ECE 232E Project 3

Reinforcement learning and Inverse Reinforcement learning

Jui Chang

Wenyang Zhu

Xiaohan Wang

Yang Tang

2. Reinforcement learning (RL)

Question 1.

In this question, we use heat maps to visualize both Reward Function 1 and Reward Function 2. We also include the coloring scale for both maps.

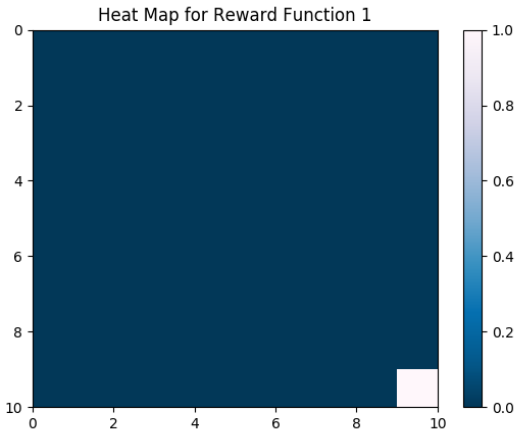

Fig1. Heat Map for Reward Function 1

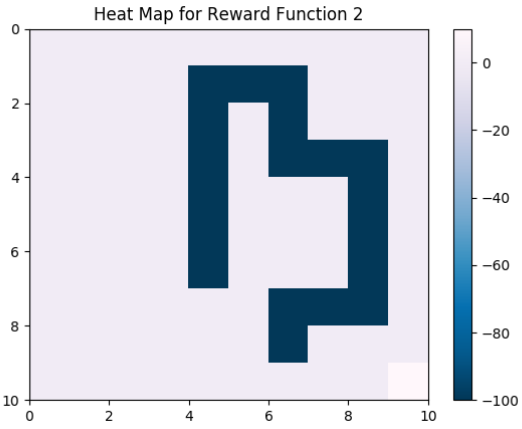

Fig2. Heat Map for Reward Function 2

3. Optimal Policy Learning using RL algorithms

Question 2.

This question defines the parameters we need: number of states = 100, number of actions = 4, w = 0.1, discount factor gamma = 0.8, reward function = reward function 1 (Question 1), and threshold = 0.01.

With the given environment, we need to realize the initialization and estimation part for the value iteration procedure. Then, a table is created to demonstrate the optimal value of each state. The table is shown below:

0.044	0.065	0.091	0.125	0.168	0.223	0.292	0.38	0.491	0.61
0.065	0.088	0.122	0.165	0.219	0.289	0.378	0.491	0.633	0.788
0.091	0.122	0.165	0.219	0.289	0.378	0.491	0.636	0.818	1.019
0.125	0.165	0.219	0.289	0.378	0.491	0.636	0.82	1.052	1.315
0.168	0.219	0.289	0.378	0.491	0.636	0.82	1.054	1.352	1.695
0.223	0.289	0.378	0.491	0.636	0.82	1.054	1.353	1.733	2.182
0.292	0.378	0.491	0.636	0.82	1.054	1.354	1.735	2.22	2.807
0.38	0.491	0.636	0.82	1.054	1.353	1.735	2.22	2.839	3.608
0.491	0.633	0.818	1.052	1.352	1.733	2.22	2.839	3.629	4.635
0.61	0.788	1.019	1.315	1.695	2.182	2.807	3.608	4.635	4.702

Fig3. Table representing the optimal value of each state

Question 3.

In order to show figure 3 in a more visualized way, we also generate its heat map across the 2D grid. The heat map is shown below:

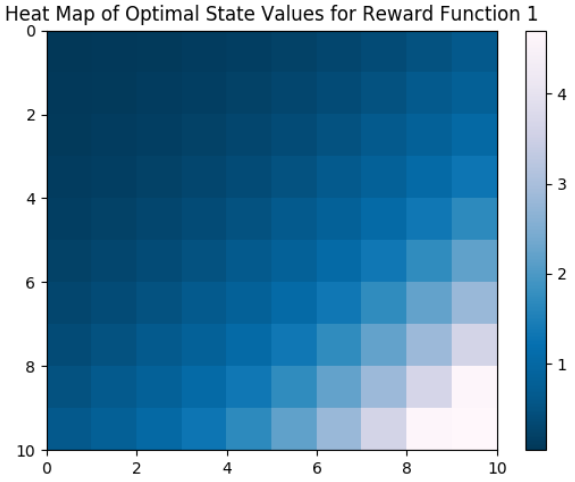

Fig4. Heat Map for Fig3

Question 4.

The higher optimal state value a state has, the higher reward it can obtain. The reason for the occurrence of this optimal state values representation in Fig4 is also described here.

The initial reward function only has value on the last state (number of state = 99). According to the update function for values in the estimation step, we can find that each state value is influenced by its surrounding 4 neighbors (and itself). Therefore, the values are impacted beginning from the right-bottom part of the 2D grid (where the reward function has larger value).

On the other hand, due to the discount factor = 0.8 in the same function, there exists a decay from one state to its neighbor states. So that's the reason why the final optimal state has comparably low value for states on the top-left part of the 2D grid. Therefore, the grid follows the rule that the values on the top-left are always smaller than those on the bottom-right.

Question 5.

