Multi-armed Bandit Problems with History

Pannagadatta Shivaswamy and Thorsten Joachims
Department of Computer Science, Cornell University, Ithaca NY
{pannaga,tj}@cs.cornell.edu

1 Introduction

In a multi-armed bandit problem, at each time step, an algorithm chooses one of the possible arms and observes its rewards. The goal is to maximize the sum of rewards over all time steps (or to minimize the regret). In the conventional formulation of the problem, the algorithm has no prior knowledge about the arms. Many applications, however, provide some data about the arms even before the algorithm starts. For example: a search engine company obtained data from a small sample of paid users on its newly developed retrieval functions. The availability of such historic data leads to the question of how online learning algorithms can best use it to reduce regret. This problem is meaningful only for the case of stochastic arms [1]. We propose algorithms which show that a logarithmic amount of historic data allows them to achieve constant regret. The work by [2] assumes that historic data collected via some policy is available to *evaluate* a mapping from side information to arms. In the absence of side-information, their policy evaluation strategy reduces to choosing the arm with the highest mean reward on the historic data.

In a K armed stochastic bandit problem, random variable $X_{j,t} \in [0,1]$ ($1 \leq j \leq K$, $t \geq 1$) denotes the reward incurred when the j^{th} arm is pulled the t^{th} time. For arm j, the rewards $X_{j,t}$ are *iid* with mean μ_j and variance σ_j^2. The best arm is denoted by j^* *i.e.,* $\mu_{j^*} := \max_{1 \leq i \leq K} \mu_i$. Historic data is denoted by $X^h_{j,t} \in [0,1]$ for $1 \leq j \leq K$ and $1 \leq t \leq H_j$. The historic rewards for each arm are assumed to be drawn *iid* as well. $T_j(n)$ denotes the number of times the arm j is pulled between times 1 and n (this excludes the pulls of the arm in the historic data). The regret at time n is defined as $REG(n) := \mu_{j^*} n - \mu_j \sum_{j=1}^{K} \mathbf{E}[T_j(n)]$. The per-round regret at time n is defined as $REG(n)/n$. Mean reward of arm j during the execution of the algorithms until its n^{th} pull is defined as $\bar{X}_{j,n}$. Analogously, the joint mean reward of arm j incorporating both the historic and the online data is $\bar{X}^h_{j,n}$.

2 Algorithms

We first consider a **Naive Algorithm** in the presence of historic data: pick the arm with the maximum mean reward on the historic data and then simply play that arm in every iteration. Unfortunately, this is not a very good strategy since the regret can grow polynomially with time:

Theorem 1 *Consider a two armed bandit problem. The first arm has a fixed reward* $0.5 + \epsilon$, $\epsilon > 0$, *the second arm has a Bernoulli reward with mean* 0.5. *Assume that the two arms have* H_1 *and* H_2 *historic data points. Suppose* $H_2 = \left(\delta \log(n)/4\epsilon^2\right)$ *then the naive strategy has regret growing at the rate* $\epsilon n^{1-\delta}/2\pi$ *with* n *for any* $n > \exp(4/\delta)$.

HUCB1: UCB1 with Historic Data Our first algorithm is derived from the UCB1 algorithm [1] that pulls the arm j that maximizes $\bar{X}_{j,T_j(t-1)} + \sqrt{2\ln(t)/T_j(t-1)}$. Our extension of UCB1 for historic data – called HUCB1 – pulls arm j that maximizes $\bar{X}^h_{j,T_j(t-1)} + \sqrt{2(\ln(H_j + t))/(T_j(t-1) + H_j)}$.

Theorem 2 *For HUCB1, the expected number of pulls of any sub-optimal arm* j, *for any time horizon* n, *satisfies,* $\mathbf{E}[T_j(n)] \leq 1 + \max\left(0, 8\log(n + H_j)/(\mu_j - \mu_{j^*})^2 - H_j\right) + \mathcal{O}(1)$.

To see how much impact historic data can have on the regret, consider $n = \exp(H_j(\mu_j - \mu_{j^*})^2/8) - H_j$. In this situation $\mathbf{E}[T_j(n)] = \mathcal{O}(1)$ for HUCB1. At the same n, the regret bound for UCB1 would be linear in H_j.

