CMSC 858G: Bandits, Experts and Games 09/12/16

Lecture 4: Lower Bounds (ending); Thompson Sampling

Instructor: Alex Slivkins Scribed by: Guowei Sun,Cheng Jie

1 Lower bounds on regret (ending)

Recap from last lecture. We consider 0-1 rewards and the following family of problem instances:

$$\mathcal{I}_j = \begin{cases} \mu(i) = \frac{1}{2} & \text{for each arm } i \neq j, \\ \mu(i) = (1+\epsilon)/2 & \text{for arm } i = j \end{cases} \quad \text{for each } j = 1, 2, \ldots, K. \tag{1}$$

Here ϵ is a parameter to be adjusted in the analysis. Recall that K is the number of arms.

We considered a "bandits with predictions" problem, and proved that it is impossible to make an accurate prediction with high probability if the time horizon is too small, regardless of what bandit algorithm we use to explore and make the prediction. In fact, we proved it for at least a third of problem instances $\mathcal{I}_j$:

Lemma 1.1. *Suppose $T \leq \frac{cK}{\epsilon^2}$, for a small enough absolute constant c. Fix any deterministic algorithm for "bandits with prediction". Then there exists at least $\lceil K/3 \rceil$ arms j such that*

$$\Pr[y_T = j \mid \mathcal{I}_j] < \tfrac{3}{4}$$

Notation. $\Pr[\,\cdot \mid \mathcal{I}]$ and $\mathcal{E}[\,\cdot \mid \mathcal{I}]$ denote probability and expectation, resp., for problem instance $\mathcal{I}$. As usual, $\mu(a)$ denotes the mean reward of arm a, and $\mu^* = \max_a \mu(a)$ is the best mean reward.

1.1 From "bandits with predictions" to regret

We use Lemma 1.1 to finish our proof of the $\sqrt{KT}$ lower bound on regret.

Theorem 1.2 (Auer et al. (2002))**.** *Fix time horizon T and the number of arms K. For any bandit algorithm, there exists a problem instance such that $\mathbb{E}[R(T)] \geq \Omega(\sqrt{KT})$. In fact, this lower bound holds for a uniform distribution over the instances $\mathcal{I}_j$: for any algorithm we have*

$$\mathop{\mathbb{E}}_{j \sim uniform} [\, \mathbb{E}[R(T) | \mathcal{I}_j] \,] \geq \Omega(\sqrt{KT}). \tag{2}$$

Proof. Fix ϵ in (1) (to be adjusted later), and assume that $T \leq \frac{cK}{\epsilon^2}$, where c is the constant from Lemma 1.1. Fix a bandit algorithm. Let us interpret it as a "bandits with predictions" algorithm by saying that the prediction in each round t is simply a_t, the arm chosen in this round.

Fix round t. Denote

$$S_t = \{\text{arms } j : \Pr[a_t = j | \mathcal{I}_j] \leq \tfrac{3}{4}\},$$

the set of problem instances $\mathcal{I}_j$ with small probability of making a correct prediction in round t. Note that we can apply Lemma 1.1 to round t (then round t becomes the time horizon in Lemma 1.1), to deduce that $|S_t| \geq \frac{K}{3}$.

Let us consider a uniform distribution over problem instances $\mathcal{I}_j$. In what follows, the expectations are over the choice of $\mathcal{I}_j$ (as well as the random rewards and the randomness in the algorithm). Since the problem instances $\mathcal{I}_j$ have mean reward $\mu^* = \frac{1+\epsilon}{2}$ on the best arm a^*, and mean reward $\frac{1}{2}$ on all other arms, we have:

$$\begin{aligned}
\mathbb{E}[\mu(a_t)|a^* \in S_t] &\leq \tfrac{1}{2}\ \Pr[a_t \neq a^*|a^* \in S_t] + \tfrac{1+\epsilon}{2}\ \Pr[a_t = a^*|a^* \in S_t] \\
&= \tfrac{1}{2} + \tfrac{\epsilon}{2}\, p[a_t = a^*|a^* \in S_t] \\
&= \tfrac{1}{2} + \tfrac{\epsilon}{2}\,\tfrac{3}{4} \\
&= \mu^* - \tfrac{\epsilon}{8}. \\
\mathbb{E}[\mu(a_t)] &= \Pr[a^\star \in S_t]\, \mathbb{E}[\mu(a_t)|a^\star \in S_t] + \Pr[a^\star \in S_t]\, \mathbb{E}[\mu(a_t)|a^\star \in S_t] \\
&\leq \mu^* - p[a^* \in S_t]\, \tfrac{\epsilon}{8} \\
&\leq \mu^* - \epsilon/24.
\end{aligned}$$

