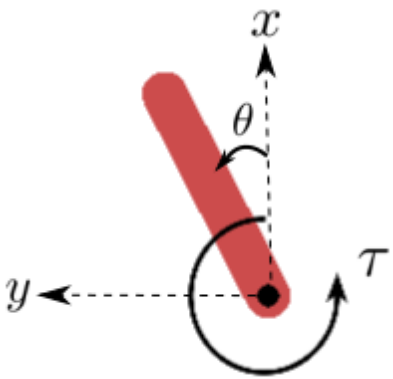
1. TASK DESCRIPTION

In this project you will implement a reinforcement learning (RL) algorithm to control and swing-up an inverted pendulum.



In the Pendulum environment, a *Pendulum* must be controlled to swing up from an angle of π to 0. Your task is to implement an off-policy RL algorithm that, by practicing on a simulator, learns a control policy for the Pendulum.

The `task4_supplementary.pdf` in the handout contains the following additional sections; Environment and Scoring Details will give a more thorough description of the Pendulum environment, along with the requirements for passing the baseline. Solution Details will guide you through a set of minimal modifications to `solution.py` for passing the task baseline. In the *Hints* section, references to relevant off-policy algorithms which could be used for solving the task are provided. We advise you to take a look at this section, but also to refresh your knowledge about this type of RL algorithms from the lectures.

SUBMISSION WORKFLOW

1. Install and start Docker (https://www.docker.com/get-started). Understanding how Docker works and how to use it is beyond the scope of the project. Nevertheless, if you are interested, you could read about Docker's use cases (https://www.docker.com/use-cases).
2. Download handout (/static/task4_handout.zip).
3. The handout contains the solution template `solution.py` in which you should implement your solution and additional python file `utils.py`, in which additional methods and functionalities are given. The template code `solution.py` is meant to guide you through the implementation and get you started on the task. `# TODO: enter your code here` markers in the solution template point out the spots where you should introduce your changes. More details about specific classes and methods in the template code are provided in the supplementary file.The `solution.py` will also generate a `.mp4` file giving a video of one of your evaluation episodes, which does not need to be submitted and serves as visualization of your method.
4. You should use Python 3.8.5 for your local development. File `requirements.txt` could be used to set up your local conda environment. You are free to use any other libraries that are not already imported in the solution template. Important: please make sure that you list these additional libraries together with their versions in the `requirements.txt`.
5. Once you have implemented your solution, run the checker in Docker:
    - if you are using Linux or MacOS (without Apple silicon), run `bash runner.sh`. On some operating systems, you might need to run `sudo bash runner.sh` if you see a Docker permission denied error. Docker might by default restrict how much memory your solution may use. Running over the memory limit will result in docker writing "Killed" to the terminal. If you encounter out-of-memory issues you can increase the limits as described in the Docker Desktop for Mac user manual (https://docs.docker.com/desktop/mac/). Running over the memory limit will result in docker writing "Killed" to the terminal.
    - if you are using Windows, open a PowerShell, change the directory to the handout folder and run `docker build --tag task4 .; docker run --rm -u $(id -u):$(id -g) -v "$(pwd):/results" task4`
    - If you are having trouble running your solution using docker locally, consider using the ETH Euler cluster to run your solution. Please follow the guide specified by *euler-guide.md* in the handout. The setup time of using the cluster means that this option is only worth doing if you really can't run your solution locally.
6. There is a known issue where the Docker container gets killed during training or evaluation of the model due to running out of memory. If this happens, you need to increase the RAM available to your Docker instance. We successfully ran everything with 8 GB of RAM.
7. If the checker fails, it will display an appropriate error message. If the check was successful, then a file called `results_check.byte` will be generated. You should upload this file together with your source code to the project server.
8. We limit submissions to the server to 40 per team, with at most 20 in a 24 hour period.

GRADING

This task has a public and private score. You only receive feedback about your performance on the public part in the form of the public score, while the private leaderboard remains secret. The purpose of this division is to prevent overfitting to the public score. We refer you to the `task4_supplementary.pdf` in the handout for more details on the grading. When handing in the task, you need to select which of your submissions will get graded and provide a short

description of your approach. This has to be done **individually by each member** of the team. We will then compare your selected submission to our baseline. Your final grade for this task is the mean of the grades of your public and private scores. We will publish 2 baselines: easy and hard that each has a public and private score. Beating the easy baseline's public/private score guarantees you a 4.0 public/private grade respectively, while beating the hard baseline's public/private score guarantees you a 6.0 public/private grade respectively. Failing to beat the easy baseline's public/private score grants you a 2.0 public/private grade. For scores better than the easy baseline but worse than the hard baseline, we will grant a grade between 4.0 to 6.0 depending on how close you are to the hard vs easy baseline. We emphasise that this will likely not be a simple linear interpolation between the hard and easy baselines. In addition, for the pass/fail decision, we consider the code and the description of your solution that you submitted. That is, your code should be runnable and reproduce your predictions (see faq) and you should include a short writeup of your submitted solution. We emphasize that the public score leaderboard is just for fun: the scores of other teams will not effect the baseline or your own grade.

⚠ Make sure that you properly hand in the task, otherwise you may obtain zero points for this task.

FREQUENTLY ASKED QUESTIONS

⊙ WHICH PROGRAMMING LANGUAGE AM I SUPPOSED TO USE? WHAT TOOLS AM I ALLOWED TO USE?

You are free to choose any programming language and use any software library. However, **we strongly encourage you to use Python**. You can use publicly available code that was not produced directly for the purposes of this course, but you should specify the source as a comment in your code.

⊙ AM I ALLOWED TO USE MODELS THAT WERE NOT TAUGHT IN THE CLASS?

Yes. Nevertheless, the baselines were designed to be solvable based on the material mentioned in the project description or taught in the class up to the second week of each task.

⊙ IN WHAT FORMAT SHOULD I SUBMIT THE CODE?

You can submit it as a single file (main.py, etc.; you can compress multiple files into a .zip) having max. size of 1 MB. If you submit a zip, please make sure to name your main file as *main.py* (possibly with other extension corresponding to your chosen programming language).

⊙ WILL YOU CHECK / RUN MY CODE?

We will check your code and compare it with other submissions. We also reserve the right to run your code. Please make sure that your code is runnable and your predictions are reproducible (fix the random seeds, etc.). Provide a readme if necessary (e.g., for installing additional libraries).

⊙ SHOULD I INCLUDE THE DATA IN THE SUBMISSION?

No. You can assume the data will be available under the path that you specify in your code.

⊙ CAN YOU HELP ME SOLVE THE TASK? CAN YOU GIVE ME A HINT?

As the tasks are a graded part of the class, **we cannot help you solve them**. However, feel free to ask general questions about the course material during or after the exercise sessions.

⊙ CAN YOU GIVE ME A DEADLINE EXTENSION?

⚠ We do not grant any deadline extensions!

⊙ CAN I POST ON MOODLE AS SOON AS HAVE A QUESTION?

This is highly discouraged. Remember that collaboration with other teams is prohibited. Instead,

- Read the details of the task thoroughly.
- Review the frequently asked questions.
- If there is another team that solved the task, spend more time thinking.
- Discuss it with your team-mates.

⊙ WHEN WILL I RECEIVE THE PRIVATE SCORES? AND THE PROJECT GRADES?

We will publish the private scores, and corresponding grades before the exam at the latest.