Comparative Evaluation of Asymptotically Optimal and Standard Upper Confidence Bound Algorithms in Multi-Armed Bandit Scenarios

Qiuyuan Lyu *

School of Computer science, Beijing University of Posts and Telecommunications, Beijing, 100876, China

* Corresponding Author Email: lvqiuyuan1101@bupt.edu.cn

Abstract. The Multi-Armed Bandit (MAB) problem stands as a cornerstone challenge in the realm of reinforcement learning, particularly in contexts where an agent must sequentially choose actions from multiple options to maximize cumulative rewards, often analogized to pulling levers on slot machines. A plethora of algorithms have been developed and rigorously examined to tackle this issue, each aiming to enhance the process of reward collection. Nonetheless, a significant research gap persists, notably in relation to the Upper Confidence Bound algorithm (UCB). This study delves into the intricacies of UCB and its asymptotically optimal variant, focusing on their performance within a movie recommendation system context. Employing this real-world application as an illustrative example, the analysis seeks to discern the nuanced distinctions between the standard UCB and its asymptotically optimal counterpart. Through detailed exploration and comparison, insights emerge regarding the influence of different parameter values on the UCB algorithm's efficacy. This in-depth examination provides a more profound comprehension of how variations in these parameters affect the algorithm's capacity to balance exploration and exploitation, a key element in effective decision-making under uncertainty. Contrary to prevailing assumptions, the findings suggest that the asymptotically optimal UCB algorithm does not universally surpass the standard UCB algorithm in practical applications. This revelation calls for a reconsideration of the effectiveness of these algorithms, highlighting the importance of a nuanced evaluation of their performance across various scenarios.

Keywords: Multi-armed bandit; Upper confidence bound algorithm; Movie recommendation.

1. Introduction

Decision-making is integral to various life aspects, ranging from consumer choices in purchasing goods to strategic maneuvers in games, and extending to movie recommendations. The complexity of these decisions often mirrors the challenges in formulating optimal strategies. Historically, decision strategies commonly involved randomly allocating an equal number of samples to each option, then observing to determine the most successful one. While effective, this approach can be suboptimal in fields like marketing or medicine, where it may significantly diminish customer engagement or patient participation during testing phases.

In scenarios with limited testing populations, random allocation may not be the most efficient strategy. The imperative to maximize information gain or optimize outcomes necessitates alternative methodologies. The uncertainty about the efficacy of each alternative prior to testing complicates the decision-making process. During testing, choices are meticulously compared, incurring a cost associated with this learning process. This cost can vary, manifesting as more compelling marketing content or superior medical care for certain test subjects. Despite these challenges, the goal remains to reach and impact the maximum number of customers, patients, or movie enthusiasts.

Typically, decision-making processes unfold sequentially within a finite timeframe. In such cases, the application of a multi-armed bandit reinforcement learning model is a promising approach. This model provides a framework for systematically studying and optimizing decision-making processes where the outcomes of each choice influence future selections. It navigates the balance between exploration and exploitation, aiming to maximize rewards in scenarios with time-limited decision-making.

In summary, employing a multi-armed bandit reinforcement learning model offers an adaptive and sophisticated approach to decision-making in contexts involving sequential choices and constrained durations. This framework is invaluable in developing strategies that learn from the uncertainties of testing and aim to achieve the primary objective of benefiting the broadest possible audience.

2. Theoretical Foundations

2.1. Conceptualizing Multi-Armed Bandits

2.1.1. Problem description

In the multi-armed bandit problem, the challenge lies in optimizing decisions when faced with a slot machine equipped with K pull bars, each associated with a distinct probability distribution, denoted as R, governing the potential rewards [1]. The inherent uncertainty stems from the unknown reward probability distribution for each pull bar. With each pull, a reward, denoted as r is drawn from the corresponding distribution. The primary objective is to maximize the cumulative reward over a series of T operations [2].

