Assignment 2 (Sol.)

Reinforcement Learning

Prof. B. Ravindran

1. In the UCB 1 algorithm, how is it ensured that the confidence bounds around the estimated values for each arm shrink?

   (a) the intervals do not shrink

   (b) the confidence intervals of each arm are made to shrink after every arm selection

   (c) in calculating the bounds we use the number of times an arm is selected in the denominator

   (d) on selecting an arm its interval is shrunk by half every time

   Sol. (c)
   The number of times arm j is selected, denoted by n_j appears in the denominator of the upper confidence bound for each arm. As n_j increases, the bound decreases.

2. After 12 iterations of the UCB 1 algorithm applied on a 4-arm bandit problem, we have $n_1 = 3$, $n_2 = 4$, $n_3 = 3$, $n_4 = 2$ and $Q_{12}(1) = 0.55$, $Q_{12}(2) = 0.63$, $Q_{12}(3) = 0.61$, $Q_{12}(4) = 0.40$. Which arm should be played next?

   (a) 1

   (b) 2

   (c) 3

   (d) 4

   Sol. (d)
   The next action, A_{13}, will be the action with the maximum upper confidence bound among the four arms. Calculating these values, we have

$$Q_{12}(1) + \sqrt{\frac{2ln12}{n_1}} = 0.55 + \sqrt{\frac{2ln12}{3}} = 1.837$$

$$Q_{12}(2) + \sqrt{\frac{2ln12}{n_2}} = 0.63 + \sqrt{\frac{2ln12}{4}} = 1.745$$

$$Q_{12}(3) + \sqrt{\frac{2ln12}{n_3}} = 0.61 + \sqrt{\frac{2ln12}{3}} = 1.897$$

$$Q_{12}(4) + \sqrt{\frac{2ln12}{n_4}} = 0.40 + \sqrt{\frac{2ln12}{2}} = 1.976$$

   Clearly, arm 4 has the highest upper confidence bound and hence will be selected by the UCB 1 algorithm.

3. In the proof of the UCB 1 algorithm, we had the following expression:

$$\{min_{0<s<m}\left(Q_s(a^*) + C_{m-1,T_{a^*}(s)}\right) \leq max_{l \leq s_i \leq m}(Q_{s_i}(i) + C_{m-1,T_i(s_i)})\}$$

What does this expression stand for?

(a) the number of instances where the minimum value among the upper confidence bound values of an optimal arm up till the m^{th} time step is less than or equal to the maximum value among the upper confidence bound values of the i^{th} arm between the times steps l and m

(b) the condition that the minimum value among the upper confidence bound values of an optimal arm up till the m^{th} time step is less than or equal to the maximum value among the upper confidence bound values of the i^{th} arm between the times steps l and m

(c) the condition that there exists an upper confidence bound value of an optimal arm that is smaller than the corresponding upper confidence value of the chosen arm i

(d) the number of instances where the upper confidence bound value of an optimal arm is smaller than the corresponding upper confidence value of the chosen arm i

Sol. (b)

4. In the initial stage of the UCB 1 algorithm, each arm is selected one time. Thereafter, the arm that is selected depends on the arm with the maximum upper confidence bound. Suppose that one of the arms has not been selected after its initial selection in the first stage of the algorithm. What happens to the upper confidence bound of this arm?

(a) it increases

(b) it decreases

(c) it remains constant until it is selected

Sol. (a)
Presence of n, the number of time steps in the numerator ensures that at each step the upper confidence bounds of each unselected arm increases.

5. Suppose that we apply the naive PAC algorithm on a 10-arm bandit problem where the required (ϵ, δ) values are: $\epsilon = 0.5$ and $\delta = 0.05$. How many iterations would the algorithm take to output an arm selection? (Note: each arm is sampled $\frac{2}{\epsilon^2} ln\left(\frac{2k}{\delta}\right)$ times).

(a) 48

(b) 21

(c) 210

(d) 480

Sol. (d)
We know that in the naive algorithm, each arm is sampled

$$l = \frac{2}{\epsilon^2} ln\left(\frac{2k}{\delta}\right)$$

times, where k is the number of arms. Thus, we have

$$l = \frac{2}{0.5^2} ln \left(\frac{2 * 10}{0.05} \right) = 8ln(400) = 47.93 = 48$$

Thus, the total number of iterations = 10 * l = 480.

6. In the proof of the Naive PAC algorithm, we have the expression

$$P(Q(a') > Q(a^*)) \leq P(Q(a') > q_*(a') + \epsilon/2 \text{ OR } Q(a^*) < q_*(a^*) - \epsilon/2)$$

Why is the LHS "less than or equal to" the RHS here?

