10 Multi-Armed Bandit

Exercise 10.1

Tell if the following statements about MAB are true or false and provide motivations for your choice.

1. The MDP corresponding to a MAB setting has no state.
2. The MDP corresponding to a MAB setting has no transition probabilities.
3. An algorithm which always uses the greedy choice (maximize the empirical expected reward) might get stuck to suboptimal solutions.
4. We are able to solve the exploration/exploitation problem by considering either upper or lower bounds on the expected rewards.
5. In a frequentist framework, if we base our sequential policy on a MAB by considering only $\hat{R}(a_i)$ we are not using all the information we have about the estimates.
6. Uncertainty over quantities is usually handled by explorative choices in the MAB setting.

Exercise 10.2

Tell if the following problems can be modeled as MAB and explain why.

1. Maze escape;
2. Pricing goods with stock;
3. Pricing goods without stock;
4. Optimal bandwidth allocation (send the largest amount of information without congesting the band);
5. Web ads placement;
6. Weather prediction (with multiple experts).

If they fit the MAB setting, is the environment adversarial or stochastic?

Exercise 10.3

State if the following are applicable to generic RL problems or MAB problems. Motivate your answers.

1. We should take into account state-action pairs one by one to estimate the value function;
2. Past actions you perform on the MDP might influence the future rewards you will gain;
3. The Markov property holds;
4. The time horizon of the episode is finite.

Exercise 10.4

The ε-greedy algorithm selects the best action except for small percentages of times $\varepsilon \in (0, 1)$, where all the actions are considered. Consider a MAB stochastic setting.

1. Is this algorithm converging to the optimal strategy (in some sense)?
2. If not, propose a scheme which has the chance of converging to the optimal solution.
3. Are we in a MAB perspective if we are using this algorithm?

Exercise 10.5

Write the formula for the minimum regret we might have on average over $T = \lceil e^{10} \rceil$ time steps in the case we have a stochastic MAB with 3 arms and expected rewards:

$$R(a_1) = 0.2 \tag{10.1}$$
$$R(a_2) = 0.4 \tag{10.2}$$
$$R(a_3) = 0.7 \tag{10.3}$$

and each distribution $\mathcal{R}(a_i)$ is Bernoulli.

Note that the KL divergence for Bernoulli variables with means p and q is:

$$KL(p, q) = p \log\left(\frac{p}{q}\right) + (1 - p) \log\left(\frac{(1 - p)}{(1 - q)}\right).$$

Hints: $\log(\frac{0.2}{0.7}) = -1.25$, $\log(\frac{0.8}{0.3}) = 0.98$, $\log(\frac{0.4}{0.7}) = -0.56$ and $\log(\frac{0.6}{0.3}) = 0.69$.

Is it possible that your algorithm achieves lower regret? If so, provide an example.

Exercise 10.6

Provide examples of either Bayesian or frequentist MAB algorithm showing the following properties:

1. It incorporates expert knowledge about the problem in the arms;
2. It provides tight theoretical lower bound on the expected regret in the stochastic setting;
3. It provides tight theoretical upper bound on the expected regret in the stochastic setting;
4. At each turn, it modifies only the statistics of the chosen arm.

Motivate your answers.

Exercise 10.7

Consider the following Höeffding bounds (supposed to hold with probability at least δ) for a MAB stochastic setting:

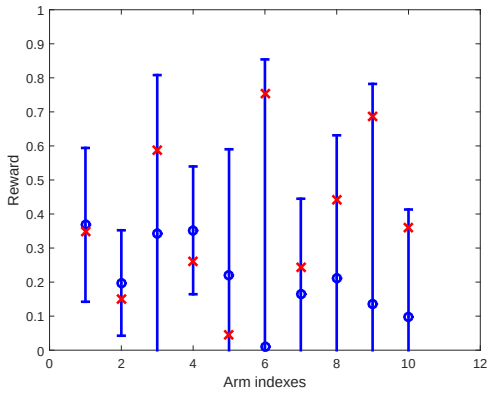

Figure 10.1: Bounds in a MAB with $N = 10$.

where the blue bars are the Höeffding bounds for the expected rewards, the blue circles are the estimated expected rewards, and the red crosses are the real expected rewards.