The crux of the dilemma lies in striking a delicate balance between exploration and exploitation. Exploration involves delving into the unknown, seeking to understand the probability distributions associated with each pull bar. On the other hand, exploitation revolves around leveraging existing knowledge and favoring pull bars that have yielded favorable results in the past.

Initiating the process from scratch adds an additional layer of complexity. The lack of prior information necessitates a strategic approach to simultaneously explore the uncharted territories of each pull bar's reward probability distribution while exploiting the knowledge gained through prior pulls. This delicate equilibrium is crucial for achieving the overarching goal of obtaining the highest cumulative reward within the specified number of operations.

Effectively addressing this exploration-exploitation trade-off requires sophisticated algorithms. Strategies such as the Upper Confidence Bound algorithm come into play, dynamically adjusting the balance between exploration and exploitation based on accumulated informatαion [3]. The algorithm assigns confidence bounds to each pull bar's estimated reward distribution, allowing for a systematic exploration of uncertain options while gradually favoring the arms that appear to be more rewarding.

2.1.2. Formal description

The multi-arm slot machine problem can be represented as a tuple$\langle A, R\rangle$, where:

A is a set of actions, one of which represents pulling a pull rod. If the multi-arm slot machine has a total of K pull bars, the action space is the set$\{a_1, \ldots, a_k\}$, we use $a_t \in A$ to represent any action;

R is the reward probability distribution, the action of pulling each tie rod a corresponds to a reward probability distribution$R(r|a)$, and the reward distribution of different tie rods is usually different.

Assuming that only one pull bar can be pulled per time move, the goal of a multi-armed slot machine is to maximize the rewards accumulated over a period of T time moves:$\max \sum_{t=1}^{T} r_t$, $r_t \sim R(\cdot|a_t)$, a_t represents the action of pulling a pull rod in the t time step, and r_t represents the reward obtained by the action a_t [4].

2.1.3. Accumulation of remorse

For each action, existing theories define its expected reward as $Q(a) = E_{r\sim R(\cdot|a)}[r]$ [5]. Therefore, there is at least one pull bar whose expected reward is not less than that of pulling any other pull bar, and they express this optimal expected reward as $Q^* = \max_{a\in A} Q(a)$ [6].

In order to more intuitively and conveniently observe the difference between the expected reward of pulling a pull rod and the expected reward of the optimal pull rod, they introduce the concept of regret. Regret is defined as the difference between the action a of pulling the current pull rod and the expected reward of the optimal pull rod,$R(a) = Q^* - Q(a)$ Cumulative regret T is the total amount of regret accumulated after the operation of the pull rod, for a complete T step

decision$\{a_1, a_2, \ldots, a_T\}$, cumulative regret is $\sigma_R = \sum_{t=1}^{T} R(a_t)$ [7]. The goal of the MAB problem is to maximize cumulative reward, which is equivalent to minimizing cumulative remorse.

2.1.4. Estimate expected rewards

In order to know which pull bar gets the higher reward, it is in need to estimate the expected reward for pulling this pull bar. Due to the randomness of the reward obtained by pulling the pull rod only once, it is necessary to pull a pull rod several times, and then calculate the expectation of the obtained multiple rewards. The algorithm flow is shown as follows.

For $\forall a \in A$, initialize the counter N (a) =0 and expect a reward estimate $\widehat{Q}(a) = 0$

For $t = 1 \rightarrow T$ do

Select a tie rod, the action is denoted as a_t

Get a reward r_t

Update counter: N (a_t) = N (a_t) + 1

Updated expected reward valuation: $\widehat{Q}(a_t) = \widehat{Q}(a_t) + \frac{1}{N(a_t)}\left[r_t - \widehat{Q}(a_t)\right]$

End for

2.2. Overview of UCB Algorithm Variants

Consider the following scenario: a two-arm slot machine with a single pull of the first pull rod, which results in a reward of zero; multiple pulls of the second pull bar, which results in a reward distribution that is generally understood. Then, someone may try pulling the first lever again to see how the rewards are distributed. Since the first pull rod has only been pulled once and there is a lot of uncertainty surrounding it, this idea is primarily based on uncertainty. A pull bar's exploration value increases with its degree of uncertainty because further investigation may reveal a high expected reward. Here, we present the uncertainty metric U (a), which gets smaller the more times an action is tried. The main challenge is estimating the uncertainty. We can combine the current expected reward valuation and uncertainty using an indeterminacy-based approach.

