SYNOPSIS OF

A Study on Online Sequential Learning using Bandits

A THESIS

submitted by

SUBHOJYOTI MUKHERJEE

for the award of the degree

of

MASTER OF SCIENCE

(by Research)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

December 2017

1 Introduction

In today's world, artificial intelligence has proved to be a game-changer in designing agents that interact with an evolving environment and make decisions on the fly. The main goal of artificial intelligence is to design artificial agents that make dynamic decisions in an evolving environment. In pursuit of these, the agent can be thought of as making a series of sequential decisions by interacting with the dynamic environment which provides it with some sort of feedback after every decision which the agent incorporates into its decision-making strategy to formulate the next decision to be made. A large number of problems in science and engineering, robotics and game playing, resource management, financial portfolio management, medical treatment design, ad placement, website optimization and packet routing can be modeled as sequential decision-making under uncertainty. Many of these real-world interesting sequential decision-making problems can be formulated as reinforcement learning (RL) problems ((Bertsekas and Tsitsiklis, 1996), (Sutton and Barto, 1998)). In an RL problem, an agent interacts with a dynamic, stochastic, and unknown environment, with the goal of finding an action-selection strategy or policy that optimizes some long-term performance measure. Every time when the agent interacts with the environment it receives a signal/reward from the environment based on which it modifies its policy. The agent learns to optimize the choice of actions over several time steps which is learned from the sequences of data that it receives from the environment. This is the crux of online sequential learning.

This is in contrast to supervised learning methods that deal with labeled data which are independently and identically distributed (i.i.d.) samples from the considered domain and train some classifier on the entire training dataset to learn the pattern of this distribution to predict the labels of future samples (test dataset) with the assumption that it is sampled from the same domain. In contrast to this, an RL agent learns from the samples that are collected from the trajectories generated by its sequential interaction with the system. For an RL agent, the trajectory consists of a series of sequential interactions whereby it transitions from one state to another following some dynamics intrinsic to the environment while collecting the reward till some stopping condition is reached. This is known as an episode. Here, for an action i_t taken by the agent at the

t-th timestep, the agent transitions from its current state denoted by $S_{i,t}$ to state $S_{i,t+1}$ and observes the reward $X_{i,t}$. An illustrative image depicting the reinforcement learning scenario is shown in Figure 1.

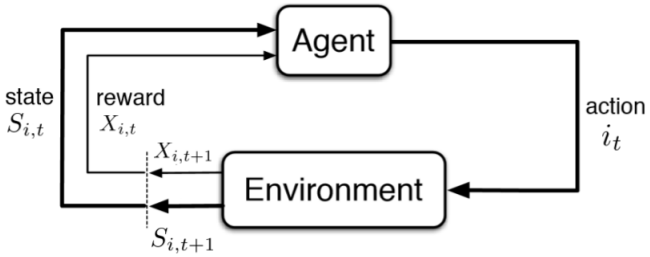

Figure 1: Reinforcement Learning

Now, for a single-step interaction, i.e., when the episode terminates after a single transition, the problem is captured by the multi-armed bandit (MAB) model. Infact the MAB model can be considered as a single looping state when the agent after taking an action, and observing the reward transitions back to the same state. That single looping state consist of several finite number of actions which are called as arms.

The name bandit originated from the concept of casino slot machine where there are levers which are termed as arms and the learner can pull one lever and observe the reward associated with that arm which is sampled from a distribution associated with the specific arm. This game is repeated T times and the goal of the learner is to maximize its profit.

2 Motivation

The MAB model fits very well in various real-world scenarios that can be modeled as sequential decision-making problems. Some of which are mentioned as follows:-

1. *Online Shop Domain:* In the online shop domain (Ghavamzadeh *et al.*, 2015), a retailer aims to maximize profit by sequentially suggesting products to online shopping customers. In this scenario, at every timestep, the retailer displays an item to a customer from a pool of items which has the highest probability of being selected by the customer. The episode ends when the customer selects or does not select a product (which will be considered as a loss to the retailer). This feedback is incorporated by the learner as a feedback from the environment and it modifies

its policy for the next suggestion. This process is repeated till a pre-specified number of times with the retailer gathering valuable information regarding the customer from this behaviour and modifying its policy to display other items to different customers.

