1 Reinforcement Learning Project

The final project of the lecture will be on developing a Reinforcement learning agent for a small game. You can form teams of 2 or 3 people. For teams of 3 there are special requirements detailed below. If you want to pursue a different task, let us know and we decide whether this is okay. You will implement different reinforcement learning algorithms and evaluate their performance on a simple task first and then apply them to a simulated laser-hockey game. We are using Open AI gym, see https://gym.openai.com/ with a custom environment. Instructions on installing gym are at https://github.com/openai/gym

The code for the laser-hockey game for this project is at the git repository https://github.com/martius-lab/laser-hockey-env We will need to change the code to bugs or change parameters, so please stay tuned. We will make releases, such that you know when things changed.

All teams will compete against each other in the game at the end. The tournament mode for different teams is not yet implemented, but we will provide it. There are intermediate checkpoints, see below, that we will discuss in the remaining recitation sessions.

For the final evaluation, you have to prepare:

(a) A report with:
  
  - description of the chosen methods and your potential modifications,
  - architecture description and implementation details
  - experimental evaluation for all the methods and environments, and a
  - discussion.

  Each team member should have one algorithm implemented and his/her independent contribution should be clearly marked in the report.

(b) A presentation ∼8 min (for 2-person teams and 10min for 3-person teams)

(c) The source code
Everything needs to be finished/presented at the “exam” day: 08.02.2019 The tournament will happen before, exact details will follow.

Requirements:

(a) Teams of two should implement 2 algorithms
(b) Teams of three should implement 3 algorithms
(c) In order to pass the exam report, presentation and code have to be handed in on time.
(d) The code has to run and the simple pendulum environment must be solved.
(e) The mark will be determined based on all parts. You are expected to deliver a nicely written report, a clear presentation, and a good performance.

1.1 Checkpoint 1: Get your algorithms up and running

Start with the Pendulum-v0 from the gym suite. Implement your algorithms of choice. I recommend to consider: Deep Q-learning (DQN) [3], Deep deterministic policy gradient (DDPG) [2], Soft/Natural Actor Critic (SAC [1]/NAC [4]), proximal policy gradient (PPO) [5], Policy Gradient by Parameter Exploration (PGPE) [6].

The version of DQN that you implemented for your last exercise would be a good starting point. Keep in mind that the full DQN also has a target Q network. Also, the Q-network outputs a value for each action at once. For algorithms using discrete actions, transform the continuous actions into a few discrete actions. The task is solved if you get on average a reward above -300. Make appropriate analysis and track your performance etc. Don’t forget this procedure during the rest of the project. Remember that you want to create a report with plots giving detail about the training, comparisons etc.

1.2 Checkpoint 2: Laser-hokey – learning to handle the puck

Start working on the laser-hockey game (https://github.com/martius-lab/laser-hockey-env). The repository provides the environment and a little notebook to see how the environment works.

In order to learn how to play laser-hokey there is a small training camp for the RL-agents to go through. These are:

**TRAIN_SHOOTING** hitting a static ball into the goal (other agent is static)

**TRAIN_DEFENCE** defending your goal against incoming shots (other agent is static)

**NORMAL** normal gameplay against another agent.

You can enable the game-modes with LaserHockerEnv(mode=NORMAL|TRAIN_SHOOTING|TRAIN_DEFENCE).
1.3 Checkpoint 3: Self-play

Let your agents play against each other in normal game mode. Make appropriate analysis and track your performance etc. Experiment with different tournament modes.

1.4 Final

We will organize a tournament to let your agents play against each other. Details will follow.

References