3. Algorithmic Analysis and Empirical Research

Asymptotically optimal UCB vs. Standard UCB: Theoretical Underpinnings.

The upper confidence bound (UCB) algorithm is a classical strategy algorithm based on uncertainty, and its idea uses a very famous mathematical principle: Hoeffding's inequality. In Hovding's inequality, $X_1, X_2, \ldots, X_n$ are n independent and equally distributed random variables with a value range of $[0, 1]$, and its empirical expectation is $\overline{x_n} = \frac{1}{n}\sum_{j=1}^{n} X_j$, then

$$P\{E[X] \geq \overline{x_n} + u\} \leq e^{-2nu^2} \tag{1}$$

Now we apply Hoeffding's inequality to the multi-arm slot machine problem. Plug $\widehat{Q_t}(a)$ into$\overline{x_t}$. The parameters $u = \widehat{U_t}(a)$ in the inequality represent the measure of uncertainty [8]. Given a probability$p = e^{-2N_t(a)U_t(a)^2}$, according to the above inequality, $Q_t(a) < \widehat{Q_t}(a) + \widehat{U_t}(a)$ at least holds the probability$1 - p$. When p is very small, it is true $Q_t(a) < \widehat{Q_t}(a) + \widehat{U_t}(a)$ with a high probability, so $\widehat{Q_t}(a) + \widehat{U_t}(a)$ is the expected reward upper bound. At this point, the upper confidence bound algorithm selects the action that expects to reward the upper bound the most. Namely$\underset{a \in A}{\operatorname{argmax}}[\widehat{Q_t}(a) + \widehat{U_t}(a)]$. According to the equation,

$$\widehat{U_t}(a) = \sqrt{-\log(p)/2N_t(a)} \tag{2}$$

Therefore, once a probability p is set, the corresponding measure of uncertainty $\widehat{U_t}(a)$ can be calculated. More intuitively, the UCB algorithm first estimates the upper bound of the expected reward of each pull bar before selecting each pull bar, so that the expected reward of pulling each pull bar has only a small probability p of exceeding this upper bound, and then selects the pull bar

with the largest upper bound of the expected reward, so as to select the pull bar most likely to obtain the maximum expected reward.

Now, let $X_1,X_2,...,X_n$ are n independent and 1-subgaussian (which means that E[Xi]=0) and $\hat{\mu} = \sum_{t=1}^{n} \frac{X_t}{n}$, then $P(\hat{\mu} \geq \varepsilon) \leq e^{-\frac{n\varepsilon^2}{2}}$, Equating the right-hand side with δ and solving for ε leads to

$$P(\hat{\mu} \geq \sqrt{-2\log(1/\delta)/n}) \leq \delta \tag{3}$$

This analysis immediately suggests a definition of "as large as plausibly possible". Using the theory above, we can say that when the learner is deciding what to do in round t it has observed $T_i(t-1)$ samples from arm i and observed rewards with an empirical mean of $\hat{\mu}_i(t-1)$ for it. Then a good candidate for the largest plausible estimate of the mean for arm i is

$$\hat{\mu}_i(t-1) + \sqrt{2\log(1/\delta)/T_i(t-1)} \tag{4}$$

The value of $1-\delta$ is called the confidence level and different choices lead to different algorithms, each with their pros and cons, and sometimes different analysis. If we choose $1/\delta = n$, then algorithm is called the standard UCB. If we choose$1/\delta = 1 + t\log^2(t)$, t=1, 2... Then algorithm is called the asymptotically optimal UCB.

