TASK 4

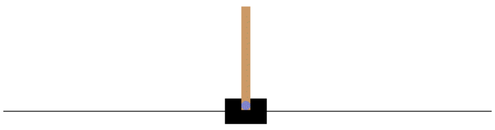
Implementing an Off-policy RL algorithm.

You are in the group Binarybandits consisting of ahiropedi (ahiropedi@student.ethz.ch), mwylie (mwylie@student.ethz.ch) and sverreault (sverreault@student.ethz.ch).

1. READ THE TASK DESCRIPTION | 2. SUBMIT SOLUTIONS | 3. HAND IN FINAL SOLUTION

1. TASK DESCRIPTION

In this project you will implement a reinforcement learning (RL) algorithm to balance a pole by controlling the cart it is installed on.



In the Cartpole environment, a pendulum is linked to a cartpole and initialized in a vertical, inverted position. The cart moves on a frictionless rail, and can be controlled to keep the pendulum upright. Your task is to implement an off-policy RL algorithm that, by practicing on a simulator, learns a control policy for the Cartpole.

The `task4_supplementary.pdf` in the handout contains the following additional sections; Environment and Scoring Details will give a more thorough description of the Cartpole 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. 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.
2. Download handout .
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. After setting $save\_video$ to $True$, 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. 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 is a pass — fail project and you will not receive a grade for this task. You need to pass 3 / 4 projects in the PAI course to be eligible for taking the exam. Your algorithm will make predictions on a held out test set, or (for tasks 3 and 4) obtain a single score for its interactions with the environment.

When handing in the task, you need to select which of your submissions will get graded and provide a public link to a maximum one minute long video explaining your approach. Make sure that we can access the link till the first of February of the following year. Every student has to submit an individual 1 minute long video. This has to be done **individually by each member** of the team. For generating the video link, please use the ETH polybox service, upload your video there and generate a shareable link (see this documentation for more detail). Submissions that are not handed-in, do not have a video that we can access till the first of February of the following year or have a video exceeding the one minute threshold will not be graded.

We will compare your selected submission to our public baseline, which is visible on the public leaderboard. You pass this project if your submitted solution beats our public baseline. At the end of the semester, we will award one team with a certificate and prize for their performance on this task. The selection criteria are based on the team's performance on our public and private leaderboards and the creativity of their solution. The prize will be disclosed at the end of the semester. The private leaderboards are based on a separate test score on an undisclosed test set or (for task 3 and 4) environment. 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. Your model should generalize well to the private part of the test set. The creativity of your solution will be evaluated by our TAs based on your video submission.

> ⚠ Make sure that you properly hand in the task, otherwise you will fail this task.

We refer you to the `task4_supplementary.pdf` in the handout for more details on the grading.

PLAGIARISM

The use of **open-source** libraries is allowed and encouraged, except code that could reasonably be considered a solution to this or previous years' PAI projects. We do not allow copying the work of other groups / students outside the group (including work produced by students in previous versions of this course). **Publishing project solutions online is not allowed and use of solutions from previous years in any capacity is considered plagiarism**. Among the code, including those of previous years, we search for similar solutions in order to detect plagiarism. If we find strong evidence for plagiarism, we reserve the right to let the respective students or the entire group fail in the PAI 2024 course and take further disciplinary actions. Although not strictly forbidden, we discourage the use of Github Copilot or similar code/language generation tools for writing code. **We expect that if such tools are used, the tools used are stated in the video submission explaining the solution (see above). While it will have no effect on your grade or if a solution passes or fails, it may affect the awarding of prizes for best solutions**. We discourage these tools because we feel that the best way to understand the material is to write the code yourself referring to just the lecture material, source papers and documentation of any libraries used. The projects are designed in a way that you should be able to complete them in a reasonable amount of time using this approach. For the purposes of disclosing what generative AI tools you used to write code, we don't need you to disclose using e.g. basic code autocompletion such as those used in the default setup of Sublime Text 3. By submitting the solution, you agree to abide by the plagiarism guidelines of PAI2024.

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 (pass/fail) 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.