In this part, we implement the computation step for the value iteration algorithm and get the optimal actions. Then, we visualize the actions using the arrows in the state table. The figure is shown below:

↓	→	→	→	→	→	→	→	↓	↓
↓	→	→	→	→	→	↓	↓	↓	↓
↓	↓	↓	→	→	↓	↓	↓	↓	↓
↓	↓	↓	→	↓	↓	↓	↓	↓	↓
↓	↓	↓	→	↓	↓	↓	↓	↓	↓
↓	↓	→	→	→	→	↓	↓	↓	↓
↓	→	→	→	→	→	→	↓	↓	↓
↓	→	→	→	→	→	→	→	↓	↓
→	→	→	→	→	→	→	→	→	↓
→	→	→	→	→	→	→	→	→	→

Fig5. Optimal Action for each state

This optimal policy follows our intuition. The upper states from the table tend to take optimal actions to go down, and the left states from the table tend to go right. We can see from Fig5 that every state follows this form. The reason for this form can be described as: since the last state has the highest optimal value, all the other states tend to take actions to come nearer to the last state in order to get a higher reward. In other word, all states tend to go to their neighbor states with higher optimal values in Fig3.

Therefore, it is possible for the agent to compute the optimal action to take by observing the optimal values of its neighboring states.

Question 6.

From this question on, we will use the reward function 2 (in Question 1) as our reward function. The other parameters remain unchanged.

With the same initialization and estimation procedure in Question 2, a table is created to demonstrate the optimal value of each state. The table is shown below:

0.647	0.791	0.821	0.525	-2.386	-4.237	-1.923	1.128	1.591	2.035
0.828	1.018	1.062	-1.879	-6.755	-8.684	-6.373	-1.298	1.925	2.607
1.061	1.313	1.446	-1.635	-6.758	-13.917	-9.653	-5.515	-0.135	3.355
1.358	1.689	1.944	-1.243	-6.339	-7.983	-7.947	-9.434	-1.918	4.387
1.734	2.168	2.586	-0.736	-5.847	-3.258	-3.241	-7.434	1.715	9.16
2.211	2.778	3.413	-0.038	-5.114	-0.553	-0.488	-2.984	6.583	15.354
2.816	3.553	4.479	3.024	2.48	2.88	-0.466	-4.911	12.688	23.296
3.584	4.539	5.793	7.288	6.719	7.241	0.931	12.366	21.159	33.483
4.558	5.795	7.397	9.439	12.008	12.889	17.097	23.014	33.778	46.529
5.727	7.316	9.388	12.045	15.452	19.824	25.498	36.158	46.583	47.311

Fig6. Table representing the optimal value of each state

Question 7.

In order to show figure 6 in a more visualized way, we also generate its heat map across the 2D grid. The heat map is shown below:

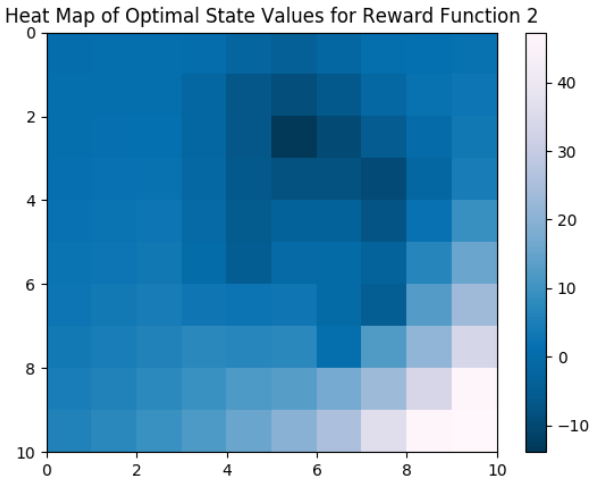

Fig7. Heat Map for Fig6

Question 8.

From Fig2, we can find that the initial reward function has a high value on the last state (number of state = 99) and several low values on some states on the right-half plane of the 2D grid.

Similar to the explanation in Question 4, the values are impacted beginning from the right-bottom part of the 2D grid (where the reward function has the much larger value than all the others). Besides, the decay also exists. So, there is a ladder modularity existing in the right-bottom part of the values grid.

Moreover, there is an unclosed circle-like negative values on the right-half of the grid. Influenced by these neighbors with negative values, the states inside the unclosed circle tend to have lower optimal values. This is the reason why minimum values occurs at near state 52 (also the relatively right half-plane in the heat map) in Fig7.

Question 9.

In this part, we again implement the computation step for the value iteration algorithm and get the optimal actions, just as what we did in question 5. Then, we visualize the actions using the arrows in the state table. The figure is shown below:

↓	↓	↓	←	←	→	→	→	→	↓
↓	↓	↓	←	←	↑	→	→	→	↓
↓	↓	↓	←	←	↓	→	→	→	↓
↓	↓	↓	←	←	↓	↓	↑	→	↓
↓	↓	↓	←	←	↓	↓	↓	→	↓
↓	↓	↓	←	←	↓	↓	←	→	↓
↓	↓	↓	↓	↓	↓	←	←	→	↓
↓	↓	↓	↓	↓	↓	←	↓	↓	↓
→	→	→	↓	↓	↓	↓	↓	↓	↓
→	→	→	→	→	→	→	→	→	→

Fig8. Optimal Action for each state

Since the final state has the largest optimal value, the overall trend is that the upper state tends to go down and the left-part state tends to go right. However, since there exist negative optimal values in the right-half plane, the trend for optimal actions is interrupted by these points as well. Within the neighborhood of these low values, the neighbors tend to flow to states with higher values, but not these low values. So, the overall mode is interrupted for these points. Therefore, this optimal policy still follows our intuition in this question.

4. Inverse Reinforcement learning (IRL)

4.1 IRL algorithm

Question 10.