1. Which arm would a UCB1 algorithm choose for the next round?
2. Do you think that Figure 10.1 might be the results obtained by running UCB1 for several rounds?

3. Which arm will UCB1 converge to if $T \to \infty$?
4. Which arm is the one which we pulled the most so far?

Motivate your answers.

Exercise 10.8

Consider the Thompson Sampling algorithm. Assume to have the following posterior distributions $Beta_i(\alpha_t, \beta_t)$ for arms $\mathcal{A} = \{a_1, \ldots, a_5\}$ rewards, which are distributed as Bernoulli r.v.:

$$
\begin{aligned}
a_1 &: \quad \alpha_t = 1 \quad \beta_t = 5 \\
a_2 &: \quad \alpha_t = 6 \quad \beta_t = 4 \\
a_3 &: \quad \alpha_t = 11 \quad \beta_t = 23 \\
a_4 &: \quad \alpha_t = 12 \quad \beta_t = 33 \\
a_5 &: \quad \alpha_t = 28 \quad \beta_t = 21
\end{aligned}
$$

From these distribution you extract the following samples for the current round:

$$
\begin{aligned}
\hat{r}(a_1) &= 0.63 \\
\hat{r}(a_2) &= 0.35 \\
\hat{r}(a_3) &= 0.16 \\
\hat{r}(a_4) &= 0.22 \\
\hat{r}(a_5) &= 0.7
\end{aligned}
$$

1. Which arm would the TS algorithm play for the next round?
2. What considerations can be done in the case the real distributions of the arm rewards are Gaussian. Is the standard Bernoulli TS algorithm a viable option?
3. Assume we started the TS algorithm with uniform $Beta(1, 1)$ priors. What would UCB1 have chosen in the case of Bernoulli rewards for the next round?

Exercise 10.9

Tell whether the following problems can be modeled as MAB or an MDP (or none). Explain why.

1. Web ads display;
2. Play “Space Invaders”;

3. Weather prediction (with multiple experts);
4. Maze escape.

Exercise 10.10

Consider the UCB1 and the Thompson Sampling algorithms. Tell if the following statements are true for the two aforementioned algorithms and motivate your answers.

1. It requires the knowledge of a pair of conjugate prior/posterior distributions;
2. It manages automatically the exploration/exploitation tradeoff;
3. It relies on the "optimism in the face of uncertainty" paradigm;
4. It is able to incorporate a priori knowledge about the problem.

Exercise 10.11

Consider the Thompson Sampling algorithm. Assume to have the following posterior distributions $Beta_i(\alpha_t, \beta_t)$ for arms $\mathcal{A} = \{a_1, \ldots, a_5\}$ rewards, which are distributed as Bernoulli r.v.:

$$
\begin{array}{llll}
a_1\text{:} & \alpha_t = 1 & \beta_t = 6 & \hat{r}(a_1) = 0.13 \\
a_2\text{:} & \alpha_t = 4 & \beta_t = 6 & \hat{r}(a_2) = 0.55 \\
a_3\text{:} & \alpha_t = 31 & \beta_t = 31 & \hat{r}(a_3) = 0.4 \\
a_4\text{:} & \alpha_t = 8 & \beta_t = 1 & \hat{r}(a_4) = 0.90 \\
a_5\text{:} & \alpha_t = 5 & \beta_t = 7 & \hat{r}(a_5) = 0.95
\end{array}
$$

where $\hat{r}(a_i)$ are random samples extracted from the posterior distributions.

1. Which arm would play the TS algorithm in the next round?
2. Do you think that there is an arm that is more promising to be the best one?
3. What is the UCB1 bound for arms a_3 and a_5? Assume that the Bayesian setting started from uniform $Beta(1, 1)$ priors.

Exercise 10.12

For each one of the following statements, tell if it is true for the UCB1 and/or TS algorithms.

1. It relies on the assumption to know the family of the arms reward distributions (e.g., Bernoulli);

2. At each round, it modifies the statistics of all the arms;
3. It incorporates knowledge about the arms;
4. It is a randomized algorithm.

