Optimization & Statistical Inference Lab

Introduction to Multi-Armed Bandits

From Alexsandrs Slivkins
Microsoft Research NYC

20183160 Kim Tae Hyeon

Optimization & Statistical Inference Lab

Contents

Where to use

1. News

Problem a user visits a news site, the site presents it with a news header, and a user either clicks on this header or not.

Goal to maximize the number of clicks

Assumption

each user is drawn independently from a fixed distribution over users

In each round, the click happens independently with a probability that depends only on the chosen header.

2. Ad selection

Problem for a user, if ad a is displayed, the website observes whether the user clicks on the ad, in which case the advertiser pays some amount $v_a \in [0,1]$

Goal to maximize the paid amount (ad = arm, reward = paid amount)

Assumption

The paid amount v_a depends only on the displayed ad, but does not change over time.

Introduction

Problem

Problem protocol: Multi-armed bandits

In each round $t \in [T]$:

1. Algorithm picks arm $a_t \in \mathcal{A}$.
2. Algorithm observes reward $r_t \in [0, 1]$ for the chosen arm.

1. The restriction to the interval [0,1]
2. The reward for each action is i.i.d. (independent and identically distributed).
3. Reward distribution is the Bernoulli distribution.

$$R(t) = \mu^* \cdot t - \sum_{s=1}^{t} \mu(a_s)$$

Definition of Regret at round t.
R(t) is a random variable, so we should talk about expected regret E[R(T)].

Algorithm

1. Uniform exploration
2. Adaptive exploration
    - 2.1. Successive elimination algorithm
    - 2.2. UCB1 algorithm
    - 2.3. Thompson sampling

Uniform exploration

1 Exploration phase: try each arm N times;
2 Select the arm $\hat{a}$ with the highest average reward (break ties arbitrarily);
3 Exploitation phase: play arm $\hat{a}$ in all remaining rounds.

Algorithm 1.1: Explore-First with parameter N.

$N = (T/K)^{2/3} \cdot O(\log T)^{1/3}$ Put this to N

Theorem 1.3. *Explore-first achieves regret* $\mathbb{E}[R(T)] \leq T^{2/3} \times O(K \log T)^{1/3}$, *where* K *is the number of arms.*

Its performance in the exploration phase is terrible.

Uniform exploration

& epsilon greedy

```
for each round t = 1, 2, ... do
    Toss a coin with success probability ε_t;
    if success then
        explore: choose an arm uniformly at random
    else
        exploit: choose the arm with the highest average reward so far
end
```

Algorithm 1.2: Epsilon-Greedy with exploration probabilities $(\epsilon_1, \epsilon_2, \ldots)$.

Theorem 1.4. *Epsilon-greedy algorithm with exploration probabilities* $\epsilon_t = t^{-1/3} \cdot (K \log t)^{1/3}$ *achieves regret bound* $\mathbb{E}[R(t)] \leq t^{2/3} \cdot O(K \log t)^{1/3}$ *for each round t.*

It is the same regret as in uniform exploration, but it holds for all rounds t.
To get (epsilon t) ε_t from $\varepsilon_t \sim t^{-1/3}$

Uniform exploration

With epsilon greedy

Both exploration-first and epsilon-greedy have a big flaw that the exploration schedule does not depend on the history of the observed rewards

Adaptive exploration

Successive Elimination Algorithm

Alternate two arms until $\mathtt{UCB}_t(a) < \mathtt{LCB}_t(a')$ after some even round t;
Then abandon arm a, and use arm a' forever since.

Algorithm 1.3: "High-confidence elimination" algorithm for two arms

This approach extends to $K > 2$ arms as follows:

Initially all arms are set "active";
Each phase:
    try all active arms (thus each phase may contain multiple rounds);
    deactivate all arms a s.t. $\exists$arm a' with $\mathtt{UCB}_t(a) < \mathtt{LCB}_t(a')$;
Repeat until end of rounds.

Algorithm 1.4: Successive Elimination algorithm

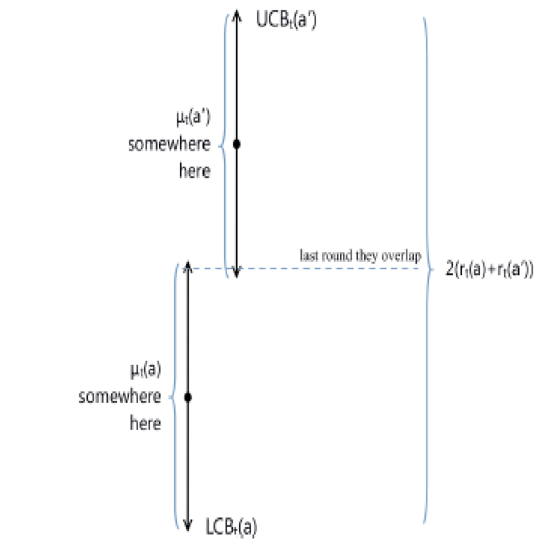

Figure 1.2: t is the last round that the two confidence intervals still overlap

Adaptive exploration

Successive Elimination Algorithm

Lemma 1.6. *For two arms, Algorithm 1.3 achieves regret* $\mathbb{E}[R(t)] \leq O(\sqrt{t \log T})$ *for each round* $t \leq T$.

Theorem 1.8. *Successive Elimination algorithm achieves regret*

$$\mathbb{E}[R(t)] = O(\sqrt{Kt \log T}) \quad \textit{for all rounds } t \leq T.$$

Theorem 1.9. *Successive Elimination algorithm achieves regret*

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

The existence of logarithmic regret bounds is benefit of adaptive exploration compared to non-adaptive exploration.

