Chapter 14
Reinforcement Learning

1. Let there be three universities. The students of X university prefer to do master degree in university X is 60%. 30% of university X go to university Y and 10% of the students go to university Z. Similarly, the number of students of university Y registered in the same Y university and university X and Y respectively are 50%, 30% and 20% respectively. All the students of Z university prefer the same university for masters, and none go for university X and Y. Find the transition matrix. Predict for two years if the distribution is (0.5 0.3 0.2)

Solution:

The starting distribution is (0.5 0.3 0.2). The Markov chain is shown In Fig 15.9 and transition matrix is given as

The prediction after a year is given as

$$u^{(n)} = uP^n$$

$$= \begin{pmatrix} 0.5 & 0.3 & 0.2 \end{pmatrix} \begin{pmatrix} 0.6 & 0.3 & 0.1 \\ 0.5 & 0.3 & 0.2 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 0.45 \\ 0.24 \\ 0.31 \end{pmatrix}$$

The hold of the universities will be after a year would be 45%, 24% and 31%.

The second-year prediction would be

$$P = \begin{pmatrix} 0.45 & 0.24 & 0.31 \end{pmatrix} \begin{pmatrix} 0.6 & 0.3 & 0.1 \\ 0.5 & 0.3 & 0.2 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 0.395 \\ 0.207 \\ 0.403 \end{pmatrix}$$

Thus, the hold of the universities will be 39.5%,20.7% and 40.3%.

2. Consider the following grid game where a robot can move,

	Block	
		Goal

Name the empty grids and write states, actions and episodes.

Solution:

One can identify the nodes using names as follows

A	Block	B
C	D	Goal
E	F	G

One can assign the index for these states as state values

State Index Table

Locations	State
A	0
Block	1
B	2
C	3
D	4
Goal	5
E	6
F	7
G	8

The actions of the robot would be like moving from G-D, D-A like that. The actions are next possible locations a robot can move on. The destinations also can be encoded in indexes same as the state. So, the possible actions are {0,1,2,3,4,5,6,7,8}. The episode would be said A, C, D, Goal. E, C, D, Goal. There can be many episodes possible.

3. If the reward is 10, for the 10^{th} move with a discount factor of 0.3, what is the reward.

Solution:

The reward happens at 10^{th} step and it is 10. Therefore, the discounted reward is

$$(0.3)^{10} \times 10 = 5.9049 \times 10^{-5}$$

4. Show the value function for the agent in position (2,1) in Table 15.6. The grid actions are UP with a probability of 0.8, RIGHT and LEFT with the probability of 0.1. How values are determined and write the Bellman equation?

Table 15.6 Grid Positions

(1,3)	(2,3)	(3,3)
(1,2)	(2,2)`	(3,2)
(1,1)	(2,1) Agent Position	(3,1)

Solution: The agent is in position (2,1). The actions are up, left and right with probability 0.1, 0.1 and 0.1 respectively. So, the Bellman equation of the position (2,1) is

$$v_\pi(2,1) = r(2,1) + \gamma[0.1 \times v_\pi(2,2) + 0.1 \times v_\pi(1,1) + 0.1 \times v_\pi(3,1)]$$

Thus, the value of the position depends on the value of the other locations. Initially, the values are initialized with value, and the equations are formed for every location and solved.

5. Using dynamic programming, find all the total number of paths from school to home in the following grid position as shown in Fig. 15.20 using the allowed actions right and down.

$$\begin{bmatrix} Home & - & - \\ - & - & - \\ - & - & School \end{bmatrix}$$

Fig. 15.20: Initial Board Configurations

and in the game as shown in Fig. 15.21, coins are given as c. Show the paths where the maximum amount of coins can be collected.

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array}\begin{bmatrix} Home & 0 & 0 \\ 0 & c & c \\ 0 & 0 & School \end{bmatrix}$$
$$\begin{array}{ccc} 0 & 1 & 2 \end{array}$$

Fig. 15.21: Initial Coin Positions

Solution:

To solve this problem, one must use the dynamic programming first the total number of paths that are possible from school to home. Let us start a table called Q-Table with the indexes as shown below

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$\begin{array}{ccc} 0 & 1 & 2 \end{array}$$

Initially the Q-Table is initialized to zero. Let us discuss about the formulation of the recursive formula (or Bellman equation) to solve this problem. Let us take a grid position, Q(0,3). Since the allowed operations are right and bottom, the tiles with be Q(0,2) or Q(1,3). In other words, the recursive formulation can be done as follows:

$$Q(i, j) = 1 + \max\left(Q(i+1, j) + Q(i, j+1)\right)$$

The last row can have only right operation (as only right and down) are permitted operations as per the problem. Therefore, this is base condition and its value is 1. Similarly, the third column can have only down operation and therefore, it is also a base condition and its value is 1. This results in the Q-table as follows:

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

Fig. 15.23a: Base Conditions

The values for the rest of the positions can be computed as

Q(1,1) = 1+ max { Q(1+1,1) + Q(1,1+1) as per recursive equation

= 1 + max (Q(2,1),Q(1,2)} = 1 + 1 = 2.

