Finite-Time Analysis of the Multiarmed Bandit Problem

Intelligent System for Pattern Recognition - Midterm 4

Roberto Esposito
r.esposito8@studenti.unipi.it

University of Pisa

31-05-2022

Introduction

In Reinforcement Learning there is the **exploration** versus **exploitation** dilemma that consists of a search for a balance between exploring new states to find helpful actions while taking the empirically best action most of the times.
The following trade-off can be easily described by the **multi-armed bandit problem**.
A **K-armed bandit problem** is defined by the random variables $X_{i,n}$ for $1 \leq i \leq K$ and $n \geq 1$ where the index i corresponds to a slot machine. The following variables $X_{i,s}$ and $X_{j,t}$ are **independent** $\forall 1 \leq i < j \leq K$ and $\forall s, t \geq 1$. Furthermore the rewards of the machine i $(X_{i,1}, X_{i,2}, \dots)$ are **i.i.d.**.

Given a policy A the **regret** of A is defined as follows:

$$\mu^* n - \mu_j \sum_{j=1}^{K} \mathbb{E}[T_j(n)]$$

where $T_j(n)$ is the number of times the machine j has been played by the policy the first n plays, $\mathbb{E}[\,]$ is the expectation, μ_j is the expectation of the machine j and $\mu^* = \max \mu_i$, with $1 \leq i \leq K$.
In simpler words the regret is the **expected loss** due to the fact that the policy does not always play the best machine.

Asymptotic Logarithmic Regret

Lai and Robbins, in [1], found a family of reward distributions for which it is satisfied the following inequality:

$$\mathbb{E}[T_j(n)] \leq \left(\frac{1}{D(p_j||p^*)} + o(1) \right) \ln n$$

where $o(1) \to 0$ if $n \to \infty$ and $D(p_j||p^*)$ is the Kullback-Leibler divergence (it shows how much a probability distribution is different with respect to another).
In other words the inequality says that the **best machine** is played **asymptotically** more often than the others.

Auer et al. [2] show four different policies, simple and efficient to compute, that can achieve **logarithmic regret** uniformly **over-time** instead of being asymptotic:

- **UCB1**: it comes from the index-based policy of Agrawal (shown in [3]);
- **UCB2**: it is a bit more complicated than the first one and the plays are divided into epochs;
- ϵ_n-**GREEDY**: it plays with probability $1 - \epsilon_n$ the machine with the highest average reward and with probability ϵ_n a randomly chosen machine;
- **UCB1-NORMAL**: it refers to a special case in which the reward probabilities are normally distributed and it is achieved the logarithmic regret.

ϵ_n-GREEDY

Let's analyze in details the ϵ_n-**GREEDY**. If it is not chosen properly the value of ϵ_n then it causes a linear growth in the regret instead of logarithmic. For this reason it is useful to let go to 0 the value of ϵ with a **fixed rate**. It turns out that a rate equals to $\frac{1}{n}$ allows to have a logarithmic bound on the regret.
In order to execute the following policy it is needed to choose the parameters: $c > 0$ and $0 < d < 1$. After that we need to define the sequence of $\epsilon_n \in [0, 1]$ with $n = 1, 2, \ldots$ as:

$$\epsilon_n \stackrel{def}{=} \min \left\{ 1, \frac{cK}{d^2 n} \right\}$$

At iteration n, it is picked the machine i_n (the one with the highest current average reward) with probability $1 - \epsilon_n$ and with probability ϵ_n it is picked a random machine. Furthermore $\forall K > 1$ and $\forall P_1, \ldots, P_K$ (reward distributions) with support in $[0, 1]$ if the policy is run with input parameter $0 < d \leq \min_{i:\mu_i < \mu^*} \Delta_i$ where Δ_i is $(\mu^* - \mu_i)$, then the probability that after any number $n \geq (cK/d)$ of plays the policy chooses a **sub-optimal machine** j is at most:

$$\frac{c}{d^2 n} + 2 \left(\frac{c}{d^2} \ln \frac{(n-1)d^2 e^{1/2}}{cK} \right) \left(\frac{cK}{(n-1)d^2 e^{1/2}} \right)^{c/(5d^2)} + \frac{4e}{d^2} \left(\frac{cK}{(n-1)d^2 e^{1/2}} \right)^{c/2}$$

Experiments

In the paper they compare the results obtained with three policies (UCB1-TUNED, UCB2 and ϵ_n-GREEDY) on Bernoulli reward distributions with different parameters and for two different values of K ($K = 2$ and $K = 10$).

