CPSC 422

Practice Midterm Exam Questions

February 2013

The exam covers what we did before the midterm break, covering the following sections of the textbook:

- Agents and control: Chapter 1 and Sections 2.1-2.3
- Decision processes: Section 9.5
- Reinforcement Learning: Section 11.3 (except 11.3.7)
- Approximate Inference and time. Section 6.4 and 6.5.

This set of practice questions is meant to give some example questions to make sure you understand the material. This is longer than the exam. The exam may be nothing like this. But if you can do this, you should have no problems with the exam. **You may bring in one letter sized piece of paper with anything written on it. You may not use calculators, phones, robotic assistants or other electronic aids.**

Some important points (that students often forget):

- Read and answer the question. You will not get marks for writing things (whether they are true or not) that are not relevant to the question.
- Use proper English in full sentences. You will not get marks if we cannot work out what you are saying.
- If a question asks about a particular instance of a problem, make sure your answer refers to that instance. Writing a general formula that you may have copied from the sheet you can bring in, is not worth any marks. (The questions are usually asking to apply a formula to a particular case, to make sure you understand it).

This practice midterm concentrates on what was not covered in assignments. You should also expect some questions about what you should have learned from

doing your assignment (e.g., dimensions of complexity, value iteration for MDPs, parameters of reinforcement learning, designing features, HMMs).

1. Answer the following questions. Use proper English. Be concise in your answers. (You will lose marks for stating irrelevant facts). You must use your own words (text from the textbook or another source copied onto your crib sheet will not get any marks).
   (a) Give the intuition behind the notion of a transduction? Why do we define *causal* transductions?
   (b) Why does an agent have a belief state?
   (c) Explain why an agent controller needs a command function and a state transition function, but not other functions.

2. Suppose a Q-learning agent, with fixed α, and discount γ, was in state 34 did action 7, received reward 3 and ended up in state 65. What value(s) get updated? Give an expression for the new value. (You need to be as explicit as possible.)

3. In temporal difference learning (e.q. Q-learning), to get the average of a sequence of k values, we let $\alpha_k = 1/k$. Explain why it may be advantageous to keep α_k fixed in the context of reinforcement learning.

4. Explain what happens in reinforcement learning if the agent always chooses the action that maximizes the Q-value. Suggest two ways to force the agent to explore.

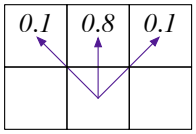
5. In MDPs and reinforcement learning, explain why we often discount future rewards.

6. What is the main difference between asynchronous value iteration and standard value iteration? Why does asynchronous value iteration often work better than standard value iteration?

7. What is the relationship between asynchronous value iteration and Q-learning?

8. Consider a 5×5 grid game similar to the simple game shown in class. The agent can be at one of the 25 locations, and there can be a treasure at one of the corners or no treasure.

   In this game the "up" action has the dynamics given by:

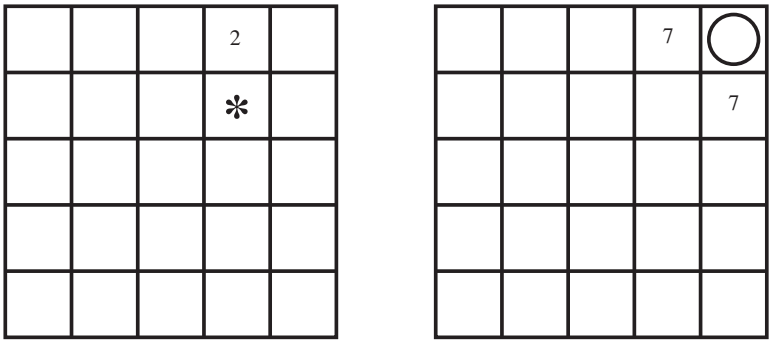


That is the agent goes up with probability 0.8 and goes up-left with probability 0.1 and up-right with probability 0.1.

If there is no treasure, a treasure can appear with probability 0.2. When it appears, it appears randomly at one of the corners, and each corner has an equal probability of treasure appearing. The treasure stays where it is until the agent lands on the square where the treasure is, and the agent gets an immediate reward of $+10$, and the treasure disappears in the next state transition. The agent and the treasure move simultaneously so that if the agent arrives at a square at the same time the treasure appears at the same time, it gets the reward.

If the agent lands in the centre square at the top (whether or not there is a treasure), with probability 0.2 it receives an immediate reward of -10, and with probability 0.8 it receives no immediate reward.