We can get the equivalent LP using block matrices:

$$\begin{aligned} &\text{maximize} \quad c^T x \\ &\text{subject to} \quad Dx \preccurlyeq b, \forall a \in \mathcal{A} \backslash a_1 \end{aligned}$$

where

$$c = \begin{bmatrix} \mathbf{1}_{|\mathcal{S}|\times 1} \\ -\boldsymbol{\lambda}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \end{bmatrix}$$

$$x = \begin{bmatrix} \boldsymbol{t} \\ \boldsymbol{u} \\ \boldsymbol{R} \end{bmatrix}$$

$$D = \begin{bmatrix} \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & \mathbf{0} & \left(\boldsymbol{P}_a - \boldsymbol{P}_{a_1}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{a_1}\right)^{-1} \\ \mathbf{0} & \mathbf{0} & \left(\boldsymbol{P}_a - \boldsymbol{P}_{a_1}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{a_1}\right)^{-1} \\ \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \end{bmatrix}$$

$$b = \begin{bmatrix} \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ (\boldsymbol{R}_{\boldsymbol{max}})_{|\mathcal{S}|\times 1} \\ (\boldsymbol{R}_{\boldsymbol{max}})_{|\mathcal{S}|\times 1} \end{bmatrix}$$

4.2 Performance measure

Question 11.

In this question, we use the optimal policy of the expert in question 5 to fill $O_E(s)$ values. Then we sweep λ from 0 to 5 with 500 evenly distributed points. For each λ, we use the optimal policy of the agents (with extracted reward function 1) to fill $O_A(s)$ values. Next, we calculate the Accuracy with each value of λ and plot λ against Accuracy as below:

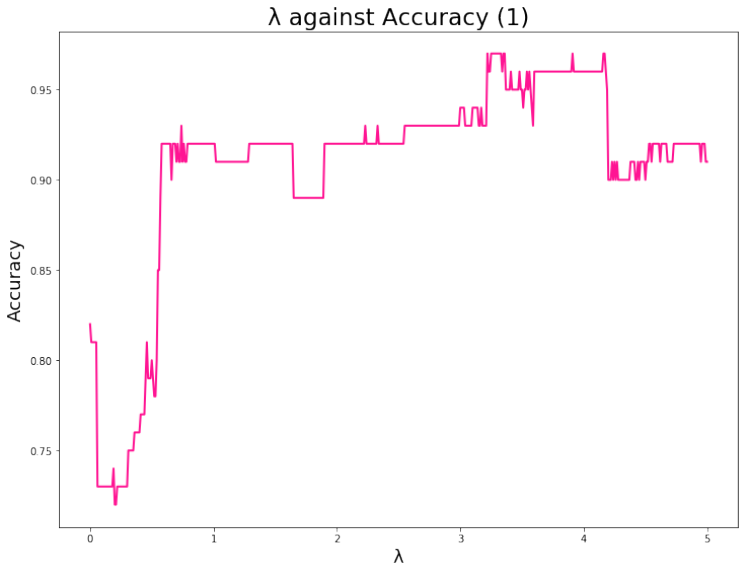

Fig9. λ against Accuracy with Extracted Reward Functions (1)

From the figure above, we find that the curve raised up to about 0.97 first and then decreased. Besides, there is a sharp decrease at $\lambda \approx 3.22$, which indicates that the maximum effect of adjustable penalty coefficient λ happens around this area.

Question 12.

From the plot in the last question, we can get the value of λ with maximum accuracy:

$$\lambda_{max} \approx 3.22$$

Question 13.

In this question, we set $\lambda = \lambda_{max} \approx 3.22$ to compute the extracted reward function 1. The heat maps of the ground truth reward and the extracted reward are as below:

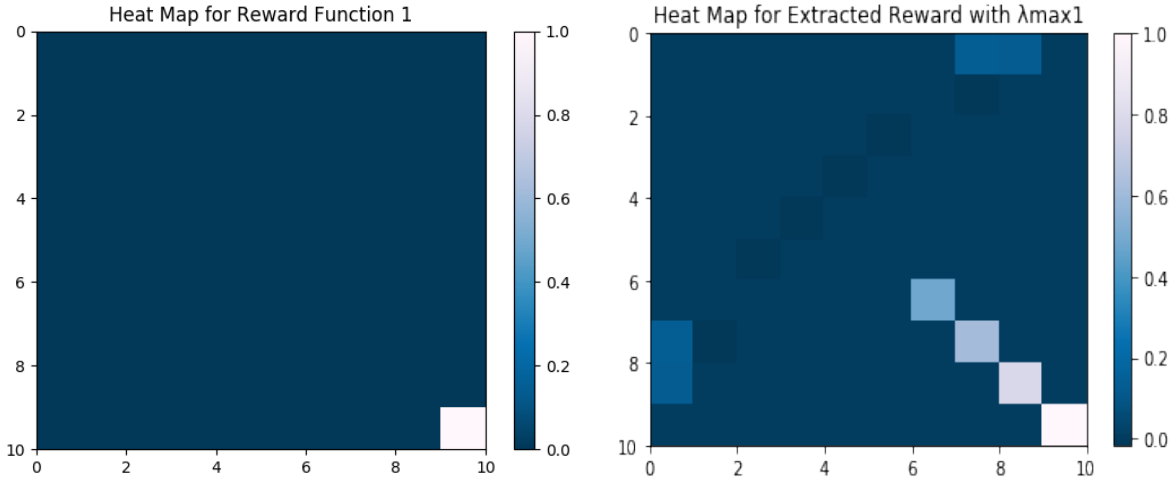

Fig10. Heat Maps of Ground Truth Reward and the Extracted Reward 1

From the figures above, we find that the extracted reward function 1 is similar to the ground truth Reward Function 1. The maximum rewards both appear at state 99. The difference is that there is a gradual change from $R_{max} = 1$ at the right-bottom corner in the heat map of the extracted reward, while for the ground truth Reward Function 1, the values change sharply from 0 to 1 at the right-bottom corner. Besides, there are five states in the center with negative values of reward, while in the ground truth one, all rewards are positive.

Question 14.

