Introduction of Stochastic Bandits

Dick Jessen William

October 31, 2021

Abstract

In this report, we will take a look on the problem of Stochastic Bandits with IID rewards. We will take a look on algorithms and analyze their performance. This report follows from the chapter 1 of the book of Aleksandrs Slivkins [22].

1 Introduction

First, lets define the problem of Stochastic Bandits. The algorithm has K actions (arms) to choose every round, and there are T rounds, for a given K and T. Each round, the algorithm chooses an action and gains a reward depending of the action. For now, assume all three of the following are true.

1. The algorithm only get to see the reward gained from the chosen action every round.
2. The reward for each action a is independent and identically distributed (IID) from the distribution D_a.
3. Rewards are bounded in the interval $[0, 1]$.

Our interest is the mean reward vector $\mu \in [0,1]^K$, with $\mu(a) = \mathbb{E}[D_a]$. We define the set of all allowed arms as A, and the best mean reward as $\mu^* = \max_{a \in A} \mu(a)$. The difference $\Delta(a) = \mu^* - \mu(a)$ is called the gap of arm a. An optimal arm a is any arm with zero gap. Note that there may be more than one optimal arm.

Now, we will introduce the notion of regret of an algorithm. We do this by comparing the difference between the expected payout when we always choose the optimal arm and our expected payoff of an algorithm. In other words, we define $R(T)$ as the regret at round T as

$$R(T) = \mu^* \cdot T - \sum_{t=1}^{T} \mu(a_t).$$

This number denotes how much the algotithm is missing out not knowing the best arm. Because a_t, the chosen arm in round T, is random (depending on our algorithm), $R(T)$ is also a random variable. Hence, we will talk about the expected regret, $\mathbb{E}(R(T))$.

2 Our First Algorithm: Explore-First

We will start with a simple idea. First, try every arm N times, for some N. This step is called the exploration step. Then, find the arm with the best average $\hat{a}$ based on observation. Finally, we pick $\hat{a}$ for the rest of the round. The third step is called the exploitation step.

Algorithm 1 Explore-First

Try each arm N times
$\hat{a} \leftarrow$ arm with highest average
Play $\hat{a}$ until end of round.

The parameter N will be fixed beforehand, and we will choose it as a function based on T and K to minimize regret. We will see the regret of this algorithm.

Suppose the average reward for action a after the exploration phase is $\overline{\mu}(a)$. We want the average reward to be a good estimate of the true expected rewards. To see this, we use Hoeffding's Inequality [10], which is stated below (proof omitted).

Theorem 1. *Let $Z_1, Z_2, \cdots, Z_n$ be independent bounded random variables with $Z_i \in [a, b]$ for all i. Then,*

$$P\left(\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) \geq nt\right) \leq e^{-\frac{2nt^2}{(b-a)^2}} \text{ and } P\left(\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) \leq -nt\right) \leq e^{-\frac{2nt^2}{(b-a)^2}}.$$

Denote $rad = \sqrt{2 \log T / N}$. Now, using the inequality above, we see that

$$P(|\overline{\mu}(a) - \mu(a)| \leq rad) = 1 - P(\overline{\mu}(a) - \mu(a) \geq rad) - P(\overline{\mu}(a) - \mu(a) \leq -rad) \geq 1 - \frac{2}{T^4}. \quad (1)$$

Hence, we see that $\overline{\mu}(a)$ is a good approximation of $\mu(a)$. In other words, $\mu(a)$ lies on $[\overline{\mu}(a) - rad, \overline{\mu}(a) + rad]$ with high probability. Next, define the clean event as the event when for every arms satisfies the inequalities above, and define the bad event otherwise. The good thing about this classification is that we can only consider the clean event in our analysis, as the probability of a bad event happening is very small. This causes a larger constant in our big-O analysis, but simplifies our analysis considerably.

