CPSC 502 — Fall 2013

Assignment 4

Due: 10:00pm, Tuesday November 5 2013.

This can be done in groups of size 1, 2 or 3. Working alone is not recommended. A group of size n can choose any $n + 1$ questions from questions 1-5. All members of the group need to be able to explain the group's answer. Please look at all of the questions, as the exam will assume that you have thought about all of the questions. Everyone should do question 6 (it is worth marks). Please post questions to the Connect web site.

Question 1

The following story is fictional. Any similarity to real people is purely coincidental.

Sam wanted to make an informed decision about whether it was worthwhile to exercise or if it was better to relax. Sam suspected that it was about the long-term benefits, not only the short-term rewards. So Sam decided to model this as an MDP.

Sam being very tired after finishing assignment 3, decided to only model whether Sam was "fit" or "unfit" at each week, with two possible actions at each week: Sam can either exercise or relax that week.

Based on Sam's experience, Sam estimated that:

- If Sam is fit and relaxes, there is a 30% chance Sam will become unfit the next week (i.e. $P(s' = \textit{unfit} \mid s = \textit{fit}, a = \textit{relax}) = 0.3$).
- If Sam is unfit and exercises, there is a 20% chance Sam will become fit the next week.
- If Sam is fit and exercises, Sam will remain fit the next week, with probability 0.99 (allowing for injury).
- If Sam is unfit and relaxes, Sam will remain unfit the next week.

Sam estimated his (immediate) rewards to be: $R(\textit{fit}, \textit{relax}) = 10$, $R(\textit{fit}, \textit{exercise}) = 8$, $R(\textit{unfit}, \textit{relax}) = 5$, $R(\textit{unfit}, \textit{exercise}) = 0$, independently of the resulting state. (Sam always enjoys relaxing more than exercising... But when Sam is fit, Sam feels much better overall.)

Sam picked $\gamma = 0.7$, in order to reflect his life philosophy of "living the moment".

(a) How many policies are there? List them.

(b) Explain the first two steps of value iteration (using both the V and Q values). Start with $V_0[s] = 0$ for all states. You don't need to write down the algorithm, only explain what it calculates. Do so at a detailed level as you may expect in an example for a tutorial on value iteration.

(c) What is the optimal policy? What are the values of the states for this policy? What are the Q-values of state-action pairs for this policy?

How did you calculate them? (You do not need to submit your calculations (or code), just explain what you did, and your results.)

(d) Sam decided to think more seriously about the long term (i.e., increase γ) and the short term (i.e., decrease γ), and realized, that the optimal policy π^* depends on γ. What policies are possible as γ changes? Explain why. Give values for γ that result in each different policy.

You may adapt the code at `http://www.cs.ubc.ca/~poole/cs502/2013/as4/mdp_vi.py` which does value iteration.

Question 2

Suppose we have a system that observes a person's TV watching habits in order to recommend other TV shows the person may like. Suppose that we have characterized each show by whether it is a comedy, whether it features doctors, whether it features lawyers, and whether it has guns. Suppose we have the following data about the person's likes for various TV shows:

Example	*Comedy*	*Doctors*	*Lawyers*	*Guns*	*Likes*
e_1	false	true	false	false	false
e_2	true	false	true	false	true
e_3	false	false	true	true	true
e_4	false	false	true	false	false
e_5	false	false	false	true	false
e_6	true	false	false	true	false
e_7	true	false	false	false	true
e_8	false	true	true	true	true
e_9	false	true	true	false	false
e_{10}	true	true	true	false	true
e_{11}	true	true	false	true	false
e_{12}	false	false	false	false	false
e_{13}	true	true	false	false	true

We want to use this data to learn the value of *Likes* as a function of the values of the other variables.

You may find the AIspace decision tree applet (`http://aispace.org/dTree/` — this is "Likes TV" sample data set) useful for this assignment. (Before you start, see if you can see the pattern in what shows the person likes.) The applet can do all of the computation you need, but won't actually solve the assignment.

(a) Suppose the error is the sum of absolute errors. Give the optimal decision tree with only one node (i.e., with no splits). What is the error of this tree?

(b) Do the same as part (a), but with the sum of squares error.

(c) Suppose the error is the sum of absolute errors. Give the optimal decision tree of depth 2 (i.e., the root node is the only node with children). For each leaf in the tree, give the examples that are filtered to that node. What is the error of this tree?

(d) Do the same as part (c) but with the sum of squares error.

(e) What is the smallest decision tree (in terms of number of nodes) that correctly classifies all examples?

(f) Suggest a way that an algorithm may be able to find the smallest tree.

Question 3

After running 1000000 steps of the tiny game at `http://artint.info/demos/rl/tGame.html`, Chris was suspicious that Q-learning converges to the correct Q-values where $\alpha_k = \frac{1}{k}$, the empirical average.

Chris then had some hypotheses about which method converges faster (or at all) to the correct Q-values:

(a) $\alpha_k = \frac{1}{k}$

(b) $\alpha_k = \frac{10}{9+k}$

(c) $\alpha_k = 0.1$.

(d) $\alpha_k = 0.1$ for the first 1,000 steps, $\alpha_k = 0.01$ for the next 1,000 steps, $\alpha_k = 0.001$ for the next 1,000 steps, $\alpha_k = 0.0001$ for the next 1,000 steps, and so on.

Either construct the simplest possible RL example you can construct that exhibits slow convergence for the empirical average, or modify the open-source code for the tiny game (α is used in `TGameQController.java`) to investigate how well each of these works.

This is intended to allow an open-ended investigative question. You can get full marks by either answering the specifics of the question or helping Chris in some other way to understand the convergence of the empirical average. Write something interesting with some (theoretical or empirical) evidence.

Question 4

Give a possible exam question (perhaps with sub-parts) that would be good to test students about decision making and/or learning as covered in class. It should be worth 10 marks, and take students approximately 10 minutes to complete in an exam setting. It must be clear what the question is asking for and must be self-contained. Give a solution.

Question 5

On the wiki `http://wiki.ubc.ca/Course:CPSC:Artificial_Intelligence` create pedagogical or real-world examples that use useful for students to learn about decision-theoretic planning, or an aspect of learning. Please add references for real-world examples. You will need to login with your CWL to edit. This is intended to be an open-ended creative question. This is a cooperative question, as anyone can edit other people's questions. It is possible to gain credit by improving other's contributions. Please help to build a useful resource.

It can be worth multiple questions in this part; please justify any claim of how many questions your contribution is worth. It is even possible to do the whole assignment just by creating useful resources.

Question 6

For each question in this assignment, say how long you spent on it. Was this reasonable? What did you learn?