
Machine Learning Applications
for Particle Accelerators
HOME
CONFERENCES
CONTACT US

Exercises

Exercise 1

Build a Gaussian Process to model

$$f(x) = \sin(x).$$

For this first exercise, build the GP with only `numpy` and `scipy` (i.e. do not use `scikit-learn`, `BoTorch`,...).

Choose a prior with any kernel and $\mu(x) = 0$, define a method to compute the posterior and plot the posterior and the true function with 5 test points.

Tipp:

e.g. RBF kernel

covariance matrix element $K_{ij} = k_{RBF}(x_i, x_j)$,...build entire matrix.

Exercise 1a

Build your own Bayesian Optimisation algo.

Use the GP and data from Exercise 1 and optimise it between $[0,2\pi]$.

- Define the Acquisition Function UCB method for a single data point
  - e.g. `def acq_ucb(x):…`

Iteratively find the optimum of $f(x)$ by optimising the Acquisition Function at each iteration.

Exercise 2

Solve Exercise 1 and 1a with BoTorch.

Exercise 3

Optimise the RMS of the AWAKE e^- in the horizontal plane with Bayesian Optimisation.

Start with a random corrector settings (method `initialise_trajectory_random()`)

Work with bounds $\pm 300 \mu\mathbf{rad}$

- **Electron beam trajectory steering problem for CERN's AWAKE**
  Advanced Proton Driven Plasma Wakefield Acceleration Experiment

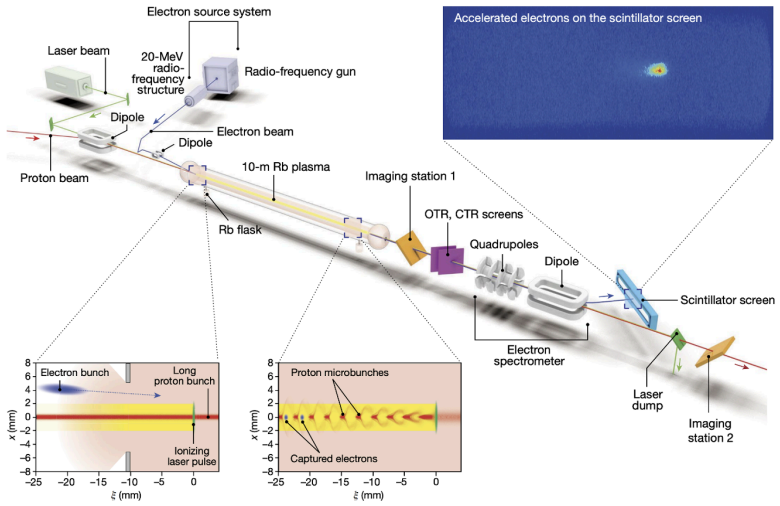
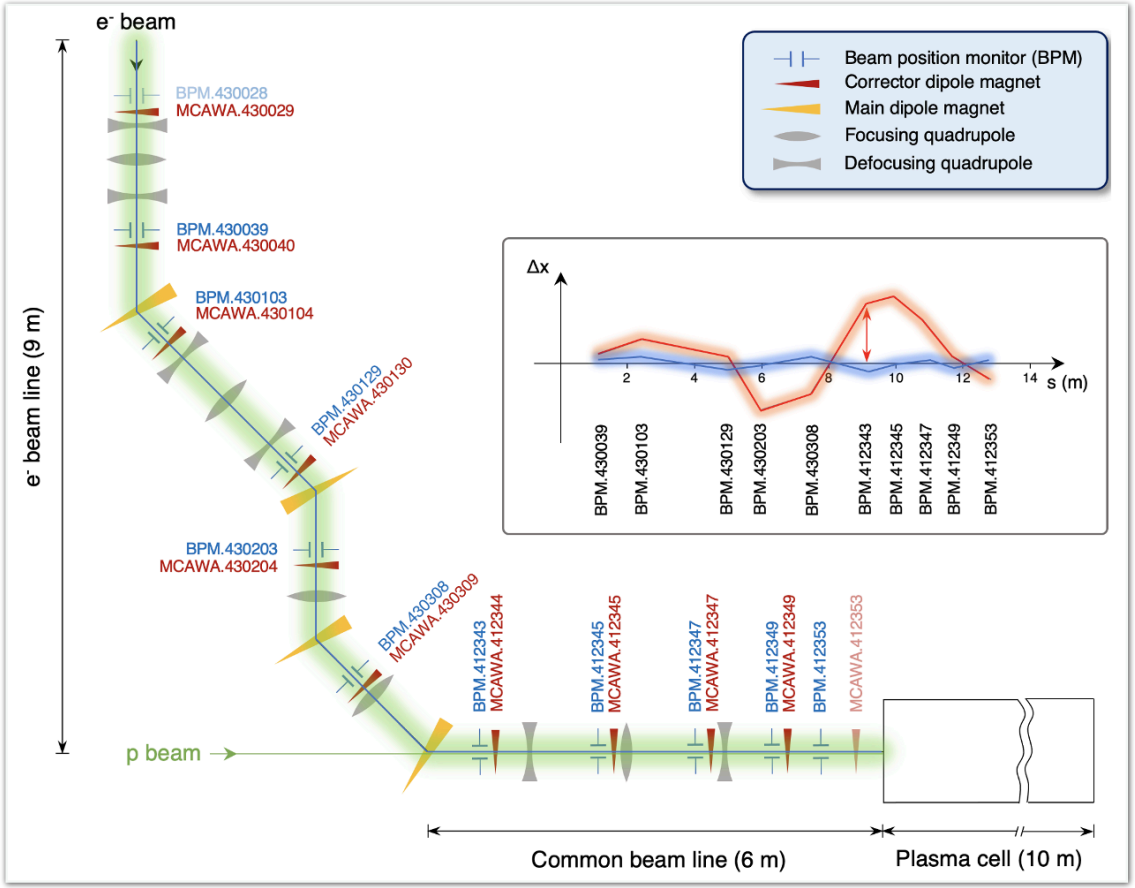


image by AWAKE Collaboration

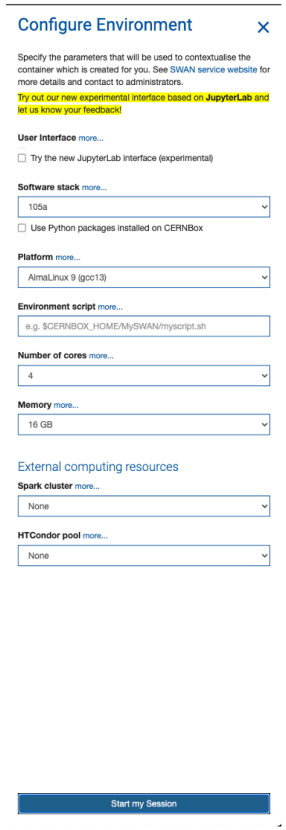
Exercise 3 - instructions (1)



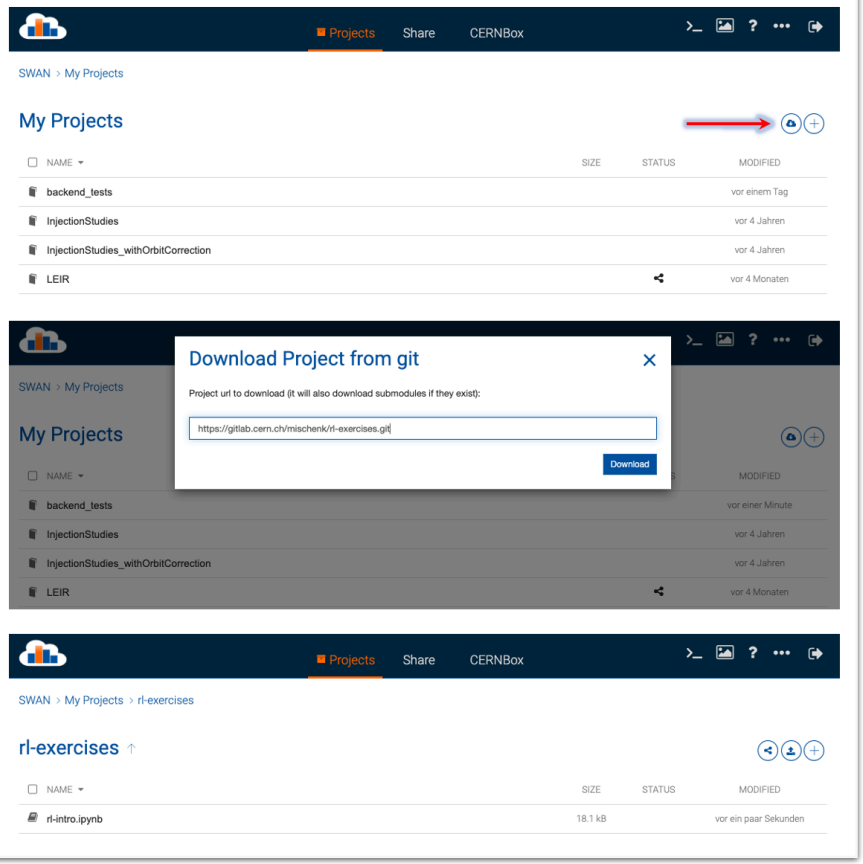
Exercise 3 - instructions (2)

Install the following package and work with swan; **choose awake-machine-demo.ipynb**

1 *Go to: swan.cern.ch and pick configuration*



2 *Clone from: https://gitlab.cern.ch/mischenk/rl-exercises.git*



Exercise 4

The goal of this exercise is to understand Q-learning.

Choose the project rl-intro.ipynb

Exercise 5

Implement model-free RL (based on SB3) with a Gymnasium environment to optimise the RMS of the AWAKE e^- trajectory in the horizontal plane.

Tipps:

Each episode should start with a new random trajectory and the agent needs to correct below RMS threshold (e.g. 1.5 mm).

The units of the example are: actors in `radian`, positions in `m`

Truncate the episodes: e.g. maximum number of iterations = 50