Now summing up over all rounds, we obtain $\mathbb{E}[R(T)] \geq \epsilon T/24$. Using $\epsilon = \sqrt{\frac{cK}{T}}$, we obtain (2). □

1.2 Per-instance lower bound

In addition to Theorem 1.2, there is another fundamental lower bound on regret. This lower bound applies to any given problem instance and talks about $\log(T)$ regret with an instance-dependent constant, complementing the $\log(T)$ *upper* bound that we proved for algorithms UCB1 and Successive Elimination. We will present and discuss this lower bound without giving a proof.[1]

As before, we focus on 0-1 rewards. For a particular problem instance, we view $\mathbb{E}[R(t)]$ a function of t, and we are interested in how this function grows with t. We start with a simpler and slightly weaker version of the lower bound:

Theorem 1.3. *No algorithm can achieve regret $\mathbb{E}[R(t)] = o(c_\mathcal{I}\ \log t)$ for all problem instances $\mathcal{I}$, where the "constant" $c_\mathcal{I}$ can depend on the problem instance but not on the time t.*

A stronger version is stated as follows:

Theorem 1.4. *Fix K, the number of arms. Consider an algorithm such that*

$$\mathbb{E}[R(t)] \leq O(C_{\mathcal{I},\alpha}\ t^\alpha) \quad \textit{for each problem instance } \mathcal{I} \textit{ and each } \alpha > 0. \tag{3}$$

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:

$$\textit{There exists time } t_0 \textit{ such that for any } t \geq t_0 \quad \mathbb{E}[R(t)] \geq C_\mathcal{I} \ln(t), \tag{4}$$

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

Remark 1.5. The assumption (3) is necessary to rule out trivial counterexamples: *e.g.,* an algorithm which always play arm 1 will have zero regret on a problem instance for which arm 1 is the best arm. Thus, we cannot hope to prove (4) (or any non-trivial lower bound on regret that applies to every problem instance) without ruling out such trivial algorithms.

[1] The proof is also based on a KL-divergence technique. It can be found in the original paper (Lai and Robbins, 1985), as well as in the survey (Bubeck and Cesa-Bianchi, 2012).

Let us specify how the instance-dependent constant $C_{\mathcal{I}}$ in (4) can be chosen. Let $\Delta(a) = \mu^* - \mu(a)$ be the "badness" of arm a.

Theorem 1.6. *For each problem instance $\mathcal{I}$ and any algorithm that satisfies* (3)*,*

(a) the bound (4) *holds with*

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

(b) for each $\epsilon > 0$, the bound (4) *holds with*

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

Remark 1.7. The lower bound from part (a) is similar to the upper bound achieved by UCB1 and Successive Elimination: $R(T) \leq \sum_{a\colon \Delta(a)>0} \frac{O(\log T)}{\Delta(a)}$. In particular, we see that the upper bound is optimal up to a constant factor when μ^* is bounded away from 0 and 1 (*e.g.,* when $\mu^* \in [\frac{1}{4}, \frac{3}{4}]$).

Remark 1.8. Part (b) is a stronger (*i.e.,* larger) lower bound which implies the more familiar form in part (a). Several algorithms in the literature are known to come arbitrarily close to this lower bound. In particular, a version of Thompson Sampling (discussed soon) achieves regret

$$R(t) \leq (1+\delta)\, C_{\mathcal{I}}\, \ln(t) + C'_{\mathcal{I}}/\epsilon^2, \quad \forall \delta > 0,$$

where $C_{\mathcal{I}}$ is from part (b) and $C'_{\mathcal{I}}$ is some other instance-dependent constant.

2 Discussion: bandits with initial information

Sometimes some information about the problem instance is known to the algorithm beforehand; informally, we refer to it as "initial information". When and if such information is available, one would like to use it to improve algorithm's performance. Using the "initial information" has been a subject of much recent work on bandits.

