Multi-Armed Bandits

Myeonghun Yu

University of California—San Diego

January 26, 2024

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

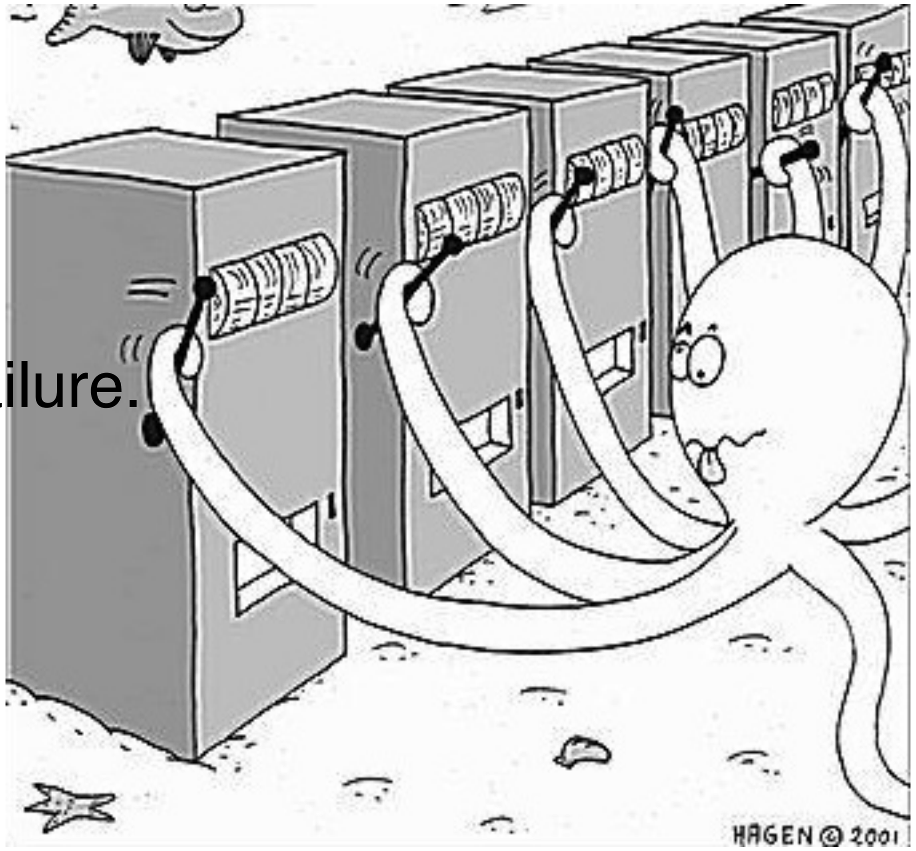
Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

Introduction

Motivating example (Bernoulli Bandit)



There are A actions

If we pick an action, we receive either a success or a failure.

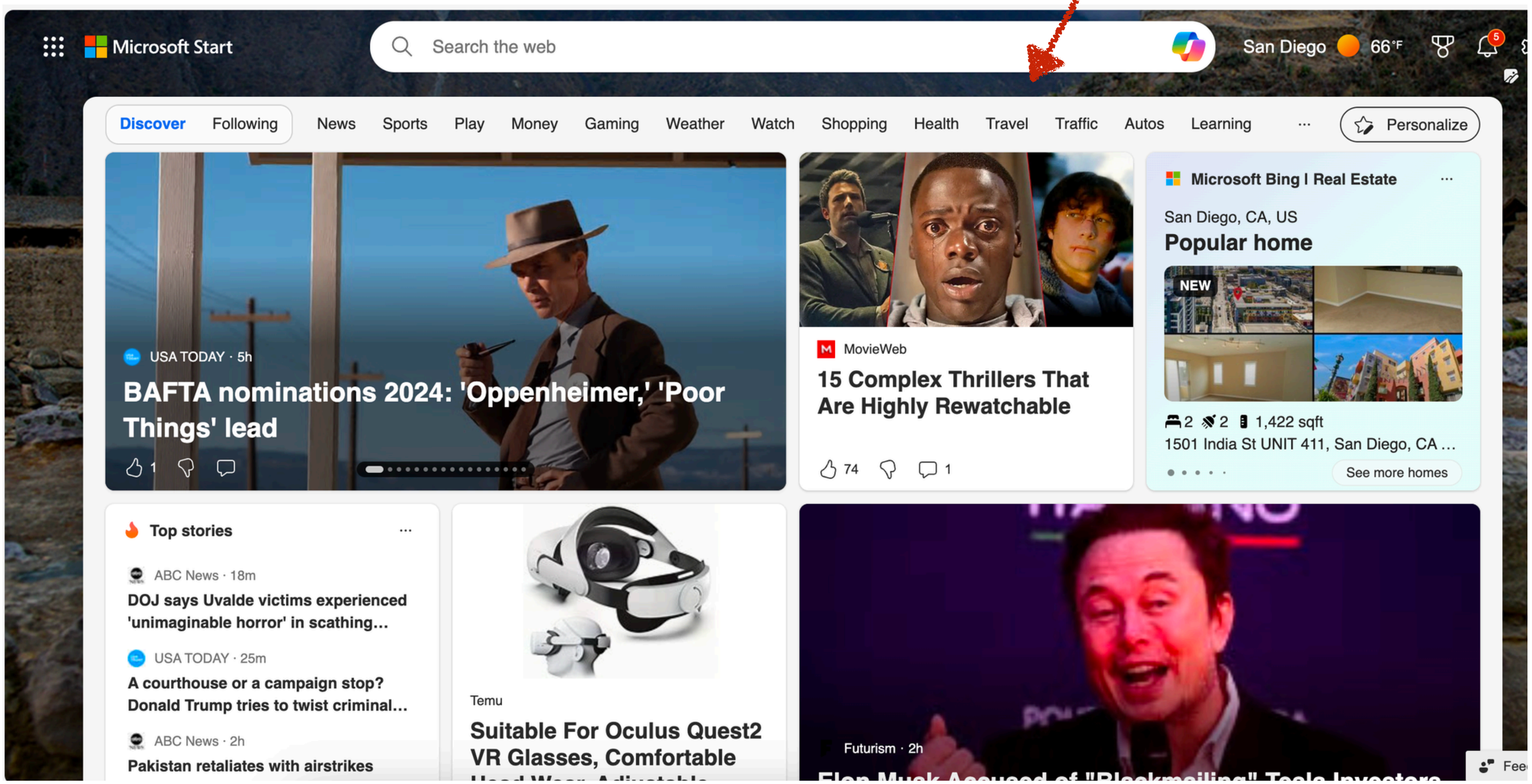
Each action a $(1 \leq a \leq k)$ produces a success with unknown probability $\theta_a \in [0,1]$.

Want to maximize the cumulative number of successes over T periods.

Introduction

Motivating example (Bernoulli Bandit)

Contextual bandit…



Web-browser (Edge) should choose which banner ads (arms) should be displayed.

A success is associated either with a click on the ad.

θ_a represents the click rate among the population of users who uses this browser.

Introduction

More formally….

We consider a stochastic bandit, which is a collection of distributions $\{\mathbb{P}_a : a \in \mathscr{A}\}$, where $\mathscr{A}$ is the set of available actions.