Exercise 10.13

Consider the Thompson Sampling algorithm. Assume to have the following posterior distributions $Beta_i(\alpha_t, \beta_t)$ for arms $\mathcal{A} = \{a_1, \ldots, a_5\}$ rewards, which are distributed as Bernoulli random variables with mean μ_i, and you extracted from them the samples $\hat{r}(a_i)$:

$$
\begin{array}{llllll}
a_1\text{:} & \alpha_t = 1 & \beta_t = 5 & \hat{r}(a_1) = 0.63 & \mu_1 = 0.1 \\
a_2\text{:} & \alpha_t = 6 & \beta_t = 4 & \hat{r}(a_2) = 0.35 & \mu_2 = 0.5 \\
a_3\text{:} & \alpha_t = 11 & \beta_t = 23 & \hat{r}(a_3) = 0.16 & \mu_3 = 0.3 \\
a_4\text{:} & \alpha_t = 12 & \beta_t = 25 & \hat{r}(a_4) = 0.22 & \mu_4 = 0.2 \\
a_5\text{:} & \alpha_t = 38 & \beta_t = 21 & \hat{r}(a_5) = 0.7 & \mu_5 = 0.6
\end{array}
$$

1. How much pseudo-regret the TS algorithm accumulated so far, assuming we started from uniform $Beta(1, 1)$ priors?
2. Which one of the arm reward posteriors is the most peaked one?
3. What would UCB1 have chosen for the next round? Assume Bernoulli rewards and that in the Bayesian setting we started from uniform $Beta(1, 1)$ priors?

Exercise 10.14

Tell if the following statements about Multi-Armed Bandit are true or false and provide motivations for your choice.

1. The MDP corresponding to a MAB setting has no actions.
2. One of the Bayesian approaches to solve the MAB setting resorts to a sampling procedure from posterior distributions.
3. The use of an ε-greedy algorithm to solve the MAB setting is a viable solution and provides an expected pseudo-regret of order $O(\log(\log(T)))$, where T is the number of rounds.

Exercise 10.15

Tell if the following statements about the Thompson Sampling (TS) algorithm are true or false. Motivate your answers.

1. The more an arm has been pulled, the more the posterior distribution of its reward is peaked;
2. The TS algorithm uses specific information about the reward being Bernoulli distributed;
3. The TS algorithm is a deterministic algorithm.

Exercise 10.16

Tell if the following statements about MAB are true or false. Motivate your answers.

1. In a setting with Gaussian rewards we can use the UCB1 algorithm to have sublinear regret.
2. An arm with a large upper confidence bound is likely to turn out to be the optimal one at the end of the MAB time horizon.
3. A MAB algorithm can be used in an MDP as long as the reward corresponding to the actions have the same distribution in every state.
4. The Thompson Sampling algorithm can be applied only if we have bernoulli rewards (success/failures) as rewards.
5. The design of MAB algorithm different from UCB1 might provide an expected pseudo-regret of order log(log(T)), T being the time horizon.
6. The upper bound on the regret of UCB1 scales linearly with the number of arms.
7. There exist situations in which we are able to obtain a regret lower than the one prescribed by the lower bound for MAB.
8. It is generally a viable option to use an epsilon-greedy exploration strategy on a MAB setting.
9. The selection of the arm provided by TS is stochastic.

Exercise 10.17

Tell if the following statements about MAB are true or false. Motivate your answers.

1. In a setting with Gaussian rewards we can use the TS algorithm to have sublinear regret.
2. The selection of the arm provided by UCB1 is deterministic.
3. In the presence of prior information on the arms rewards, we should use the UCB1

algorithm.

4. The upper bound on the regret of UCB1 and TS scales logarithmically with the time horizon.

5. The design of MAB algorithms different from TS might provides an expected pseudo-regret of order log(log(T)), T being the time horizon.

