Imperial College
London

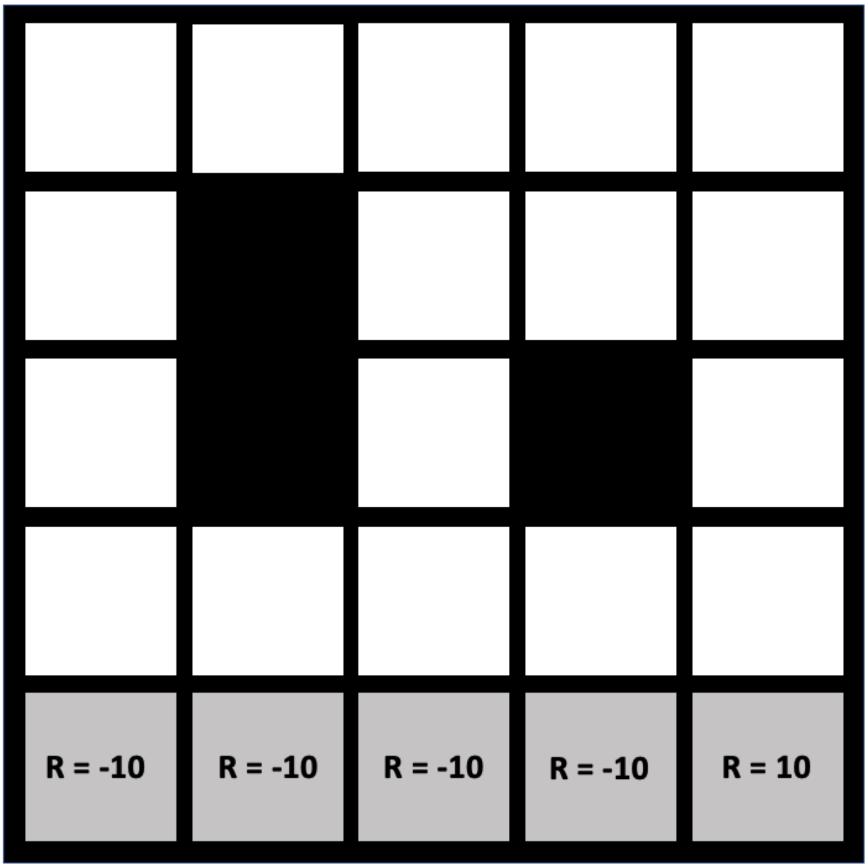
October 20, 2022

Reinforcement Learning – Prof Aldo Faisal & Dr Paul Bilokon
Assignment design: Prof Aldo Faisal & Manon Flageat

Lab Assignment 2: Policy evaluation, policy iteration and value iteration

Grid World

In this lab assignment, we are going to focus on the GridWord environment illustrated below. On this figure, each white square represents a state with associated reward 0, and each grey square an absorbing state with associated reward indicated on the figure. A black square stands for an obstacle.



An agent is moving on this grid, trying to get the best possible reward. To do so, it can choose at any time-step between four actions:

- a_0 = going north of its current state
- a_1 = going east of its current state
- a_2 = going south of its current state
- $a_2$3 = going west of its current state

The chosen action has a probability p to succeed and lead to the expected direction. If it fails it has equal probability to lead to any other direction. For example, if the agent chooses the action a_0 =

going north, it is going to succeed and go north with probability p, and it has probability $\frac{1-p}{3}$ to go east, probability $\frac{1-p}{3}$ to go south and probability $\frac{1-p}{3}$ to go west. p is given as a hyperparameter, defining the environment.

If the outcome of action leads the agent to a wall or an obstacle, its effect is to keep the agent at his current place. For example, if the agent chooses the action a_1 = going east and there is a wall east but no wall in the other directions, it is going to stay in place in his current state with probability p and to go north, south or west with probability $\frac{1-p}{3}$.

Jupyter Notebook

This lab assignment is based on the provided Jupyter Notebook (you can also find it directly on Google Colab: `https://colab.research.google.com/drive/1kSS1Se2cSbqMM2yP6VtkJDrEYjzOzQBX?usp=sharing`), which already defined most of the GridWord structure. In the following questions, you will be asked to progressively complete the functions of the GridWorld class indicated with the tag ”[Action required]”.

The Notebook has the following structure:

1. `GraphicsGridWorld` class definition: allow all the graphical visualisation of the GridWorld. You DO NOT NEED to read or understand this class, you can simply call its methods when needed.
2. Utility function definition: some functions used across all code, you DO NOT NEED to read or understand them.
3. `GridWorld` class definition: this class is the main GridWorld class, we encourage you to try to understand it, as you will be asked to complete it in the following questions.
4. Questions sections: each question of this lab assignment corresponds to one code block that will allow you to test your code and visualise your results.

Question 1: Grid World definition

Using the Grid World presentation above, build the environment by implementing the methods:

- `fill_in_transition`
- `fill_in_reward`

Question 2: Policy evaluation implementation

Fill in the `policy_evaluation` method of the `GridWorld` class to perform policy evaluation of a given policy in the Grid World environment. Use the example code provided for Question 2 to test your code and visualise your value function.

Question 3: Impact of gamma on the policy evaluation

Use the example code provided for Question 3 to visualise the impact of the discount factor `gamma` on the policy evaluation algorithm. You can investigate similarly the impact of the `threshold` value, or the initialisation of the policy in the policy evaluation, and the structure of the grid (obstacle localisation, reward values, etc).

Question 4: Policy iteration implementation

Fill in the `policy_iteration` method of the `GridWorld` class to perform policy iteration and find the optimal policy in the Grid World environment. Use the example code provided for Question 4 to test your code and visualise your value function.

Question 5: Impact of gamma on the policy iteration

Use the example code provided for Question 5 to visualise the impact of the discount factor `gamma` on the **policy iteration** algorithm. Try to investigate similarly the impact of the other parameters.

Question 6: Value iteration implementation

Fill in the `value_iteration` method of the `GridWorld` class to perform value iteration and find the optimal policy in the Grid World environment. Use the example code provided for Question 6 to test your code and visualise your value function.

Question 7: Impact of gamma on the value iteration

Use the example code provided for Question 7 to visualise the impact of the discount factor `gamma` on the **value iteration** algorithm. Try to investigate similarly the impact of the other parameters.