2. *Medical Treatment Design:* Another interesting domain that MAB model was first studied was for the medical treatment design (Thompson, 1933),(Thompson, 1935). Here at every timestep, the learner chooses to administer one out of several treatments sequentially on a stream of patients who are suffering from the same ailment (say). Let's also assume that there is a single treatment which will be able to alleviate the patients from their disease. Here, the episode ends when the patient responds well or does not respond well to the treatment whereby the learner modifies its policy for the suggestion to the next patient. The goal of the learner is to quickly converge on the best treatment so that whenever a new patient comes with the same ailment, the learner can suggest the best treatment which can relive the patient of its ailment with a high probability.

3. *Financial Portfolio Management:* In financial portfolio management MAB models can also be used. Here, the learner is faced with the choice of selecting the most profitable stock option out of several stock options. The simplest strategy where we can employ a bandit model is this; at the start of every trading session the learner suggests a stock to purchase worth Re 1, while at the closing of the trading session it sells off the stock to witness its value after a day's trading. The profit recorded is treated as the reward revealed by the environment and the learner modifies its policy for the next day. Let's assume that no new stock options are being introduced over the considered time horizon and there is a single best stock option which if selected for perpetuity will always give the best returns. Then, the goal of the learner is reduced to identifying the best stock option as quickly as possible.

4. *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 learner must suggest the product variant(s) whose qualities 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).

5. *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 learner has to explore as many channels as possible to suggest the best possible set of channel(s) whose qualities are above a particular threshold τ. The threshold may depend on the subscription level of the customer such that 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 set of suggestions within a very small number of attempts.

6. *Anomaly Detection and Classification:* MABs can also be used for anomaly detection where the goal is to seek out extreme values in the data. Anomalies may not always be naturally concentrated which was shown in Steinwart *et al.* (2005). To implement a MAB model the best possible way is to define a cut-off level τ and classify the samples above this level τ as anomalous along with a tolerance factor which gives it a degree of flexibility. Such an approach has already been mentioned in Streeter and Smith (2006) and further studied in Locatelli *et al.* (2016).

3 Objectives and Scope of Thesis

The main objectives of the thesis are as follows:-

1. The first objective of this thesis is to study the area of stochastic multi-armed bandit (SMAB) and how to minimize cumulative regret in this setup. We intend to give strong gap-dependent and gap-independent regret guarantees in the SMAB setting. We also intend to provide algorithm in the SMAB setting that outperforms the current state-of-the-art algorithms in this setting.

2. The second objective of this thesis is to study the area of thresholding bandit problem (TBP) setting where the goal is to minimize the expected loss at the end of a fixed budget provided as input. We intend to provide strong guarantees with respect to expected loss and also propose algorithm that does not require any problem complexity as an input. We also intend to provide strong empirical evaluations of the algorithm proposed for the TBP setting.

4 Contributions of Thesis

The main contributions of the thesis are as follows:-

1. We proposed a novel algorithm for the stochastic multi-armed bandit (MAB) problem. Our proposed Efficient UCB Variance method, referred to as EUCBV is an arm elimination algorithm based on UCB-Improved and UCBV strategy which takes into account the empirical variance of the arms and along with aggressive exploration factors eliminate sub-optimal arms. Through a theoretical analysis, we establish that EUCBV achieves a better gap-dependent regret upper bound than UCB-Improved, MOSS, UCB1, and UCBV algorithms. EUCBV enjoys an order optimal gap-independent regret bound same as that of OCUCB and MOSS, and better than UCB-Improved, UCB1 and UCBV. Empirically, in several considered environments EUCBV outperforms most of the state-of-the-art algorithms.

2. We proposed the Augmented-UCB (AugUCB) algorithm for a fixed-budget version of the thresholding bandit problem (TBP), where the objective is to identify a set of arms whose quality is above a threshold. A key feature of AugUCB is that it uses both mean and variance estimates to eliminate arms that have been sufficiently explored. This is the first algorithm to employ such an approach for the considered TBP. Furthermore, in numerical evaluations we establish in several considered environments that AugUCB outperforms all the algorithms that does not take into consideration the variance of the arms in their action selection strategy.

5 Summary of the Research Work

We divided the research work done into two parts, in the first part we study the stochastic multi-armed bandit setting and propose an order optimal algorithm for this setting. In the second part we study the thresholding bandit problem and propose our solution to this problem. All these have been briefly discussed below.

