesteps. In the m_i-th round an arm i can be pulled no more than n_{m_i} times. So when $n_i = n_{m_i}$, putting the value of $n_{m_i} = \dfrac{2\log{(\psi T \epsilon_{m_i}^2)}}{\epsilon_{m_i}}$ in c_i,

$$\begin{aligned}
c_i &= \sqrt{\frac{\rho \epsilon_{m_i} \log(\psi T \epsilon_{m_i}^2)}{2 n_i}} \\
&= \sqrt{\frac{\rho \epsilon_{m_i} \log(\psi T \epsilon_{m_i}^2)}{2 * 2 \log(\psi T \epsilon_{m_i}^2)}} \\
&= \frac{\sqrt{\rho \epsilon_{m_i}}}{2} \\
&\leq \sqrt{\rho \epsilon_{m_i + 1}} < \frac{\Delta_i^\tau}{4}, \text{ as } \rho \in (0, 1].
\end{aligned}$$

Again, for $i \in A^{'}$ for 1 elimination condition,

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

Also, for $i \in A^{'}$ for 2 elimination condition,

$$\begin{aligned}
\hat{r}_i - c_i &\geq r_i - 2c_i \\
&= r_i - 4c_i + 2c_i \\
&> r_i - \Delta_i^\tau + 2c_i \\
&\geq \tau + 2c_i \\
&\geq \tau + c_i
\end{aligned}$$

Since, arm elimination condition is being checked at every timestep, in the m_i-th round as soon as $n_i = n_{m_i}$, arm i gets eliminated. Applying Chernoff-Hoeffding bound and considering independence of complementary of the two events in 1,

$$\begin{aligned}\mathbb{P}\{\hat{r}_i \geq r_i - (\tau + 2c_i)\} &\leq exp(-2(\tau + 2c_i)^2 n_i) \\ &\leq exp(-2(2\tau c_i)^2 n_i) \text{ , as} (a+b)^2 \geq (ab)^2 \text{ for } a, b \in [0,1] \\ &\leq exp(-8 * \tau^2 \frac{\rho \log(\psi T \epsilon_{m_i}^2)}{2n_i} * n_i) \\ &\leq \frac{1}{(\psi T \epsilon_{m_i}^2)^{4\tau^2 \rho}}\end{aligned}$$

Similarly, $\mathbb{P}\{\hat{r}_i \leq r_i + (\tau + 2c_i)\} \leq \dfrac{1}{(\psi T \epsilon_{m_i}^2)^{4\tau^2 \rho}}$

Summing, the two up, the probability that an arm i is not eliminated on or before m_i-th round based on the 1 and 2 elimination condition is $\left(\dfrac{2}{(\psi T \epsilon_{m_i}^2)^{4\tau^2 \rho}}\right)$.

7.2.2. *Case b: For any arm i, it is either eliminated on or before round m_i or there is no arm above τ.*

For any round m, for any timestep $t \in m$ an arm $i \in B_m$ gets pulled if,

$$|\hat{r}_i - \tau| - c_i < |\hat{r}_k - \tau| - c_k, \forall k \in B_m$$

Now from reverse triangle inequality,

$$\begin{aligned}|\hat{r}_k(t) - r_k| &= |(\hat{r}_k(t) - \tau) - (r_k - \tau)| \qquad (3) \\ &\geq ||\hat{r}_k(t) - \tau| - |(r_k - \tau)|| \qquad (4) \\ &\geq |\hat{\Delta}_k^\tau(t) - \Delta_k^\tau| \qquad (5)\end{aligned}$$

Also, from Case a, we know that for any arm i, in round m_i

$$|\hat{r}_i - r_i| \leq c_{m_i} = \sqrt{\frac{\rho \log{(\psi T \epsilon_m^2)}}{2n_i}} \tag{6}$$