3.1. Comparative Performance Evaluation of UCB Variants

3.1.1. Standard UCB

Standard UCB algorithm is the basic form of UCB algorithm, which balances exploration and utilization to a certain extent. It encourages the selection of an unexplored action by adding a confidence upper bound term to improve the estimate of the action's value. However, the standard UCB can be overexplored in some cases, especially at the beginning.

3.1.2. Asymptotically Optimal UCB

Asymptotically optimal UCB is an improvement over the standard UCB, designed to overcome some of the limitations of the standard UCB. It introduces a correction factor that decreases as the number of rounds increases, thus placing more emphasis on exploration in the early stages of the algorithm and on exploitation in the later stages. This adjustment helps the algorithm better adapt to the changes in the problem and improves the long-term performance of the algorithm.

3.2. Insights into the Efficacy of UCB Algorithms

After a comparative evaluation of the performance of UCB algorithms, the author hopes to further explore the effectiveness of UCB algorithms and explore some key insights.

3.2.1. Trade-offs between exploration and exploitation

The success of UCB algorithm lies in its good trade-off between exploration and exploitation. By introducing a confidence upper bound, standard UCB encourages the selection of actions that are not frequently selected, thus better exploring the solution space of the problem in the early stages of the algorithm. Asymptotically optimal UCB further adjusts this balance, making it more flexible to adapt to changes in the problem during the evolution of the algorithm.

3.2.2. Adaptability to long-term tasks

A key advantage of UCB algorithms is their adaptability, especially when faced with long-term tasks. Asymptotically optimal UCB can better adapt to the changes of the problem at different stages by introducing correction factors. This adaptability is essential for dynamic environments in the real world, where the nature of the problem may change over time [8].

3.2.3. Robustness and robustness

The robustness of the UCB algorithm in different scenarios is also a concern. In some cases, the standard UCB may converge too quickly to certain actions, leading to overutilization. ile

asymptotically optimal UCB improves the robustness of the algorithm and slows down the overuse of certain actions by introducing the adjustment factor.

3.2.4. Sensitivity of algorithm parameters

Finally, we also need to pay attention to the sensitivity of the UCB algorithm to its parameters. For example, the choice of confidence upper bound and correction factor in UCB algorithm will affect the performance of the algorithm. By sensitivity analysis of these parameters, the algorithm can be better adjusted to meet the needs of different problems.

3.3. Case Study: Implementing UCB Algorithms in Movie Recommender Systems

Now, the author select Movie Lens as the dataset and it contains user ratings for movies. The author's main goal is to compare the performance of the following MAB algorithms: the standard UCB and the asymptotically optimal UCB algorithms. Put differently, we will explore the impact of the exploration bonus on the algorithm performance. The comparison will be in terms of the expected cumulative regret an algorithm incurs until round t, where t = 1, n. Here, n defines the horizon, i.e., the total number of rounds the algorithm is used. Depending on the data set, each movie genre can be an arm, and user ratings can be considered as the reward received when movie from a genre is rated by a user.

Choose n = 50, 000. The asymptotically optimal UCB algorithm uses the UCB index given as

$$\mathrm{UCB_i} = \hat{\mu}_i(t-1) + \frac{B}{2}\sqrt{2\log(f(t))/T_i(t-1)} \tag{5}$$

Where$f(t) = 1 + t(\log t)^2$. Here, B is the difference between the maximum possible reward value and the minimum possible reward value. For example, for the Movie Lens dataset where rewards (i.e., ratings) can be in the interval 1-5 (stars), B should be set as 4. Note that, we took B = 1 in the above since we assumed that rewards are 1-sub-Gaussian.

The standard UCB algorithm uses the index as

$$\mathrm{UCB_i} = \hat{\mu}_i(t-1) + \frac{B}{2}\sqrt{l\log(n)/T_i(t-1)} \tag{6}$$

Then the author will compare the performance of the asymptotically optimal UCB and the standard UCB with three different l values: l = 1, l = 2, and l = 4. The larger is the l value the more will the algorithm explore, while with smaller l it will more aggressively exploit.