5.1 Efficient UCB Variance: An almost optimal algorithm in Stochastic Multi-Armed Bandit setting

In this part, we deal with the stochastic multi-armed bandit (SMAB) setting. In its classical form, stochastic MABs represent a sequential learning problem where a learner is exposed to a finite set of actions (or arms) and needs to choose one of the actions at each timestep. After choosing (or pulling) an arm the learner receives a reward, which is conceptualized as an independent random draw from stationary distribution associated with the selected arm. Also, note that in SMAB, the distribution associated with each arm is fixed throughout the entire duration of the horizon denoted by T.

In SMAB setting, the learner seeks to identify the optimal arm as quickly as possible to maximize its rewards. In the pursuit of this, the learner faces the task of balancing exploitation and exploration. In other words, should the learner pull the arm which currently has the best-known estimates (exploit) or explores arms more thoroughly to ensure that a correct decision is being made. This is termed as the *exploration-exploitation dilemma*, one of the fundamental challenges of reinforcement learning. The objective

of the learner in the SMAB setting is to maximize his rewards or in other words, to minimize the cumulative regret, which is defined as follows:

$$R_T = r^*T - \sum_{i=1}^{K} r_i n_i(T),$$

where T is the number of timesteps, and $z_i(T)$ is the number of times the algorithm has chosen arm i up to timestep T. The expected regret of an algorithm after T timesteps can be written as,

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

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

5.1.1 Contributions

We propose the Efficient-UCB-Variance (henceforth referred to as EUCBV) algorithm for the stochastic MAB setting. EUCBV combines the approach of UCB-Improved (Auer and Ortner, 2010) , CCB (Liu and Tsuruoka, 2016) and UCBV (Audibert *et al.*, 2009) algorithms. EUCBV, by virtue of taking into account the empirical variance of the arms, exploration parameters and non-uniform arm selection (as opposed to UCB-Improved), performs significantly better than the existing algorithms in the stochastic MAB setting. EUCBV outperforms UCBV which also takes into account empirical variance but is less powerful than EUCBV because of the usage of exploration regulatory factor by EUCBV. Also, we carefully design the confidence interval term with the variance estimates along with the pulls allocated to each arm to balance the risk of eliminating the optimal arm against excessive optimism. Theoretically we refine the analysis of Auer and Ortner (2010) and prove that for $T \geq K^{2.4}$ our algorithm is order optimal and achieves a worst case gap-independent regret bound of $O\left(\sqrt{KT}\right)$ which is same as that of MOSS (Audibert and Bubeck, 2009) and OCUCB (Lattimore, 2015) but better than that of UCBV, UCB1 (Auer *et al.*, 2002) and UCB-Improved. Here, K is the total number of arms and T is the total number of available timesteps, termed as horizon. Also, the gap-dependent regret bound of EUCBV is better than UCB1, UCB-

Table 1: Regret upper bound of different algorithms

Algorithm	Gap-Dependent	Gap-Independent
EUCBV	$O\left(\frac{K\sigma^2_{\max}\log(\frac{T\Delta^2}{K})}{\Delta}\right)$	$O\left(\sqrt{KT}\right)$
UCB1	$O\left(\frac{K\log T}{\Delta}\right)$	$O\left(\sqrt{KT\log T}\right)$
UCBV	$O\left(\frac{K\sigma^2_{\max}\log T}{\Delta}\right)$	$O\left(\sqrt{KT\log T}\right)$
UCB-Imp	$O\left(\frac{K\log(T\Delta^2)}{\Delta}\right)$	$O\left(\sqrt{KT\log K}\right)$
MOSS	$O\left(\frac{K^2\log(T\Delta^2/K)}{\Delta}\right)$	$O\left(\sqrt{KT}\right)$
OCUCB	$O\left(\frac{K\log(T/H_i)}{\Delta}\right)$	$O\left(\sqrt{KT}\right)$

Improved and MOSS but is poorer than OCUCB. However, EUCBV's gap-dependent bound matches OCUCB in the worst case scenario when all the gaps are equal. Through our theoretical analysis we establish the exact values of the exploration parameters for the best performance of EUCBV. Our proof technique is highly generic and can be easily extended to other MAB settings. An illustrative table containing the bounds is provided in Table 1.