However, how does the "initial information" look like, what is a good theoretical way to model it? Several approaches has been suggested in the literature.

Constrained reward functions. Here the "initial information" is that the reward function[2] must belong to some family $\mathcal{F}$ of feasible reward functions with nice properties. Several examples are below:

- $\mathcal{F}$ is a product set: $\mathcal{F} = \prod_{\text{arms a}} I_a$, where $I_a \subset [0,1]$ is the interval of possible values for $\mu(a)$, the mean reward of arm a. Then each $\mu(a)$ can take an arbitrary value in this interval, regardless of the other arms.
- one good arm, all other arms are bad: *e.g.,* , the family of instances $\mathcal{I}_j$ from the lower bound proof.

[2] Recall that the *reward function* μ maps arms to its mean rewards. We can also view μ as a vector $\mu \in [0,1]^K$.

- "embedded" reward functions: each arm corresponds to a point in $\mathbb{R}^d$, so that the set of arms $\mathcal{A}$ is interpreted as a subset of $\mathbb{R}^d$, and the reward function maps real-valued vectors to real numbers. Further, some assumption is made on these functions. Some of the typical assumptions are: $\mathcal{F}$ is all linear functions, $\mathcal{F}$ is all concave functions, and $\mathcal{F}$ is a Lipschitz function. Each of these assumptions gave rise to a fairly long line of work.

From a theoretical point of view, we simply assume that $\mu \in \mathcal{F}$ for the appropriate family $\mathcal{F}$ of problem instances. Typically such assumption introduces dependence between arms, and one can use this dependence to infer something about the mean reward of one arm by observing the rewards of some other arms. In particular, Lipschitz assumption allows only "short-range" inferences: one can learn something about arm a only by observing other arms that are not too far from a. Whereas linear and concave assumptions allow "long-range" inferences: it is possible to learn something about arm a by observing arms that lie very far from a.

When one analyzes an algorithm under this approach, one usually proves a regret bound for each $\mu \in \mathcal{F}$. In other words, the regret bound is only as good as the *worst case* over $\mathcal{F}$. The main drawback is that such regret bound may be overly pessimistic: what if the "bad" problem instances in $\mathcal{F}$ occur very rarely in practice? In particular, what if most of instances in $\mathcal{F}$ share some nice property such as linearity, whereas a few bad-and-rare instances do not.

Bayesian bandits. Another major approach is to represent the "initial information" as a distribution $\mathbb{P}$ over the problem instances, and assume that the problem instance is drawn independently from $\mathbb{P}$. This distribution is called "prior distribution", or "Bayesian prior", or simply a "prior". One is typically interested in *Bayesian regret*: regret in expectation over the prior. This approach a special case of *Bayesian models* which are very common in statistics and machine learning: an instance of the model is sampled from a prior distribution which (typically) is assumed to be known, and one is interested in performance in expectation over the prior.

A prior $\mathbb{P}$ also defines the family $\mathcal{F}$ of feasible reward functions: simply, $\mathcal{F}$ is the support of $\mathbb{P}$. Thus, the prior can specify the family $\mathcal{F}$ from the "constrained rewards functions" approach. However, compared to that approach, the prior can also specify that some reward functions in $\mathcal{F}$ are more likely than others.

An important special case is *independent priors*: mean reward $(\mu(a) : a \in \mathcal{A})$ are mutually independent. Then the prior $\mathbb{P}$ can be represented as a product $\mathbb{P} = \prod_{\text{arms } a} \mathbb{P}_a$, where $\mathbb{P}_a$ is the prior for arm a (meaning that the mean reward $\mu(a)$ is drawn from $\mathbb{P}_a$). Likewise, the support $\mathcal{F}$ is a product set $\mathcal{F} = \prod_{\text{arms } a} \mathcal{F}_a$, where each $\mathcal{F}_a$ is the set of all possible values for $\mu(a)$. Per-arm priors $\mathbb{P}_a$ typically considered in the literature include a uniform distribution over a given interval, a Gaussian (truncated or not), and just a discrete distribution over several possible values.

Another typical case is when the support $\mathcal{F}$ is a highly constrained family such as the set of all linear functions, so that the arms are very dependent on one another.