In this question, we use the extracted reward function in the last question to compute the optimal values of the states. Below is the heat map of the optimal state values:

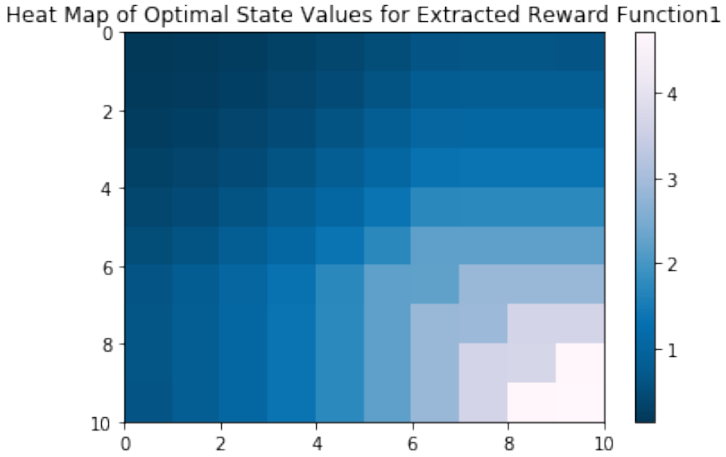

Fig11. Heat Map of Optimal State Values with Extracted Reward Function 1

Question 15.

Heat maps of optimal state values with the ground truth Reward Function 1 and the extracted reward function 1 are as below:

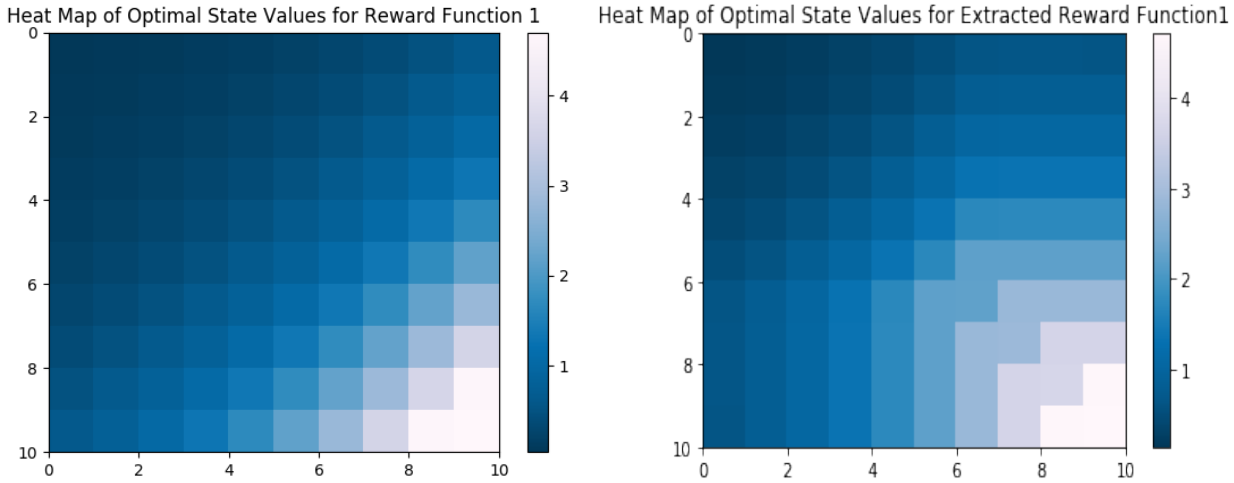

Fig12. Heat Maps of Optimal State Values with Reward Function 1 and the Extracted Reward Function 1

From the figures above, we find that the trends are same in these two heat maps, both increasing from top-left corner to right-bottom corner. In the previous question, the extracted reward function is very similar to the ground truth Reward Function 1, so the optimal state values should be same as well, both decaying from right-bottom corner to top-left corner.

However, there is one difference. In the ground truth one, the states with same value are in diagonal-direction line (from top-right to bottom-left). While in the extracted one, the states with same value are in diamond shape. It can be explained with the differences between two reward functions. There is only one state with the highest value of reward in the ground truth Reward Function 1. So this state has equal effect to other states, i.e. states 89 and 98 have the same optimal values, states 79/88/97 have the same optimal values, etc. (decay equally in diagonal line). While for the extracted reward function, there are four states with gradual decreasing rewards at the bottom-right corner (states 99/88/77/66), and they both have effects on the states around them. So these four states act like a division line and generate the diamond-shape decreasing.

Question 16.

With the extracted reward function in question 13, we compute the optimal policy of the agent as below:

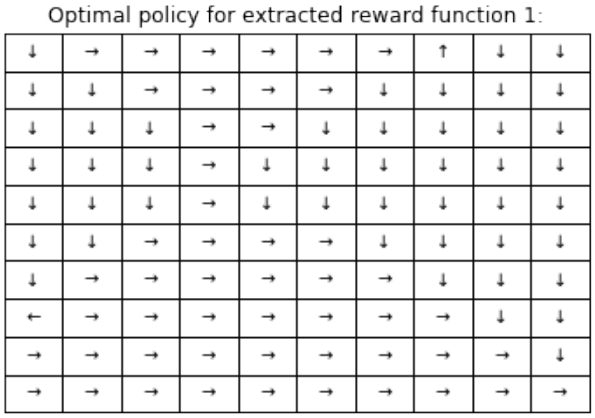

Fig13. Optimal Policy of the Agent with Extracted Reward Function 1

Question 17.

