Prof. Joschka Boedecker, Gabriel Kalweit, Maria Huegle

Reinforcement Learning
Exercise 8

1 Q-learning with Function Approximation

(a) Implement Q-learning with function approximation in `q_learning_fa.py`. Apply it on the modified Gym Mountain Car environment which you find in `mountain_car.py`. It includes a shaped reward and a deterministic initial state. Run the control loop. Create learning curves of your experiments. You can use the plotting function which is given in the script.

(b) Add experience replay and a target network. You do not update on the transition you collect in a time step, but sample a minibatch and train on that. The targets are calculated using the target network which is a slowly updated copy of the real Q-network. You can get the parameters of a network `net` via `net.parameters()`. Play around with the soft and hard update. Again, create learning curves and write a short comparison about your experiences with Q-learning – with and without experience replay and target networks.

(c) Implement Double-Q-learning. The **action for target calculation** is then based on the current actual Q-network and not the target network. The **value** estimation however is **still based on the target network**.

2 Experiences

Make a post in thread *Week 08: Off-policy Methods with Function Approximation* 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_1380029&client_id=unifreiburg`