CS 6700: Reinforcement Learning, Written Assignment #1

E Santhosh Kumar

CS16B107

1. **Answer to question 1:**
   For an opponent that did take advantage of symmetries, their policy at any two such symmetrical states would be the same. Thus, we can replace all symmetrical states of one kind in our model by a super-state (as all symmetrical states will have the same true value for value function). This reduces the state-space size. Furthermore for certain new states, this will also reduce the number of possible actions from that state as several actions may lead into the same super-state. The reduced state-space means that we can better utilize the samples we have compared to the naive approach.
   Suppose the opponent did not take advantage of symmetries, then their policies, and hence the probabilities of winning and the value function, at two symmetrical states might be different. Thus it is wrong to simplify the model by using symmetries merely based on the position of pieces on the grid. In fact, the we might be able to learn different winning strategies for symmetrical states.

2. **Answer to question 2:**
   Two updates are made for each run of the game, one for each player. Since both players use the same strategy, the policy keep adapting until it reaches a Nash equilibrium. This policy would be different from a policy that was trained against a specific fixed opponent.

3. **Answer to question 3:**
   The greedy algorithm considers only immediate rewards and does not explore other states that might have potential for better future rewards. Hence, it might so happen that the state value functions are initialized in such a way that some states of high potential never get explored. The non-greedy player(for instance an ϵ-greedy player) on the other hand might provide the right balance between exploration and exploitation, thus leading to a better strategy. Furthermore, the moves then taken (except on exploratory moves) are in fact the optimal moves against this opponent.

4. **Answer to question 4:**
   The initial action-value of 5 is wildly optimistic compared to the true expectations. Hence the first 10 actions always sample each arm once. At the end of 10 steps, it is most likely for the true optimal action to have highest action-value. Hence, it gets picked one or more times (spike in the graph), till its action value drops and it is no longer the current best arm. The above described event may happen repeatedly till all arms have action-values closer to their true expectations. This may lead to more spikes being observed.

5. **Answer to question 5:**
   The arm with maximum true expectation (optimal arm) is most likely to be the best arm of each round. Hence, even here the task is to find the optimal arm. This means that our previous algorithms work even here.
   Also, since exploration is not an issue right now (convergence of action-values of all arms happen

simultaneously, irrespective of our algorithm), choosing the greedy arm at every iteration gives the best regret value. Furthermore, the UCB algorithm also works the same as the regular greedy algorithm in this setup as the confidence bounds of all arms are the same at any time step.

6. **Answer to question 6:**
   The task now boils down to finding the arm with the maximum of the given expected values. Let the maximum expected value be μ^*. One metric that can be used to select the arm to pull at every step is the **t-statistic**. For each arm, we apply a one-tailed t-test to test the hypothesis that the observed distribution of the arm comes from a true distribution with mean more than μ^*.

7. **Answer to question 7:**

$$\begin{aligned}
\frac{\partial \pi_t(b)}{\partial \rho_t(a)} &= \frac{\partial}{\partial \rho_t(a)} \left[\frac{e^{\rho_t(b)}}{\sum_{c=1}^{n} e^{\rho_t(c)}} \right] \\
&= \frac{\frac{\partial e^{\rho_t(b)}}{\partial \rho_t(a)} \sum_{c=1}^{n} e^{\rho_t(c)} - e^{\rho_t(b)} \frac{\partial \sum_{c=1}^{n} e^{\rho_t(c)}}{\partial \rho_t(a)}}{\left(\sum_{c=1}^{n} e^{\rho_t(c)}\right)^2} (by\ quotient\ rule) \\
&= \frac{\mathbb{1}_{a=b} e^{\rho_t(b)} \sum_{c=1}^{n} e^{\rho_t(c)} - e^{\rho_t(b)} e^{\rho_t(a)}}{\left(\sum_{c=1}^{n} e^{\rho_t(c)}\right)^2} \\
&= \frac{\mathbb{1}_{a=b} e^{\rho_t(b)}}{\sum_{c=1}^{n} e^{\rho_t(c)}} - \frac{e^{\rho_t(b)} e^{\rho_t(a)}}{\left(\sum_{c=1}^{n} e^{\rho_t(c)}\right)^2} \\
&= \mathbb{1}_{a=b} \pi_t(b) - \pi_t(b) \pi_t(a) \\
&= \pi_t(b)(\mathbb{1}_{a=b} - \pi_t(a))
\end{aligned} \tag{1}$$

