

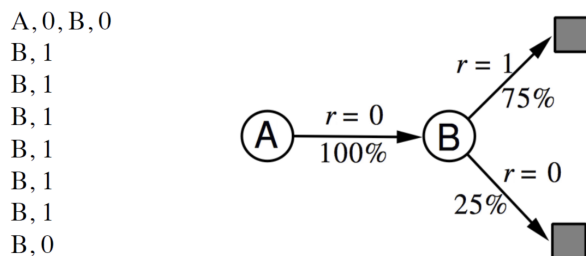
# Sample Questions

ELEN E6885: Reinforcement Learning

November 27, 2019

**Problem 1** Given a stationary policy, is it possible that if the agent is in the same state at two different time steps, it can choose two different actions? If yes, please provide an example.

**Problem 2** As shown in Figure 1, consider a A-B model that has been learnt from 8 non-discounting episodes of experience. Please show how the numbers on the model are calculated.



**Problem 3** Figure 2 shows the DynaQ algorithm. What is the functionality/purpose of the steps in red box?

**Problem 4** Consider a bandit problem in which the parameters on which the policy depends are the preferences of the actions and the action selection probabilities are determined by the softmax relationship as  $\pi(a_i; \theta) = \frac{e^{\theta_i}}{\sum_{j=1}^k e^{\theta_j}}$ , where  $k$  is the total number of actions and  $\theta_i$  is the preference value of action  $a_i$ . Show the parameter update conditions according to the REINFORCE procedure is

$$\Delta\theta_i = a(r_i - b)(1 - \pi(a_i; \theta)),$$

where  $a$  is the step size,  $r_i$  is the reward received at the  $n$ -th play and the baseline  $b$  is the reference reward defined as the average of the rewards received for all arms. Note: you only need to derive  $\frac{\partial \ln(\pi(a_i; \theta))}{\partial \theta} = (1 - \pi(a_i; \theta))$ .

**Problem 5** In Q-learning, Q value is updated as follows,

$$Q(s, a) = Q(s, a) + \alpha(r + \gamma \max_{a' \in A} Q(s', a') - Q(s, a)). \quad (1)$$

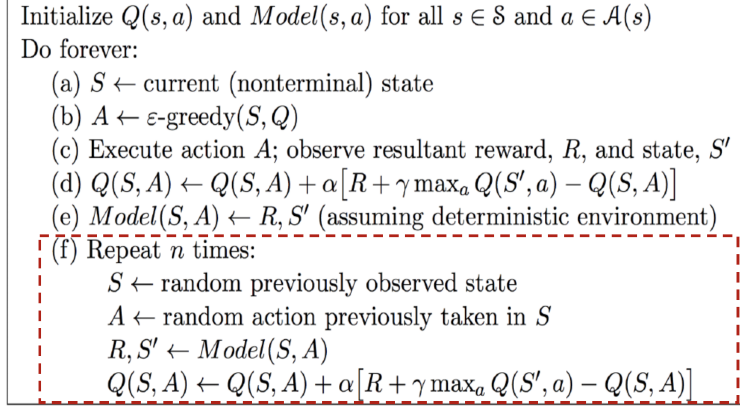


Figure 2: DynaQ

However, considering the curse of dimensionality, we can represent  $Q$  values as a function  $\hat{q}(s, a, w)$  where  $w$  are parameters of the function (e.g., neural network's weights and bias). In this approximation setting, the update rule on parameters  $w$  becomes

$$w = w + \alpha(r + \gamma \max_{a' \in \mathcal{A}} \hat{q}(s', a', w) - \hat{q}(s, a, w)) \nabla_w \hat{q}(s, a, w). \quad (2)$$

Show that the above equations (1) (2) are exactly the same when  $\hat{q}(s, a, w) = w^T x(s, a)$ , where  $x(s, a) : \mathcal{S} \times \mathcal{A} \in R^{|\mathcal{S}| |\mathcal{A}|}$  is the table look up feature vector on each state-action pair. For a given state-action pair  $(s', a')$ , the entry  $x(s, a)_{s', a'}$  in  $x(s, a)$  equals to one if  $s' = s, a' = a$  and  $x(s, a)_{s', a'} = 0$  otherwise.

**Problem 6** Consider a three-arm bandit problem with UCB1 algorithm. After selecting arms 0 – 2 once each, UCB1 will select the arm that maximizes

$$\frac{v_i}{n_i} + c * \sqrt{\ln(t)/n_i},$$

where  $v_i$  is the total payout for arm  $i$ ,  $n_i$  is the number of times arm  $i$  has been selected,  $t$  is the total number of times all arms have been selected, and  $\ln$  is the natural log function. Arm 0 has been pulled 100 times with a total payout of 564.750. Arm 1 has been pulled 1 time with a total payout of -3.978. Arm 2 has been pulled 10 times with a total payout of 34.905. Given this information, which arm will UCB1 select next?