Exercise 10.18

Consider the Thompson Sampling algorithm applied to an environment with Bernoulli rewards, in which the prior is a Beta distribution initialized as a uniform one: $Beta(1, 1)$.
For each arm $a_i \in A = \{a_0, \ldots, a_4\}$, you are provided its posterior distributions $Beta_i(alpha_t; beta_t)$, the true reward mean mu_i, and the last sample extracted from the Beta $r(a_i)$:

a_0	$\alpha_t = 8,$	$\beta_t = 6,$	$r_t = 0.46,$	$mu_0 = 0.6$
a_1	$\alpha_t = 8,$	$\beta_t = 18,$	$r_t = 0.34,$	$mu_1 = 0.3$
a_2	$\alpha_t = 6,$	$\beta_t = 9,$	$r_t = 0.33,$	$mu_2 = 0.4$
a_3	$\alpha_t = 18,$	$\beta_t = 19,$	$r_t = 0.4,$	$mu_3 = 0.5$
a_4	$\alpha_t = 4,$	$\beta_t = 14,$	$r_t = 0.2,$	$mu_4 = 0.2$

1. How much pseudo-regret did the TS algorithm accumulate so far?

2. What would UCB1 have chosen for the next round?

3. Suppose the next reward can be chosen by an adversary, but keeping the same domain: how could this adversary behave if it knows that the agent is using TS? What about UCB1 instead? (no computations are required)

Exercise 10.19

Consider a MAB algorithm choosing the arm a_t and providing the following rewards R_t over a time horizon of $T = 10$ rounds.

$R_{1,t}$	**0**	1	1	0	**0**	1	0	1	**0**	0
$R_{2,t}$	1	**0**	1	1	1	1	**1**	1	1	**1**
$R_{3,t}$	0	1	**0**	**0**	0	**0**	1	**0**	1	1
a_t	1	2	3	3	1	3	2	3	1	2

(only the reward for the chosen arm is revealed to the algorithm) Moreover, assume that the expected value for the three arms' reward are $\mu_1 = 0.3$, $\mu_2 = 0.8$, and $\mu_3 = 0.6$.

1. Compute the cumulated reward, the regret and the pseudo-regret for the algorithm over the time horizon T. (Recall that the regret is computed over realizations, while the pseudo-regret is computed over expected values);

2. Compute the values for the UCB1 bounds for at time $t = 5$ for the three arms. Do you think that the algorithm used in this setting can be the UCB1?

3. Do you think that the algorithm used in this setting might be the Thompson sampling?

Exercise 10.20

Consider the following snippet of code:

```
for t in range(1, T+1):
pulled = np.argwhere(criterion == criterion.max()).reshape(-1)
reward = rew(pulled)

n_pulls[pulled] = n_pulls[pulled] + 1
exp_payoffs[pulled] = ((exp_payoffs[pulled] *
(n_pulls[pulled] - 1.0) + reward) / n_pulls[pulled])
for k in range(0, n_options)
criterion[k] = exp_payoffs[k] + np.sqrt(2 * t / n_pulls[k])
```

1. Describe the procedure (what algorithm it is implementing) and the purpose (which kind of problem it is solving) of the above snippet of code.

2. Is it correct? In the case the algorithm is not correct, propose a modification to fix the problem. In the case the algorithm is correct, state the theoretical guarantees of such an algorithm.

3. Are there other available methods to solve this problem? Are they requiring some specific assumptions to be applied to the same setting the algorithm in the snippet is used?

Exercise 10.21

Assume to have a stochastic Multi-Armed Bandit (MAB) with 3 arms with average reward:

$$R(a_1) = 0.1,$$
$$R(a_2) = 0.6,$$
$$R(a_3) = 0.3,$$

and each distribution $\mathcal{R}(a_i)$ is a Bernoulli.

1. Write the asymptotic minimum expected pasudo-regret we might have on average over $T = e^{10}$ time steps.

2. If we apply the UCB1 algorithm, which is the upper bound on the expected pseudo-regret? Is it larger or smaller than the previous one?

3. What can we tell about the minimum regret we might have on the above problem? And about the minimum regret of UCB1?

Note that the KL divergence for Bernoulli variables with means p and q is:

$$KL(p,q) = p\log\left(\frac{p}{q}\right) + (1-p)\log\left(\frac{(1-p)}{(1-q)}\right)$$

and (if necessary) that $\log(\frac{0.1}{0.6}) = -1.8$, $\log(\frac{0.9}{0.4}) = 0.8$, $\log(\frac{0.3}{0.6}) = -0.7$, $\log(\frac{0.7}{0.4}) = 0.6$, $\log(\frac{0.1}{0.3}) = -1.1$, and $\log(\frac{0.9}{0.7}) = 0.25$.