Now, lets actually analyze the regret value of the algorithm. First, we start with the case $K = 2$. Consider the clean event. We prove that if we end up exploiting the worse arm, we will not lose too much value. Suppose that a^* is the best arm and we choose the arm $a \neq a^*$. Because we choose this arm, by the algorithm, we have $\overline{\mu}(a) > \overline{\mu}(a^*)$. Because the event is clean, $\mu(a) + rad \geq \overline{\mu}(a) > \overline{\mu}(a^*) \geq \mu(a^*) - rad$. Hence, $\mu(a^*) - \mu(a) \leq 2rad$. So, every exploitation round have adds at most $2rad$ of regret. Each exploration round contributes at most 1 to regret trivially. Hence, the total regret $R(T)$ is at most $N \cdot 1 + (T - 2N) \cdot (2rad) < N + 2rad \cdot T$. Because N is determined from the start, we can pick N that minimizes this value. For $N = T^{2/3}(\log T)^{1/3}$, we find that

$$R(T) \leq O(T^{2/3}(\log T)^{1/3}). \quad (2)$$

Finally, we analyze the bad event. If the bad event happens, note that $R(T) \leq T$ trivially. Hence, considering both cases,

$$\begin{aligned} \mathbb{E}[R(T)] &\leq \mathbb{E}[R(T)|clean] \times P(clean) + \mathbb{E}[R(T)|bad] \times P(bad) \\ &\leq \mathbb{E}[R(T)|clean] + T \times O(T^{-4}) \\ &\leq O(T^{2/3}(\log T)^{1/3}). \end{aligned}$$

To generalize to more than two arms, use the union bound for (1) over K arms and follow the argument above (note that $K \leq T$ because we need to run the arms at least once). Finally, for regret computation, we need to account for the dependence on K. Now, by similar logic as above, $R(T) \leq NK + 2rad \cdot T$. Setting $N = (T/K)^{2/3} O(\log T)^{1/3}$, we have shown the following theorem.

Theorem 2. *Explore-First Algorithm achieves regret* $\mathbb{E}[R(T)] \leq T^{2/3} \times O(K \log T)^{1/3}$.

3 Epsilon-Greedy Algorithm

The algorithm above has several problems. If many arms have a large gap, then our algorithm wastes a lot of time checking these arms. We should spread the exploration phase more uniformly over time. This can be done by the Epsilon-Greedy Algorithm.

Algorithm 2 Epsilon-Greedy

```
for each round t = 1, 2, · do
    Toss a coin with success probability ε_t.
    if succes then
        Explore: Choose an arm uniformly at random.
    else
        Exploit: Choose the arm with highest average so far.
    end if
end for
```

Now, we will consider the case when $\epsilon_t \sim t^{-1/3}$. This way, the exploration phase up to round t is on the order $t^{2/3}$. By simple calculations, we have the following theorem.

Theorem 3. *Epsilon-Greedy algorithm with* $\epsilon_t = t^{-1/3} \cdot (K \log t)^{1/3}$ *has regret bound* $\mathbb{E}[R(t)] \leq t^{2/3} O(K \log t)^{1/3}$ *for each round* t.

4 Adaptive Explorations

In this section, we will examine two algorithms with much better bounds.

First, assume $K = 2$. One good idea is to alternate arms until we are confident. Now, we will discuss how much confidence is needed.

Fix round t and arm a. Denote $n_t(a)$ as the number of rounds before t such that arm a is chosen, and $\overline{\mu}_t(a)$ be the average reward. By Hoeffding's Inequality, assuming $r_t(a) = \sqrt{2 \log T / n_t(a)}$,

$$P(|\overline{\mu}_t(a) - \mu(a)| \leq r_t(a)) \geq 1 - \frac{2}{T^4}. \quad (3)$$

However, we cannot simply use Hoeffding, because we have a random number of independent random variable. Hence, we will use a more careful argument. For each arm a, assume a reward tape as a $1 \times T$ table with cells taken independently from D_a. WLOG, the reward when we take the arm a the j-th time is the value of the j-th cell. Let $\overline{v}_j(a)$ be the average reward at arm a from the first j times the arm a is chosen. By Hoeffding's Inequality, for all j, $P(\overline{v}_j(a) - \mu(a) \leq r_t(a)) \geq 1 - \frac{2}{T^4}$. By union bound, assuming $K \leq T$, $P(\forall a \forall t |\overline{\mu}_t(a) - \mu(a)| \leq r_t(a)) \geq 1 - \frac{2}{T^2}$.