Adaptive exploration

UCB1 Algorithm

1 Try each arm once;
2 In each round t, pick $\operatorname{argmax}_{a \in \mathcal{A}} \mathtt{UCB}_t(a)$, where $\mathtt{UCB}_t(a) = \bar{\mu}_t(a) + r_t(a)$;

Algorithm 1.5: UCB1 Algorithm

$$r_t(a) = \sqrt{\frac{\alpha \cdot \ln t}{n_t(a)}}$$

Original version of confidence radius

Theorem 1.14. *Algorithm* UCB1 *satisfies regret bounds in (1.9) and (1.11).*

$$\mathbb{E}[R(t)] = O(\sqrt{Kt\log T}) \quad \textit{for all rounds } t \leq T. \tag{1.9}$$

$$\mathbb{E}[R(T)] \leq O(\log T)\left[\sum_{\textit{arms } a \textit{ with } \mu(a) < \mu(a^*)} \frac{1}{\mu(a^*) - \mu(a)}\right]. \tag{1.11}$$

Optimization & Statistical Inference Lab

Adaptive exploration

A regret bound of the form C * f(T), where f() does not depend on the mean rewards μ, and the "constant" C does not depend on T.

Instance-independent
C does not depend on T

Instance-dependent
Otherwise

$$\mathbb{E}[R(T)] \leq \underbrace{O(\log T)}_{f} \underbrace{\left[\sum_{\text{arms } a \text{ with } \mu(a) < \mu(a^*)} \frac{1}{\mu(a^*) - \mu(a)} \right]}_{C}$$

Instance independent

Theorem 2.14. *Fix K, the number of arms. Consider an algorithm such that*

$$\mathbb{E}[R(t)] \leq O(C_{\mathcal{I},\alpha}\, t^{\alpha}) \quad \text{for each problem instance } \mathcal{I} \text{ and each } \alpha > 0. \tag{2.15}$$

Here the "constant" $C_{\mathcal{I},\alpha}$ can depend on the problem instance $\mathcal{I}$ and the α, but not on time t.
Fix an arbitrary problem instance $\mathcal{I}$. For this problem instance:

$$\text{There exists time } t_0 \text{ such that for any } t \geq t_0 \quad \mathbb{E}[R(t)] \geq C_{\mathcal{I}} \ln(t), \tag{2.16}$$

for some constant $C_{\mathcal{I}}$ that depends on the problem instance, but not on time t.

Theorem 2.16. *For each problem instance $\mathcal{I}$ and any algorithm that satisfies (2.15),*

(a) the bound (2.16) holds with

$$C_{\mathcal{I}} = \sum_{a:\, \Delta(a)>0} \frac{\mu^*(1-\mu^*)}{\Delta(a)}.$$

(b) for each $\epsilon > 0$, the bound (2.16) holds with

$$C_{\mathcal{I}} = \sum_{a:\, \Delta(a)>0} \frac{\Delta(a)}{\mathtt{KL}(\mu(a),\, \mu^*)} - \epsilon.$$

Adaptive exploration

Thompson Sampling

Theorem 3.12. *Consider IID bandits with no priors. For Thompson Sampling with both approaches (i) and (ii) we have:* $\mathbb{E}[R(T)] \leq O(\sqrt{kT\log T})$.

Theorem 3.13. *Consider IID bandits with no priors. For Thompson sampling with approach (i),*

$$\mathbb{E}[R(T)] \leq (1+\epsilon)(\log T) \underbrace{\sum_{\substack{\text{arms } a \\ \text{s.t.} \Delta(a)>0}} \frac{\Delta(a)}{KL(\mu(a),\mu^*)}}_{(*)} + \frac{f(\mu)}{\epsilon^2},$$

for all $\epsilon > 0$. *Here* $f(\mu)$ *depends on the reward function* μ, *but not on the* ϵ, *and* $\Delta(a) = \mu(a^*) - \mu(a)$.

Prior work considered two such "fake priors":

(i) independent, uniform priors and 0-1 rewards,

(ii) independent, Gaussian priors and Gaussian unit-variance rewards (so each reward is distributed as $\mathcal{N}(\mu(a), 1)$, where $\mu(a)$ is the mean).

Main definition. For each round t, consider the posterior distribution for the best arm a^*. Formally, it is distribution p_t over arms given by

$$p_t(a) = \mathbb{P}[a = a^* \mid H_t] \quad \text{for each arm } a. \tag{3.1}$$

Thompson Sampling is a very simple algorithm:

$$\text{In each round } t, \text{ arm } a_t \text{ is drawn independently from distribution } p_t. \tag{3.2}$$

Sometimes we will write $p_t(a) = p_t(a|H_t)$ to emphasize the dependence on history H_t.

Alternative characterization. Thompson Sampling can be stated differently: in each round t,

1. sample reward function μ_t from the posterior distribution $\mathbb{P}_t(\mu) = \mathbb{P}(\mu|H_t)$.
2. choose the best arm $\tilde{a}_t$ according to μ_t.

Let us prove that this characterization is in fact equivalent to the original algorithm.

Lemma 3.1. *For each round t and each history* H_t, *arms* a_t *and* $\tilde{a}_t$ *are identically distributed.*

Proof. For each arm a we have:

$$\begin{aligned}
\Pr(\tilde{a}_t = a) &= \mathbb{P}_t(\text{arm } a \text{ is the best arm}) && \text{by definition of } \tilde{a}_t \\
&= \mathbb{P}(\text{arm } a \text{ is the best arm}|H_t) && \text{by definition of the posterior } \mathbb{P}_t \\
&= p_t(a|H_t) && \text{by definition of } p_t.
\end{aligned}$$

Thus, $\tilde{a}_t$ is distributed according to distribution $p_t(a|H_t)$. □