Empirically, we show that EUCBV, owing to its estimating the variance of the arms, exploration parameters and non-uniform arm pull, performs significantly better than MOSS, OCUCB, UCB-Improved, UCB1, UCBV, TS (Thompson, 1933),(Agrawal and Goyal, 2012), BU (Kaufmann *et al.*, 2012), DMED (Honda and Takemura, 2010), KLUCB (Garivier and Cappé, 2011) and Median Elimination (Even-Dar *et al.*, 2006) algorithms. Note that except UCBV, TS, KLUCB and BU (the last three with Gaussian priors) all the aforementioned algorithms do not take into account the empirical variance estimates of the arms. Also, for the optimal performance of TS, KLUCB and BU one has to have the prior knowledge of the type of distribution, but EUCBV requires no such prior knowledge. EUCBV is the first arm-elimination algorithm that takes into account the variance estimates of the arm for minimizing cumulative regret and thereby answers an open question raised by Auer and Ortner (2010), where the authors conjectured that an UCB-Improved like arm-elimination algorithm can greatly benefit by

taking into consideration the variance of the arms. Also, it is the first algorithm that follows the same proof technique of UCB-Improved and achieves a gap-independent regret bound of $O\left(\sqrt{KT}\right)$ thereby, closing the gap of UCB-Improved which achieved a gap-independent regret bound of $O\left(\sqrt{KT\log K}\right)$.

5.1.2 The EUCBV algorithm

Algorithm 1 EUCBV

Input: Time horizon T, exploration parameters ρ and ψ.
Initialization: Set $m := 0$, $B_0 := \mathcal{A}$, $\epsilon_0 := 1$, $M = \left\lfloor \frac{1}{2}\log_2 \frac{T}{e} \right\rfloor$, $n_0 = \left\lceil \frac{\log(\psi T\epsilon_0^2)}{2\epsilon_0} \right\rceil$ and $N_0 = Kn_0$.
Pull each arm once
for $t = K+1, .., T$ **do**
Pull arm $i \in \arg\max_{j\in B_m}\left\{\hat{r}_j + \sqrt{\frac{\rho(\hat{v}_j+2)\log(\psi T\epsilon_m)}{4z_j}}\right\}$, where z_j is the number of times arm j has been pulled.
Arm Elimination by Mean Estimation
For each arm $i \in B_m$, remove arm i from B_m if,

$$\hat{r}_i + \sqrt{\frac{\rho(\hat{v}_i+2)\log(\psi T\epsilon_m)}{4z_i}} < \max_{j\in B_m}\left\{\hat{r}_j - \sqrt{\frac{\rho(\hat{v}_j+2)\log(\psi T\epsilon_m)}{4z_j}}\right\}$$

if $t \geq N_m$ and $m \leq M$ **then**
Reset Parameters
$\epsilon_{m+1} := \frac{\epsilon_m}{2}$; $B_{m+1} := B_m$
$n_{m+1} := \left\lceil \frac{\log(\psi T\epsilon_{m+1}^2)}{2\epsilon_{m+1}} \right\rceil$
$N_{m+1} := t + |B_{m+1}|n_{m+1}$; $m := m+1$

end if
Stop if $|B_m| = 1$ and pull $i \in B_m$ till T is reached.
end for

The algorithm: Earlier round-based arm elimination algorithms like Median Elimination (Even-Dar *et al.*, 2006) and UCB-Improved mainly suffered from two basic problems:

(i) *Initial exploration:* Both of these algorithms pull each arm equal number of times in each round, and hence waste a significant number of pulls in initial explorations.

(ii) *Conservative arm-elimination:* In UCB-Improved, arms are eliminated conservatively, i.e, only after $\epsilon_m < \frac{\Delta_i}{2}$, where the quantity ϵ_m is initialized to 1 and halved after every round. In the worst case scenario when K is large, and the gaps are uniform

($r_1 = r_2 = \cdots = r_{K-1} < r^*$) and small this results in very high regret.

