Artificial Intelligence Lab Report 10

1st Dipean Dasgupta
202151188
BTech CSE
IIIT,Vadodara

2nd Shobhit Gupta
202151149
BTech CSE
IIIT,Vadodara

3rd Rahul Rathore
202151126
BTech CSE
IIIT,Vadodara

4th Rohan Deshpande
202151133
BTech CSE
IIIT,Vadodara

***Abstract*—To understand sequential decision making with Markov Decision process through G bike rental problem and Gridworld problem.**

I. INTRODUCTION

Learning Objective:To understand the process of sequential decision making (stochastic environment) and the connection with reinforcement learning
Problem Statements:

A. Grid World Problem

Suppose that an agent is situated in the 4x3 environment as shown in Figure 1. Beginning in the start state, it must choose an action at each time step. The interaction with the environment terminates when the agent reaches one of the goal states, marked +1 or -1. We assume that the environment is fully observable, so that the agent always knows where it is. You may decide to take the following four actions in every state: Up, Down, Left and Right. However, the environment is stochastic, that means the action that you take may not lead you to the desired state. Each action achieves the intended effect with probability 0.8, but the rest of the time, the action moves the agent at right angles to the intended direction with equal probabilities. Furthermore, if the agent bumps into a wall, it stays in the same square. The immediate reward for moving to any state (s) except for the terminal states S+ is r(s)= -0.04. And the reward for moving to terminal states is +1 and -1 respectively. Find the value function corresponding to the optimal policy using value iteration.

Find the value functions corresponding optimal policy for the following: r(s)=-2 r(s)=0.1 r(s)=0.02 r(s)=1

B. Gbike Rental Problem

You are managing two locations for Gbike. Each day, some number of customers arrive at each location to rent bicycles. If you have a bike available, you rent it out and earn INR 10 from Gbike. If you are out of bikes at that location, then the business is lost. Bikes become available for renting the day after they are returned. To help ensure that bicycles are available where they are needed, you can move them between the two locations overnight, at a cost of INR 2 per bike moved. Assumptions: Assume that the number of bikes requested and returned at each location are Poisson random variables. Expected numbers of rental requests are 3 and 4 and returns are 3 and 2 at the first and second locations respectively. No more than 20 bikes can be parked at either of the locations. You may move a maximum of 5 bikes from one location to the other in one night. Consider the discount rate to be 0.9.
Formulate the continuing finite MDP, where time steps are days, the state is the number of bikes at each location at the end of the day, and the actions are the net number of bikes moved between the two locations overnight.
Download and extract files from gbike.zip. Try to compare your formulation with the code. Before proceeding further, ensure that you understand the policy iteration clearly.

C. Problem 3

Write a program for policy iteration and resolve the Gbike bicycle rental problem with the following changes. One of your employees at the first location rides a bus home each night and lives near the second location. She is happy to shuttle one bike to the second location for free. Each additional bike still costs INR 2, as do all bikes moved in the other direction. In addition, you have limited parking space at each location. If more than 10 bikes are kept overnight at a location (after any moving of cars), then an additional cost of INR 4 must be incurred to use a second parking lot (independent of how many cars are kept there).

II. THEORY

The solutions of the given problems are as follows.

A. Grid World Problem

A mathematical model that depicts a system that varies randomly or probabilistically across time is called a stochastic process. The problems in these labs are sequential, so decisions you make affect more than just your immediate gain.

1) State: In all, there are twelve states. This corresponds to the initial state s_1, when the agent is at square (1,1). Nine non-terminal states make up the remaining states, with two target states.

2) Action: At each state, the agent can take 4 actions,

- Move up, which will move the agent one cell up, unless the agent is already in the top row, in which case the agent stays in the same cell.
- Move down, which will move the agent one cell down, unless the agent is already in the bottom row, in which case the agent stays in the same cell.
- Move left, which will move the agent one cell to the left, unless the agent is already in the leftmost column, in which case the agent stays in the same cell.

- Move right, which will move the agent one cell to the right, unless the agent is already in the rightmost column, in which case the agent stays in the same cell.

3) Value Function :

$$V_\pi(s) = \sum_a \pi(a|S) \sum_{s',r} (s', r|s, a)[r + \gamma * V_\pi(s')]$$

s is current state , a is the action and s' is the next state, r is the reward.

B. Gbike Rental Problem and Algorithm

The MDP formulation of the Gbike Rental Problem is as follows -

1) State: The states can be represented as -

$$[b_t^1, b_t^2]$$

$$b_t^i \in \{0, 1, ..., 20\}$$

b_t^1 and b_t^2 represent the number of bikes available at location 1 and location 2 respectively.