Tables of optimal policy with the ground truth Reward Function 1 and the extracted reward function 1 are as below:

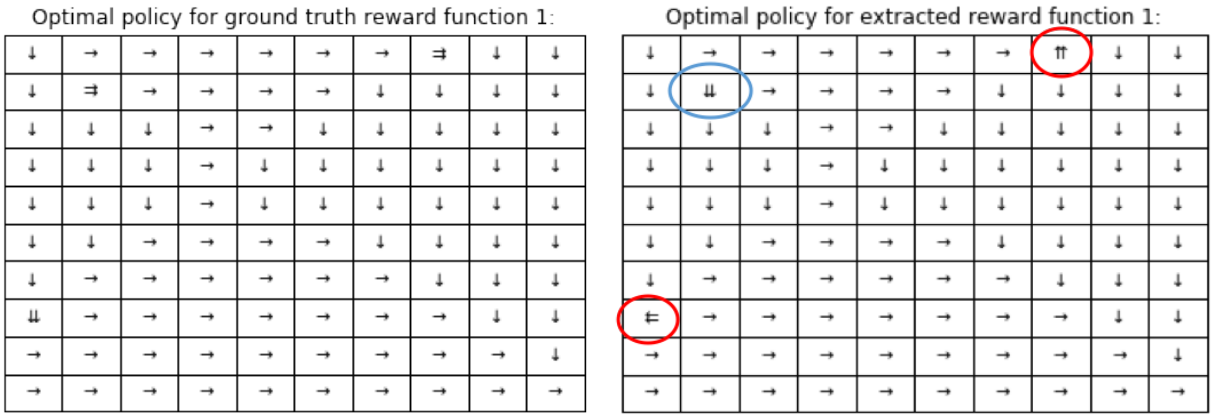

Fig14. Optimal Policy with Reward Function 1 and Extracted Reward Function 1

From the two figures above, we can find that they are almost same except 3 states (with two arrows in one cell). According to the compare in question 15, we know that the optimal state values with the extracted reward function are very close to those with the ground truth reward function. So the optimal policy should be same.

Besides, there are two kinds of differences. The two states circled in red are caused by the extracted reward function. States 7/8 and 70/80 have higher value of rewards than states around them. When values decrease to those 4 states, the values around them may be lower than the values of the states themselves. Since they are all edge states, they may choose to go out (stay in the current state) to keep the higher values. As for the state in blue circle, it may be caused by the rules of the function np.argmax(). Referring to the figure of optimal state values in question 2, we find that for one state, the values of its directions may be same, e.g. state 11 with both 0.122 for right and down directions. Thus, the differences are not true differences, but only the choice of the directions with same values.

Question 18.

In this question, we use the optimal policy of the expert in question 9 to fill $O_E(s)$ values. Then we sweep λ from 0 to 5 with 500 evenly distributed points. For each λ, we use the optimal policy of the agents (with extracted reward function 2) to fill $O_A(s)$ values. Next, we calculate the Accuracy with each value of λ and plot λ against Accuracy as below:

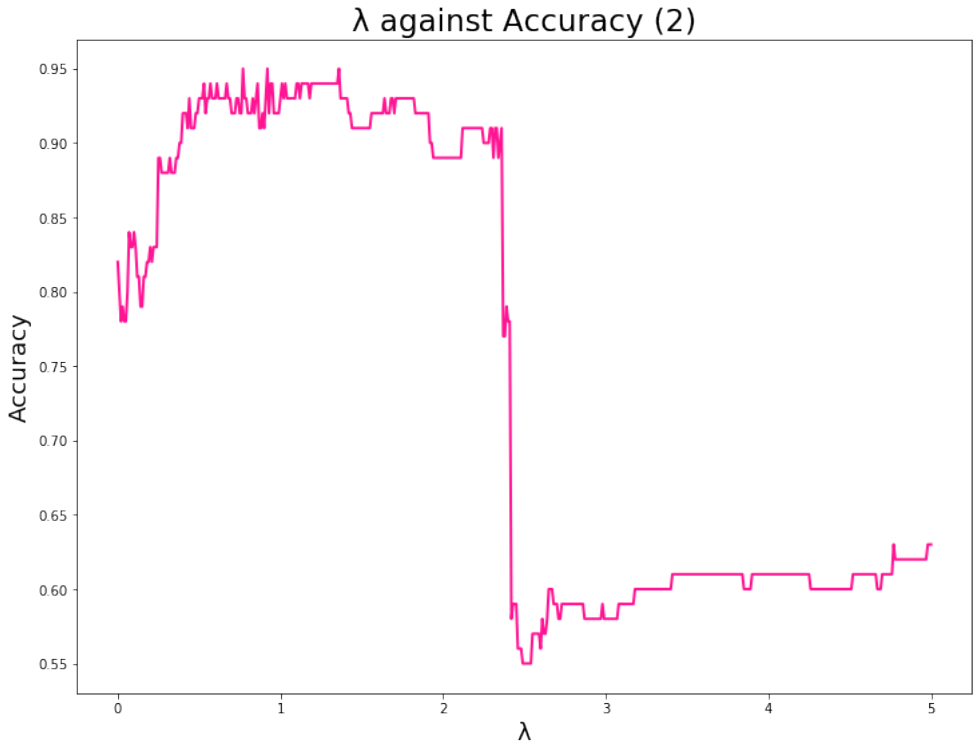

Fig15. λ against Accuracy with Extracted Reward Functions (2)

Question 19.

From the plot in the last question, we can get the value of λ with maximum accuracy:

$$\lambda_{max} \approx 0.77$$

Question 20.

