Reinforce Learning Homework

2022130247 나요한

1. Code Implementation

Explain flow of code working. The Maze class sets the environment and hyperparameters.

텍스트이(가) 표시된 사진

자동 생성된 설명

Run an episode repeated time through the run\_episode() function.

텍스트이(가) 표시된 사진

자동 생성된 설명

For every episode, go through get\_eps\_greedy\_action() -> get\_step\_reward() -> update\_Q().

텍스트이(가) 표시된 사진

자동 생성된 설명

Each function works in the following order.

텍스트이(가) 표시된 사진

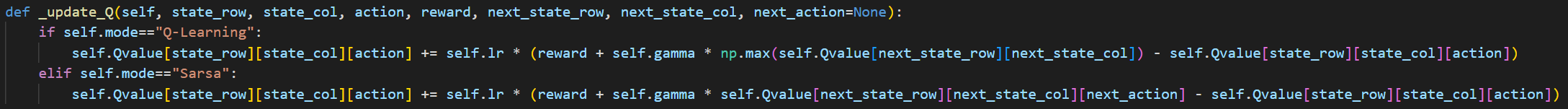
자동 생성된 설명

Depending on the value of epsilon, Action moves to Random or moves to the highest path in QTable.

텍스트이(가) 표시된 사진

자동 생성된 설명

According to the given action, the environment provides a reward for the state that moves in the corresponding direction. You will receive a reward for the next state.



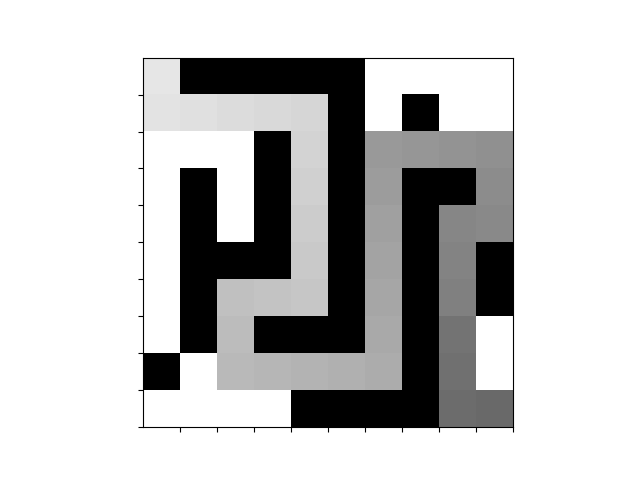
Using the given reward, update QTable. The updated Table influences the next action, so that the QTable that finally understands the environment can be obtained, and as a result, it is possible to move toward the target point with the final reward.

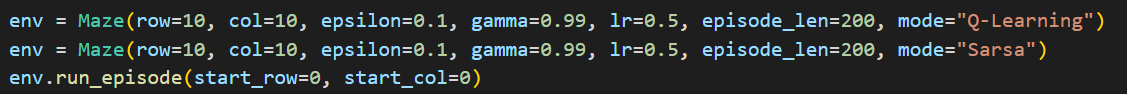
텍스트이(가) 표시된 사진

자동 생성된 설명

It is a function that shows the maze\_frame as follows, and gradually increases the color depth to express the movement path.

2. Result





The following results were obtained through the following hyperparameter settings.

2. Analysis

Through each learning, the Q Table is updated, and finally the shortest path is found.

In the case of SARSA, there is a problem that the path diverges instead of decreasing.

When an action is randomly selected in the next state through the epsilon-greedy function, even though moving from the current state to the next state can receive a good reward, it is judged as a bad policy, and there is a problem in that learning is not possible.

Q-learning is an off-policy TD control method to improve SARSA. For the behavior, a function with exploration using epsilon was added, and for the target policy, a Q function-based greedy action was selected so that the target could be learned through behavior.

If SARSA requires S\_t, A\_t, R\_t+1, S\_t+1, and A\_t+1, Q-learning does not require A\_t+1 and selects a greedy action, solving the problem of learning divergence.