The learner and the environment (Nature) **interact sequentially** over T rounds.

For each round $t \in \{1, 2, \ldots, T\}$, the learner chooses an action $A_t \in \mathscr{A}$.

The environment samples a reward $R_t \in \mathbb{R}$ from a distribution $\mathbb{P}_{A_t}$ and reveals it to the learner.

Introduction

More formally….

We consider a stochastic bandit, which is a collection of distributions $\{\mathbb{P}_a : a \in \mathscr{A}\}$, where $\mathscr{A}$ is the set of available actions.

The (unknown) conditional distribution $R_t \,|\, A_1, R_1, \ldots, R_{t-1}, A_t$ is $\mathbb{P}_{A_t}$.

The (learner-chosen) conditional law of action A_t given $A_1, X_1, \ldots, A_{t-1}, X_{t-1}$ is

$$\pi_t(\,\cdot\,|\, A_1, X_1, \ldots, A_{t-1}, X_{t-1})$$

Introduction

Regret

We measure the learner's performance via regret to the best action

$$a^\star \in \arg\min_{a \in \mathcal{A}} \mathbb{E}[R_t \,|\, A_t = a] = \mathbb{E}_a[R] = \mu_a,$$

$$\mathrm{Reg}(\pi) = T \cdot \mathbb{E}[R \,|\, A = a^\star] - \sum_{t=1}^{T} \mathbb{E}[R_t]$$

Here, π is implicitly included in the RHS, that is, R_t is generated by following the policy π.

Introduction

Regret

$$\mathrm{Reg}(\pi) = T \cdot \mathbb{E}[R \,|\, A = a^\star] - \sum_{t=1}^{T} \mathbb{E}[R_t]$$

Goal: Develop algorithms that enjoy *sublinear regret*, i.e.

$$\frac{1}{T}\mathrm{Reg}(\pi) \to 0, \qquad T \to \infty\,.$$

Important Principle: Exploit vs Explore

We do not know the reward for each arm at the initial time.

$\Rightarrow$ Algorithms should discover the action/arm with the largest mean using the data.

Introduction

Simple Greedy-Algorthim which exemplifies the need for exploration.

At time t, we compute an empirical estimate for the reward mean of an action a

$$\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{s \le t} R_s 1(A_s = a), \qquad N_a(t) = \sum_{s \le t} 1(A_s = a)\,.$$

$$A_t = \arg\max_{a \in \mathcal{A}} \hat{\mu}_a(t-1)$$

$\mathcal{A} = \{1,2\}$. Decision 1 gives $1/2$, and Decision 2 gives $\mathrm{Ber}(3/4)$.

After initializing by playing each decision a single time to ensure $N_a > 0$, the algorithm will get stuck on Decision 1 with probability 1/4, leading to regret $\Omega(T)$.

Introduction

Decomposition of the Regret

$$\mathrm{Reg}(\pi) = T \cdot \underbrace{\mathbb{E}[R \,|\, A = a^\star]}_{=:\mu_{a^\star} =: \mu^\star} - \sum_{t=1}^{T} \mathbb{E}[R_t]$$

Define $\Delta_a = \mu^\star - \mu_a$, sub optimality gap or action gap or immediate regret.

$$\mathrm{Reg}(\pi) = \sum_{a \in \mathcal{A}} \Delta_a \mathbb{E}\left\{N_a(T)\right\} \qquad N_a(t) = \sum_{s \le t} 1(A_s = a)$$

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

Explore-Then-Commit Algorithm

Explore-Then-Commit (ETC) Algorithm

Explore the problem by playing each arm a fixed number of times, then exploits.

1: **Input** m.
2: In round t choose action

$$A_t = \begin{cases} (t \bmod k) + 1, & \text{if } t \leq mk; \\ \text{argmax}_a \hat{\mu}_a(mk), & t > mk. \end{cases}$$

(ties in the argmax are broken arbitrarily)

Algorithm 1: Explore-then-commit.

$$\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{s \leq t} R_s 1(A_s = a), \qquad N_a(t) = \sum_{s \leq t} 1(A_s = a) \, .$$

Intuitively…

When m is too small (too exploiting), then an estimate of mean of each arm is not reliable.

When m is too large (too exploring), then we waste times for choosing obviously wrong choice.

Explore-Then-Commit Algorithm

Explore-Then-Commit (ETC) Algorithm

When bandits are 1-subGaussian and $1 \leq m \leq T/k$. Recall $\Delta_a = \mu^\star - \mu_a$.

$$\mathrm{Reg}(\pi_{\mathrm{ETC}}) \leq m \sum_{a=1}^{k} \Delta_a + (T - mk) \sum_{a=1}^{k} \Delta_a \exp\left(- \frac{m\Delta_a^2}{4} \right).$$

This illustrates rigorously the trade-off between exploration and exploitation.

If m is large $\Rightarrow$ The policy explores for too long $\Rightarrow$ The first term increases.

If m is small $\Rightarrow$ The policy exploits too early $\Rightarrow$ It may choose wrong arms, so the second term increases.

Explore-Then-Commit Algorithm

Explore-Then-Commit (ETC) Algorithm

When bandits are 1-subGaussian and $1 \leq m \leq T/k$. Recall $\Delta_a = \mu^\star - \mu_a$.

$$\mathrm{Reg}(\pi_{\mathrm{ETC}}) \leq m \sum_{a=1}^{k} \Delta_a + (T - mk) \sum_{a=1}^{k} \Delta_a \exp\left(- \frac{m\Delta_a^2}{4} \right).$$

If we assume that there are only two arms, and 1 is optimal, $\Delta = \Delta_2$, then

$$\mathrm{Reg}(\pi_{\mathrm{ETC}}) \leq \Delta + C\sqrt{T}$$

when we choose an optimal choice of m as

$$m = \max\left\{ 1, \left\lceil \frac{4}{\Delta^2} \log\left(\frac{T\Delta^2}{4} \right) \right\rceil \right\}$$

Explore-Then-Commit Algorithm

Explore-Then-Commit (ETC) Algorithm

When bandits are 1-subGaussian and $1 \leq m \leq T/k$. Recall $\Delta_a = \mu^\star - \mu_a$.

$$\mathrm{Reg}(\pi_{\mathrm{ETC}}) \leq \Delta + C\sqrt{T} \qquad m = \max\left\{1, \left\lceil \frac{4}{\Delta^2} \log\left(\frac{T\Delta^2}{4}\right) \right\rceil \right\}$$

⚠️ Caveat…..

The regret bound is close to optimal, but to achieve this, we need to know

1. The knowledge of the horizon T, so it is not an online setting.
2. The knowledge of the sub optimality gap Δ, which is not (obviously) unknown.