The event $\forall a \forall t |\overline{\mu}_t(a) - \mu(a)| \leq r_t(a)$ will be called the clean event for subsequent analysis. We denote $UCB_t(a) = \overline{\mu}_t(a) + r_t(a)$ and $LCB_t(a) = \overline{\mu}_t(a) - r_t(a)$ as the upper confidence and lower confidence bounds, respectively.

4.1 Sucessive Elimination

Now, we write the idea of high-confidence elimination algorithm.

Algorithm 3 High-confidence Elimination, K=2

Alternate two arms until $UCB_t(a) < LCB_t(a')$ for some even round t.
Use arm a' until the end.

It is obvious that the "abadoned" arm is not the best arm. Hence, we only to need to calculate regret before throwing out one arm. Suppose that t is the last round before we abadon an arm. Then, $\Delta = |\mu(a) - \mu(a')| \leq 2(r_t(a) + r_t(a'))$. Because we are alternating before time t, $n_t(a) = \frac{1}{2}$. Hence, $\Delta \leq 2(r_t(a) + r_t(a')) \leq 4\sqrt{2\log(T)/\lfloor t/2 \rfloor} = O(\sqrt{\log(T)/t})$. Hence, total regret $R(t)$ is at most $\Delta \times t = O(\sqrt{t \log T})$. Finally, we take care of the bad event. This is easy, because

$$\begin{aligned}
\mathbb{E}[R(T)] &\leq \mathbb{E}[R(T)|clean] \times P(clean) + \mathbb{E}[R(T)|bad] \times P(bad) \\
&\leq \mathbb{E}[R(T)|clean] + t \times O(T^{-2}) \\
&\leq O(\sqrt{t \log T}).
\end{aligned}$$

Hence, we have proven the following.

Theorem 4. *For $K = 2$, High-confidence Elimination achieves regret $\mathbb{E}[R(t)] \leq O(\sqrt{t \log T})$ for $t \leq T$. This is stronger than Explore-First.*

Now, we move on to general K.

Algorithm 4 Successive Elimination

All arms are set to active
loop
  Play each arm once.
  Deactivate all arms a such that if t is the current round, there exists another arm a' such that $UCB_t(a) < LCB_t(a')$ deactivation rule.
end loop

To analyze this, we only need to focus on the clean event (bad event is negligible). Let a^* be an optimal arm, and note that this arm will stay active. Fix any arm a that is not optimal. Consider the last round t such that the deactivation rule was invoked and a is still active. Similar to above, the confidence intervals of these arms is still overlapping. Because the algorithm alternate active arms and both a and a^* are active before round t, $r_t(a) = r_t(a^*)$. Hence, $\Delta(a) \leq 2(r_t(a^*) + r_t(a)) = 4r_t(a)$. By construction of t, $n_T(a) \leq 1 + n_t(a)$. Hence, for any non-optimal arm a,

$$\Delta(a) \leq O(r_T(a)) = O(\sqrt{\log T / n_T(a)}). \tag{4}$$

Denote the contribution of arm a to regret at round t as $R(t, a)$. This can be bound by noting that $R(t, a) = n_t(a) \cdot \Delta(a) \leq n_t(a) \cdot O(\sqrt{\log T / n_t(a)}) = O(\sqrt{n_t(a) \log T})$. Summing over every arm, we obtain that $R(t) = \sum_{a \in A} R(t, a) \leq O(\log T) \sum_{a \in A} \sqrt{n_t(a)}$. Because $f(x) = \sqrt{x}$ is concave, using Jensen's inequality, we have $\frac{1}{K} \sum_{a \in A} \sqrt{n_t(a)} \leq \sqrt{\frac{1}{K} \sum_{a \in A} n_t(a)} = \sqrt{\frac{t}{k}}$. Hence, we conclude that $R(t) \leq O(\sqrt{Kt \log T})$. We have proven the following theorem.