The prior can substantially restrict the set of feasible functions that we are likely to see even if it has "full support" (*i.e.,* if $\mathcal{F}$ includes all possible functions). For simple example, consider a prior such that the reward function is linear with probability 99%, and with the remaining probability it is drawn from some distribution with full support.

The main drawback — typical for all Bayesian models — is that the Bayesian assumption (that the problem instance is sampled from a prior) may be very idealized in practice, and/or the "true" prior may not be fully known.

Hybrid approach. One can, in principle, combine these two approaches: have a Bayesian prior over some, but not all of the uncertainty, and use worst-case analysis for the rest. To make this more precise, suppose the reward function μ is fully specified by two parameters, θ and ω, and we have a prior on θ but nothing is known about ω. Then the hybrid approach would strive to prove a regret bound of the following form:

For each ω, the regret of this algorithm in expectation over θ is at most ... .

For a more concrete example, arms could correspond to points in $[0,1]$ interval, and we could have $\mu(x) = \theta \cdot x + \omega$, for parameters $\theta, \omega \in \mathbb{R}$, and we may have a prior on the θ. Another example: the problem instances $\mathcal{I}_j$ in the lower bound are parameterized by two things: the best arm a^* and the number ϵ; so, *e.g.,* we could have a uniform distribution over the a^*, but no information on the ϵ.

3 Bayesian bandits and Thompson Sampling

We consider Bayesian bandits, and discuss an important algorithm for this setting called *Thompson Sampling* (also known as *posterior sampling*). It is the first bandit algorithm in the literature (Thompson, 1933). It is a very general algorithm, in the sense that it is well-defined for an arbitrary prior, and it is known to perform well in practice. The exposition will be self-contained; in particular I will introduce Bayesian concepts as I need them.

3.1 Preliminaries

Bayesian bandits. To recap, Bayesian bandit problem is defined as follows. We start with "bandits with IID rewards" which we have studied before, and make an additional *Bayesian assumption*: the bandit problem instance $\mathcal{I}$ is drawn initially from some known distribution $\mathbb{P}$ over problem instances (called the *prior*). The goal is to optimize *Bayesian regret*, defined as

$$\mathop{\mathbb{E}}_{\mathcal{I}\sim\mathbb{P}} \left[\, \mathbb{E}[R(T)|\mathcal{I}] \,\right],$$

where the inner expectation is the (expected) regret for a given problem instance $\mathcal{I}$, and the outer expectation is over the prior.

Simplifications. We make several assumptions to simplify presentation.

First, we assume that the (realized) rewards come from a *single-parameter family* of distributions: specifically, there is a family of distributions $\mathcal{D}_\nu$ parameterized by a number $\nu \in [0,1]$ such that the reward of arm a is drawn from distribution $\mathcal{D}_\nu$, where $\nu = \mu(a)$ is the mean reward of this arm. Typical examples are 0-1 rewards and Gaussian rewards with unit variance. Thus, the reward distribution for a given arm a is completely specified by its mean reward $\mu(a)$. It follows that the problem instance is completely specified by the reward function μ, and so the prior $\mathbb{P}$ is a distribution over the reward functions.

Second, we assume that there are only finitely many arms, the (realized) rewards can take only finitely many different values, and the prior $\mathbb{P}$ has a finite support. Then we can focus on concepts and arguments essential to Thompson Sampling, rather than worry about the intricacies of probability densities, integrals and such. However, all claims stated below hold hold for arbitrary priors, and the exposition can be extended to infinitely many arms.

Third, we assume that the best arm a^* is unique for each reward function in the support of $\mathbb{P}$.

Notation. Let $\mathcal{F}$ be the support of $\mathbb{P}$, *i.e.,* the set of all feasible reward functions. For a particular run of a particular algorithm on a particular problem instance, let $h_t = (a_t, r_t)$ be the history for round t, where a_t is the chosen arm and r_t is the reward. Let $H_t = (h_1, h_2\,, \ldots\,, h_t)$ be the history up to time t. Let $\mathcal{H}_t$ be the set of all possible histories H_t. As usual, $[t]$ denotes the set $\{1, 2\,, \ldots\,, t\}$.

