Reinforcement Learning
Exercise 3

Mathias Niepert, Vinh Tong

April 26, 2024

1 Proofs (5P)

a) Show that the Bellman **optimality** operator $\mathcal{T}$ is a γ-contraction. This is similar to but not the same as the Bellman **expectation backup** operator from lecture 3 slide 20. Be able to explain all the steps! (3P)

$$(\mathcal{T}v)(s) = \max_a \sum_{s',r} p(s', r|s, a)[r + \gamma v(s')] \tag{1}$$

b) Asuming a general finite MDP (S, A, R, p, γ) where rewards are bounded: $r \in [r_{\min}, r_{\max}]$ for all $r \in R$. Prove the following equations. (2P)

$$\frac{r_{\min}}{1-\gamma} \leq v(s) \leq \frac{r_{\max}}{1-\gamma} \tag{2}$$

$$|v(s) - v(s')| \leq \frac{r_{\max} - r_{\min}}{1-\gamma} \tag{3}$$

2 Value Iteration (5P)

As in the previous exercise sheet, we will use the FrozenLake environment from gym (`https://www.gymlibrary.dev/environments/toy_text/frozen_lake/`). The code template can be found on Ilias in *ex03-dynp/ex03-dynp.py*. It has been tested with gym version 0.18.0 (but should also be stable with version 0.18.3).

a) Implement the value iteration algorithm (see lecture 3 slide 28) in the function *value_iteration*. Use the values for γ and θ in the code. Initialize the value function $V(s)$ to 0 for all states.

How many steps does it need to converge? (1P)

What is the optimal value function? (2P)

b) Compute the optimal policy from the value function. (2P)