Explore-Then-Commit Algorithm

Explore-Then-Commit (ETC) Algorithm

When bandits are 1-subGaussian and $1 \leq m \leq T/k$. Recall $\Delta_a = \mu^\star - \mu_a$.

$$\mathrm{Reg}(\pi_{\mathrm{ETC}}) = \sum_{a=1}^{k} \Delta_a \mathbb{E}\{N_a(T)\} \leq m \sum_{a=1}^{k} \Delta_a + (T - mk) \sum_{a=1}^{k} \Delta_a \exp\left(-\frac{m\Delta_a^2}{4} \right).$$

Proof

$$\begin{aligned}
\mathbb{E}\{N_a(T)\} &= m + (T - mk)\mathbb{P}(A_{mk+1} = a) \\
&\leq m + (T - mk)\mathbb{P}\left\{ \hat{\mu}_a(mk) \geq \max_{j \neq a} \hat{\mu}_j(mk) \right\}
\end{aligned}$$

$$\begin{aligned}
\mathbb{P}\left\{ \hat{\mu}_a(mk) \geq \max_{j \neq a} \hat{\mu}_j(mk) \right\} &\leq \mathbb{P}\{\hat{\mu}_a(mk) \geq \hat{\mu}_1(mk)\} \\
&= \mathbb{P}\{\hat{\mu}_a(mk) - \mu_a - \hat{\mu}_1(mk) + \mu_1 \geq \Delta_a\} \\
&\leq \exp\left(-\frac{m\Delta_a^2}{4} \right)
\end{aligned}$$

Outline

ϵ-Greedy Algorithm

ϵ-Greedy Algorithm

Let $\epsilon \in (0,1)$ be the exploration parameter.

1. At each time $t+1, \quad t \geq 0$, we compute the estimated reward values for each arm $1 \leq a \leq k$,

$$\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{s \leq t} R_s 1(A_s = a), \qquad N_a(t) = \sum_{s \leq t} 1(A_s = a)\,.$$

2. With probability $1 - \epsilon$, the algorithm choose the greedy decision

$$A_{t+1} = \arg\max_{a \in \mathcal{A}} \hat{\mu}_a(t)$$

3. With probability ϵ,

$$A_{t+1} \sim \mathrm{Unif}(\{1,2,\ldots,k\})$$

ϵ-Greedy Algorithm

ϵ-Greedy Algorithm

It allows (forces) the learner to get information uniformly for all arms.

But the algorithm continually explores all arms, even though we may expect or be certain to rule out some actions with very low reward after a relatively small amount of explorations.

ϵ-Greedy Algorithm

Sublinearity of ϵ-Greedy Algorithm

Assume that $\mu^\star = \mu_1 \in [0,1]$, and R_t is subGaussian. Then, for any T, by choosing ϵ appropriately, the ϵ-Greedy algorithm ensures that with probability at least $1-\delta$,

$$\widehat{\text{Reg}} = T \cdot \mu^\star - \sum_{t=1}^{T} \mathbb{E}_{A_t \sim \pi^t}\left(R_{A_t}\right) \lesssim k^{1/3} T^{2/3} \log^{1/3}(kT/\delta)$$

Proof

For convenience, we denote $\hat{A}_{t+1} \in \arg\max_{a \in \mathscr{A}} \hat{\mu}_t(a)$. Then,

$$\widehat{\text{Reg}} = (1-\epsilon)\sum_{t=1}^{T} \mu^\star - \mu_{\hat{A}_t} + \epsilon \sum_{t=1}^{T} \mathbb{E}_{A_t \sim \text{Unif}}(\mu^\star - \mu_{A_t})$$

$$\leq \sum_{t=1}^{T} \mu^\star - \mu_{\hat{A}_t} + \epsilon T$$

ϵ-Greedy Algorithm

Sublinearity of ϵ-Greedy Algorithm

$$
\begin{aligned}
\widehat{\text{Reg}} &= (1-\epsilon)\sum_{t=1}^{T} \mu^\star - \mu_{\hat{A}_t} + \epsilon \sum_{t=1}^{T} \mathbb{E}_{A_t \sim \text{Unif}}(\mu^\star - \mu_{A_t}) \\
&\le \sum_{t=1}^{T} \mu^\star - \mu_{\hat{A}_t} + \epsilon T
\end{aligned}
$$

Now, fix t. By the definition of $\hat{A}_t$, we get

$$
\begin{aligned}
\mu^\star - \mu_{\hat{A}_t} &= \mu_1 - \hat{\mu}_a(t-1) + \hat{\mu}_a(t-1) - \hat{\mu}_{\hat{A}_t}(t-1) + \hat{\mu}_{\hat{A}_t}(t-1) - \mu_{\hat{A}_t} \\
&\le 2 \max_{a \in \mathcal{A}} |\, \mu_a - \hat{\mu}_a(t-1)\, |
\end{aligned}
$$

ϵ-Greedy Algorithm

Sublinearity of ϵ-Greedy Algorithm

Now, we show that the event

$$\mathscr{E}_t := \left\{ \max_{a \in \mathscr{A}} |\mu_a - \hat{\mu}_a(t)| \lesssim \sqrt{\frac{k \log(kT/\delta)}{\epsilon t}} \right\}$$

occurs for all t with probability at least $1 - \delta$.

$$\widehat{\text{Reg}} \lesssim \sum_{t=1}^{T} \sqrt{\frac{A \log(AT/\delta)}{\epsilon t}} + \epsilon T$$

$$\leq \sqrt{\frac{AT \log(AT/\delta)}{\epsilon}} + \epsilon T$$

$$\epsilon \asymp \left(\frac{k \log(kT/\delta)}{T} \right)^{1/3}$$

ϵ-Greedy Algorithm

Sublinearity of ϵ-Greedy Algorithm

$$\mathscr{E}_t := \left\{ \max_{a \in \mathscr{A}} |\mu_a - \hat{\mu}_a(t)| \lesssim \sqrt{\frac{k \log(kT/\delta)}{\epsilon t}} \right\}$$

Note the following Hoeffding's inequality:

$$\frac{1}{N} \sum_{t=1}^{N} Z_i - \mathbb{E}[Z] \lesssim \sigma \sqrt{\frac{\log(T/\delta)}{2N}}$$

with probability at least $1 - \delta$, where $N \in \{1,2,\ldots,T\}$ is a random variable.

Now, recall $N_a(t) = \sum_{s \le t} 1(A_s = a)$. Then, with probability at least $1 - \delta$, for all a and t uniformly

$$|\mu_a - \hat{\mu}_a(t)| \le \sqrt{\frac{2 \log(2AT^2/\delta)}{N_a(t-1)}}$$

ϵ-Greedy Algorithm

Thus, it suffices to show that $N_a(t) = \sum_{s \le t} 1(A_s = a)$ is sufficiently large.

Define $e_t \in \{0,1\}$ to be a random variable whose value indicates whether the algorithm explore uniformly at step t.

$$m_a(t) = \sum_{s \le t} 1(A_s = a, e_s = 1) \le N_a(t)$$

which counts the number of $s \le t$ such that we chose a with the exploration step at time s.

ϵ-Greedy Algorithm

$$m_a(t) = \sum_{s \le t} 1(A_s = a, e_s = 1)$$

Let $Z_a(t) = 1(A_t = a, e_t = 1)$, so that $m_a(t) = \sum_{s \le t} Z_a(s)$. Note that $Z_a(t) \sim \mathrm{Ber}(\epsilon/k)$.

Using Bernstein's inequality, and $\mathbb{E}\{Z_a(t)\} = \epsilon t/k$, we have with $1 - 2e^{-u}$,

$$\left| m_a(t) - \frac{\epsilon t}{k} \right| \le \sqrt{2\mathrm{Var}(\mathrm{Ber}(\epsilon/k)tu} + \frac{u}{e} \le \frac{\epsilon t}{2k} + \frac{4u}{3}$$

Setting $u = \log(2kT/\delta)$, and taking union bound, we have for all a and t,

$$m_a(t) \ge \frac{\epsilon t}{2A} - \frac{4\log(2kT/\delta)}{3}$$

ϵ-Greedy Algorithm

$$N_a(t) \geq m_a(t) \geq \frac{\epsilon t}{2A} - \frac{4 \log(2kT/\delta)}{3} \gtrsim \frac{\epsilon t}{k}$$

Thus, we have

$$|\mu_a - \hat{\mu}_a(t)| \leq \sqrt{\frac{2 \log(2AT^2/\delta)}{N_a(t-1)}} \lesssim \sqrt{\frac{k \log(kT/\delta)}{\epsilon t}},$$

which establishes that $\mathscr{E}_t$ occurs with high probability.

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

Upper Confidence Bound Algorithm

Motivation

UCB algorithm is based on the principle of optimism in the face of uncertainty.

In the presence of uncertainty, we take an optimistic view as if the environment is as nice as possible.

Suppose that with ***high probability***, Tesla's stock will increase 5% in a best-case scenario, and decrease -10% in a worst-case scenario.

For an (extremely) optimistic person, she will have a long position.

For an (extremely) pessimistic person, she will have a short position.

In the multi-armed setting, we assign to each arm a value, called the upper confidence bound which is an overestimate of the unknown mean μ_a with high probability.

Upper Confidence Bound Algorithm

Algorithm

1. At round t, the learner calculate the upper confidence bound for each arm a:

$$\mathrm{UCB}_a(t-1,\delta) = \begin{cases} \infty & \text{if } N_a(t-1) = 0 \\ \hat{\mu}_a(t-1) + \sqrt{\dfrac{2\log(1/\delta)}{N_a(t-1)}} & \text{otherwise.} \end{cases}$$

2. Choose action $A_t \in \arg\max_{a \in \mathcal{A}} \mathrm{UCB}_a(t-1,\delta)$.

3. Observe reward R_t, and update upper confidence bounds.

Upper Confidence Bound Algorithm

Algorithm

$$\mathrm{UCB}_a(t-1,\delta) = \begin{cases} \infty & \text{if } N_a(t-1) = 0 \\ \hat{\mu}_a(t-1) + \sqrt{\dfrac{2\log(1/\delta)}{N_a(t-1)}} & \text{otherwise.} \end{cases}$$

The algorithm will choose arm a at round t if

(i) it is promising because $\hat{\mu}_a(t-1)$ is large, or

(ii) it is not well explored because $N_a(t-1)$ is small.

Upper Confidence Bound Algorithm

Theoretical Guarantee

Assume the random variables are subGaussian, and choose the confidence level $\delta = 1/T^2$.

$$\text{Reg} \leq 3\sum_{a=1}^{k} \Delta_a + \sum_{a:\Delta_a>0} \frac{16\log(T)}{\Delta_a}$$

Introduction of some notation

Let $(R_{ta})_{t\geq 1, 1\leq a\leq k}$ be a collection of independent random variables, $R_{ta} \sim R \,|\, A = a$

$$\hat{\mu}_{as} = \frac{1}{s}\sum_{t=1}^{s} X_{ta}$$

Then, the reward in round t is $R_t = R_{N_a(t)A_t}$, $\hat{\mu}_a(t) = \hat{\mu}_{aN_a(t)}$.

Upper Confidence Bound Algorithm

Proof

As it is before, we start from $\mathrm{Reg} = \sum_{a=1}^{k} \Delta_a \mathbb{E}\{N_a(t)\}$, so it suffices to bound the expectation of counts.

$$G_a = \left\{ \mu_1 < \min_{1 \le t \le T} \mathrm{UCB}_1(t, \delta) \right\} \bigcap \left\{ \hat{\mu}_{au_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} < \mu_1 \right\}$$

Here, u_a is a constant to be determined later.

1. Under G_a, μ_1 is never underestimated by the upper confidence bound for all time.

2. Under G_a, after u_a observations of rewards from the arm a, the UCB is below the mean of the best arm 1.

Upper Confidence Bound Algorithm

Proof

$$G_a = \left\{ \mu_1 < \min_{1 \le t \le T} \mathrm{UCB}_1(t, \delta) \right\} \bigcap \left\{ \hat{\mu}_{au_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} < \mu_1 \right\}$$

Note that when G_a occurs, arm a will be selected at most u_a times, $N_a(T) \le u_a$.

$\because$ Suppose $N_a(T) > u_a$, then $\exists t \in [T]$ s.t. $N_a(t-1) = u_a$ and $A_t = a$.

$$\begin{aligned} \mathrm{UCB}_a(t-1,\delta) &= \hat{\mu}_a(t-1) + \sqrt{\frac{2\log(1/\delta)}{N_a(t-1)}} \\ &= \hat{\mu}_a(t-1) + \sqrt{\frac{2\log(1/\delta)}{u_a}} \\ &< \mu_1 < \mathrm{UCB}_1(t-1,\delta) \end{aligned}$$

Then… the arm a cannot be chosen at the round t, contraction.

Upper Confidence Bound Algorithm

Proof $$G_a = \left\{ \mu_1 < \min_{1 \le t \le T} \mathrm{UCB}_1(t,\delta) \right\} \bigcap \left\{ \hat{\mu}_{au_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} < \mu_1 \right\}$$

Calculate the probability $\mathbb{P}(G_a^c)$.

$$\mathbb{P}\left\{ \mu_1 \ge \min_{t \in [T]} \mathrm{UCB}_1(t,\delta) \right\} \le \mathbb{P}\left[\bigcup_{s \in [T]} \left\{ \mu_1 \ge \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}} \right\} \right]$$

$$\le \sum_{s=1}^{T} \mathbb{P}\left\{ \mu_1 \ge \hat{\mu}_{1s} + \sqrt{\frac{2\log(1/\delta)}{s}} \right\} \le n\delta \,.$$

Upper Confidence Bound Algorithm

Proof $$G_a = \left\{ \mu_1 < \min_{1 \le t \le T} \mathrm{UCB}_1(t, \delta) \right\} \bigcap \left\{ \hat{\mu}_{au_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} < \mu_1 \right\}$$

Calculate the probability $\mathbb{P}(G_a^c)$.