The result is shown in the Fig1:

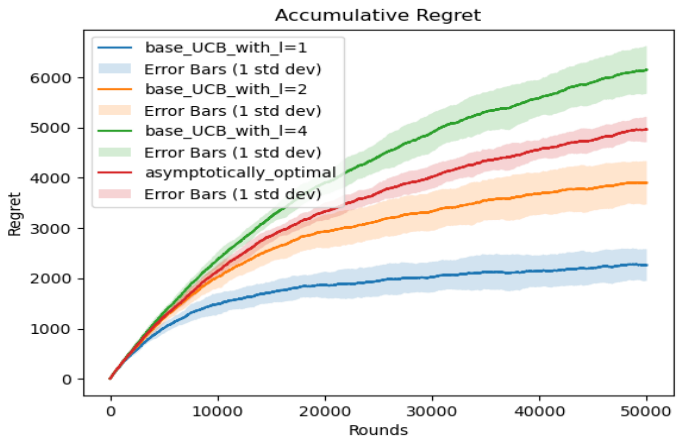

Fig 1. Comparative Analysis of Accumulative Regret in Multi-Armed Bandit Algorithms over Multiple Rounds (Photo/Picture credit: Original).

It can be seen that the comparison of cumulative regret values is as follows: the standard ucb when l=4 is greater than the asymptotically optimal ucb, the asymptotically optimal ucb is greater than the standard ucb when l=2, and the standard ucb when l=1 has the smallest cumulative regret value. This shows that the performance comparison of these four algorithms is: standard ucb when l=1 > standard

ucb when l=2 > asymptotically optimal ucb> standard ucb when l=4. This result is quite different from the author's previous estimate, which shows that in some cases too much exploration will not necessarily reduce the regret value, on the contrary, active use will greatly reduce the probability of accumulating high regret value. Moreover, it can be noted that the error bar of the asymptotically optimal ucb algorithm is the smallest, which indicates that the algorithm has higher stability

At the same time, the author compare the standard ucb when l=4 (represented by ucb in the Fig 2) and the asymptotically optimal ucb with the other two common algorithms explore-then-commit (ETC) and Thompson Sampling (TS) in the multi-armed bandit problem. The following results are obtained.

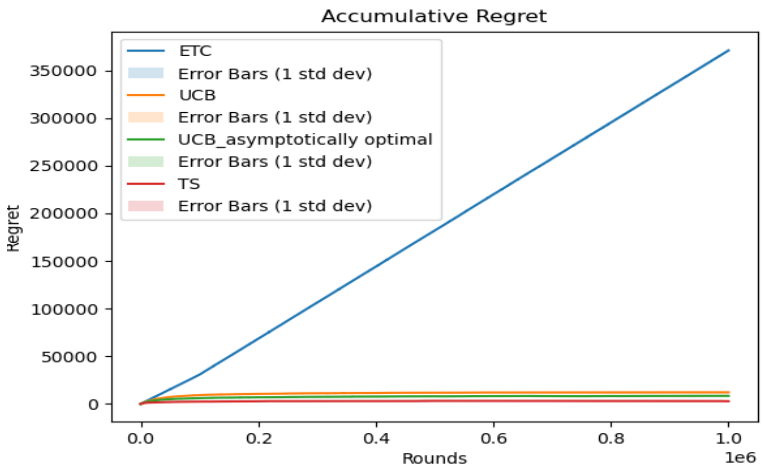

Fig 2. Comparative Analysis of Accumulative Regret in Multi-Armed Bandit Algorithms over Multiple Rounds (Photo/Picture credit: Original).

It can be found that when horizon n is close to infinity, the performance of the other three algorithms except etc is very close, and the logarithmic trend can be observed.