The EUCBV algorithm, which is mainly based on the arm elimination technique of the UCB-Improved algorithm, remedies these by employing exploration regulatory factor ψ and arm elimination parameter ρ for aggressive elimination of sub-optimal arms. Along with these, similar to CCB (Liu and Tsuruoka, 2016) algorithm, EUCBV uses optimistic greedy sampling whereby at every timestep it only pulls the arm with the highest upper confidence bound rather than pulling all the arms equal number of times in each round. Also, unlike the UCB-Improved, UCB1, MOSS and OCUCB algorithms (which are based on mean estimation) EUCBV employs mean and variance estimates (as in Audibert *et al.* (2009)) for arm elimination. Further, we allow for arm-elimination at every time-step, which is in contrast to the earlier work (e.g., Auer and Ortner (2010); Even-Dar *et al.* (2006)) where the arm elimination takes place only at the end of the respective exploration rounds.

5.2 Augmented UCB for Thresholding Bandits Problem

In this part we study another setting called pure-exploration thresholding multi-armed bandits which are unlike their traditional (exploration vs. exploitation) counterparts, the SMABs, where the objective is to minimize the cumulative regret. In pure-exploration problems a learning algorithm, until time T, can invest entirely on exploring the arms without being concerned about the loss incurred while exploring; the objective is to minimize the probability that the arm recommended at time T is not the best arm. In this chapter, we further consider a combinatorial version of the pure-exploration MAB, called the thresholding bandit problem (TBP). Here, 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. It is important to emphasize that the *thresholding* bandit problem is different from the *threshold* bandit setup studied in Abernethy *et al.* (2016), where the learner receives an unit reward whenever the value of an observation is above a threshold.

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 threshold 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 a learning algorithm (under consideration) 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 mis-classification* (i.e., error), that occurs either if a good arm is rejected or a bad arm is accepted as a good one.

5.2.1 Contributions

We propose the Augmented UCB (AugUCB) algorithm for the fixed-budget setting of a specific combinatorial, pure-exploration, stochastic MAB called the thresholding bandit problem. AugUCB essentially combines the approach of UCB-Improved, CCB (Liu and Tsuruoka, 2016) and APT (Locatelli *et al.*, 2016) algorithms. Our algorithm takes into account the empirical variances of the arms along with mean estimates; to the best of our knowledge this is the first variance-based algorithm for the considered TBP. Thus, we also address an open problem discussed in Auer and Ortner (2010) of designing an algorithm that can eliminate arms based on variance estimates. In this regard, note that both CSAR (Chen *et al.*, 2014) and APT are not variance-based algorithms.

Our theoretical contribution comprises proving an upper bound on the expected loss incurred by AugUCB. In Table 2 we compare the upper bound on the losses incurred by the various algorithms, including AugUCB. The terms H_1, H_2, $H_{CSAR,2}, H_{\sigma,1}$ and $H_{\sigma,2}$ represent various problem complexities. We note that, for all $K \geq 8$, we have

$$\log\left(K \log K\right) H_{\sigma,2} > \log(2K) H_{\sigma,2} \geq H_{\sigma,1}.$$

Thus, it follows that the upper bound for UCBEV (Gabillon *et al.*, 2011) is better than that for AugUCB. However, implementation of UCBEV algorithm requires $H_{\sigma,1}$ as input, whose computation is not realistic in practice. In contrast, our AugUCB algorithm requires no such complexity factor as input. Proceeding with the comparisons, we emphasize that the upper bound for AugUCB is, in fact, not comparable with that of APT and CSAR; this is because the complexity term $H_{\sigma,2}$ is not explicitly comparable with either H_1 or $H_{CSAR,2}$. However, through extensive simulation experiments we find that AugUCB significantly outperforms both APT, CSAR and other non variance-based algorithms. AugUCB also outperforms UCBEV under explorations where non-optimal values of $H_{\sigma,1}$ are used. In particular, we consider experimental scenarios comprising large number of arms, with the variances of arms in S_τ being large. AugUCB, being variance based, exhibits superior performance under these settings.

5.2.2 The Augmented UCB algorithm

The algorithm: The Augmented-UCB (AugUCB) algorithm is presented in Algorithm 2. 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

Table 2: AugUCB vs. State of the art

Algorithm	Upper Bound on Expected Loss
AugUCB	$\exp\left(-\frac{T}{4096\log(K\log K)H_{\sigma,2}}+\log(2KT)\right)$
UCBEV	$\exp\left(-\frac{1}{512}\frac{T-2K}{H_{\sigma,1}}+\log(6KT)\right)$
APT	$\exp\left(-\frac{T}{64H_1}+2\log((\log(T)+1)K)\right)$
CSAR	$\exp\left(-\frac{T-K}{72\log(K)H_{CSAR,2}}+2\log(K)\right)$

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 details 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).

