Reinforcement Learning: An Introduction - Chapter 1

Thomas Hopkins

Reading Notes

- "the reward function must necessarily be unalterable by the agent" :
  - What if reward functions were evolved over a population of individuals?
  - This could be similar to how animal's receive reward through pleasure/pain. The pleasure/pain signal changing through evolution.
  - Each individual in a population has its own (evolved) reward function and learns to act in that environment using rewards derived from that function
  - Possible experiment: Population of individuals in a maze, rewards for states are evolved, some states "kill" the individual while others have no effect
    - Now clearly, the reward can just be defined to make the "kill" states -1 with every other state 0 (or 1)
    - But, more complex problems don't have clear reward functions and so evolving them might produce better alternatives than ones humans could define [1].

[1] https://dl.acm.org/doi/abs/10.1145/2001858.2001957

Exercise 1.1

I think that the reinforcement learning algorithm would converge to the optimal policy of always forcing a tie. This is because it will know what moves to make against a poor opponent as well as a good opponent. It will incrementally make better decisions and have to play tougher games to win. I think it would learn a more general way of playing rather than overfitting to some specific fault in the fixed opponent's play.

Exercise 1.2

You could take advantages of symmetry which will improve performance by allowing the agent to learn faster. This can be achieved by backing up states as well as symmetrical states in one step of learning. If an opponent's policy is different for symmetric states then we should not take advantage of the symmetry to do backups and do them normally instead. This is because the backed up value will oscillate between the values given based on the opponent's actions.

Exercise 1.3

If the reinforcement learning player is greedy, then it would always play the move it thinks is

best. It would probably learn to play worse than a non-greedy player since it would not explore enough of the state space to have an accurate estimate of the value for each state. Over time, however, the agent may converge to the optimal policy of always ending the game with a tie since it will lose enough to make the states it thinks are best have very low value, leading to other action selections.

Exercise 1.4

When we learn from exploratory moves, we are introducing the bias of making non-optimal moves in the action selection process. This will lead to the value of the state to have a different probability of winning than if the backups were not made. It is best that we do not backup the states after exploratory moves since this set will represent the actual probabilites of winning. For example, if the probabilites of winning were determined on the basis of counts (i.e. +1 for every state that led to a win) then backing up exploratory states would change this count. This is an issue since the value no longer represents an estimate of true value of that state (the probability of winning the game from that state). If we continue making exploratory moves at the same rate, it will be preferable to backup exploratory states since this will account for the randomness involved in exploration. Not backing up exploratory states will result in more wins since the value of the states will not be biased in favor of potentially choosing exploratory moves.

Exercise 1.5

We can look further ahead using a model of the environment to make better estimates of the value of each state.