We assume that u_a is chosen large enough that

$$\Delta_a - \sqrt{\frac{2\log(1/\delta)}{u_a}} \ge \frac{1}{2}\Delta_a\,.$$

We choose the smallest integer satisfying the inequality, so that $u_a = \left\lceil \frac{8\log(1/\delta)}{\Delta_a^2} \right\rceil$

$$\mathbb{P}\left\{ \hat{\mu}_{au_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} \ge \mu_1 \right\} = \mathbb{P}\left\{ \hat{\mu}_{au_a} - \mu_a \ge \Delta_a - \sqrt{\frac{2\log(1/\delta)}{u_a}} \right\}$$

$$\le \mathbb{P}(\hat{\mu}_{au_a} - \mu_a \ge \frac{1}{2}\Delta_a) \le \exp\left(- \frac{u_a \Delta_a^2}{8} \right).$$

Upper Confidence Bound Algorithm

Proof $$G_a = \left\{ \mu_1 < \min_{1 \le t \le T} \mathrm{UCB}_1(t, \delta) \right\} \bigcap \left\{ \hat{\mu}_{a u_a} + \sqrt{\frac{2\log(1/\delta)}{u_a}} < \mu_1 \right\}$$

Calculate the probability $\mathbb{P}(G_a^c)$. $\le n\delta + \exp\left(-\frac{u_a \Delta_a^2}{8} \right)$

$$\begin{aligned}
\mathbb{E}\{N_a(T)\} &= \mathbb{E}\{1(G_a)N_a(T)\} + \mathbb{E}\{1(G_a^c)N_a(T)\} \le u_a + \mathbb{P}(G_a^c)T \\
&\le u_a + T\left\{ T\delta + \exp\left(-\frac{u_a \Delta_a^2}{8} \right) \right\} \\
&\le 3 + \frac{16 \log T}{\Delta_a^2}.
\end{aligned}$$

Upper Confidence Bound Algorithm

Bound without inverse of gaps

$$\text{Reg} \le 8\sqrt{Tk\log(T)} + 3\sum_{a=1}^{k} \Delta_a \,.$$

Recall that we obtain

$$\mathbb{E}\{N_a(T)\} \le 3 + \frac{16\log T}{\Delta_a^2}$$

For a truncation level $\Delta > 0$ which will be determined later, we have

$$\begin{aligned}
\text{Reg} = \sum_{a=1}^{k} \Delta_a \mathbb{E}\{N_a(T)\} &= \sum_{a:\Delta_a<\Delta} \Delta_a \mathbb{E}\{N_a(T)\} + \sum_{a:\Delta_a\ge\Delta} \Delta_a \mathbb{E}\{N_a(T)\} \\
&\le T\Delta + \sum_{a:\Delta_a\ge\Delta} \left\{ 3\Delta_a + \frac{16\log T}{\Delta_a} \right\} \\
&\le T\Delta + \frac{16k\log T}{\Delta} + 3\sum_{a} \Delta_a
\end{aligned}$$

Upper Confidence Bound Algorithm

Comparison with ETC algorithm

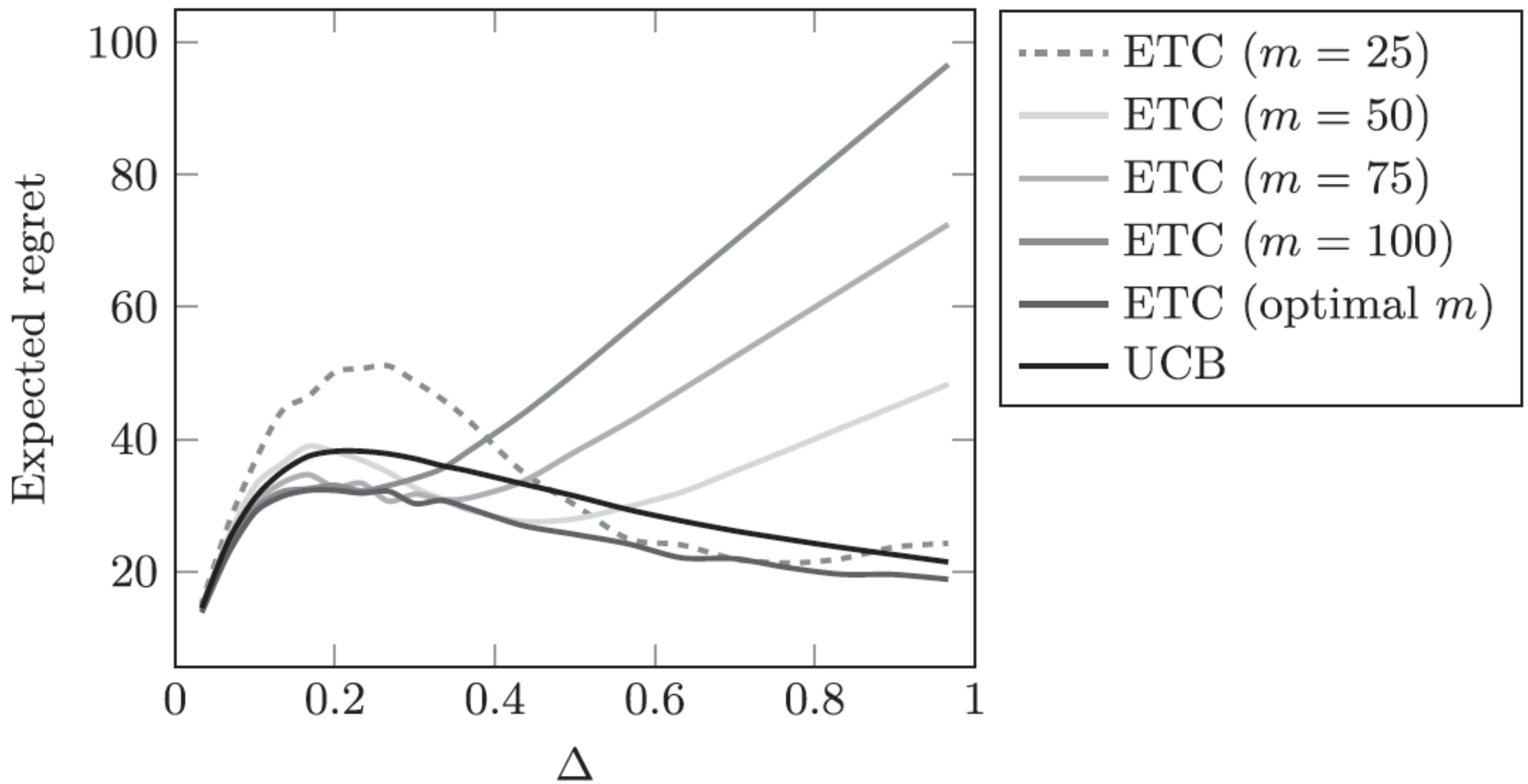


Figure 7.1 Experiment showing universality of UCB relative to fixed instances of ETC

Upper Confidence Bound Algorithm

UCB does not necessitate the knowledge of the true sub-optimality gaps.

But still… the algorithm has to choose $\delta = 1/T^2$, which means that the horizon (end of the round) must be known in advance.

Thus, the algorithm is not appropriate to the online setting.

Upper Confidence Bound Algorithm

Improved UCB algorithm

$$\mathrm{UCB}_a(t-1,\delta) = \begin{cases} \infty & \text{if } N_a(t-1) = 0 \\ \hat{\mu}_a(t-1) + \sqrt{\dfrac{2\log(1+t\log^2(t))}{N_a(t-1)}} & \text{otherwise.} \end{cases}$$

$\log(1/\delta) \Rightarrow \log(1+t\log^2(t))$

The algorithm will choose arm a at round t if

(i) it is promising because $\hat{\mu}_a(t-1)$ is large, or

(ii) it is not well explored because $N_a(t-1)$ is small.