Theorem 5. *Successive Elimination algorithm achieves regret $\mathbb{E}[R(t)] = O(\sqrt{Kt \log T})$ for all rounds $t \leq T$.*

We can also note that rearranging (4), we have $n_T(a) \leq O(\frac{\log T}{\Delta(a)^2})$. Hence, $R(T, a) = \Delta(a) \cdot n_T(a) \leq \Delta(a) \cdot O(\frac{\log T}{\Delta(a)^2}) = O(\frac{\log T}{\Delta(a)})$. Summing all arms, we have this result.

Theorem 6. *Successive Elimination algorithm achieves regret*

$$\mathbb{E}[R(t)] = O(\log T) \left[\sum_{arms\ a\ with\ \mu(a)<\mu(a^*)} \frac{1}{\mu(a^*) - \mu(a)} \right].$$

Note that the bound is logarithmic in T, with a constant at most $O(K/\Delta)$, with $\Delta = \min_{\text{suboptimal arms a}} \Delta(a)$.

4.2 Optimism Under Uncertainity

The next algorithm have a idea of optimism under uncertainity: assume that each arm is as good as it can possibily be given the observations so far, and choose the best arm based on that. This algorithm is called UCB1.

Algorithm 5 UCB1

Try each arm once
for e **do**ach round $t = 1, 2, \cdots T$
  Pick some arm that maximizes $UCB_t(a)$.
end for

To analyze this algorithm, we focus on the clean event. Recall that a^* is an optimal arm and a_t is the arm used on round t. According to the algorithm, $UCB_t(a_t) \geq UCB_t(a^*)$. In a clean event, $\mu(a_t) + r_t(a_t) \geq \overline{\mu_t}(a_t)$ and $UCB_t(a^*) \geq \mu(a^*)$. Hence, $\mu(a_t) + 2r_t(a_t) \geq \overline{\mu_t}(a_t) + r_t(a_t) = UCB_t(a_t) \geq UCB_t(a^*) \geq \mu(a^*)$. So, $\Delta(a_t) = \mu(a^*) = \mu(a_t) \leq 2r_t(a_t) = 2\sqrt{2 \log T / n_t(a_t)}$. For each arm a, consider the last round t when the arm is chosen by the algorithm. Using this bound, we can prove (4). Continuing the proof similar with the one in section 4.1 gives us this theorem.

Theorem 7. *UCB1 satisfies regret bounds similar to Theorem 5 and Theorem 6.*

5 Bandits with Prior Knowledge

In some cases, some information about the mean reward vector μ may be known to the algorithm before. We can encode this information by a constraint on μ or a bayesian prior. Such model can allow nontrivial regret bounds that are independent of K, and even works for $K = \infty$.

5.1 Constrained μ

The canonical modelling embeds arms into $\mathbb{R}^d$, for positive integer $d \in \mathbb{N}$. Then, μ is a function on a subset of $\mathbb{R}^d$ that maps arms into their mean rewards. The constraint is that μ is on some family of well-behaved functions. Some example include

1. Linear functions: $\mu(a) = w \cdot a$ for a fixed but unknown $w \in \mathbb{R}^d$.
2. Concave functions: The set of arms is a convex subset and μ'' exists and negative.
3. Lipschitz functions: $|\mu(a) = \mu(a')| \leq L \cdot ||a - a'||_2$ for all arms a, a' and some constant L.

This made arms dependant into each other, so by observing the reward on an arm, we can deduce something about the mean reward on the other arm. In particular, Lipschitz condition only allows local inferences (we can only learn about a by observing arms that are not too far from a'). On the contary, Linearity and concavity allows as to learn about a by looking a very far a'.