6 Conclusions

In this thesis, we studied two complex bandit problems, the stochastic multi-armed bandit (SMAB) with the goal of cumulative regret minimization and pure exploration stochastic thresholding bandit problem (TBP) with the goal of expected loss minimization. For the first problem, we devised a novel algorithm called Efficient UCB Variance (EUCBV) which enjoys an order optimal regret bound and performs superbly in diverse stochastic environments. In the second part, the thresholding bandit problem, we came up with the novel algorithm called Augmented UCB (AugUCB) which is the first algorithm to use variance estimation for the considered TBP setting and also empirically outperforms most of the other algorithms.

Algorithm 2 AugUCB

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

$$M = \left\lfloor \frac{1}{2}\log_2\frac{T}{e} \right\rfloor;\ \ \psi_0 = \frac{T\epsilon_0}{128\Big(\log(\frac{3}{16}K\log K)\Big)^2}; \ell_0 = \left\lceil \frac{2\psi_0\log(T\epsilon_0)}{\epsilon_0} \right\rceil;\ \ N_0 = K\ell_0$$

Pull each arm once

for $t = K+1, .., T$ **do**
  Pull arm $j \in \arg\min_{i\in B_m}\Big\{|\hat{r}_i - \tau| - 2s_i\Big\}$
  $t \leftarrow t+1$
  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
    $\epsilon_{m+1} \leftarrow \frac{\epsilon_m}{2}$; $B_{m+1} \leftarrow B_m$
    $\psi_{m+1} \leftarrow \frac{T\epsilon_{m+1}}{128(\log(\frac{3}{16}K\log K))^2}$; $\ell_{m+1} \leftarrow \left\lceil \frac{2\psi_{m+1}\log(T\epsilon_{m+1})}{\epsilon_{m+1}} \right\rceil$
    $N_{m+1} \leftarrow t + |B_{m+1}|\ell_{m+1}$; $m \leftarrow m+1$
  end if
end for
Output: $\hat{S}_\tau = \{i : \hat{r}_i \geq \tau\}$.

References

1. **Abernethy, J. D.**, **K. Amin**, and **R. Zhu**, Threshold bandits, with and without censored feedback. *In Advances In Neural Information Processing Systems*. 2016.

2. **Agrawal, S.** and **N. Goyal**, Analysis of thompson sampling for the multi-armed bandit problem. *In COLT*. 2012.

3. **Audibert, J.-Y.** and **S. Bubeck**, Minimax policies for adversarial and stochastic bandits. *In COLT*. 2009.

4. **Audibert, J.-Y.** and **S. Bubeck**, Best arm identification in multi-armed bandits. *In COLT-23th Conference on Learning Theory-2010*. 2010.

5. **Audibert, J.-Y.**, **R. Munos**, and **C. Szepesvári** (2009). Exploration–exploitation trade-off using variance estimates in multi-armed bandits. *Theoretical Computer Science*, **410**(19), 1876–1902.

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

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

8. **Bertsekas, D. P.** and **J. N. Tsitsiklis** (1996). Neuro-dynamic programming (optimization and neural computation series, 3). *Athena Scientific*, **7**, 15–23.

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

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

11. **Even-Dar, E.**, **S. Mannor**, and **Y. Mansour** (2006). Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *The Journal of Machine Learning Research*, **7**, 1079–1105.

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

13. **Garivier, A.** and **O. Cappé** (2011). The kl-ucb algorithm for bounded stochastic bandits and beyond. *arXiv preprint arXiv:1102.2490*.

14. **Ghavamzadeh, M.**, **S. Mannor**, **J. Pineau**, **A. Tamar**, *et al.*, *Bayesian reinforcement learning: a survey*. World Scientific, 2015.

15. **Honda, J.** and **A. Takemura**, An asymptotically optimal bandit algorithm for bounded support models. *In COLT*. Citeseer, 2010.

16. **Kaufmann, E.**, **O. Cappé**, and **A. Garivier**, On bayesian upper confidence bounds for bandit problems. *In AISTATS*. 2012.