In this question, we set $\lambda = \lambda_{max} \approx 0.77$ to compute the extracted reward function 2. The heat maps of the ground truth reward and the extracted reward are as below:

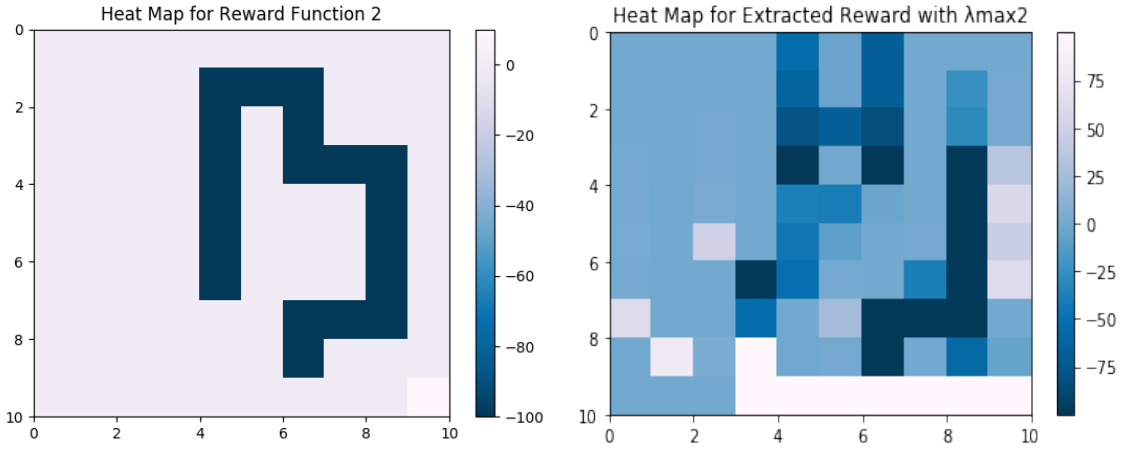

Fig16. Heat Maps of Ground Truth Reward and the Extracted Reward 2

From the figures above, we find that the extracted reward function 2 is different from the ground truth Reward Function 2. The same thing is that they both have the largest reward at state 99.

There are mainly three differences. 1) the coloring scales differ. The values of reward in the ground truth reward function are from -100 to 0, while the extracted reward function has values from -100 to 100, which fits the LP inequality $|R_i| \leq R_{max} = 100$. 2) there is a ring-like shape in the ground truth reward function with the dark blue color. While in the extracted reward function, there is a half ring-like shape in the center. 3) in the ground truth one, most values of reward are 0. While in the extracted one, the states in the right last row have much higher rewards than other states.

Question 21.

In this question, we use the extracted reward function in the last question to compute the optimal values of the states. Below is the heat map of the optimal state values:

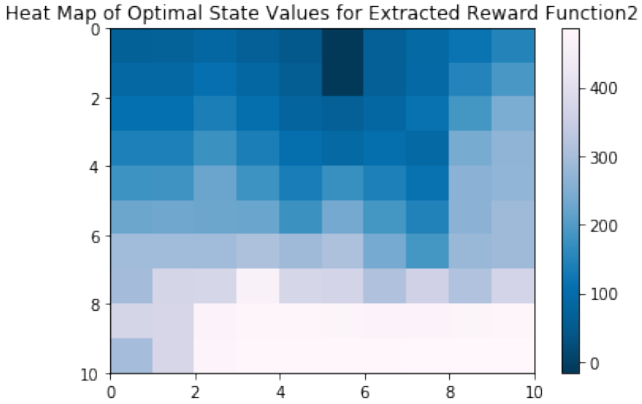

Fig17. Heat Map of Optimal State Values with Extracted Reward Function 2

Question 22.

Heat maps of optimal state values with the ground truth Reward Function 2 and the extracted reward function 1 are as below:

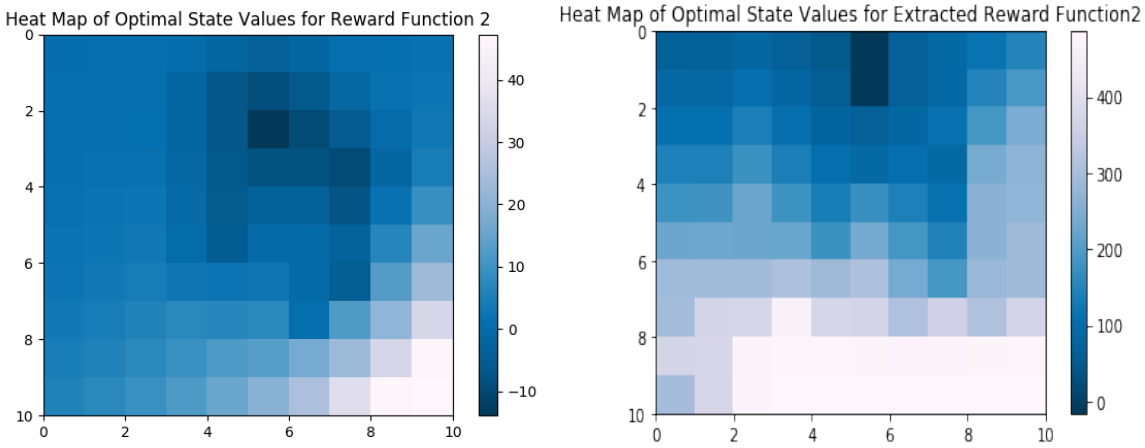

Fig18. Heat Maps of Optimal State Values with Reward Function 2 and the Extracted Reward Function 2

From the two figures above, we can get the similarities and differences as below:

1) Similarities: the optimal state values are both decrease from the right-bottom at state 99; there are ladder modularity in both two maps since the two reward functions both have a ring-like shape in the center.
2) Differences: First, the coloring scales differ greatly. In the ground truth one, the state values are from -15 to 45. While in the extracted one, the state values are from -10 to 500. It's caused by the difference of reward function with higher upper bound; Second, in the ground truth one, the values decay from bottom-right to top-left, with a ladder modularity in the upper center. While in the extracted one, it tends to decay from the bottom to top, with the smallest values at state 50 and 51. It can be also explained from the reward function. In the extracted reward map, the states in the last row have much higher rewards than other states. Thus, the decay trend is from bottom to top.

Question 23.

