Prof. Joschka Boedecker, Gabriel Kalweit, Maria Huegle, Andreas Saelinger

Reinforcement Learning
Exercise 9

# 1 Offline $\lambda$-Return Algorithm

Consider the grid-world depicted in Figure 1. In each episode, the agent starts in a random cell of the grid-world and is allowed to move from its present position to one of the four adjacent cells (reliable, deterministic transitions) in each time step. The black cells mark blocked cells which cannot be entered by the agent. For each transition, the agent gets a reward of $-1$. Assume no discount.

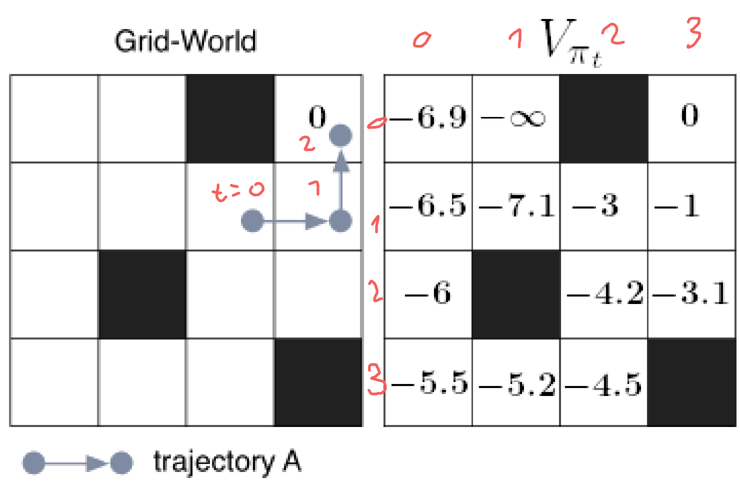


Figure 1: Grid.

(a) On the basis of $V_{\pi_t}$ determine the TD-errors for the two steps of trajectory A (as depicted in the left part of the figure).

(b) Based on the TD-errors and the initial value function $V_t = V_{\pi_t}$, calculate $V_{t+1}$ using the Offline $\lambda$-Return Algorithm with $\lambda = 1$. Use a learning rate of $\alpha = \frac{1}{2}$.

# 2 TD($\lambda$)

Implement the TD($\lambda$)-algorithm (with neural networks as function approximator) as defined in the lecture in `td_lambda.py`. You find implementation details in the script. Evaluate the optimal policy (defined in `get_action(s)`) on the gridworld environment from Exercise 3. You again find an implementation in `gridworld.py`.

## 3 Experiences

Make a post in thread *Week 09: Eligibility Traces* in the forum[1], where you provide a brief summary of your experience with this exercise and the corresponding lecture.

[1] `https://ilias.uni-freiburg.de/goto.php?target=frm_1837317&client_id=unifreiburg`