Sample space. Consider a fixed bandit algorithm. While we defined $\mathbb{P}$ as a distribution over reward functions μ, we can also treat it as a distribution over the sample space

$$\Omega = \{(\mu, H_\infty) : \ \mu \in \mathcal{F},\ H_\infty \in \mathcal{H}_\infty\},$$

the set of all possible pairs (μ, H_t). This is because the choice of μ also specifies (for a fixed algorithm) the probability distribution over the histories. (And we will do the same for any distribution over reward functions.)

Bayesian terminology. Given time-t history H_t, one can define a conditional distribution $\mathbb{P}_t$ over the reward functions by $\mathbb{P}_t(\mu) = \mathbb{P}[\mu|H_t]$. Such $\mathbb{P}_t$ is called the *posterior distribution.* The act of deriving the posterior distribution from the prior is called *Bayesian update.*

Say we have a quantity $X = X(\mu)$ which is completely defined by the reward function μ, such as the best arm for a given μ. One can view X as a random variable whose distribution $\mathbb{P}_X$ is induced by the prior $\mathbb{P}$. More precisely, $\mathbb{P}_X$ is given by $\mathbb{P}_X(x) = \mathbb{P}[X_\mu = x]$, for all x. Such $\mathbb{P}_X$ is called the *prior distribution* for X. Likewise, we can define the conditional distribution $\mathbb{P}_{X,t}$ induced by the posterior $\mathbb{P}_t$: it is given by $\mathbb{P}_{X,t}(x) = \mathbb{P}[X = x|H_t]$ for all x. This distribution is called *posterior distribution* for X at time t.

3.2 Thompson Sampling: definition

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

$$p_t(a) = \mathbb{P}[a = a^\star \mid H_t] \quad \text{for each arm } a. \tag{5}$$

Thompson Sampling is a very simple algorithm:

$$\text{In each round } t\text{, arm } a_t \text{ is drawn independently from distribution } p_t. \tag{6}$$

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 $p_t(a|H_t)$. □

Independent priors. Things get simpler when we have independent priors. (We will state some properties without a proof.) Then for each arm a we have a prior $\mathbb{P}_a$ for the mean reward $\mu(a)$ for this arm, so that the "overall" prior is the product over arms: $\mathbb{P}(\mu) = \prod_{\text{arms a}} \mathbb{P}^a(\mu(a))$. The posterior $\mathbb{P}_t$ is also a product over arms:

$$\mathbb{P}_t(\mu) = \prod_{\text{arms a}} \mathbb{P}^a_t(\mu(a)), \quad \text{where} \quad \mathbb{P}^a_t(x) = \mathbb{P}[\mu(a) = x | H_t]. \tag{7}$$

So one simplification is that it suffices to consider the posterior on each arm separately.

Moreover, the posterior $\mathbb{P}^a_t$ for arm a does not depend on the observations from other arms and (in some sense) it does not depend on the algorithm's choices. Stating this formally requires some care. Let S^a_t be the vector of rewards received from arm a up to time t; it is the n-dimensional vector, $n = n_t(a)$, such that the j-th component of this vector corresponds to the reward received the j-th time arm a has been chosen, for $j \in [n_t(a)]$. We treat S^a_t as a "summary" of the history of arm a. Further, let $Z^a_t \in [0,1]^t$ be a random vector distributed as t draws from arm a. Then for a particular realization of S^a_t we have

$$\mathbb{P}^a_t(x) := \mathbb{P}[\mu(a) = x | H_t] = \mathbb{P}^a\left[\mu(a) = x \mid Z^a_t \text{ is consistent with } S^a_t\right]. \tag{8}$$

Here two vectors v, v' of dimension n and n', resp., are called *consistent* if they agree on the first $\min(n, n')$ coordinates.

One can restate Thompson Sampling for independent priors as follows:

1. for each arm a, sample mean reward $\mu_t(a)$ from the posterior distribution $\mathbb{P}^a_t$.
2. choose an arm with maximal $\mu_t(a)$ (break ties arbitrarily).

3.3 Computational aspects

While Thompson Sampling is mathematically well-defined, the arm a_t may be difficult to compute efficiently. Hence, we have two distinct issues to worry about: algorithm's statistical performance (as expressed by Bayesian regret, for example), and algorithm's running time. It is may be the first time in this course when we have this dichotomy; for all algorithms previously considered, computationally efficient implementation was not an issue.