For each experiment are tracked two metrics: the first one is the **percentage of plays of the optimal machine** and the second one is the **actual regret** (the difference between the reward of the optimal machine and the reward of the machine played). The plot reported in the next slides are run in a semi-logarithmic scale with $100,000$ plays.

The **UCB2** is pretty **insensitive** to the choice of the parameter α. On the other hand for the ϵ_n-GREEDY there were two values to be set: the first one is c for which was difficult to find a value that fit the distribution and the second one is d set to $\mu^* - \max_{i:\mu_i<\mu^*} \mu_i$.

Comparison between policies

Performing the comparison of all the policies it turns out:

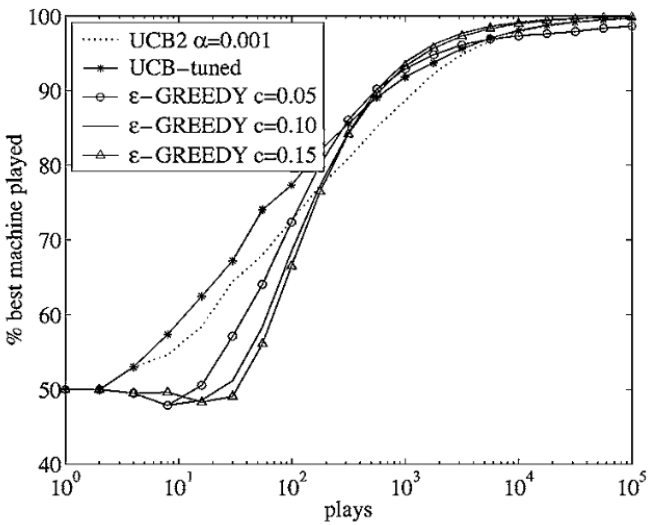
- If the ϵ_n-GREEDY is optimally tuned it **performs always better** than the others, but if the policy is not well tuned its performance decreases rapidly;
- Most of the times the UCB1-TUNED performs **similarly** to the ϵ_n-GREEDY when it is tuned optimally; The following policy is a variant of the UCB1 in which the confidence bound of the machines is changed and it turns out that this new policy is highly better than the UCB1;
- The policy UCB2 performs like the UCB1-TUNED, but **slightly worse**.

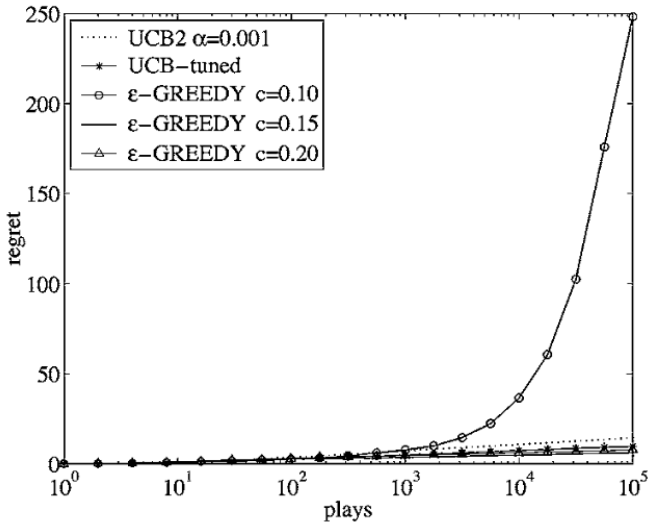
In the table below are reported the reward expectations of the Bernoulli either for the 2-Armed Bandit either for the 10-Armed Bandit.

	1	2	3	4	5	6	7	8	9	10
1	0.9	0.6								
2	0.9	0.8								
3	0.55	0.45								
11	0.9	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6
12	0.9	0.8	0.8	0.8	0.7	0.7	0.7	0.6	0.6	0.6
13	0.9	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
14	0.55	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45