With the extracted reward function in question 20, we compute the optimal policy of the agent as below:

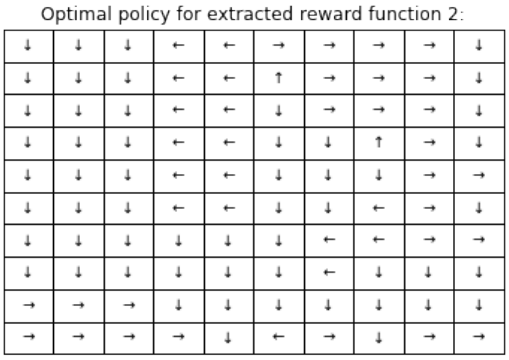

Fig19. Optimal Policy of the Agent with Extracted Reward Function 2

Question 24.

Tables of optimal policy with the ground truth Reward Function 2 and the extracted reward function 2 are as below:

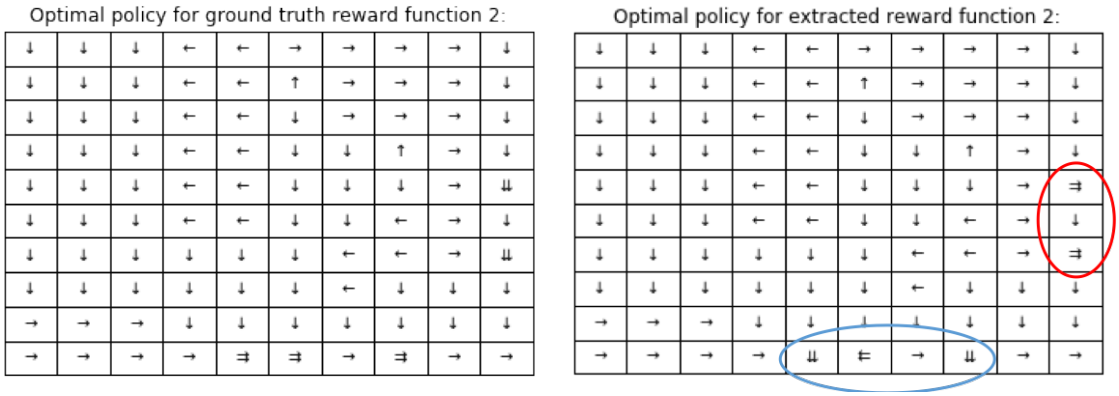

Fig20. Optimal Policy with Reward Function 2 and Extracted Reward Function 2

In the figures above, the cells with double arrows are those with different optimal policies from the ground truth one. The optimal policies of these two figures are generally same with the accuracy of 95%.

While there are mainly two major parts of differences circled in red and blue respectively. The causes are explained in the next question.

Question 25:

1. Identify and explain two discrepancies

From the figure 20, we can find two major discrepancies highlighted in red and blue circles respectively. The discrepancy in red circle may be caused by the threshold of the RL algorithm. From figures 21 to 22, we find that the reward and the optimal state values from state 93 to 96 are close. It indicates that the possibilities of choosing four directions are close, and if we can reduce the value of the threshold ϵ, the optimal policy in this area may be same.

Values of extracted reward for each state

-0.0	-0.0	0.0	0.0	0.0	0.98649	0.0	60.63883	-0.0	-0.0
-0.25592	-0.34036	-0.45463	-0.60675	-0.79396	-4e-05	-0.0	0.0	79.45452	0.0
-0.0	-0.0	0.0	0.55817	2.60525	49.58761	-0.0	-0.63435	3.55427	-0.0
0.0	-0.0	-0.0	-2e-05	-0.0	-0.0	-100.0	-52.74234	100.0	99.34432
-53.39088	-64.04725	-78.21272	-100.0	-37.86464	-43.86289	-50.33834	-2e-05	-1.93307	99.25023
-3.20155	-3.20155	-67.99856	-1e-05	-40.29398	-9.20465	0.0	23.77905	0.0	99.28426
-67.99856	-67.99856	-83.23728	-100.0	-3.76632	-0.0	0.0	-99.36215	-100.0	100.0
-0.0	-0.41186	0.0	-0.0	-0.0	0.0	-38.36437	-100.0	0.0	99.76051
0.0	-24.88328	-29.10458	-100.0	-100.0	-100.0	-100.0	-100.0	-58.48974	100.0
0.0	0.0	0.0	36.53881	58.98255	45.59437	62.3829	-0.0	-4.95665	100.0

Fig21. Values of extracted reward for each state

Optimal state values for extracted reward function 2

68.052	68.372	84.047	64.737	46.696	-16.03	64.967	89.423	116.555	148.796
86.453	86.879	107.725	82.737	60.017	-16.03	64.967	89.938	147.659	189.888
110.325	110.893	138.74	106.666	77.723	64.956	84.014	115.164	187.721	244.308
140.797	141.555	178.021	137.552	104.958	89.412	104.958	89.422	240.163	268.653
179.682	180.675	226.048	178.719	136.743	172.828	142.26	115.154	262.562	271.111
228.072	229.305	228.978	225.01	177.29	236.516	186.372	145.769	263.949	287.841
290.963	290.963	290.963	309.025	287.933	307.968	238.124	186.9	280.554	288.608
295.478	371.277	373.294	463.512	372.501	367.86	310.574	358.425	311.372	366.582
371.371	374.45	471.373	476.267	478.232	473.689	471.895	470.538	473.473	476.733
296.799	375.009	475.816	482.968	483.085	483.043	482.148	484.665	484.366	485.798

Fig22. Optimal state values for extracted reward function 2

While another discrepancy in blue circle may be caused by the discrepancy of the reward function. In the extracted reward map, the states in the last row have much higher rewards than the other states. Thus, those states tend to stay in current places to keep higher state values. This discrepancy can be fixed by modifying the formulation of the LP in question 10:

$$\begin{aligned} &\text{maximize} \quad c^T x \\ &\text{subject to} \quad Dx \preccurlyeq b, \forall a \in \mathcal{A} \backslash a_1 \end{aligned}$$

where

$$c = \begin{bmatrix} \mathbf{1}_{|\mathcal{S}|\times 1} \\ -\boldsymbol{\lambda}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \end{bmatrix}$$

$$x = \begin{bmatrix} \boldsymbol{t} \\ \boldsymbol{u} \\ \boldsymbol{R} \end{bmatrix}$$

$$D = \begin{bmatrix} \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & \mathbf{0} & \left(\boldsymbol{P}_a - \boldsymbol{P}_{a_1}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{a_1}\right)^{-1} \\ \mathbf{0} & \mathbf{0} & \left(\boldsymbol{P}_a - \boldsymbol{P}_{a_1}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{a_1}\right)^{-1} \\ \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \\ \mathbf{0} & \mathbf{0} & -\boldsymbol{I}_{|\mathcal{S}|\times|\mathcal{S}|} \end{bmatrix}$$

$$b = \begin{bmatrix} \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ \mathbf{0}_{|\mathcal{S}|\times 1} \\ (\boldsymbol{R_{max}})_{|\mathcal{S}|\times 1} \\ (-\boldsymbol{R_{min}})_{|\mathcal{S}|\times 1} \end{bmatrix}$$

$\boldsymbol{R_{max}}$ – the maximum value of the ground truth reward;

$\boldsymbol{R_{min}}$ – the minimum value of the ground truth reward.

2. Fix two discrepancies

We change threshold ϵ (only for extracted reward) to 0.00001, and then re-run the optimal policy as below:

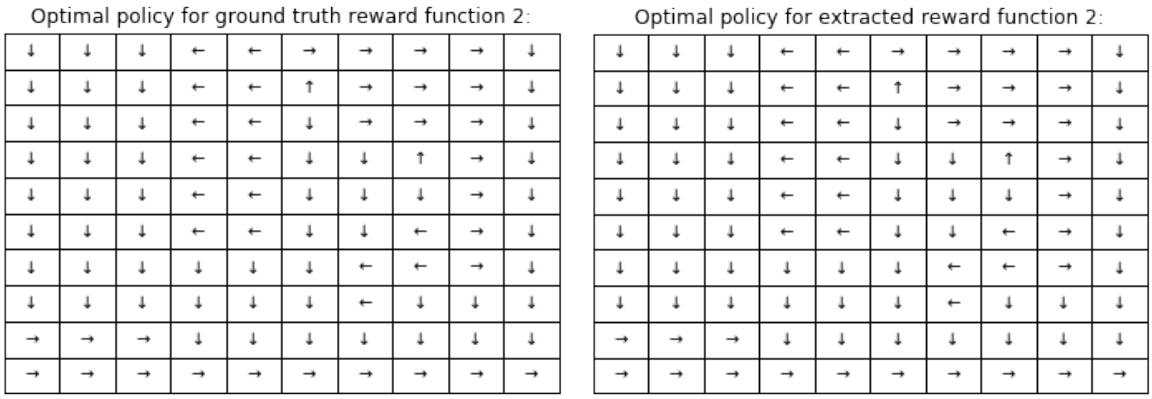

Fig23. Optimal policy with $\epsilon = 0.00001$

From the figures above, we find that the policies for each state are all same as the ground truth ones, i.e. the maximum accuracy is 100%.

Next, we change thresholds in RL algorithms for both ground truth reward and extracted reward, then recompute the maximum accuracy as below. The maximum accuracy is 100%, which indicates that our modifications are effective.

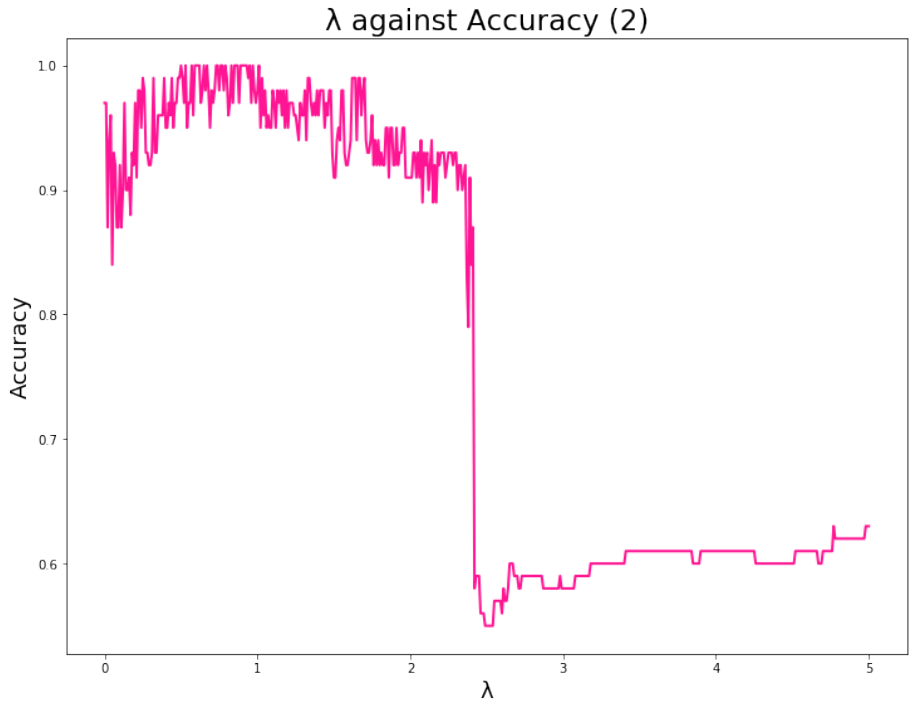

Fig24. Accuracy curve with $\epsilon = 0.00001$