Ideally, we'd like to have both a good regret bound (RB) and a computationally fast implementation (FI). However, either one of the two is interesting: an algorithm with RB but without FI can serve as a proof-of-concept that such regret bound can be achieved, and an algorithm with FI but without RB can still achieve good regret in practice. Besides, due to generality of Thompson Sampling, techniques developed for one class of priors can potentially carry over to other classes.

A brute-force attempt. To illustrate the computational issue, let us attempt to compute probabilities $p_t(a)$ by brute force. Let $\mathcal{F}_a$ be the set of all reward functions μ for which the best arm is a. Then:

$$p_t(a|H_t) = \frac{\mathbb{P}(a^* = a \,\&\, H_t)}{\mathbb{P}(H_t)} = \frac{\sum_{\mu \in \mathcal{F}_a} \mathbb{P}(\mu) \cdot \Pr[H_t|\mu]}{\sum_{\mu \in \mathcal{F}} \mathbb{P}(\mu) \cdot \Pr[H_t|\mu]}.$$

Thus, $p_t(a|H_t)$ can be computed in time $|\mathcal{F}|$ times the time needed to compute $\Pr[H_t|\mu]$, which may be prohibitively large if there are too many feasible reward functions.

Sequential Bayesian update. Faster computation can sometimes be achieved by using the alternative characterization of Thompson Sampling. In particular, one can perform the Bayesian update *sequentially*: use the prior $\mathbb{P}$ and round-1 history h_1 to compute round-1 posterior $\mathbb{P}_1$; then treat $\mathbb{P}_1$ as the new prior, and use $\mathbb{P}_1$ and round-2 history h_2 to compute round-2 posterior $\mathbb{P}_2$; then treat $\mathbb{P}_2$ as the new prior and so forth. Intuitively, the round-t posterior $\mathbb{P}_t$ contains all relevant information about the prior $\mathbb{P}$ and the history H_t; so once we have $\mathbb{P}_t$, one can forget the $\mathbb{P}$ and the H_t. Let us argue formally that this is a sound approach:

Lemma 3.2. *Fix round t and history H_t. Then $\mathbb{P}_t(\mu) = \mathbb{P}_{t-1}(\mu|h_t)$ for each reward function μ.*

Proof. Let us use the definitions of conditional expectation and posterior $\mathbb{P}_{t-1}$:

$$\mathbb{P}_{t-1}(\mu|h_t) = \frac{\mathbb{P}_{t-1}(\mu \,\&\, h_t)}{\mathbb{P}_{t-1}(h_t)} = \frac{\mathbb{P}(\mu \,\&\, h_t \,\&\, H_{t-1})/\mathbb{P}(H_{t-1})}{\mathbb{P}(h_t \,\&\, H_{t-1})/\mathbb{P}(H_{t-1})} = \frac{\mathbb{P}(\mu \,\&\, H_t)}{\mathbb{P}(H_t)} = \mathbb{P}_t(\mu).$$

□

With independent priors, one can do the sequential update for each arm separately:

$$\mathbb{P}^a_t(\mu(a)) = \mathbb{P}^a_{t-1}(\mu(a)|h_t),$$

and only when this arm is chosen in this round.

3.4 Example: 0-1 rewards and Beta priors

Assume 0-1 rewards and independent priors. Let us provide some examples in which Thompson Sampling admits computationally efficient implementation.

The full t-round history of arm a is denoted H^a_t. We summarize it with two numbers:

$\alpha_t(a)$: the number of 1's seen for arm a till round t,

$n_t(a)$: total number of samples drawn from arm a till round t.

If the prior for each arm is a uniform distribution over finitely many possible values, then we can easily derive a formula for the posterior.

Lemma 3.3. *Assume 0-1 rewards and independent priors. Further, assume that prior $\mathbb{P}^a$ is a uniform distribution over $N_a < \infty$ possible values, for each arm a. Then $\mathbb{P}^a_t$, the t-round posterior for arm a, is given by a simple formula:*

$$\mathbb{P}^a_t(x) = C \cdot x^\alpha (1-x)^{n-\alpha}$$

for every feasible value x for the mean reward $\mu(a)$, where $\alpha = \alpha_t(a)$ and $n = n_t(a)$, and C is the normalization constant.