Upper Confidence Bound Algorithm

Improved UCB algorithm

The improved UCB algorithm satisfies

$$\text{Reg} \lesssim \sum_{a=1}^{k} \Delta_a + \sqrt{kT \log T}$$

Note that this algorithm does not require the knowledge of the true suboptimality gaps nor the horizon.

Can we remove the logarithmic term in the regret bound?

Upper Confidence Bound Algorithm

MOSS algorithm

$$\mathrm{UCB}_a(t-1,\delta) = \begin{cases} \infty & \text{if } N_a(t-1)=0 \\ \hat{\mu}_a(t-1) + \sqrt{\dfrac{4}{N_a(t-1)} \log^+\left(\dfrac{T}{kN_a(t-1)}\right)} & \text{otherwise.} \end{cases}$$

Then, it can be shown that

$$\mathrm{Reg} \lesssim \sqrt{kT} + \sum_{a=1}^{k} \Delta_a \,.$$

However, the algorithm is not an ultimate one because

1. it is suboptimal relative to UCB in certain regimes;
2. the variance of the regret of the algorithm is usually too large, so it is unstable.

Outline

Thompson Sampling

History

Thompson Sampling is the first algorithm for bandits proposed by Thompson [1933].

Thompson only considers Bernoulli case with two arms without theoretical guarantees, but Thompson argued the validity intuitively and gave hand-calculated empirical analysis.

For almost 8 decades, it is not popular (unknown) until a large number of authors independently rediscovered the algorithm and establish theoretical guarantees after 2000s.

Thompson Sampling

Simple Example Consider the Bernoulli bandits setting with k arms.

$R \mid A = a \sim \mathrm{Ber}(\mu_a)$ for $1 \le a \le k$.

The learner has a prior belief over each μ_a, e.g., $\mu_a \sim \mathrm{Beta}(\alpha_a, \beta_a)$, which are independent among a.

If $A_t = a$, then we update the distribution of μ_a by the Bayes' rule, remaining the other distributions of $a' \neq a$ the same.

$$(\alpha_a, \beta_a) = \begin{cases} (\alpha_a, \beta_a) & \text{if } A_t \neq a \\ (\alpha_a, \beta_a) + (R_t, 1 - R_t) & \text{if } A_t = a \end{cases}$$

Sample $\hat{\mu}_a \sim \mathrm{Beta}(\alpha_a, \beta_a)$ for each $1 \le a \le k$, then

$$A_{t+1} = \arg\min_{1 \le a \le k} \hat{\mu}_a \, .$$

Thompson Sampling

Difference with the previous algorithm?

$$(\alpha_a, \beta_a) = \begin{cases} (\alpha_a, \beta_a) & \text{if } A_t \neq a \\ (\alpha_a, \beta_a) + (R_t, 1 - R_t) & \text{if } A_t = a \end{cases}$$

Suppose that for all a, $(\alpha_a, \beta_a) = (1,1)$ at the initial step. (Uniform distribution)

Then, $\alpha_a + \beta_a = \sum_{s \leq t} 1(A_s = a) + 2, \quad \alpha_a = \sum_{s \leq t} R_s 1(A_s = a) + 1.$

Thus, Greedy-Algorithm just choose $A_t \in \arg\max_a (\alpha_a - 1)/(\alpha_a + \beta_a - 2)$.

Note that $\mathbb{E}\{Z\} = \alpha_a/(\alpha_a + \beta_a), \quad Z \sim \text{Beta}(\alpha_a, \beta_a)$.

Thompson Sampling

Follow-the-perturbed-leader algorithm

General form of Thompson Sampling (in a Frequentist perspective)

0. Choose $F_{1,1}, F_{2,1}, \ldots, F_{k,1}$ to be the (prior) cumulative distribution functions of the mean reward.

1. For $1 \leq t \leq T$

2. Sample $\theta_a(t) \sim F_{a,t}$ independently for each a.

3. Choose $A_t = \arg\max_a \theta_a(t)$.

4. The reward R_t reveals, and update

$$F_{a,t+1} = F_{a,t} \quad \text{if} \quad a \neq A_t \quad F_{A_t,t+1} = \text{Update}(F_{A_t,t}, A_t, R_t) \quad \text{if} \quad a = A_t$$

Thompson Sampling

Regret bound of Thompson Sampling

Assume the arm 1 is optimal, and let $\epsilon \in \mathbb{R}$ be arbitrary, $a \neq 1$. Then,

$$\mathbb{E}\{N_a(T)\} \leq 1 + \mathbb{E}\left\{ \sum_{s=0}^{T-1} \left(\frac{1}{G_{1,s}} - 1 \right) \right\} + \mathbb{E}\left\{ \sum_{s=0}^{T-1} 1(G_{a,s} > 1/T) \right\}$$

where $G_{a,s} = G_{a,s}(\epsilon) = 1 - F_{a,s}(\mu_1 - \epsilon)$.

The first sum is related to the likelihood that the the sample from the $F_{1,s}$ is nearly optimistic.

$$G_{1,s} = \mathbb{P}(Z > \mu_1 - \epsilon), \quad Z \sim F_{1,s}.$$

Thus, if $G_{1,s}$ is large (the summand in the first sum is small), it is likely to get larger $\theta_1(s)$ with large possibility of $A_s = 1$

Thompson Sampling

Regret bound of Thompson Sampling

Assume the arm 1 is optimal, and let $\epsilon \in \mathbb{R}$ be arbitrary, $a \neq 1$. Then,

$$\mathbb{E}\{N_a(T)\} \leq 1 + \mathbb{E}\left\{\sum_{s=0}^{T-1}\left(\frac{1}{G_{1,s}} - 1\right)\right\} + \mathbb{E}\left\{\sum_{s=0}^{T-1} 1(G_{a,s} > 1/T)\right\}$$

where $G_{a,s} = G_{a,s}(\epsilon) = 1 - F_{a,s}(\mu_1 - \epsilon)$.

The second sum measures the likelihood that the sample from arm a is close to μ_1.

$$G_{a,s} = \mathbb{P}(Z > \mu_1 - \epsilon), \;\; Z \sim F_{a,s}.$$

Thus, if $G_{a,s}$ is small (the summand in the second sum is small), it is likely that $A_s \neq a$.

Thompson Sampling

$$\mathbb{E}\{N_a(T)\} \leq 1 + \mathbb{E}\left\{ \sum_{s=0}^{T-1} \left(\frac{1}{G_{1,s}} - 1 \right) \right\} + \mathbb{E}\left\{ \sum_{s=0}^{T-1} 1(G_{a,s} > 1/T) \right\}$$

Proof

Let $\mathscr{F}_t = \sigma(A_1, R_1, \ldots, A_t, R_t)$ and $E_a(t) = \{\theta_a(t) \leq \mu_1 - \epsilon\}$.

$$\mathbb{P}(\theta_1(t) > \mu_1 - \epsilon \,|\, \mathscr{F}_{t-1}) = G_{1,N_1(t-1)}$$

$$\mathbb{E}\{N_a(T)\} = \mathbb{E}\left\{ \sum_{t=1}^{T} 1(A_t = a) \right\}$$

$$= \mathbb{E}\left\{ \sum_{t=1}^{T} 1(A_t = a, E_a(t)) \right\} + \mathbb{E}\left\{ \sum_{t=1}^{T} 1(A_t = a, E_a^c(t)) \right\}$$

Thompson Sampling

Recall $\mathscr{F}_t = \sigma(A_1, R_1, \ldots, A_t, R_t)$ and $E_a(t) = \{\theta_a(t) \le \mu_1 - \epsilon\}$.

