
CS6700 : Reinforcement Learning

Written Assignment #2

Deadline: 30-May-2020

- This is an individual assignment. Collaborations and discussions are strictly prohibited.
 - Be precise with your explanations. Unnecessary verbosity will be penalized.
 - Check the Moodle discussion forums regularly for updates regarding the assignment.
 - **Please start early.**
-

AUTHOR : RANA PRAGNESHKUMAR RAJUBHAI Name.

ROLL NUMBER : ME17S301

1. (3 points) Consider a bandit problem in which the policy parameters are mean μ and variance σ of normal distribution according to which actions are selected. Policy is defined as $\pi(a; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(a-\mu)^2}{2\sigma^2}}$. Derive the parameter update conditions according to the REINFORCE procedure (assume baseline is zero).

Solution: The REINFORCEMENT parameter update equation can be written as,

$$\Delta\theta_n = \alpha(R_n - b_n) \frac{\partial \ln \pi(a_n; \theta)}{\partial \theta_n} \quad (1)$$

The parameters for Gaussian policy is given as μ and σ and policy is given as,

$$\pi(a; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(a-\mu)^2}{2\sigma^2}} \quad (2)$$

Update of μ :

$$\begin{aligned} \frac{\partial \ln \pi(a_n; \mu_n, \sigma_n)}{\partial \mu_n} &= \frac{\partial}{\partial \mu_n} \left\{ \frac{-(a_n - \mu_n)^2}{2\sigma_n^2} \right\} \\ &= \frac{a_n - \mu_n}{\sigma_n^2} \end{aligned} \quad (3)$$

Considering baseline performance to zero($b=0$), update rule will be ,

$$\mu_{n+1} = \mu_n + \alpha R_n \left(\frac{a_n - \mu_n}{\sigma_n^2} \right) \quad (4)$$

similarly, Update of σ :

$$\begin{aligned} \frac{\partial \ln \pi(a_n; \mu_n, \sigma_n)}{\partial \sigma_n} &= \frac{\partial}{\partial \sigma_n} \left\{ -\ln(\sqrt{2\pi\sigma_n}) \right\} + \frac{\partial}{\partial \sigma_n} \left\{ -\frac{(a_n - \mu_n)^2}{2\sigma_n^2} \right\} \\ &= -\frac{1}{\sigma_n} + \frac{-(a_n - \mu_n)^2}{2\sigma_n^3} \\ &= \frac{1}{\sigma_n} \left\{ \left(\frac{a_n - \mu_n}{\sigma_n} \right)^2 - 1 \right\} \end{aligned} \quad (5)$$

update rule for the variance will be ,

$$\sigma_{n+1} = \sigma_n + \alpha R_n \frac{1}{\sigma_n} \left\{ \left(\frac{a_n - \mu_n}{\sigma_n} \right)^2 - 1 \right\} \quad (6)$$

2. (6 points) Let us consider the effect of approximation on policy search and value function based methods. Suppose that a policy gradient method uses a class of policies that do not contain the optimal policy; and a value function based method uses a function approximator that can represent the values of the policies of this class, but not that of the optimal policy.

- (a) (2 points) Why would you consider the policy gradient approach to be better than the value function based approach?

Solution: For non-existence of optimal policy, value function based approximation does not converge as it optimal policy and it may get stuck in non-optimal policy infinitely whereas, the policy gradient has better convergence criteria. The action is picked based on the state which maximized the reward. Parameter is denoted by parameter θ .

$$\pi(a|a, \theta = \Pr A_t = a, S_t, \theta_t = \theta) \quad (7)$$

Eq.- shows that, the policy is denotes the probability of taking action based on state(s) and parameter(θ).

- (b) (2 points) Under what circumstances would the value function based approach be better than the policy gradient approach?

Solution:

- (c) (2 points) Is there some circumstance under which either of the method can find the optimal policy?

Solution:

3. (4 points) Answer the following questions with respect to the DQN algorithm:

- (2 points) When using one-step TD backup, the TD target is $R_{t+1} + \gamma V(S_{t+1}, \theta)$ and the update to the neural network parameter is as follows:

$$\Delta \theta = \alpha (R_{t+1} + \gamma V(S_{t+1}, \theta) - V(S_t, \theta)) \nabla_{\theta} V(S_t, \theta) \quad (8)$$

Is the update correct ? Is any term missing ? Justify your answer

Solution:

- (2 points) Describe the two ways discussed in class to update the parameters of target network. Which one is better and why?

Solution:

4. (4 points) Experience replay is vital for stable training of DQN.
- (a) (2 points) What is the role of the experience replay in DQN?

Solution:

- (b) (2 points) Consequent works in literature sample transitions from the experience replay, in proportion to the TD-error. Hence, instead of sampling transitions using a uniform-random strategy, higher TD-error transitions are sampled at a higher frequency. Why would such a modification help?

Solution:

5. (3 points) We discussed two different motivations for actor-critic algorithms: the original motivation was as an extension of reinforcement comparison, and the modern motivation is as a variance reduction mechanism for policy gradient algorithms. Why is the original version of actor-critic not a policy gradient method?

Solution:

6. (4 points) This question requires you to do some [additional reading](#). Dietterich specifies certain conditions for safe-state abstraction for the MaxQ framework. I had mentioned in class that even if we do not use the MaxQ value function decomposition, the hierarchy provided is still useful. So, which of the safe-state abstraction conditions are still necessary when we do not use value function decomposition?

Solution:

7. (3 points) Consider the problem of solving continuous control tasks using Deep Reinforcement Learning.

- (a) (2 points) Why can simple discretization of the action space not be used to solve the problem? In which exact step of the DQN training algorithm is there a problem and why?

Solution:

- (b) (1 point) How is exploration ensured in the DDPG algorithm?

Solution:

8. (3 points) Option discovery has entailed using heuristics, the most popular of which is to identify bottlenecks. Justify why bottlenecks are useful sub-goals. Describe scenarios in which a such a heuristic could fail.

Solution: