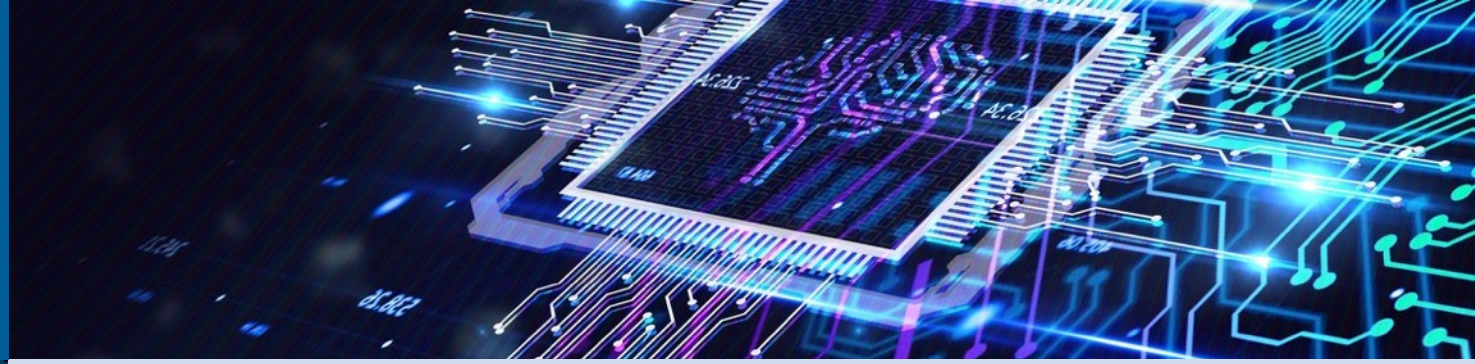
CSC

ICT Solutions for Brilliant Minds




# Machine learning on supercomputers, Part 2: Workflows for using Jupyter Notebooks and GPUs on Puhti

Mats Sjöberg, CSC

# “How can you use GPUs with Jupyter Notebooks on Puhti?”

Unfortunately the answer is a bit complicated…

- GPUs are an expensive resource - bought by taxpayers - and it’s CSC’s responsibility to use them efficiently
- Interactive use is very inefficient, as most of the time you’re editing or looking at the code and not running any computation
- During that time the GPU is still reserved and cannot be used by anyone else

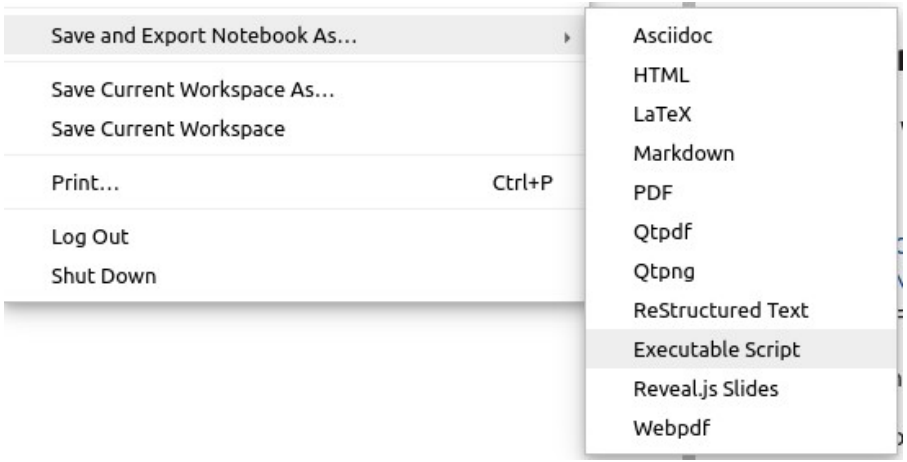
# “How can you use GPUs with Jupyter Notebooks on Puhti?”

Still, it’s possible to select the **gpu** partition when launching a Jupyter session, with some caveats:

- Only 1 GPU can be used
- You might have queue for a long time…

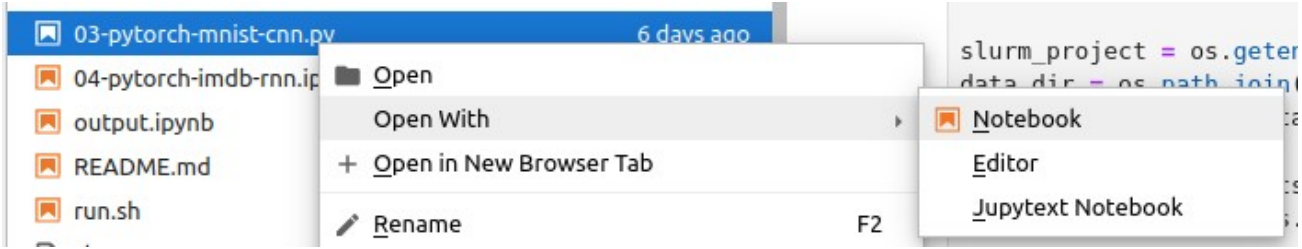
# Alternative workflow 1

- Work interactively using the **interactive** partition (just CPUs)
  - Less queueing
  - With enough CPU cores you can easily debug and test things like data loading and even small-scale training
  - Tip: with interactive you can also test "Local disk" (NVMe) for data-intensive workloads
- When you're ready for the real run:
  - Save notebook as a Python script
  - Run as usual with Slurm
- Cumbersome if you need to go back and forth between script and Notebook a lot



# Alternative workflow 2

- Use the **interactive** partition (as with workflow 1)
- Work interactively directly with the Python script:
  - Open the script “as a Notebook”
  - You can run cells interactively, but it’s still a Python script
  - … which can be submitted with Slurm
- No going back and forth between Notebook and script!



# Alternative workflow 3

- Use the **interactive** partition (as previously)
- Work interactively with the real Jupyter Notebook
- Run the notebook directly with Slurm using Papermill: https://papermill.readthedocs.io/

```
python3 -mpapermill your-notebook.ipynb output.ipynb
```