Suppose we are doing asynchronous value iteration and have the following value for each state:

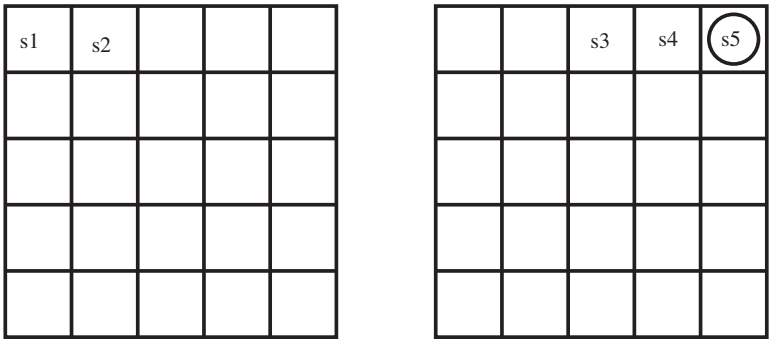


where the left grid shows the values for the states where there is no treasure and the right grid shows the values of the states when there is a treasure at the top-right. All other values are zero.

Consider the next step of asynchronous value iteration. For state s_{13}, which is marked by $*$ in the above figure, and the action a_2 which is "up", what value is assigned to $Q[s_{13}, a_2]$ on the next iteration of value iteration? You need to show all working, but don't need to do any arithmetic (i.e., leave it as an expression). Explain each terms in your expression.

9. Consider a grid world, similar (but simpler) to the simple game used in the class. The agent can move up, right, down or left. A prize can appear at one of the corners. Monsters can (stochastically) attack at certain locations.

   Suppose that the agent steps through the state space in the order of steps given in the diagram below, (i.e., going from $s1$ to $s2$ to $s3$ to $s4$ to $s5$), each time doing a "right" action. In this diagram the right grid represents those states where there is a treasure in the top-right position and the states on the left represent states where there is no treasure.



   You can assume that this is the first time the robot has visited any of these states. All Q-values are initialized to zero. Assume that the discount is 0.9. Do not include variable names in the arithmetic expressions and do not evaluate the arithmetic. Explain the answers.

   (a) Suppose the agent received a reward of -10 entering state s_3 and received a reward of $+10$ on entering the state $s5$, and no other rewards. What Q-values are updated during Q-learning based on this sequence of experiences? Explain what values they get assigned. You should assume that $\alpha_k = 1/k$.

   (b) Suppose that, at some later time in the same run of Q-learning, the robot revisits the same states: $s1$ to $s2$ to $s3$ to $s4$ to $s5$, and hasn't visited any of these states in between (i.e, this is the second time visiting any of these states). Suppose this time, the agent receives a reward of $+10$ on entering the state $s5$, and every other reward was zero. Assume that $\alpha_k = 1/k$. What Q-values have their values changed? What are their new values?

   (c) How would your answers to the previous parts change if it was using SARSA instead of Q-learning?

10. Reinforcement Learning

(a) In the "simple game" with treasures that appear in the corners, there are features that are the x-distance the current treasure and features that are the y-distance to the current treasure. Chris thought that these were not useful as they don't depend on the action. However, when she removed them she found that the controller performed very poorly. Explain to Chris how these features help.

(b) Why do people use Q-learning and SARSA for games, but don't use them for robots? What else can be used for robots? Explain why.

(c) For a model-based reinforcement learner that reasons in terms of individual states with 1000 states and 4 actions at every state, what data structures are required and what is their size?

(d) Which of the following reinforcement algorithms will find the optimal policy, given enough time. Which ones will actually follow the optimal policy? Explain why.

i) Q-learning with fixed α and 80% exploitation.

ii) Q-learning with fixed $\alpha_k = 1/k$ and 80% exploitation.

iii) Q-learning with $\alpha_k = 1/k$ and 100% exploitation.

iv) SARSA-learning fixed $\alpha_k = 1/k$ and 80% exploitation.

v) SARSA-learning fixed $\alpha_k = 1/k$ and 100% exploitation.

vi) Feature-based SARSA learning with soft-max action selection.

vii) A model based-reinforcement learner with 50% exploitation.

11. Suppose we have a stationary hidden Markov model with 10 states where, at every time, there is a Boolean observation. What probabilities need to be specified to model this HMM? How many numbers need to be specified?

12. For a robot in a complex environment, which of the following methods would be suitable to implement robot localization, the problem of determining the location of a robot given the history of sensor observations? For each unsuitable method, specify why it would not be suitable. For each suitable method, specify what needs to be stored to implement the method, and what each of the stored items represents.

(a) variable elimination

(b) rejection sampling

(c) importance sampling

(d) particle filtering