HUCB3: an ϵ-greedy algorithm Arguably the simplest bandit algorithm is UCB3 that has two parameters $c > 0$ and $0 < d < \min_{j \neq j^*} \mu_{j^*} - \mu_j$. The historic version of UCB3–called HUCB3–is stated below:

- Define: $\epsilon_n^j := 1/K$ for $n \leq cK/d^2$ and $\epsilon_n^j := \left(K(e^{\frac{H_j d^2}{c}} - 1) + \frac{d^2 n}{c}\right)^{-1}$ for $n > cK/d^2$. At iteration n, let $j_n = \arg\max_j \bar{X}^h_{j,T_j(n-1)}$; Play arm j_n with probability $(1 - \sum_{j=1}^K \epsilon_n^j)$. Play arm j with probability ϵ_n^j.

We have the following guarantee on the behavior of HUCB3:

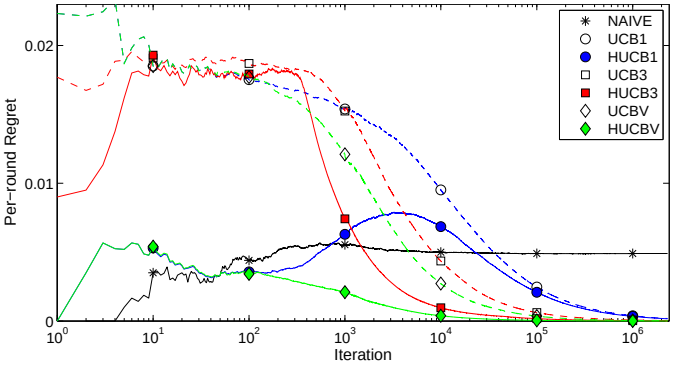
Theorem 3 *HUCB3 admits the following bound for any sub-optimal arm j, for any $n > cK/d^2$,*

$$\mathbf{E}[T_j(n)] \leq c/d^2 \log\left(\left(\frac{cK}{d^2}\left(\exp\left(\frac{H_j d^2}{c}\right) - 1\right) + n\right) / \frac{cK}{d^2}\exp\left(\frac{H_j d^2}{c}\right)\right) + \mathcal{O}(1).$$

To see how the above bound changes with historic data, suppose $H_j = \frac{c}{d^2}\log(nd^2/cK)$, then $\mathbf{E}[T_j(n)] = \mathcal{O}(1)$. This again shows that already a logarithmic amount of historic data suffices to achieve constant regret.

In addition, we have another algorithm known as HUCBV which exploits sample variance. This algorithm is based on the recent UCBV [3]. The advantage of these algorithms is that their regret bound can be significantly tighter compared to the algorithms that do not exploit the sample variance of the rewards. Due to space constraints, we cannot provide the details of this algorithm here. However, the regret bound for HUCBV also shows that a constant regret can be achieved with only logarithmic amount of historic data.

3 Experiments



Experiments were conducted on a large-scale real-world dataset [4] containing about 2.4 million instances. Each instance corresponds to a URL and has more than 3.2 million features associated with it. The label of an instance indicates whether the URL is malicious or not. Five different SVM classifiers were trained using a subset of twenty thousand examples. Predictions were then obtained on all the remaining instances for all the five classifiers. The five classifiers were then used as the arms of a multi-armed bandit problem. The reward was simply one when the prediction of the classifier matched the true URL reputation label and zero when it did not. The parameters of the algorithms (that control explore-exploit trade-off) were tuned with a validation set. The total amount of historic data was fixed at 2000, uniformly split into 400 per arm. The algorithms were then run on about 2.3 million instances and the per-round regret was noted after each iteration for each algorithm. The experiments were repeated 200 times by randomly permuting the instances. A different set of historic data was selected for each run. The results of this experiment are shown in the Figure. This experiment shows how a comparably small amount of historic data can help achieve a substantial improvement in regret. Among the three proposed algorithms, HUCBV has the best performance.

4 Conclusions

We studied the problem of multi-armed bandit problems with historic data. It was shown that the Naive strategy can suffer polynomial regret with logarithmic data where as our proposed algorithms can achieve constant regret with logarithmic amount of data. Experiments on a real-world dataset showed the significant difference a tiny amount of historic data can make on the regret.

References

[1] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[2] J. Langford, A. Strehl, and J. Wortman, "Exploration scavenging," in *ICML*, 2008.

[3] J.-Y. Audibert, R. Munos, and C. Szepesvári, "Variance estimates and exploration function in multi-armed bandit," research report 07-31, Certis - Ecole des Ponts, 2007.

[4] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying suspicious urls: An application of large-scale online learning," in *ICML*, 2009.

Topic: learning algorithms/learning theory **Preference: oral/poster**