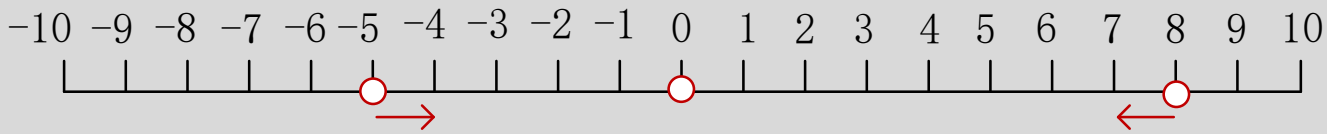
Mini Project Briefing

Shiyu Zhao

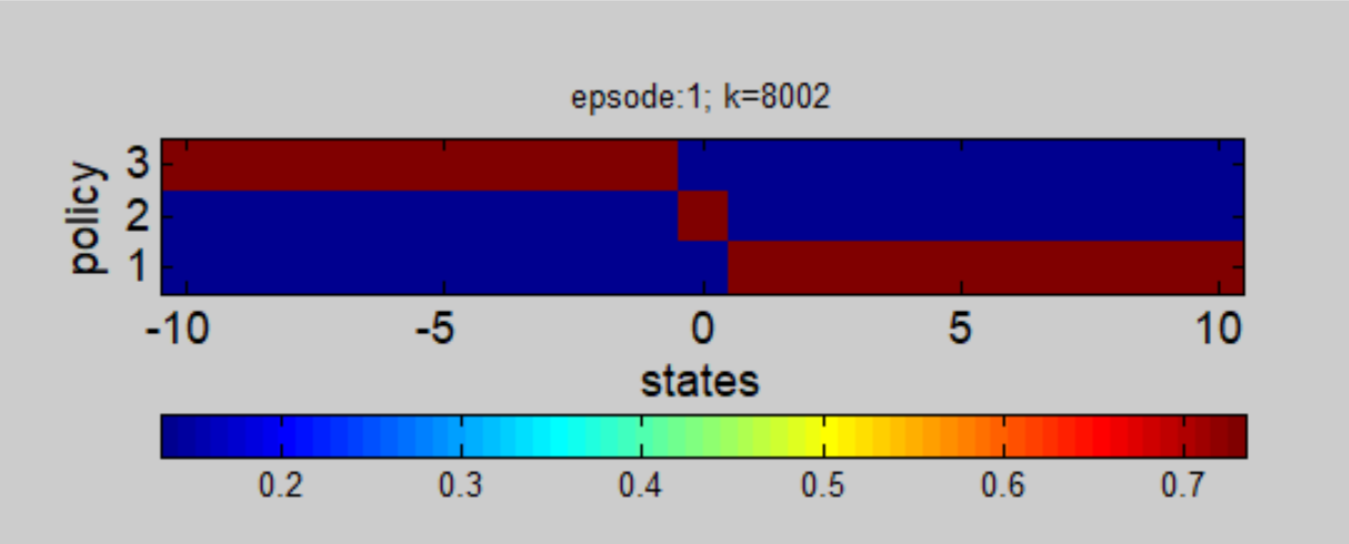
School of Engineering, Westlake University

Spring 2019



- States: $s = i$, where $i = -10, -9, \ldots, 9, 10$.
- Action: each state has 3 actions: moving leftward one unit ($a = 1$), not moving ($a = 2$), moving rightward one unit ($a = 3$).
- The aim is to find the optimal policy $\pi^*(s, a)$ such that the agent could reach $s = 0$ starting from an arbitrary initial state.
- Environment and reward (you can design your own reward):
  - If the agent attempts to move out of the interval $[-10, 10]$, the reward is $r = -10$. In particular, if $s = 10$ and $a = 3$, then the consequent state is $s' = 10$. If $s = -10$ and $a = 1$, then the consequent state is $s' = -10$.
  - Entering the target area gets a reward of 100. Otherwise, for every step get a reward of -1.

The optimal policy is



Mathematical expression:

$$s_{k+1} = s_k - \text{sign}(s_k)$$

Use the following methods:

- Model-based: implement at least one of the following two algorithms
  - Value iteration
  - Policy or generalized policy iteration
- Model-free: implement at least one of the following two algorithms
  - Monte Carlo exploring starts
  - Monte Carlo ε-greedy

Requirements

Report:

- Please submit one single PDF version of your report. The report must elaborate your work to show your understanding of the algorithms. The report may consist of mathematical and graphical contents, along with concise descriptions.
- You can use Word, Latex, or other tools to write the report. Font size: 12. Margin: at least 2 cm. Line spacing: 1.5. The report should be in English. There is not page limitation. However, an overlong report should be definitely avoided.

Source code:

- The code should be straightforward to run. A brief description of the usage of your code should be given in a text file. The code should be clearly structured and annotated.
- You may choose what ever programming language you like. However, you should not use existing implementations of the above algorithms. Instead, you must develop your own algorithms. Preliminary operations such as matrix-vector and probability operations are allowed.

Submission:

- Please submit two files: the first is your report, and the second is a zip file of your source code. The name of your report should be "*RL_student number*_report". The name of the zip file of your source code should be "*RL_student number*_source code".
- Please do NOT include your name anywhere in your submission. But make sure that you include your student number correctly!
- Way to submit: please send the submissions by email to the teaching assistant (Jianan Li, lijianan@westlake.edu.cn). If your submission does not meet the above requirements, it will be returned to you for revision.
- Time to submit: 23:59 June 7, 2020.
- Late submission: penalty of 10% grade per day

Marking criteria

Marking criteria:

- 40% marks for model-based algorithms: you can implement either value iteration or policy (or generalized) policy iteration.
- 40% marks for model-free algorithms: you can implement either the Monte Carlo exploring starts or Monte Carlo ε-greedy.
- 20% marks for report and code: The report must be clearly rewritten and organized.
  Results, description, and discussion should be sufficient to show your understanding of the topics.

Plagiarism

- Every student must work independently to complete the entire project. Discussion on certain technical problems among the students is allowed. However, note that this is not a group project. You could NOT work as a group to do the project.
- Plagiarism is prohibited. Once it is detected in the report or source code, it will be investigated carefully and the corresponding students will be punished accordingly.