We usually prove regret bounds for all μ in one family, This allows bounds for infinite arms, and only depends on T and d. However, we only can prove results as good as the worst case over one family. This will be an underestimate if bad cases occure very rarely and an overestimate if μ belongs to a family is a strong assumption.

5.2 Bayesian Bandits

Here, μ is drawn independently from a distribution $\mathbb{P}$, the Bayesian prior. We are interested in Bayesian regret: regret in expectation over $\mathbb{P}$. This is a special case of Bayesian approach, with many uses in statistics and machine learning. $\mathbb{P}$ implicitly defines the family of feasible μ, and specifies whether and to which extent some μ in the family are more likely than the others. However, sampling assumption may be very idealized and $\mathbb{P}$ may not fully known to the algorithm.

6 Literature Review

Most of the algorithms here can be generalized to multi-armed bandits. Successive Elimination is from Even-Der et al. [12], and UCB1 is from Auer et al. [4]. Explore-First and Epsilon-Greedy are well known for a long time. The original UCB1 has confidence radius $r_t(a) = \sqrt{\alpha \ln t / n_t(a)}$, with $\alpha = 2$.

- Optimality: Our bounds for Successive Elimination is near optimal. Audibert and Bubeck [3] manages to shave the $\log T$ factor to get $O(\sqrt{KT})$ for theorem 5. Theorem 6 is optimal up to the constant factor. Garvier and Cappe [14] managed to get the lower bound of $\frac{1}{2 \ln 2}$ in 2011.
- High-proability regret: To find the upper bound of $\mathbb{E}[R(T)]$, we find a high-probability upper bound of $R(T)$. This is common for regret bounds from clean event analysis, but it take more work for more advanced bandit scenarios.
- Regret for all rounds: Suppose we do not know T. We would like to get similar bound for t becomes very large.
  - If there exists an upper bound of T, we use this as our T. Because the regret depend on T logarithmically, it will not overestimate too much.
  - Use UCB1 with $r_t(a) = \sqrt{\frac{2 \log t}{n_t(a)}}$ [4]. We do not need T at all here.
  - Use a doubling trick. We run phases, with phase i have 2^i rounds. This has been done in many other analysis [17]. However, "resetting" is not very practical.
- Instantaneous regret: Define instantaneous regret at round t as $\Delta(a_t) = \mu^* - \mu(a_t)$, with a_t is the arm chosen in this round. We might want to have regret spread uniformly across rounds to avoid spikes in regret. We would like an upper bound on regret that decreases monotonically over time.
- Bandits with predictions: Another goal of a bandit problem is to output a prediction a_t^* after each round t. The algorithm's goal is now the accuracy of a_t^*, not the total reward. We can either minimize instantaneous regret $\mu^* - \mu(a_t^*)$ or maximize probability of choosing best arm (maximize $P(a_t^* = a^*)$). The first one is called pure-exploration bandits, and the latter is called best-arm identification. Good algorithms for total regret, such as UCB1 and Successive Elimination, aso works well in this case. However, improvements are possible [21, 7].
- Available arms: There are some cases when some arms may not be available. Kleinberg et al. [20] allows arms to be inactive for some rounds. Now, we focus on the selecting the best "active" arm. More extremely, [9] examines the case when arms are permanently disabled, with randomized schedule known by the algorithm. Here, UCB1 works well for both cases.
- Partial feedback: Here, instead of numeric reward, the algorithm recieve partial feedback. In dueling bandits problem [23], we choose two arms, and recieve info on which reward is higher each round. This has applications in web search optimizations. Here, action corresponds to search results and numerical reward is only a crude approximation of user satisfaction. However, we can determine which slates is better by using interleaving [16]. The works of [1] and [11] extends on this model.