   Given the arm played at time t is A_tm the REINFORCE method gives the update at each step as,

$$\begin{aligned}
\Delta \rho_t(a) &= \alpha (r_t - \bar{r_t}) \frac{\partial \ln\big(\pi_t(A_t)\big)}{\partial \rho_t(a)} \\
&= \alpha (r_t - \bar{r_t}) \frac{1}{\pi_t(A_t)} \frac{\partial \pi_t(A_t)}{\partial \rho_t(a)} \\
&= \alpha (r_t - \bar{r_t}) \frac{1}{\pi_t(A_t)} \pi_t(A_t)(\mathbb{1}_{a=A_t} - \pi_t(a)) \ \ (from\ equation\ 1) \\
&= \alpha (r_t - \bar{r_t})(\mathbb{1}_{a=A_t} - \pi_t(a))
\end{aligned}$$

   Thus the update equation is

$$\rho_{t+1}(a) = \rho_t(a) + \alpha (r_t - \bar{r_t})(\mathbb{1}_{a=A_t} - \pi_t(a)) \ \ ,\ for\ all\ a$$

8. **Answer to question 8:**
   Let the mean and standard deviation of the normal distribution at time t be μ_t and σ_t respectively. Then the policy is given by

$$\pi_t(a) = \frac{1}{\sqrt{2\pi}\sigma_t} e^{-\frac{(a-\mu_t)^2}{2\sigma_t^2}}$$

$$\frac{\partial \ln\big(\pi_t(a)\big)}{\partial \mu_t} = \frac{1}{\pi_t(a)}.\frac{1}{\sqrt{2\pi}\sigma_t} e^{-\frac{(a-\mu_t)^2}{2\sigma_t^2}} \frac{(a-\mu_t)}{\sigma_t^2} = \frac{1}{\pi_t(a)}.\pi_t(a)\frac{a-\mu_t}{\sigma_t^2} = \frac{a-\mu_t}{\sigma_t^2}$$

$$\begin{aligned}
\frac{\partial \ln\big(\pi_t(a)\big)}{\partial \sigma_t} &= \frac{1}{\pi_t(a)}.\frac{1}{\sqrt{2\pi}} \left(\frac{-1}{\sigma_t^2} e^{-\frac{(a-\mu_t)^2}{2\sigma_t^2}} + \frac{1}{\sigma_t} e^{-\frac{(a-\mu_t)^2}{2\sigma_t^2}} \frac{(a-\mu_t)^2}{\sigma_t^3} \right) \\
&= \frac{1}{\pi_t(a)}.\frac{1}{\sqrt{2\pi}\sigma_t} e^{-\frac{(a-\mu_t)^2}{2\sigma_t^2}} \left(\frac{(a-\mu_t)^2}{\sigma_t^3} - \frac{1}{\sigma_t} \right) \\
&= \frac{1}{\pi_t(a)}.\pi_t(a) \left(\frac{(a-\mu_t)^2}{\sigma_t^3} - \frac{1}{\sigma_t} \right) \\
&= \frac{(a-\mu_t)^2}{\sigma_t^3} - \frac{1}{\sigma_t}
\end{aligned}$$

Therefore, the update equations are given by (for baseline 0)

$$\mu_{t+1} = \mu_t + \alpha r_t \frac{\partial \ln\big(\pi_t(a)\big)}{\partial \mu_t} = \mu_t + \alpha r_t \left(\frac{a-\mu_t}{\sigma_t^2} \right)$$

$$\sigma_{t+1} = \sigma_t + \alpha r_t \frac{\partial \ln\big(\pi_t(a)\big)}{\partial \sigma_t} = \sigma_t + \alpha r_t \left(\frac{(a-\mu_t)^2}{\sigma_t^3} - \frac{1}{\sigma_t} \right)$$