Comparison on distribution 1

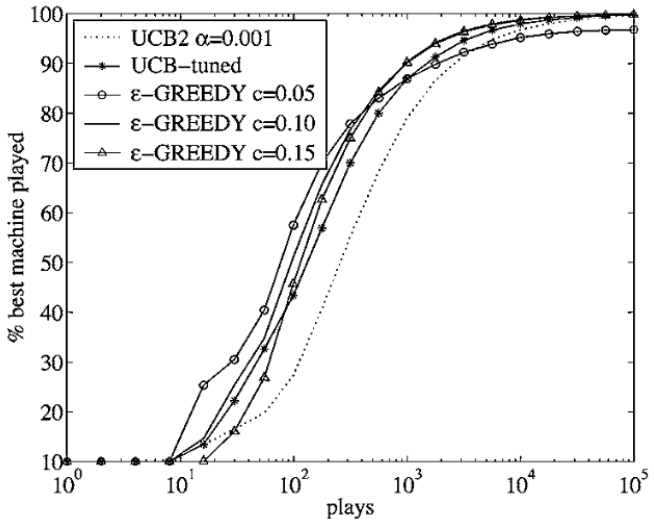
In the following figure is reported the comparison between UCB2, UCB1-TUNED and ϵ_n-GREEDY with different parameters. The two figures represent the analysis of the allocation strategy using the first distribution reported in the table before (the first one uses as metrics the percentage of plays of the optimal machine and the second one uses the actual regret). In this case the policies are tested on two machines so we have $K = 2$.

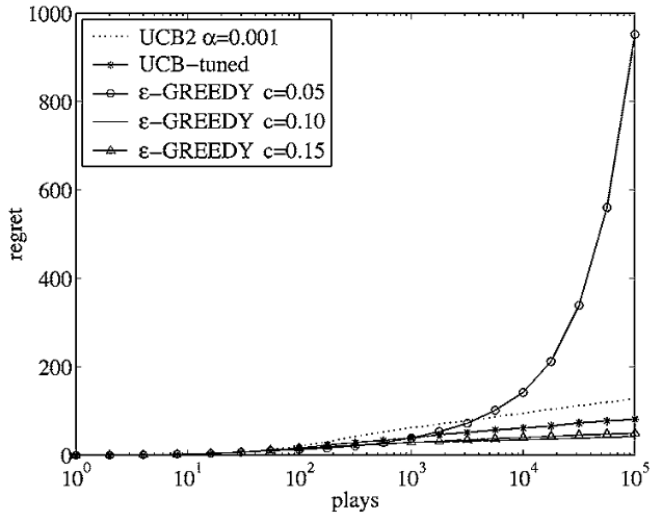




Comparison on distribution 11

Another case is the one reported in the figure below. It is the comparison with ten machines ($K = 10$) and the parameters are the ones reported in the fourth row of the table.
As we can see in both cases the policies perform quite similar, but the ϵ_n-GREEDY can be either the best one either the worse one with respect to the value of c.
For example when $c = 0.05$ for the ϵ_n-GREEDY the regret grows highly with an increasing number of plays instead if $c = 0.10$ or $c = 0.15$ it grows logarithmic.





Conclusion

In conclusion we can observed the following things:

- The UCB2 during the experiments seems to be one of the worse and for different values of α it seems that the performance does not change. The choice of α does not help to modify the behavior of the policy;
- The ϵ_n-GREEDY performs better than the others, but it is too sensitive to the value of c. Its performance could have been improved performing a search for the best value of that parameter since have been tested just three possible values for c;
- The performance of the experiments depends highly on the variance of the reward distributions. If the reward has a low variance and Δ_i is large then the policies perform well, but if the reward has high variance and Δ_i small then it performs badly;
- Furthermore the author of the paper could have tested the behavior of these policies even with other distributions and not just that of Bernoulli for example using a Poisson distribution (the tests are not run for the $UCB1 - NORMAL$).

Bibliography

[1] T.L Lai and Herbert Robbins. "Asymptotically Efficient Adaptive Allocation Rules". In: *Adv. Appl. Math.* 6.1 (1985), pp. 4–22. ISSN: 0196-8858. DOI: 10.1016/0196-8858(85)90002-8. URL: https://doi.org/10.1016/0196-8858(85)90002-8.

[2] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. "Finite-Time Analysis of the Multiarmed Bandit Problem". In: *Mach. Learn.* 47.2–3 (2002), pp. 235–256. ISSN: 0885-6125. DOI: 10.1023/A:1013689704352. URL: https://doi.org/10.1023/A:1013689704352.

[3] Rajeev Agrawal. "Sample Mean Based Index Policies with O(log n) Regret for the Multi-Armed Bandit Problem". In: *Advances in Applied Probability* 27.4 (1995), pp. 1054–1078. ISSN: 00018678. URL: http://www.jstor.org/stable/1427934 (visited on 05/27/2022).