3. Exercise Sheet

Exercise 1 Monte Carlo Method - Understanding

Think about the following questions and give possible solutions to the problems stated:

- In Monte Carlo method, if we start with a deterministic π, some/many $(s,\ a)$-pairs will never be visited! How can we make sure that (almost) all pairs are visited?
- The influence of updates shrinks with an increasing number of episodes when using the Monte Carlo Method. Why does this happen? What can we do to resolve this problem?
- The Monte Carlo method needs an end of an episode to calculate the update using the return of this episode. What do we do in never-ending games? How can we solve such problems?
- During the lecture we discussed that the Monte Carlo Method updates its value estimation for each state. We usually discuss state spaces with a finite number of states. What can we do if the state-space is infinite?

Exercise 2 Monte Carlo Method - Application

You want to estimate the time needed to get home. Therefore, you start to record the times needed for each part of your travel. After 3 days you created the following list.

Travel checkpoint	Wednesday	Thursday	Friday
Start at office	6:00	5:30	1:00
Reached the car	6:10	5:35	1:03
Leaving the unversity complex	6:15	5:45	1:10
Getting on the highway	6:20	5:50	1:15
Leaving the highway	6:45	6:20	2:30
Arrive at home	6:50	6:30	2:35

a) Use the Monte Carlo Method to update the value estimate for each episode. Use the first method as initial values for $V(s)$.

b) Use the Constant-α Monte Carlo Method to update the value estimate for each episode. Compare the differences of setting α to $\alpha = 0.1$ and $\alpha = 0.5$.

(Hint: use the traveled minutes as reward and the remaining time till you arrive at home as your return function.)

Exercise 3 Temporal Difference Learning - Application

a) Apply Temporal Difference Learning ($TD(0)$) with $\alpha = 0.5$ for the example in Task 2 of this exercise sheet.

b) Explain the differences of the error calculation in Temporal Difference Learning and the Monte Carlo Method.

Exercise 4 Q-Learning

Consider again the Tic-Tac-Toe game (3x3) and an opponent making random moves except to prevent the completion of a row.

a) State the problem of learning an optimal Tic-Tac-Toe strategy as a Q-Learning problem. What are states, actions, state transitions (function), and rewards in this non-deterministic Markov decision process?

b) Give an example for a state together with possible actions and state transitions in this state!

c) Can the learner determine an optimal strategy, if it plays against an opponent always making the optimal move?

The following task can be solved in pairs of two. Please make sure that your solution includes the name of both group members as a comment at the top of the file

Exercise 5 Programming Exercise - Reinforcement Learning

OpenAI Gym is a toolkit for developing and comparing reinforcement learning algorithms. The open-source library gives you access to a standardized set of environments (learning tasks). Your task will be to implement the learning procedure of Monte Carlo Method and Temporal Difference Learning for the rather simple CartPole-v1 environment (see: `https://gym.openai.com/envs/CartPole-v1/`). This environment is rather simple and learning algorithms may converge fast. Feel free to explore other environments after you are done!

Joining the Google Colab project

- Introduction to Google Colab:
  `https://colab.research.google.com/notebooks/welcome.ipynb`
- Join the Google Colab project:
  `https://colab.research.google.com/drive/1rmjdGfWsC1w-fOkICvknaGSsOUQ8Gp49`

Accessing all files on Github

- Link to files on GitHub; Requirements: Python 3.5 or higher, packages: gym, numpy
  `https://github.com/ADockhorn/GymCartPoleCIG`

CartPole-v1 basics

A pole is attached by an un-actuated joint to a cart, which moves along a frictionless track. The pendulum starts upright, and the goal is to prevent it from falling over by increasing and reducing the cart's velocity.

- **Rewards:** The reward is 1 for every step taken, including the termination step.
- **Starting State:** All observations are assigned a uniform random value in $[-0.05, 0.05]$
- **Termination:** pole angle $> 12°$, cart position > 2.4, episode length > 200

In this example you have two actions available

Num	Action
0	Push cart to the left
1	Push cart to the right

The observation is an array containing 4 values.

index	observation	Min	Max
0	cart position	-4.8	4.8
1	cart position differential	$-\infty$	∞
2	pole angle	-24	24
3	pole velocity at tip	$-\infty$	∞

Task

a) Discretize the values of the observations space and store a four-dimensional matrix $V(s)$.

b) Update the matrix using either the Monte Carlo Method or the Constant$-\alpha$ MC Method.

c) Implement Temporal Difference Learning to update the value estimates in $V(s)$ over time.