This results in the following Fig. 15.23b.

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 0 & 0 & 1 \\ 0 & 2 & (1) \\ 1 & (1) & 1 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

Fig. 15.23b: First Iteration

It can be observed that the 2 is the resultant of the sum of the right + bottom position. Using this shortcut, one can quickly calculate the other values also as

Q(1,0) = 1+ max { Q(1+1,0) + Q(1,0+1) as per recursive equation

= 1 + max (Q(2,0),Q(1,1)} = 1 + 2 = 3.

This is shown in Fig. 15.23c.

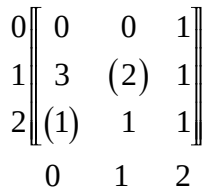


Fig. 15.23c: Second Iteration

And Q(0,1) can be computed as

Q(0,1) = 1+ max { Q(0+1,0) + Q(0,1+1) as per recursive equation

= 1 + max (Q(1,0),Q(0,2)} = 1 + 2 = 3.

This is shown in Fig. 15.23d.

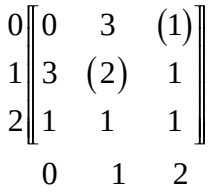


Fig. 15.23d: Third Iteration

Ans finally the starting position Q(1,0) can be computed as

Q(0,0) = 1+ max { Q(0+1,0) + Q(0,1+1) as per recursive equation

= 1 + max (Q(1,0),Q(0,2)} = 3 + 3 = 6

This is shown in Fig. 15.23e.

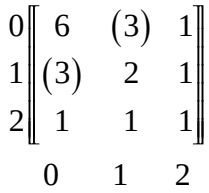


Fig. 15.23e: Fourth Iteration

So totally six paths are available from start to the destination. The six paths are given as shown in Fig. 5.24.

$$\left[\begin{matrix}\rightarrow & \rightarrow & \rightarrow \\ 0 & 0 & \downarrow \\ 1 & 0 & \text{school}\end{matrix}\right] \left[\begin{matrix}\rightarrow & \downarrow & \\ 0 & \rightarrow & \downarrow \\ 1 & & \text{school}\end{matrix}\right] \left[\begin{matrix}\rightarrow & \downarrow & \\ 0 & \downarrow & \\ 0 & \rightarrow & \text{school}\end{matrix}\right]$$

$$\left[\begin{matrix}\downarrow & 0 & 0 \\ \downarrow & 0 & 0 \\ \downarrow & \rightarrow & \text{school}\end{matrix}\right] \left[\begin{matrix}\downarrow & & \\ \rightarrow & \rightarrow & \downarrow \\ 1 & & \text{school}\end{matrix}\right] \left[\begin{matrix}\downarrow & & \\ \rightarrow & \downarrow & \\ 0 & \rightarrow & \text{school}\end{matrix}\right]$$

It can be observed that two paths that gather maximum coins is

$$\left[\begin{matrix}\rightarrow & \downarrow & \\ 0 & \rightarrow & \downarrow \\ 1 & & \text{school}\end{matrix}\right]$$

Fig. 15.24: All optimal Paths

6. Using delayed reward of $\gamma = 0.7$, compute the utility values for navigating a robot in the start position as shown in Fig. 15.11 and navigate it to the Goal state that is associated with the reward +100. All other states are initialized with the reward 0. Assume the allowed operations are UP, RIGHT, LEFT and Down. Diagonal movement is not allowed.

0	0	Goal 100
0	0	0
Start	0	0

Fig. 15.11: Robot (or agent) Navigation Problem

Solution: The discount factor that is given is $\gamma = 0.7$.

The tiles that can reach the goal state that is associated with the reward 100 can be identified. With the discount factor of 1, the utility can be computed as $\gamma \times 100 = 0.7 \times 100 = 70$. This results in the following Fig. 15.12.

0	70	Goal 100
0	0	70
Start	0	0

Fig. 15.12: Robot (or agent) Navigation after First Step

The tiles that can reach the goal state that is associated with the reward 100 in two steps can be identified. With the discount factor of 1, the utility can be computed as.

$\gamma^2 \times 100 = 0.7 \times 0.7 \times 100 = 49$ This results in the following Fig. 15.13.