17. **Lattimore, T.** (2015). Optimally confident ucb: Improved regret for finite-armed bandits. *arXiv preprint arXiv:1507.07880*.

18. **Liu, Y.-C.** and **Y. Tsuruoka** (2016). Modification of improved upper confidence bounds for regulating exploration in monte-carlo tree search. *Theoretical Computer Science*.

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

20. **Steinwart, I.**, **D. Hush**, and **C. Scovel** (2005). A classification framework for anomaly detection. *Journal of Machine Learning Research*, **6**(Feb), 211–232.

21. **Streeter, M. J.** and **S. F. Smith** (2006). Selecting among heuristics by solving thresholded k-armed bandit problems. *ICAPS 2006*, 123.

22. **Sutton, R. S.** and **A. G. Barto**, *Reinforcement learning: An introduction*. MIT press, 1998.

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

24. **Thompson, W. R.** (1935). On the theory of apportionment. *American Journal of Mathematics*, **57**(2), 450–456.

7 Proposed Contents of the Thesis

In chapter 1 (table 3), we discuss on Reinforcement Learning and its connection to bandits, we give an overview of the various types of bandits available in the literature and also discuss about the main objectives of the thesis and our contributions.

1. Introduction to Bandits
1.1 Reinforcement Learning
1.2 Connection between Reinforcement Learning and Bandits
1.3 Why study Bandits?
1.4 Motivation
1.5 Types of Information Feedback
1.6 Different types of Bandits
1.7 Objectives of Thesis
1.8 Contributions of Thesis
1.9 Outline of the Thesis

Table 3: Introduction to thesis

In chapter 2 (table 4) we give a detailed overview of the stochastic multi-armed bandit setting and the latest available algorithms in this setting. In the next chapter 3 (table 4) we introduce our algorithm Efficient UCB Variance (EUCBV) for the stochastic multi-armed bandit setting. We give theoretical guarantees on the performance of EUCBV and also show in numerical simulations that it indeed performs very well as compared to the state-of-the-art algorithms.

2. Stochastic Multi-armed Bandits	**3. Efficient UCB Variance: An almost optimal algorithm in SMAB setting**
2.1 Introduction to SMAB	3.1 Introduction
2.2 Notations and assumptions	3.2 Our Contributions
2.3 Problem Definition	3.3 Algorithm: Efficient UCB Variance
2.4 Motivation	3.4 Main Results
2.5 Related Work in SMAB	3.5 Proofs
2.6 Summary	3.6 Experiments
	3.7 Summary
	3.8 Appendix B

Table 4: Part 1. Stochastic MAB problem

In the subsequent chapter 4 (table 5) we introduce a new variant of pure exploration multi-armed stochastic bandit called the thresholding bandit problem. We analyze the

connections between thresholding bandit problem and pure exploration problem and also discuss several existing algorithms in both the settings that are relevant to carefully analyze the thresholding bandit problem. Then in chapter 5 (table 5) we introduce our solution for the thresholding bandit problem, called the Augemented UCB (AugUCB) algorithm. We analyze our algorithm AugUCB and derive theoretical guarantees for it as well as show in numerical experiments that it indeed outperforms several state-of-the-art algorithms in the thresholding bandit setting.

4. Thresholding Bandits	**5. Augmented UCB for Thresholding Bandit Problem**
4.1 Introduction to TBP	5.1 Introduction
4.2 Notations and assumptions	5.2 Our Contributions
4.3 Problem Definition	5.3 Augmented-UCB Algorithm
4.4 Motivation	5.4 Theoretical Results
4.5 Related Work in Pure Exploration	5.5 Numerical Experiments
4.6 TBP connection to Pure Exploration	5.6 Summary
4.7 Related Work in TBP	
4.8 Summary	

Table 5: Part 2. Thresholding Bandit Problem

Finally, in chapter 6, we conclude with a brief summarization on the work done in the thesis and discuss future directions.

8 Publications

8.1 Papers in Refereed Journals

8.2 Presentations in Conferences

1. Presented *Thresholding Bandit with Augmented UCB* at the **Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017**, Melbourne, Australia, August 19-25.

2. To present *Efficient UCBV: An Almost Optimal Algorithm using Variance Estimates* at the **Thirty-Second Association for the Advancement of Artificial Intelligence, AAAI 2018**, New Orleans, Louisiana, USA, February 2-7.