CS 312: Artificial Intelligence Laboratory
Lab 8 Report

S V Praveen - 170010025

April 24, 2020

1 MDP Description

The game High-Low is a card game played with an infinite deck containing three types of cards: 2,3,4. You start with a 3 showing(,ie, the start state), and say either higher or lower. Then, a new card is flipped; if you say higher and the new card is higher in value than your current card, you win the points shown on the new card. Similarly, if you say lower and the new card is lower in value than your current card, you win the points shown on the new card. If the new card is the same value as your current card, you don't get any points. Otherwise, the you get a penalty for guessing wrong. Your current card is discarded and the new card becomes your current card.

The states are given by,

$$S = \{2, 3, 4\} \tag{1}$$

The actions are given by,

$$A = \{HIGH, LOW\} \tag{2}$$

Let p2, p3 and p4 be the different proportions of 2, 3, and 4 cards in the deck. Let this be represented by the proportion matrix M,

$$M = \{p2, p3, p4\} \tag{3}$$

The transition probabilities,P for this problem from the current state s to the new state s' using action a is given by,

$$P(s, a, s') = M[s' - 2] \tag{4}$$

Note: $s' - 2$ helps to index the zero-indexed matrix M appropriately.

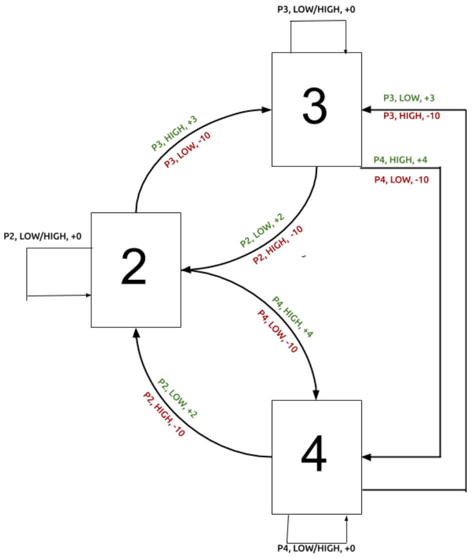
The given MDP has an infinite time horizon so $N = \infty$.

The Rewards, R, is given by

$$R(s, a, s') = \begin{cases} 0 & \text{if s = s'} \\ PENALTY & \text{if s'> s, a = LOW} \\ PENALTY & \text{if s'< s, a = HIGH} \\ s' & \text{otherwise} \end{cases}$$

where PENALTY $\in \mathbb{Z}^{-}. Here, PENALTY = -10$.

2 State Transition Graph



3 Optimal Policy

An easy way to verify if the algorithm gives us the optimal policy is to test it on trivial cases. We set the p2 = p3 = 0 and p4 = 1. Thus we have a deck of only 4's and the expected policy is [HIGH, HIGH, HIGH/LOW]. Since the algorithm does indeed give us this optimal policy we say that it works optimally. Similarly, we test it for p2=p4=0 and p3=1 followed by testing for p3=p4=0 and p2=1. The algorithms give us the expected policy in each of these cases.

4 Experimental Results on varying γ

Let H be High, L be Low.

4.1 Results in Value Iteration

Input	**Policy Found**		
	γ=0.1	**γ=0.5**	**γ=1**
input1	[H, L, L]	[H, L, L]	[H, L, L]
input2	[H, H, L]	[H, H, L]	[H, H, L]
input3	[H, H, L]	[H, H, L]	[H, H, L]

4.2 Results in Policy Iteration

Input	**Policy Found**		
	γ**=0.1**	γ**=0.5**	γ**=1**
input1	[H, L, L]	[H, L, L]	[H, L, L]
input2	[H, H, L]	[H, H, L]	[H, H, L]
input3	[H, H, L]	[H, H, L]	[H, H, L]

5 Comparison of Policy Iteration and Value Iteration

Clearly, both policy iteration and value iteration converge to give us optimal policies to for the given Markov decision process. However, it is important to note that policy iteration gives us the solution quicker as it converges faster. As seen in the examples, policy iteration converges in merely 2 iterations whereas value iteration takes 100 iterations to arrive at the optimal policy.

6 Conclusion

While value iteration involves finding an optimal value function and then extracting the policy, Policy iteration combines policy evaluation with policy improvement to make it converge faster. Thus, policy iteration is preferred to solve the High-Low problem and this concept can be extended to solve other Markov Decision processes.