$A_t' = \arg\max_{a \neq 1} \theta_a(t)$.

$$
\begin{aligned}
\mathbb{P}(A_t = 1, E_a(t) \,|\, \mathscr{F}_{t-1}) &\ge \mathbb{P}\{A_t' = a, E_a(t), \theta_1(t) \ge \mu_1 - \epsilon \,|\, \mathscr{F}_{t-1}\} \\
&= \mathbb{P}\{\theta_1(t) \ge \mu_1 - \epsilon \,|\, \mathscr{F}_{t-1}\}\mathbb{P}\{A_t' = a, E_a(t) \,|\, \mathscr{F}_{t-1}\} \\
&\ge \frac{G_{1,N_1(t-1)}}{1 - G_{1,N_1(t-1)}}\mathbb{P}(A_t = a, E_a(t) \,|\, \mathscr{F}_{t-1})\,.
\end{aligned}
$$

Here, the last inequality follows by the observation that if $\{A_t = a\} \cap E_a(t)$ occurs, then $\{A_t' = a\} \cap E_a(t) \cap \{\theta_1(t) \le \mu_1 - \epsilon\}$. That is,

$$
\mathbb{P}(A_t = a, E_a(t) \,|\, \mathscr{F}_{t-1}) \le [1 - \mathbb{P}\{\theta_1(t) > \mu_1 - \epsilon \,|\, \mathscr{F}_{t-1}\}]\mathbb{P}(A_t' = a, E_a(t) \,|\, \mathscr{F}_{t-1})
$$

Thompson Sampling

$$\mathbb{P}(A_t = 1, E_a(t) \,|\, \mathcal{F}_{t-1}) \geq \frac{G_{1,N_1(t-1)}}{1 - G_{1,N_1(t-1)}} \mathbb{P}(A_t = a, E_a(t) \,|\, \mathcal{F}_{t-1}) \,.$$

Thus, summing up the probabilities, we have

$$\mathbb{E}\left[\sum_{t=1}^{T} 1\{A_t = a, E_a(t)\}\right] \leq \mathbb{E}\left[\sum_{t=1}^{T} \left(\frac{1}{G_{1,N_1(t-1)}} - 1\right) 1(A_t = 1)\right]$$

$$\leq \mathbb{E}\left\{\sum_{s=0}^{T-1} \left(\frac{1}{G_{1,s}} - 1\right)\right\}$$

Here, the last step follows from the fact that if $N_1(t-1) = s, 1(A_t = 1)$, then $N_1(t) = s + 1 \neq s$.

Thompson Sampling

$$\mathbb{E}\{N_a(T)\} = \mathbb{E}\left\{\sum_{t=1}^{T} 1(A_t = a, E_a(t))\right\} + \mathbb{E}\left\{\sum_{t=1}^{T} 1(A_t = a, E_a^c(t))\right\}$$

Now, to bound the second term, define the following subset

$$\mathcal{T} = \{t \in [T] : 1 - F_{a,N_a(t-1)}(\mu_1 - \epsilon) > 1/T\}$$

and recall that $G_{a,s} = 1 - F_{a,s}(\mu_1 - \epsilon)$. Then,

$$\sum_{t \in \mathcal{T}} 1(A_t = a) \leq \sum_{s=1}^{T} 1\{G_{a,s-1} > 1/T\}$$ $\because$ by definition... only one s

$E_a(t) = \{\theta_a(t) \leq \mu_1 - \epsilon\}$

$$\mathbb{E}\left[\sum_{t \notin \mathcal{T}} 1\{E_a^c(t)\}\right] \leq \mathbb{E}\left(\sum_{t \notin \mathcal{T}} 1/T\right)$$ $\because$ by the definition of $\mathcal{T}$ and $E_a^c(t)$

Thompson Sampling

Now, to bound the second term, define

$$\mathbb{E}\left[\sum_{t=1}^{T} 1\{A_t = a, E_a^c(t)\}\right\} \leq \mathbb{E}\left\{\sum_{t \in \mathcal{T}} 1(A_t = a)\right\} + \mathbb{E}\left[\sum_{t \notin \mathcal{T}} 1\{E_a^c(t)\}\right]$$

$$\leq \mathbb{E}\left[\sum_{s=0}^{T-1} 1\{1 - F_{a,s}(\mu_1 - \epsilon) > 1/T\}\right] + \mathbb{E}\left(\sum_{t \notin \mathcal{T}} 1/T\right)$$

$$\leq \mathbb{E}\left\{\sum_{s=0}^{T-1} 1(G_{a,s} > 1/T)\right\} + 1.$$

Thompson Sampling

$$\mathbb{E}\{N_a(T)\} \leq 1 + \mathbb{E}\left\{ \sum_{s=0}^{T-1} \left(\frac{1}{G_{1,s}} - 1 \right) \right\} + \mathbb{E}\left\{ \sum_{s=0}^{T-1} 1(G_{a,s} > 1/T) \right\}$$

How…. can we use this general result?

One example

Choose $F_{a,1} = \delta_\infty$ to be the Dirac measure at infinity and let $\mathrm{Update}(F_{a,t}, A_t, R_t)$ be the cumulative distribution function of the Gaussian $\mathcal{N}(\hat{\mu}_a(t), 1/t)$. Moreover, assume that the reward follows a sub-gaussian distribution. Then,

$$\mathrm{Reg} \lesssim \sqrt{kT \log T}$$

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

Exp3 Algorithm

Adversarial Bandits

Abandon almost all assumptions on the data-generating process compared to the stochastic bandit setting.

A k-armed adversarial bandit is an arbitrary sequence of reward vectors $(r_t)_{t=1}^{T}$, where $r_t \in [0,1]^k$

In each round, the learner chooses a distribution over the actions $[k]$, and receives $R_t = r_{t,A_t}$, that is, A_t-th component of the vector r_t.

The regret for given reward vectors is the expected loss in revenue of the policy relative to ***the best fixed action***.

$$\text{Reg}(\pi, x) = \max_{a \in [k]} \sum_{t=1}^{T} r_{t,a} - \mathbb{E}\left(\sum_{t=1}^{T} R_t \right)$$

Exp3 Algorithm

Adversarial Bandits

The worst-case regret over all environment is

$$\mathrm{Reg}(\pi) = \sup_{\mathbf{r} \in [0,1]^{T \times k}} \mathrm{Reg}(\pi, \mathbf{r})$$

By the adversarial property, it can be shown that $\mathrm{Reg}(\pi) \geq T(1 - 1/k)$ for any deterministic algorithm such as ETC, UCB, Greedy, and Thompson.

Thus, the sublinear worst-case regret is only attainable by using a randomized policy.

