Course 2, Module 5
Planning, Learning & Acting

CMPUT 397
Fall 2019

Finishing the demo

2. An agent is in a 4-state MDP, $\mathcal{S} = \{1, 2, 3, 4\}$, where each state has two actions $\mathcal{A} = \{1, 2\}$. Assume the agent saw the following trajectory,

$$
\begin{aligned}
&S_0 = 1, A_0 = 2, R_1 = -1, \\
&S_1 = 1, A_1 = 1, R_2 = 1, \\
&S_2 = 2, A_2 = 2, R_3 = -1, \\
&S_3 = 2, A_3 = 1, R_4 = 1, \\
&S_4 = 3, A_4 = 1, R_5 = 100, \\
&S_5 = 4
\end{aligned}
$$

and uses Tabular Dyna-Q with 5 planning steps for each interaction with the environment.

(a) Once the agent sees S_5, how many Q-learning updates has it done with **real experience**? How many updates has it done with **simulated experience**?

$$
\begin{aligned}
&S_0 = 1, A_0 = 2, R_1 = -1, \\
&S_1 = 1, A_1 = 1, R_2 = 1, \\
&S_2 = 2, A_2 = 2, R_3 = -1, \\
&S_3 = 2, A_3 = 1, R_4 = 1, \\
&S_4 = 3, A_4 = 1, R_5 = 100, \\
&S_5 = 4
\end{aligned}
$$

(b) Which of the following are possible (or not possible) simulated transitions $\{S, A, R, S'\}$ given the above observed trajectory with a deterministic model and random search control?

i. $\{S = 1, A = 1, R = 1, S' = 2\}$

ii. $\{S = 2, A = 1, R = -1, S' = 3\}$

iii. $\{S = 2, A = 2, R = -1, S' = 2\}$

iv. $\{S = 1, A = 2, R = -1, S' = 1\}$

v. $\{S = 3, A = 1, R = 100, S' = 5\}$

3. Modify the Tabular Dyna-Q algorithm so that it uses Expected Sarsa instead of Q-learning. Assume that the target policy is ϵ-greedy. What should we call this algorithm?

Tabular Dyna-Q

Initialize $Q(s,a)$ and $Model(s,a)$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$
Loop forever:
- (a) $S \leftarrow$ current (nonterminal) state
- (b) $A \leftarrow \varepsilon$-greedy(S,Q)
- (c) Take action A; observe resultant reward, R, and state, S'
- (d) $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma \max_a Q(S',a) - Q(S,A)\big]$
- (e) $Model(S,A) \leftarrow R, S'$ (assuming deterministic environment)
- (f) Loop repeat n times:
  - $S \leftarrow$ random previously observed state
  - $A \leftarrow$ random action previously taken in S
  - $R, S' \leftarrow Model(S,A)$
  - $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma \max_a Q(S',a) - Q(S,A)\big]$

4. Consider an MDP with two states $\{1, 2\}$ and two possible actions: $\{\text{stay}, \text{switch}\}$. The state transitions are deterministic, the state does not change if the action is "stay" and the state switches if the action is "switch". However, rewards are randomly distributed:

$$P(R\,|S=1, A=\text{stay}) = \begin{cases} 0 & \text{w.p. } 0.4 \\ 1 & \text{w.p. } 0.6 \end{cases}, \quad P(R\,|S=1, A=\text{switch}) = \begin{cases} 0 & \text{w.p. } 0.5 \\ 1 & \text{w.p. } 0.5 \end{cases}$$

$$P(R\,|S=2, A=\text{stay}) = \begin{cases} 0 & \text{w.p. } 0.6 \\ 1 & \text{w.p. } 0.4 \end{cases}, \quad P(R\,|S=2, A=\text{switch}) = \begin{cases} 0 & \text{w.p. } 0.5 \\ 1 & \text{w.p. } 0.5 \end{cases}$$

(a) How might you learn the reward model? Hint: think about how probabilities are estimated. For example, what if you were to estimate the probability of a coin landing on heads? If you observed 10 coin flips with 8 heads and 2 tails, then you can estimate the probabilities by counting: $p(\text{heads}) = \frac{8}{10} = 0.8$ and $p(\text{tails}) = \frac{2}{10} = 0.2$.

(b) Modify the tabular Dyna-Q algorithm to handle this MDP with stochastic rewards.

(b) Modify the tabular Dyna-Q algorithm to handle this MDP with stochastic rewards.

Tabular Dyna-Q

Initialize $Q(s,a)$ and $Model(s,a)$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$
Loop forever:
- (a) $S \leftarrow$ current (nonterminal) state
- (b) $A \leftarrow \varepsilon$-greedy(S,Q)
- (c) Take action A; observe resultant reward, R, and state, S'
- (d) $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma \max_a Q(S',a) - Q(S,A)\big]$
- (e) $Model(S,A) \leftarrow R, S'$ (assuming deterministic environment)
- (f) Loop repeat n times:
  - $S \leftarrow$ random previously observed state
  - $A \leftarrow$ random action previously taken in S
  - $R, S' \leftarrow Model(S,A)$
  - $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma \max_a Q(S',a) - Q(S,A)\big]$