Assignment 10

Machine Learning, Summer term 2015

Prof. Ulrike von Luxburg, Morteza Alamgir, Tobias Lang, Norman Hendrich

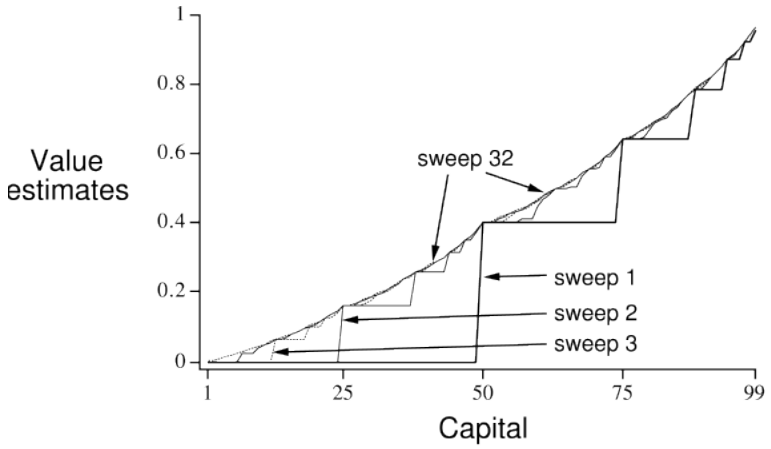
June 25, 2015

Assignment 10.1 (Example reinforcement learning tasks, 3+3 points) Devise and describe two example tasks of your own that might be solved by the reinforcement learning approach. For both of your examples, identify and describe the states and actions, and give a suitable rewards-function. Also estimate the size of the state-space: how many dimensions do you need to represent the (s, a) state-action space, and how many states are there in total? Try to make your examples as different from each other as possible, using different contexts and application areas.

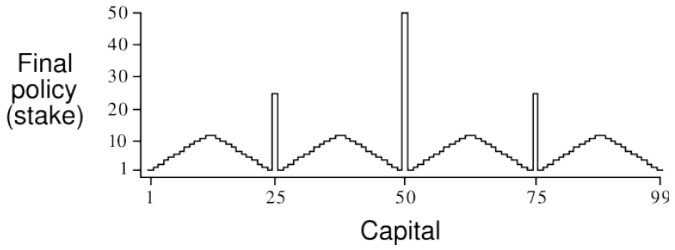
Assignment 10.2 (Gambler's Problem, 3+11 points)

A gambler has the opportunity to make bets on the outcomes of a sequence of coin flips (this is example 4.3 from the Sutton and Barto book). If the coin comes up heads, he wins as many dollars as he has staked on that flip; if it is tails, he loses his stake. The game ends when the gambler wins by reaching his goal of \$100, or loses by running out of money. On each flip, the gambler must decide what portion of his capital to stake, in integer numbers of dollars.

This problem can be formulated as an undiscounted, episodic, finite MDP. The state is the gambler's capital $s \in \{1, 2, \ldots, 99\}$, and the actions are stakes, $a \in \{0, 1, \ldots, \min(s, 100 - s)\}$. The reward is zero on all transitions except those on which the gambler reaches his goal, when it is $+1$. The state-value function then gives the probability of winning from each state. A policy is a mapping from levels of capital to stakes. The optimal policy maximizes the probability of reaching the goal. Let p denote the probability of the coin coming up heads. If p is known, then the entire problem is known and it can be solved by any of the learning algorithms discussed in the lecture. The figure shows the change in the value function over successive sweeps of value iteration, and the final policy found, for the case of $p = 0.4$.



The solution to the gambler's problem for $p = 0.4$. The graph shows the value function found by successive sweeps of value iteration.



The graph shows the final policy for $p = 0.4$.

10.2.a: Why does the optimal policy for the gambler's problem have such a curious form? In particular, for capital of 50 it bets it all on one flip, but for capital of 51 it does not. Why is this a good policy?

10.2.b: Implement a program for the gambler's problem and solve it for $p = 0.25$, $p = 0.4$, and $p = 0.55$. Plot the value function and the policy found by your program. Note: you may find it convenient to introduce two dummy states corresponding to termination with capital of 0 and 100, giving them values of 0 and 1 respectively.