Exp3 Algorithm

Exponential-weighted algorithm for Exploration and Exploitation (Exp3) Algorithm

We need to determine $P_{t,a} = \mathbb{P}_\pi(A_t = a \,|\, A_1, R_1, \ldots, A_{t-1}, R_{t-1})$

Let $\hat{R}_{s,a}$ be any unbiased estimator of $R_{s,a}$ and let $\hat{S}_{t,a} = \sum_{s=1}^{t} \hat{R}_{s,a}$.

Then, we determine the probability with exponentially weighting with some learning rate $\eta > 0$.

$$P_{t,a} = \frac{\exp(\eta \hat{S}_{t-1,a})}{\sum_{a' \in \mathcal{A}} \exp(\eta \hat{S}_{t-1,a'})}$$

In the following, we will choose $\hat{R}_{t,a} = 1 - \frac{1(A_t = a)}{P_{ta}}(1 - R_t)$. ≤ 1

Exp3 Algorithm

Exponential-weighted algorithm for Exploration and Exploitation (Exp3) Algorithm

Input: n, k, η

Set $\hat{S}_{0i} = 0$ for all i

for $t = 1, \ldots, n$ **do**

  Calculate the sampling distribution P_t:

$$P_{ti} = \frac{\exp\left(\eta \hat{S}_{t-1,i}\right)}{\sum_{j=1}^{k} \exp\left(\eta \hat{S}_{t-1,j}\right)}$$

  Sample $A_t \sim P_t$ and observe reward X_t

  Calculate $\hat{S}_{ti}$:

$$\hat{S}_{ti} = \hat{S}_{t-1,i} + 1 - \frac{\mathbb{I}\{A_t = i\}(1 - X_t)}{P_{ti}}$$

end for

Exp3 Algorithm

Regret Analysis of Exp3

Let $r \in [0,1]^{T \times k}$, $\eta = \sqrt{\log(k)/(Tk)} \in (0,1)$. Then,

$$\mathrm{Reg}(\pi, x) \le 2\sqrt{kT\log(k)}$$

Proof

Note that $\mathrm{Reg}(\pi, x) = \max_{1 \le a \le k} \mathrm{Reg}_a$, $\quad \mathrm{Reg}_a = \sum_{t=1}^{T} r_{t,a} - \mathbb{E}\left(\sum_{t=1}^{T} R_t\right)$.

Thus, for the remainder of the proof, we fix a, say 1

$$\mathbb{E}(\hat{S}_{T,a}) = \sum_{t=1}^{T} r_{t,a}, \quad \text{and} \quad \mathbb{E}_{t-1}(R_t) = \sum_{a=1}^{k} P_{t,a} r_{t,a} = \sum_{a=1}^{k} P_{t,a} \mathbb{E}_{t-1}(\hat{R}_{t,a}).$$

Exp3 Algorithm

Define $\hat{S}_T = \sum_t \sum_a P_{t,a}\hat{R}_{t,a}$. Then, by the above property,

$$Reg_a = \mathbb{E}(\hat{S}_{T,a}) - \mathbb{E}\left(\sum_t \sum_a P_{t,a}\hat{R}_{t,a}\right) = \mathbb{E}(\hat{S}_{T,a} - \hat{S}_T)$$

To bound the RHS, let $W_t = \sum_{a=1}^{k} \exp(\eta \hat{S}_{t,a})$. By convention, $\hat{S}_{0,a} = 0, W_0 = k$.

$$\exp(\eta \hat{S}_{T,1}) \leq \sum_{a=1}^{k} \exp(\eta \hat{S}_{T,1}) = W_T = W_0 \Pi_{t=1}^{T} \frac{W_t}{W_{t-1}}$$

Exp3 Algorithm

The ratio is written as

$$\frac{W_t}{W_{t-1}} = \sum_{a=1}^{k} \frac{\exp(\eta \hat{S}_{t-1,a})}{W_{t-1}} \exp(\eta \hat{R}_{t,a}) = \sum_{a=1}^{k} P_{t,a} \exp(\eta \hat{R}_{t,a}) .$$

Using $e^x \leq 1 + x + x^2$, for $x \leq 1$ and $1 + x \leq e^x$ for $x \in \mathbb{R}$,

$$\begin{aligned} \frac{W_t}{W_{t-1}} &\leq 1 + \eta \sum_{a=1}^{k} P_{t,a} \hat{R}_{t,a} + \eta^2 \sum_{a=1}^{k} P_{t,a} \hat{R}_{t,a}^2 \\ &\leq \exp\left(\eta \sum_{a=1}^{k} P_{t,a} \hat{R}_{t,a} + \eta^2 \sum_{a=1}^{k} P_{t,a} \hat{R}_{t,a}^2 \right) . \end{aligned}$$

Exp3 Algorithm

$$\exp(\eta \hat{S}_{T,1}) \leq k \exp\left(\eta \hat{S}_T + \eta^2 \sum_t \sum_a P_{t,a} \hat{R}_{t,a}^2\right).$$

$$\hat{S}_{T,1} - \hat{S}_T \leq \frac{\log(k)}{\eta} + \eta \sum_t \sum_a P_{t,a} \hat{R}_{t,a}^2 .$$

Exp3 Algorithm

$$\hat{S}_{T,1} - \hat{S}_T \leq \frac{\log(k)}{\eta} + \eta \sum_t \sum_a P_{t,a} \hat{R}_{t,a}^2 . \qquad \hat{R}_{t,a} = 1 - \frac{1(A_t = a)}{P_{ta}}(1 - R_t) .$$

Let $Y_t = 1 - R_t, y_{t,a} = 1 - r_{t,a}$. Then,

$$\mathbb{E}\left(\sum_{a=1}^{k} P_{t,a} \hat{R}_{t,a}^2 \right) = \mathbb{E}\left[\sum_{a=1}^{k} \left\{ P_{t,a} - 2 \cdot 1(A_t = a) Y_t + \frac{1(A_t = a) Y_t^2}{P_{t,a}} \right\} \right]$$

$$= \mathbb{E}\left[1 - 2Y_t + \mathbb{E}_{t-1}\left\{ \sum_{a=1}^{k} \frac{1(A_t = a) Y_t^2}{P_{t,a}} \right\} \right]$$

$$= \mathbb{E}\left[1 - 2Y_t + \mathbb{E}_{t-1}\left\{ \sum_{a=1}^{k} \frac{1(A_t = a) y_{t,a}^2}{P_{t,a}} \right\} \right]$$

$$= \mathbb{E}\left\{ 1 - 2Y_t + \sum_{a=1}^{k} y_{t,a}^2 \right\} = \mathbb{E}\left\{ (1 - Y_t)^2 + \sum_{a \neq A_t} y_{t,a}^2 \right\} \leq k .$$

Outline

Introduction

Explore-Then-Commit Algorithm

ϵ-Greedy Algorithm

Upper Confidence Bound Algorithm

Thompson Sampling

Exp3 Algorithm

One More Thing

One More Thing….

What is the limitation of the previous analysis?

Most analysis only concerns about the bound for the regret expectation.

Actually, many well-developed algorithm turn to yield regrets which have heavy-tails.

Thus, non-asymptotic analysis of previous algorithms, or proposing well-behaving algorithm in a non-asymptotic viewpoint might be another interesting problem.