  Another model is called partial modeling [5], posits that the outcome for choosing an arm a is a pair of (reward,feedback), where the feedback is an arbitary message. By assuming IID, the outcome is chosen independently for some distribution D_a for all possible outcome. Hence, the feedback can be more than bandit feedback (include rewards for other unchosen arms) or less (only states whether the reward is nonzero). A special case is when the structure of the feedback is defined as a graph, where the feedback for choosing a includes the rewards for every arms adjacent to a. ([2])
- Relaxed benchmark: For large K, it may be hard to find the best arm. Instead, we ignore the best $\epsilon > 0$ arms. We want to find regret relatice to the best remaining arm, which is $\epsilon-$regret. We see that $\frac{1}{\epsilon}$ is an effective number of arms. We will bound $\epsilon-$regret that depends ont $\frac{1}{\epsilon}$, but not on K. A result by Kleinberg [19] achieves a $\epsilon-$regret of $\frac{T^{2/3} \operatorname{polylog} T}{\epsilon}$ using the doubling trick.
- Bandits with initial information: The initial discussion are done by Kleinberg and Flaxman et al. [18, 13]. Recent advances are made by Bubeck et al. [6, 8]. A complete treatment is made by Hazan in 2015 [15].

References

[1] Nir Ailon, Zohar Karnin, and Thorsten Joachims. Reducing dueling bandits to cardinal bandits, 2014.

[2] Noga Alon, Nicolò Cesa-Bianchi, Ofer Dekel, and Tomer Koren. Online learning with feedback graphs: Beyond bandits. 02 2015.

[3] Jean-Yves Audibert and Sébastien Bubeck. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11:2785–2836, 10 2010.

[4] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 05 2002.

[5] Gábor Bartók, Dean Foster, Dávid Pál, Alexander Rakhlin, and Csaba Szepesvári. Partial monitoring—classification, regret bounds, and algorithms. *Mathematics of Operations Research*, 39:967–997, 11 2014.

[6] Sébastien Bubeck, Ofer Dekel, Tomer Koren, and Yuval Peres. Bandit convex optimization: sqrtT regret in one dimension, 2015.

[7] Sébastien Bubeck, Remi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12, 05 2011.

[8] Sébastien Bubeck, Ronen Eldan, and Yin Tat Lee. Kernel-based methods for bandit convex optimization. *CoRR*, abs/1607.03084, 2016.

[9] Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. Mortal multi-armed bandits. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2009.

[10] John Duchi. Cs229 supplemental lecture notes hoeffding's inequality.

[11] Miroslav Dudík, Katja Hofmann, Robert E. Schapire, Aleksandrs Slivkins, and Masrour Zoghi. Contextual dueling bandits. *Journal of Machine Learning Research*, 40(2015), January 2015. 28th Conference on Learning Theory, COLT 2015 ; Conference date: 02-07-2015 Through 06-07-2015.

[12] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 06 2006.

[13] Abraham Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *CoRR*, cs.LG/0408007, 2004.

[14] Aurélien Garivier and Olivier Cappé. The kl-ucb algorithm for bounded stochastic bandits and beyond. 02 2011.

[15] Elad Hazan. Introduction to online convex optimization. *CoRR*, abs/1909.05207, 2019.

[16] Katja Hofmann, Lihong Li, and Filip Radlinski. Online evaluation for information retrieval. *Foundations and Trends® in Information Retrieval*, 10(1):1–117, 2016.

[17] Irvan and Yu. Sublinear algorithms in t -interval dynamic networks.

[18] Robert Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. 01 2004.

[19] Robert Kleinberg. Anytime algorithms for multi-armed bandit problems. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA '06, page 928–936, USA, 2006. Society for Industrial and Applied Mathematics.

[20] Robert Kleinberg, Alexandru Niculescu-Mizil, and Yogeshwer Sharma. Regret bounds for sleeping experts and bandits. volume 80, pages 425–436, 01 2008.

[21] Shie Mannor, John Tsitsiklis, Kristin Bennett, and Nicol Cesa-bianchi. The sample complexity of exploration in the multi-armed bandit problem. 07 2004.

[22] Aleksandrs Slivkins. Introduction to multi-armed bandits. *CoRR*, abs/1904.07272, 2019.

[23] Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. volume 78, 01 2009.