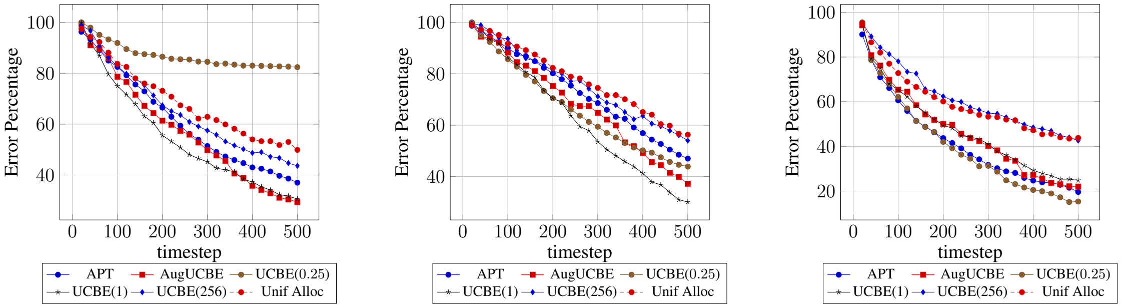
Now, combining 5 and 6 we can see that,

$$\begin{aligned}&|\hat{\Delta}_i^\tau(t) - \Delta_i^\tau| \leq c_{m_i} \\ &\Rightarrow \Delta_i^\tau - c_{m_i} \leq \hat{\Delta}_i^\tau(t) \leq \Delta_i^\tau + c_{m_i}\end{aligned}$$

Since, at time t in round m_i the arm i is pulled, so,

$$\hat{\Delta}_i^\tau - c_i < \hat{\Delta}_k^\tau - c_k, \forall k \in A'$$

■



(*a*) Experiment 1: Threshold Bandit with Arithmetic Progression

(*b*) Experiment 2: Threshold Bandit with Geometric Progression

(*c*) Experiment 3: Threshold Bandit with 3 Group Setting

Figure 1: Experiments with thresholding bandit

8. Experimental Run:

In this section we compare the empirical performance of AugUCB against APT, Unifirm Allocation and UCBE algorithm. The threshold τ is set at 0.5 for all experiments. Each algorithm is run independently a 1000 times for 500 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 1000 runs is plotted against 500 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 $a_i = 4^i \frac{T-K}{H}$ for $i \in \{-1, 0, 4\}$ and $H = \sum_{i=1}^{K} \frac{1}{\Delta_i^2}$ is defined as the problem complexity. Then for each timestep $t = 1, 2, .., T$ we pull the arm that maximizes $\{|\hat{r}_i - \tau| - \sqrt{\frac{a_i}{n_i}}\}$, where n_i is the number of times the arm i is pulled till $t - 1$ timestep. Also, APT is run with $\epsilon = 0$, which denotes the precision with which the algorithm suggests the best set of arms. So when ϵ is set to 0 APT has to suggest the exact set of arms above the threshold. For AugUCB we take $\psi = K^2 T$ and we initialize $\rho = \dfrac{1}{2^m}$ for $m = 0, 1, 2, ..., \gamma$. The high value of ψ helps in improved exploration whereas we decrease ρ sufficiently after every round to facilitate arm elimination.

The first experiment is conducted on a testbed of 10 arms involving Bernoulli reward distribution with expected rewards of the arms $r_{1:4} = 0.2 + (0 : 3) * 0.05$, $r_5 = 0.45$, $r_6 = 0.55$ and $r_{7:10} = 0.65 + (0 : 3) * 0.05$. The means are set as arithmetic progression. In this experiment we see that AugUCB performs better than all the other algorithms mentioned. Only UCBE(1) catches up with AugUCB and that is because it has access to the exact problem complexity. The result is shown in Figure 1a.

The second experiment is conducted on a testbed of 10 arms with the means set as Geometric Progression. The testbed involves Bernoulli reward distribution 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 $r_{7:10} = 0.6 + (0.2)^{5-(1:4)}$. AugUCB, APT, Uniform Allocation and UCBE with the same settings as experiment 1 are run on this testbed. The result is shown in Figure 1b. Here, we see that AugUCB beats APT with only UCBE(1) performing at par with AugUCB.

The third experiment is conducted on a testbed of 10 arms with the means divided into 3 groups. Again the testbed involves Bernoulli 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 $r_{8:10} = 0.9$. AugUCB, APT, Uniform Allocation and UCBE with the same settings as experiment 1 are run on this testbed. The result is shown in Figure 1c. Here, also we see that AugUCB beats APT.

9. Conclusion and Future work

To be written.

References

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

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.

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.

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.

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

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multi-armed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002b.

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

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

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

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.

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.

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.

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.

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.

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

Tor Lattimore. Optimally confident ucb: Improved regret for finite-armed bandits. *arXiv preprint arXiv:1507.07880*, 2015.

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

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

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

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.

10. Appendix