CPSC 633-600 Homework 2, part II of II (Total 50 points)
Reinforcement Learning

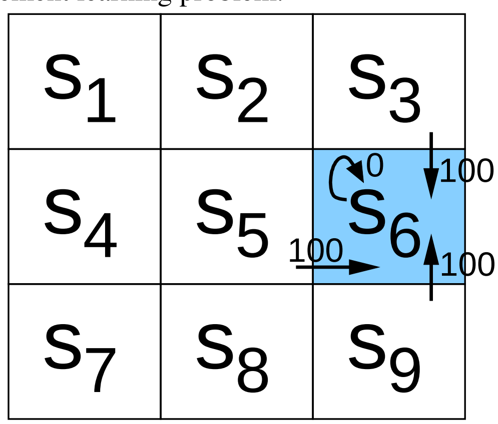
See course web page for the **due date**.
Use **elearning.tamu.edu** to submit your assignments, or submit a hard copy.

Instructor: Yoonsuck Choe

March 7, 2013, revised March 8, 2013 (bold)

1 Deterministic Case

Consider the following reinforcement learning problem.



- There are 9 states, and the actions are $\{up, down, left, right\}$. Legal actions are those that go to the immediate neighbor, horizontally or vertically (but not diagonally). Treat State 6 (s_6) as having no legal action.
- The rewards for all action are 0, except for all actions that lead into s_6, which are 100.
- In all cases, assume $\gamma = 0.9$.

Problem 1 (Program: 10 pts): Write a Q-learning algorithm to learn the $Q(s, a)$ values for the above example. Use the algorithm in slide05.pdf, Mitchell slide page 18 (pdf page 22). Stop learning when change in the Q table is 0 for the past 10 iterations. Note: use a random policy to select action a given current state s (take care to check if the random action chosen is a legal one).

(1) Include your code.

(2) Show resulting Q table (9×4 matrix).

(3) Show a plot showing sum(abs($Q_{t+1} - Q_t$)) over the iterations t.

Problem 2 (Program: 10 pts): Modify the program from problem 1 so that the exploration policy is ϵ-greedy. Initialize your Q table with a very small random number to break the initial tie (rand * 0.0001).

(1) Include your code.

(2) Test $\epsilon \in \{0.0, 0.2, 0.5, 1.0\}$. Note: $\epsilon = 0.0$ is the greedy policy, and $\epsilon = 1.0$ is the random policy.

If `rand() < epsilon`, choose random action. Otherwise, choose `[val, a] = max(Q(s,:))`.

(3) Show resulting Q tables for all 4 cases (9×4 matrix).

(4) Show plots showing sum(abs($Q_{t+1} - Q_t$)) over the iterations t for all four cases.

(5) Discuss the effect of ϵ on the quality of the learned Q-table.

2 Stochastic Case

Consider a stochastic version of the reinforcement learning problem posed in Section 1. Modify the rules so that:

- $\delta(s, a)$ is stochastic: The probability of landing in the intended direction is 0.76. The probability of landing in one of n unintended legal direction is $\frac{0.24}{n}$.
- Example 1 : If you are in s_5 and a was right, probability of landing in s_6 is 0.76, and ending up in s_2, s_4, or s_8 is 0.8 each.
- Example 2: If you are in s_1 and a was down, probability of landing in s_2 is 0.76, and ending up in s_4 is 0.24.

Problem 3 (Program: 10 pts): Repeat problem 1, with the stochastic version of the task (random policy). In addition to all the requirements, keep a running estimate of $E[r(s, a)]$ for states s_3, s_5, and s_9 and report the **final** values. Use the learning rule in slide05.pdf, Mitchell slide page 31 (pdf page 35).

Estimating $E[r(s, a)]$ throughout the learning run:

$$E[r(s, a)] = \frac{\sum_{\forall \text{visits to } (s,a)} r}{visits(s, a)}$$

Problem 4 (Program: 10 pts): Repeat problem 2, with the stochastic version of the task (ϵ-greedy policy with the four different ϵ values). In addition to all the requirements, keep a running estimate of $E[r(s, a)]$ for states s_3, s_5, and s_9 and report the values. Use the learning rule in slide05.pdf, Mitchell slide page 31 (pdf page 35).

Problem 5 (Written: 5 pts): For states s_3, s_5, and s_9, manually compute $E[r(s, a)]$ (using the exact probabilities **[note: it relates with $P(s'|s, a)$ and the reward depending on state outcome s']**) and compare those to the estimated values from problem 3 and problem 4. Are the results similar?

Problem 6 (Written: 5 pts): For states s_3, s_5, and s_9, using the estimated $E[r(s, a)]$ and all the estimated $Q(s, a)$ values from your result in problem 3 above, see if the following holds:

$$Q(s, a) = E[r(s, a)] + \gamma \sum_{s'} P(s'|s, a) \max_{a'} Q(s', a')$$