(a) because the occurrence of at least one of the events on the RHS is a necessary but not sufficient condition for the event on the LHS to occur

(b) because the event on the LHS does not require that the events on the RHS occur

(c) because the event on the LHS requires that both the events on the RHS occur

Sol. (a)
It should be clear that for $Q(a')$ to be greater than $Q(a^*)$, at least one of the events on the RHS must occur, but even then, $Q(a')$ may not be greater than $Q(a^*)$. This implies that the probability of the LHS is less than or equal to the probability of the RHS.

7. Suppose we have a 64-arm bandit problem. We apply the median elimination algorithm where the (ϵ, δ) values are: $\epsilon = 0.5$ and $\delta = 0.01$. How many times do we sample arms in the first round? (Note: in each round, each arm is sampled $\frac{1}{\epsilon_l^2/2} ln \left(\frac{3}{\delta_l} \right)$ times).

(a) 52404

(b) 818

(c) 52416

(d) 819

Sol. (c)
We know that in the median elimination algorithm, each arm is sampled

$$\frac{1}{\epsilon_l^2/2} ln \left(\frac{3}{\delta_l} \right)$$

times. In the first round, $\epsilon_1 = \epsilon/4$ and $\delta_1 = \delta/2$. Substituting the values of ϵ and δ, we have number of samples

$$\frac{32}{0.5^2} ln \left(\frac{6}{0.01} \right) = 818.81 = 819$$

This is the number of times each arm is pulled in the first round. Thus, the total number of sampled arms in the first round is 64 * 819 = 52416.

8. Continuing with the previous example, what is the number of samples in the third round? Also, what is the total number of rounds required to identify the (ϵ, δ)-optimal arm?

(a) 50384, 6

(b) 50384, 7

(c) 50378, 6

(d) 201514, 7

Sol. (a)
From the initial and update conditions, we know that $\epsilon_l = (\frac{3}{4})^{l-1}\frac{\epsilon}{4}$ and $\delta_l = \frac{\delta}{2^l}$. Now, in any given round, each arm is sampled

$$\frac{1}{\epsilon_l^2/2} ln\left(\frac{3}{\delta_l}\right) = \frac{2}{((\frac{3}{4})^{l-1}\frac{\epsilon}{4})^2} ln\left(\frac{3*2^l}{\delta}\right)$$

times. Substituting the values of ϵ and δ, with $l = 3$, we have

$$\frac{2}{((\frac{3}{4})^2\frac{0.5}{4})^2} ln\left(\frac{3*2^3}{0.01}\right) = 3148.65 = 3149$$

This is the number of times each arm is pulled in the third round. The number of arms in the third round is 16 (32 of 64 eliminated at the end of the first round and 16 eliminated at the end of the second round). Thus, the total number of sampled arms in the third round is 16 * 3149 = 50384.

The number of rounds required to identify the (ϵ, δ)-optimal arm is $log_2(64) = 6$.

9. Consider a bandit problem in which there is a single optimal arm, a^*. Applying the median elimination algorithm to this problem, is it possible that arm a^* is eliminated in the first round? In case it is possible, does this mean that the algorithm cannot output an arm that is ϵ-close to a^*?

(a) no

(b) yes, no

(c) yes, yes

Sol. (b)
It is possible that after the first round of sampling, the estimated payoff of the optimal arm is below the median estimated payoff and hence is eliminated. However, the algorithm can still output an arm that is ϵ-close to a^*. This is because in each round the probability that the best arm is more than ϵ_l away from the best arm in the previous round is less than δ_l. Thus, even if a^* has been eliminated in the first round, it is highly probable (since δ and consequently δ_l are typically small) that an arm that is ϵ_l-close to a^* remains in the second round. Also, the use of ϵ_l and δ_l in each round (with each of these values getting smaller as the rounds progress) implies that over all rounds, the probability of eliminating all arms that are ϵ-close to a^* is less than δ, which is again, typically a small value. Hence, even if the best arm is eliminated in the first round, it is still possible that the algorithm outputs an arm that is ϵ-close to a^*.

10. We know that there is an exploration/exploitation dilemma in reinforcement learning problems. Considering the Thompson sampling algorithm for solving bandit problems, which step of the algorithm ensures that we perform exploration?

(a) initialisation of each arm's reward distribution

(b) updating the reward distribution based on observing actual reward for an arm

(c) sampling the expected payoff of each arm from the corresponding reward distribution

(d) identifying the correct arm given the set of expected payoffs for each arm sampled from each arm's reward distribution

Sol. (c)
At each step, we sample the expected payoffs of each arm from their corresponding reward distributions. This introduces an element of exploration since even an arm for which the mean of the distribution is less than the mean of the distribution of another arm can be selected by the algorithm if the expected payoff sampled for the former arm is larger than the expected payoff sampled for the latter arm.