Automated Planning

Markov Decision Processes Assignment
Assigned: 27 October
Due: Thursday, 03 November, 10:40am

Prof. Felipe Meneguzzi

October 26, 2016

1 MDPs in pymdptoolbox - Class Assignment

In this practical, we will look at how MDP planning is implemented in a mathematical toolkit, and track the calculation of the rewards for each state via value iteration[1].

First, download source code with the MDP functions from Moodle at `http://moodle.pucrs.br`.[2] The package contains the pymdptoolbox and the additional python files to help you with this assignment. Second, install the pymdptoolbox toolkit. To install the toolkit you must have the following dependencies: **NumPy and SciPy**. If you are using Linux the following command will install both dependencies:

```
sudo apt-get install python-numpy python-scipy python-cvxopt
```

If you are using Windows, Numpy and Scipy provide a number of Windows distributions at . If you are using Mac OSX, you can install the dependencies following this guide at `http://sourabhbajaj.com/mac-setup/Python/numpy.html`. Once you have the dependencies installed, you can install pymdptoolbox by executing the following command on the main folder:

```
python setup.py install
```

Additionally, if have pip available, you can install using pip with the following command:

```
pip install pymdptoolbox
```

Once you have downloaded the class toolkit:

- Open a terminal to execute python or open a python IDE.
- Open the `ap_mdp_lab.py` file in your favorite text editor and check out the code.
- Analyze and understand the code. You can execute with ‘‘`python ap_mdp_lab.py`’’.
- The output will be the policy for an MDP grid domain.

[1] This should be more dynamic than programming it in class.

[2] Credits to the group at the university of Toulouse that made the MDPToolbox toolkit available: `http://www7.inra.fr/mia/T/MDPtoolbox/MDPtoolbox.html`. And to Steven Cordwell for porting it to Python: `https://github.com/sawcordwell/pymdptoolbox` Note that the version of this toolbox **has been modified** to conform with the MDP model we saw in class.

1. Study the code in `ap_mdp_lab.py` and answer the following questions.
   (a) What is the policy generated if we change the discount factor of the grid domain to 0.1?

(b) Add the following line in `ap_mdp_lab.py` before `vi.run()`:

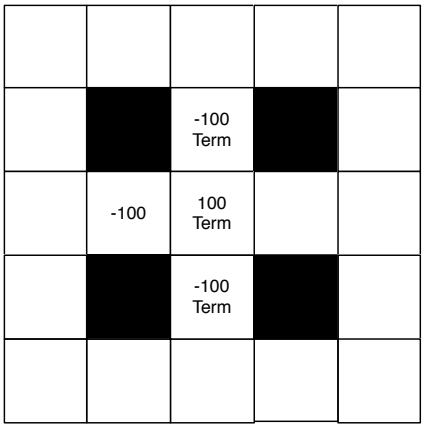
```
vi.verbose = True
```

What is the variation for each of the first three iterations with the discount factor of 0.9 and how much iterations the algorithm takes to converge?

(c) How does changes to the discount factor affect the variation of the state values over time?

2 Other MDPs - Home Assignment

2. (2 points) The scenario below has an interesting structure whereby the positive rewarding terminal state is partially surrounded by negatively-rewarding states. Program this scenario in pymdptoolbox and compute the optimal policy with a discount factor of 0.99,



3. Define two new 5 by 5 domains with multiple obstacles and an interesting geometry and following the guidelines below. Calculate the policy with discount factor 0.99, and then try to explain intuitively the reason for the resulting policies, given the initial parameters. You can program these scenarios with any programming language you like (e.g. there are classes in Java that will help you do this `http://code.google.com/p/aima-java/`), but also including pymdptoolbox. These two scenarios must have the following characteristics:
   (a) (3 points) A scenario with one (or more) terminal states with positive rewards and at least one other state with the same amount of, but negative reward and no terminal states with negative rewards.

   (b) (3 points) A scenario with one terminal state with a negative reward and at least one non-terminal state with a positive reward.

3 Grading

In order to properly evaluate your work and thought process, you will write a 2-page report in the AAAI conference format explaining your encoding and experiments, and answer the questions posed above. These guidelines are to be followed **exactly**. **Reports that are less than two pages of actual content, or not in format will receive 0 marks for the report criterion.** This report will be included in the deliverables of **Part B** of the assignment. The formatting instructions are available at the AAAI 2017 website[3]. The report must have the following sections:

- An introduction with your understanding of the problem domain, outlining the remainder of the paper;
- A section explaining your scenarios and answering the questions from Section 2
- One conclusion section, where you will summarize your experience in encoding MDPs and analyzing the resulting policies.

Grading will take consider elements of your encoding, experimentation and reporting of the work done. The criteria, as well as their weight in the final grade is as follows:

- Questions (80%) — correctness of the questions domain knowledge encoding, in relation to the domain specification from Section 2;
- Overall report readability (20%) — how accessible and coherent your explanation of your scenarios and their solutions is.

[3] `http://www.aaai.org/Publications/Templates/AuthorKit17.zip`