4. Challenges and Limitations

4.1. Sensitivity to Problem Characteristics:

One notable challenge is the sensitivity of both asymptotically optimal UCB and standard UCB algorithms to the characteristics of the underlying problem.

The performance of these algorithms may vary significantly based on the nature of the reward distributions associated with each arm. Challenges arise when the assumptions made by the algorithms do not align with the actual properties of the problem, potentially leading to suboptimal outcomes [9].

4.2. Exploration-Exploitation Dilemma:

The exploration-exploitation dilemma remains a fundamental challenge in Multi-Armed Bandit problems. While both UCB algorithms aim to balance exploration and exploitation, determining an optimal strategy in dynamic environments or scenarios with changing reward structures remains a non-trivial task. The challenge lies in adapting the algorithms to strike an appropriate balance under varying conditions [10].

4.3. Computational Complexity:

Another limitation to be considered is the computational complexity associated with these algorithms, especially in scenarios where real-time decision-making is crucial.

Both asymptotically optimal UCB and standard UCB involve calculations of confidence bounds and adjustment factors, which can be computationally demanding. This may hinder their application in resource-constrained environments [11].

4.4. Need for Adaptive Parameter Tuning:

Both UCB algorithms involve parameters such as exploration coefficients and adjustment factors. An ongoing challenge is the need for adaptive parameter tuning that can dynamically respond to changes in the problem environment. Selecting appropriate parameter values remains a task that demands careful consideration and often requires domain-specific knowledge [12].

5. Conclusion

This study presents a comparative analysis of the Asymptotically Optimal Upper Confidence Bound (AO-UCB) algorithm versus the standard Upper Confidence Bound algorithm within the framework of the Multi-Armed Bandits problem. The investigation, which encompasses both experimental and theoretical analysis, yields several significant insights:

Firstly, the AO-UCB algorithm exhibits exceptional performance in scenarios characterized by extreme conditions, such as limited resources or heightened complexity. In these environments, AO-UCB consistently surpasses the standard UCB, suggesting its potential advantages in practical settings where a balance between exploration and exploitation is crucial.

Secondly, while the standard UCB algorithm may not match the enhanced capabilities of AO-UCB in more demanding situations, it maintains notable efficacy in less complex scenarios with ample resources. The simplicity and interpretability of the standard UCB render it a viable option in certain contexts.

However, it is important to acknowledge the limitations and challenges encountered by both algorithms. The AO-UCB, despite its theoretical superiority, may not fully realize its potential in dynamic or unpredictable environments. Conversely, the standard UCB is prone to premature convergence to suboptimal solutions in complex scenarios. Therefore, the selection of an appropriate algorithm for practical applications necessitates a comprehensive evaluation of both the problem's characteristics and the algorithms' performance.

References

[1] Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. MIT Press.

[2] Robbins, H. (1952). Some aspects of the sequential design of experiments. Bulletin of the American Mathematical Society.

[3] Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. Machine learning.

[4] Berry, D. A., & Fristedt, B. (1985). Bandit problems: Sequential allocation of experiments. CRC Press.

[5] Florescu, I. (2014). Probability and stochastic processes. John Wiley & Sons.

[6] Agrawal, R. (1995). Sample mean based index policies with O (log n) regret for the multi-armed bandit problem. Advances in applied probability.

[7] Bubeck, S., & Cesa-Bianchi, N. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. Foundations and Trends® in Machine Learning.

[8] Hoeffding, W. (1963). Probability Inequalities for Sums of Bounded Random Variables. Journal of the American Statistical Association.

[9] Russo, D. (2016). A Tutorial on Thompson Sampling. Foundations and Trends® in Machine Learning.

[10] Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples, Biometrika.

[11] Lai, T., & Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. Advances in Applied Mathematics.

[12] Garivier, A., Kaufmann, E., & Cappe, O. (2016). A lower bound for the regret of an optimal learning algorithm. Comptes Rendus Mathematique.