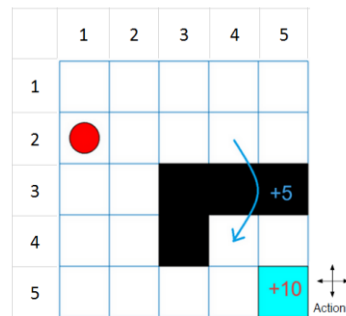


COM 762: Deep Learning and Its Application

The objective of this project is to provide students with practical experience of developing a Q-learning agent for solving a grid world environment by reinforcement learning and Python. The grid world environment is illustrated below:



which has the following configuration and rules:

1. the grid world is 5-by-5 and bounded by borders, with four possible actions (North = 1, South = 2, East = 3, West = 4).
2. the agent begins from cell [2,1] (second row, first column)
3. the agent receives a reward +10 if it reaches the terminal state at cell [5,5] (blue).
4. the environment contains a special jump from cell [2,4] to cell [4,4] with a reward of +5 (a special treatment may be necessary).
5. The agent is blocked by obstacles (black cells).
6. All other actions result in -1 reward.
7. The iteration function below will be used to calculate state values.

$$V(S_t) \leftarrow V(S_t) + \alpha [V(S_{t+1}) - V(S_t)]$$

The specific tasks are below:

- (a) detail the task and process of exploration and exploitation for solving grid world problem
- (b) Describe the formulation of policy, environment, observation (state or grid cell or position) and action, etc. in the context of the grid world problem.
- (c) create a value table (or Q-table) using the state and action specifications from the grid world environment, set the learning rate to be 1 and various values in between [0, 1] to run the code.

- (d) create a Q-learning agent using this table representation and configure the epsilon-greedy exploration.
- (e) train the agent with the following options: train for 100 episodes, where each episode should have the ability to control unlimited paths from starting position to terminal or jump-end position (for instance, stop training when the agent receives an average cumulative reward greater than 10 over 30 consecutive episodes.)
- (f) Visualize the state values in each of the grid cells with the board layout.

Happy Coding!