49	70	Goal 100
0	49	70
Start	0	49

Fig. 15.13: Robot (or agent) Navigation after Second Step

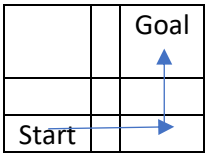
The tiles that can reach the goal state that is associated with the reward 100 in two steps can be identified. With the discount factor of 1, the utility can be computed as

$\gamma^3 \times 100 = 0.7 \times 0.7 \times 0.7 \times 100 = 34.3 \approx 34$ This results in the following Fig. 15.14.

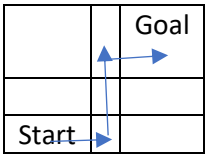
49	70	Goal 100
34	49	70
Start	51	49

Fig. 15.14: Robot (or agent) Navigation after Third Step

Now, the robot (or agent) can be navigated from the start position to the goal state by looking for highest utility. For example, from start, one can go top, top , right and goal. Six such possible paths are given in the following Fig.



b.



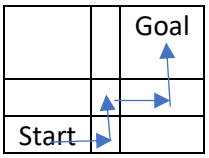


Fig. 15.15 Paths that are possible based on utility

7. Using dynamic programming and find the optimal path from start to destination as shown in Fig. 15.25. The grids have initial rewards associated with the grid. The discount factor and delayed rewards are avoided for simplicity. Find the optimal path that fetches the maximum reward for agent or robot navigation. The allowed operations are only RIGHT and DOWN.

$$\begin{bmatrix} 1 & 2 & goal(4) \\ 3 & 4 & 5 \\ Start(0) & 1 & 1 \end{bmatrix}$$

Solution

To solve this problem, one must use the dynamic programming first the total number of paths that are possible from school to home. Let us start a table called Q-Table with the indexes as follows in Fig. 15.26.

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$\begin{array}{ccc} 0 & 1 & 2 \end{array}$$

Fig. 15.26: Initial Q-Table

Initially the Q-Table is initialized to zero. Let us discuss about the formulation of the recursive formula (or Bellman equation) to solve this problem. Let us take a goal position, Q(0,3). Since the allowed operations are right and top, the tiles with be Q(0,1) and (1,2). In other words, the recursive formulation can be done as follows:

$$Q(i, j) = Q(i, j) + \max\left(Q(i, j-1) + Q(i+1, j)\right)$$

From start, one can go right or top. So, the next entries in the table are

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array}\begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$
$$\begin{array}{ccc} 0 & 1 & 2 \end{array}$$

Next, Q(2,2) can be computed as per the recursive equation as

$$Q(2,2) = 1 + \max\left(\left(Q(2,1) + Q(1,2)\right)\right) = 1 + \max(1,5) = 6$$

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array}\begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 0 \\ 0 & 1 & 6 \end{bmatrix}$$
$$\begin{array}{ccc} 0 & 1 & 2 \end{array}$$

c.

In the next iteration,

$$Q(1,1) = 0 + \max\left(\left(Q(1,0) + Q(2,1)\right)\right) = 0 + \max(3,0) = 3$$

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 0 & 0 & 0 \\ 3 & 3 & 0 \\ 0 & 1 & 6 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

d.

The next iteration gives

$$Q(1,2) = 4 + \max(3,6) = 10$$

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 0 & 0 & 0 \\ 3 & 3 & 10 \\ 0 & 1 & 6 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

e.

The next iteration gives Q(0,0) and hence Q(0,-1) = 0 .

$$Q(0,0) = 1 + \max\left(\left(Q(0,-1) + Q(1,0)\right)\right) = 4$$

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 4 & 0 & 0 \\ 3 & 3 & 10 \\ 0 & 1 & 6 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

f.

The next iteration gives

$$Q(0,1) = 1 + \max\left(\left(Q(0,0) + Q(1,1)\right)\right) = 1 + \max(4,3) = 6$$

$$\begin{matrix} 0 \\ 1 \\ 2 \end{matrix}\begin{bmatrix} 4 & 6 & 0 \\ 3 & 3 & 10 \\ 0 & 1 & 6 \end{bmatrix}\begin{matrix} \\ \\ \\ \end{matrix}$$
$$\begin{matrix} 0 & 1 & 2 \end{matrix}$$

g.

And finally gives

$$\begin{array}{c|ccc} 0 & 4 & 6 & 14 \\ 1 & 3 & 3 & 10 \\ 2 & 0 & 1 & 6 \\ \hline & 0 & 1 & 2 \end{array}$$

h.

Fig. 15.27a-h: Steps of the Path Finding

Hence 14 is the best reward for the optimal path. How can one find the optimal path? It can be found by noting the path that gave the largest reward. Therefore, it is 0, 3,4, 6, and 14. The cost is 14.