CSE 250A Assignment 8

This assignment is due at the start of the last lecture in 250A, which is on Thursday December 6, 2012. Instructions are the same as for previous assignments. You must work in partnership with one other student, but you may keep the same partner or change partners, as you wish.

1, 2, 3. Markov decision processes

Do problems 7.2, 7.3, and 7.4 on the assignment written by Lawrence Saul that is available at `http://cseweb.ucsd.edu/classes/fa11/cse250A-a/hw/hw7/hw7.pdf`.

4. Least squares policy iteration (LSPI)

Do either part (a) or part (b). Part (a) is more restricted, while part (b) is more interesting. To be fair to students who have no previous experience with Python, the available credit for both parts is the same, but we will be appreciative of those who do (b).

(a) The LSPI algorithm will be explained in class and in the online lecture notes. Implement LSPI in the programming language of your choice. Make your implementation short, straightforward, and readable. Use a standard library, or language primitives, for matrix and vector operations. Construct a toy test case to show that your implementation works. Show that the policy learned by LSPI is better than a simple policy such as a pure greedy policy.

Notes: By designing and coding the test case yourself, you will show that you understand the specification of the LSPI algorithm. Make the test case simple enough that you can follow the execution of the algorithm and be convinced that it is correct. The test case does not have to be interesting. Submit the code for LSPI and for the test case. Also submit a transcript of running LSPI on the test case. Write notes explaining what the output should be, and what it is.

(b) Implement LSPI in Python, in the PyBrain software environment, and apply it to the cart-pole domain. PyBrain includes an implementation of this domain and of the neural fitted Q iteration (NFQI) algorithm. Compare experimentally the performance of LSPI and NFQI in the cart-pole domain. The most important dimension on which to compare algorithms is the number of (s, a, r, s') training events needed in order to learn a successful policy.

Write a report describing your results. In the report, describe the experiments that you did to obtain the results, and the implementation work that made the experiments possible. The focus of attention should be the results; the purpose of explaining the work you did is to make the reader feel comfortable that the results are reliable. Show results in a way that is easy to understand.

Notes: This assignment is open-ended. You will need to use your own judgment repeatedly. For a general overview of PyBrain, see `http://jmlr.csail.mit.edu/papers/volume11/schaul10a/schaul10a.pdf`. Install the software from `http://pybrain.org`. For an overview of how to implement an RL method inside PyBrain, see `http://simontechblog.blogspot.com/2010/08/pybrain-reinforcement-learning-tutorial_15.html`. As is unfortunate but typical, there is no guarantee that documentation is up-to-date and free of errors. PyBrain includes implementations of the cart-pole domain and of the NFQI algorithm Start by getting these to work together. Then implement LSPI to have the same interface, as much as possible, as NFQI.