Reinforcement Learning Tutorial 3, Week 6

—

Monte Carlo / TD Prediction / Reward Shaping

Pavlos Andreadis, Sanjay Rakshit

February 2020

Overview: The following tutorial questions relate to material taught in week 3 of the 2019-20 Reinforcement Learning course. They aim at encouraging engagement with the course material and facilitating a deeper understanding.

Problem 1 - Modelling & Monte Carlo

Consider the simple maze problem in the figure below, comprised of 8 states $s_1, \cdots, s_8$, numbered from the bottom left to the top right. The agent can move from any state to any adjacent state (e.g. from s_1 to either s_4 or s_2), without error. Our goal is to follow the shortest path (from any state) to s_8. Upon arrival to a new state, the agent receives a reward dependent only on that new state. We assign s_8 a reward of 10, and penalise arrival to any other state with -1.

The arrows in the figure below summarise the policy π_0 which we will be evaluating in *Part b* of this question. Essentially, assume a deterministic policy for states s_2, s_3, s_5, s_6, s_7, as indicated by the respective arrow. Further assume a 50% chance of moving in either direction for states s_1, s_4.

$(s6, \rightarrow, -1)$	$(s7, \rightarrow, -1)$	$(s8, +10)$
$(s4, \uparrow, \downarrow, -1)$		$(s5, \uparrow, -1)$
$(s1, \uparrow, \rightarrow -1)$	$(s2, \rightarrow, -1)$	$(s3, \uparrow, -1)$

Part a

- Should s_8 be defined as a terminating state? Why?
- Should s_8 be defined as an absorbing state? Why?

From here on, assume a discount factor of $\gamma = 1$.

Part b

Assuming the starting state $S_0 = s_1$ and the policy π_0 outlined above, list the 2 shortest possible trajectories our agent can follow (stopping at state 8). Further to that, consider the trajectory:

$$(s_1, up), -1, (s_4, down), -1, (s_2, right), -1, (s_3, up), -1, (s_5, up), +10, (s_8).$$

For each of those trajectories, carry out an iteration of policy evaluation using First-visit Monte Carlo (where it is implied that you average across samples as opposed to using some other learning rate), computing the *action value function*. Start from an initial evaluation of 0 across state-action pairs and go through the trajectories in any order.

Part c

Perform 1-step of greedy policy improvement on policy π_0 (assuming no access to the model), based on the evaluation from Part b.

Problem 2 - TD Prediction

Use the trajectory

$$(s_1, right), -1, (s_2, right), -1, (s_3, up), -1, (s_5, up), +10, (s_8).$$

to run 1 iteration of Temporal Difference policy evaluation (use the SARSA update rule) on the policy π_1 you computed for Problem 1c. Assume a step size of $\alpha = 0.1$ (you are assuming that the action that would be taken at each time-step is the one indicated in the trajectory).

Problem 3 - Discussion: Reward Shaping

Assume the *problem as described in Problem 1*, but where after evaluating a deterministic policy π_2 (as given in the figure below) (e.g. using TD(0)), we have the below estimation of the state-value function:

$(s6, \rightarrow, v = 0)$	$(s7, \rightarrow, v = 0)$	$(s8)$
$(s4, \downarrow, v = 6)$		$(s5, \uparrow, v = 10)$
$(s1, \rightarrow, v = 7)$	$(s2, \rightarrow, v = 8)$	$(s3, \uparrow, v = 9)$

We are frustrated that our agent has not learnt to "go up" when in state s_4, and decide to, instead of running the process further, apply reward shaping to

the model, adding +2 reward to the state visits for states s_6, s_7, hoping that this will help the agent learn to take the shortest route from s_4.

1. What do you think would be the optimal policy for this modified MDP (with rewards for arriving at states s_6 and s_7 of +1)? Would the episodes terminate?
2. If instead of +2 to the above rewards, we instead add +1 (rewards for arriving at states s_6 and s_7 of 0), what would be the optimal policy?
3. In the above two models, would the calculated state-value function, after convergence, be representative of the original problem? Why?
4. When can reward shaping be a useful tool, and how?