2) Action: Sign convention -
If bikes transferred from location 1 to location 2 $\rightarrow$ +ve
If bikes transferred from location 2 to location 1 $\rightarrow$ -ve
The action $A(s_t)$ is in the range -

$$\{-min(min(5, b_t^2), 20 - b_t^1), min(min(5, b_t^1), 20 - b_t^2)\}$$

3) Transition Probability and Reward: By performing action $A_t = A(s_t)$, the state is transfered from

$$[b_t^1, b_t^2] \ to \ [b_{t+1}^1, b_{t+1}^2]$$

The chance node being

$$s_t, A_t$$

with reward $-2|A_t|$.
After performing the action A_t, the number of bikes remaining at the locations are as follows -

$$b_t^1 - A_t, b_t^2 + A_t$$

The next day, bikes are rented and returned, changing the number of bikes at each location, the final state is as follows -

$$[b_{t+1}^1 = min(b_t^1 - A_t - (min(b_t^1 - A_t, r_{t+1}^1)) + R_{t+1}^1, 20),$$
$$b_{t+1}^2 = min(b_t^2 - A_t - (min(b_t^2 - A_t, r_{t+1}^2)) + R_{t+1}^2, 20)]$$

Where,

$$requests \ \rightarrow [r_{t+1}^1, r_{t+1}^2]$$
$$returns \ \rightarrow [R_{t+1}^1, R_{t+1}^2]$$

Transition Probability -

$$P(r^1)P(r^2)P(R^1)P(R^2)$$

Where, Probability distribution is Poisson Distribution
Reward - from $s_t \rightarrow s_{t+1}$

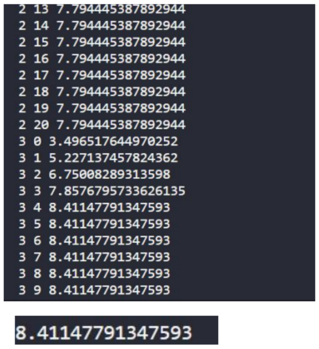
$$10min(b_t^1 - A_t, r_{t+1}^1) + 10min(b_t^2 - A_t, r_{t+1}^2) - 2|A_t|$$

Algorithm
The basic steps of policy iteration are as follows:

Initialization: Establish a starting policy that outlines the course of action to be followed at every MDP stage. Policy evaluation: Calculate the value function, which indicates the expected total reward that can be attained by adhering to the policy from each state, given the current policy. Enhance the policy by making it more avaricious in relation to the existing value function. This means that the action that maximizes the predicted total reward from every state should be chosen. Till convergence, repeat steps two and three.

III. Experiments and Analysis

Observations:

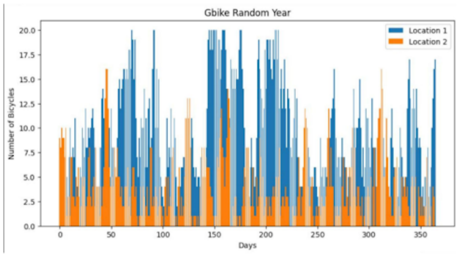


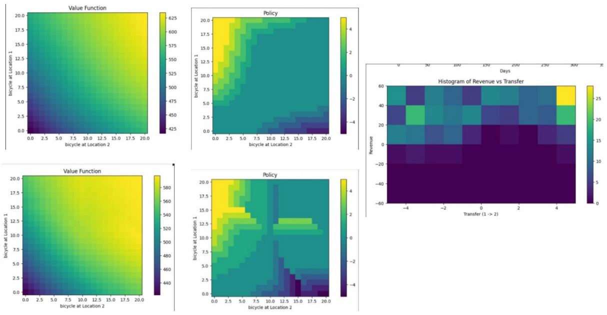
```
2 13 7.794445387892944
2 14 7.794445387892944
2 15 7.794445387892944
2 16 7.794445387892944
2 17 7.794445387892944
2 18 7.794445387892944
2 19 7.794445387892944
2 20 7.794445387892944
3 0 3.496517644970252
3 1 5.227137457824362
3 2 6.75008289313598
3 3 7.8576795733626135
3 4 8.41147791347593
3 5 8.41147791347593
3 6 8.41147791347593
3 7 8.41147791347593
3 8 8.41147791347593
3 9 8.41147791347593
```

```
8.41147791347593
```

A. Problem 2 and Problem 3

The MDP is formulated, and algorithm is mentioned in the theory section.





IV. CONCLUSION

From the observations of problem 1, we can assess that

- The optimal value function and optimal policy can be computed by iteratively applying the Bellman equation and updating the value function until convergence.
- The formulation of Value function has been done.
- The Gridworld problem provides an example how MDPs can be used to model decision-making problems, and how value iteration can be used to compute the optimal policy.

The formulation of MDP for problem 2 has been done. From the observations of problem 3, we can assess that

- We use a value alpha to track the weights assigned to the observed rewards when the rewards are not stationary. We prioritize the most recent prizes. Furthermore, the graph indicates that the average reward received is really near to the highest payout that is possible. Before evaluating the outcomes, we performed 10,000 iterations.

V. All Lab Codes

All the codes of the 10 labs performed can be accessed from the link: Github

References

[1] Reinforcement Learning: an introduction by R Sutton and A Barto (Second Edition) (Chapter 1-2)