Proof. Fix arm a, round t, and a feasible value x for the mean reward $\mu(a)$. Fix a particular realization of history H_t, and therefore a particular realization of the summary S^a_t. Let A denote the event in (8): that the random vector Z_t^a is consistent with the summary S_t^a. Then:

$$\begin{aligned}
\mathbb{P}_t^a(x) &= \mathbb{P}[\mu(a)=x|H_t] && \text{By definition of the arm-}a\text{ posterior}\\
&= \mathbb{P}^a[\mu(a)=x \mid A] && \text{by (8)}\\
&= \frac{\mathbb{P}^a(\mu(a)=x \text{ and } A)}{\mathbb{P}^a(A)}\\
&= \frac{\mathbb{P}^a[\mu(a)=x]\cdot \Pr(A \mid \mu(a)=x)}{\mathbb{P}^a(A)}\\
&= \frac{\frac{1}{N_a}\cdot x^\alpha(1-x)^{n-\alpha}}{\mathbb{P}^a(A)}\\
&= Cx^\alpha(1-x)^{n-\alpha}
\end{aligned}$$

for some normalization constant C. □

One can prove a similar result for a prior that is uniform over a $[0,1]$ interval; we present it without a proof. Note that in order to compute the posterior for a given arm a, it suffices to assume 0-1 rewards and uniform prior for this one arm.

Lemma 3.4. *Assume independent priors. Focus on a particular arm a. Assume this arm gives 0-1 rewards, and its prior $\mathbb{P}^a$ is a uniform distribution on $[0,1]$. Then the posterior $\mathbb{P}_t^a$ is a distribution with density*

$$f(x) = \tfrac{(n+1)!\,\alpha!}{(n+\alpha)!}\cdot x^\alpha\,(1-x)^{n-\alpha},\quad \forall x\in[0,1],$$

where $\alpha=\alpha_t(a)$ and $n=n_t(a)$.

A distribution with such density is called a *Beta distribution* with parameters $\alpha+1$ and $n+1$, and denoted $\texttt{Beta}(\alpha+1, n+1)$. This is a well-studied distribution, *e.g.,* see the corresponding Wikipedia page. $\texttt{Beta}(1,1)$ is the uniform distribution on the $[0,1]$ interval.

In fact, we have a more general version of Lemma 3.4, in which the prior can be an arbitrary Beta-distribution:

Lemma 3.5. *Assume independent priors. Focus on a particular arm a. Assume this arm gives 0-1 rewards, and its prior $\mathbb{P}^a$ is a Beta distribution $\texttt{Beta}(\alpha_0, n_0)$. Then the posterior $\mathbb{P}_t^a$ is a Beta distribution $\texttt{Beta}(\alpha+\alpha_0, n+n_0)$, where $\alpha=\alpha_t(a)$ and $n=n_t(a)$.*

Remark 3.6. By Lemma 3.4, starting with $\texttt{Beta}(\alpha_0, n_0)$ prior is the same as starting with a uniform prior and several samples of arm a. (Namely, n_0-1 samples with exactly α_0-1 1's.)

3.5 Example: Gaussian rewards and Gaussian priors

Assume that the priors are Gaussian and independent, and the rewards are Gaussian, too. Then the posteriors are also Gaussian, and their mean and variance can be easily computed in terms of the parameters and the history.

Lemma 3.7. *Assume independent priors. Focus on a particular arm a. Assume this arm gives rewards that are Gaussian with mean $\mu(a)$ and standard deviation $\hat{\sigma}$. Further, suppose the prior $\mathbb{P}^a$ for this arm is Gaussian with mean μ_0^a and standard deviation σ_0^a. Then the posterior $\mathbb{P}_t^a$ is a Gaussian whose mean and variance are determined by the known parameters (μ_0^a, σ_0^a, $\hat{\sigma}$), as well as the average reward and the number of samples from this arm so far.*

Next lecture: regret bounds for Thompson Sampling.

References

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002. Preliminary version in *36th IEEE FOCS*, 1995.

Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning*, 5(1), 2012.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient Adaptive Allocation Rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

William R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285294, 1933.