


Amherst College High-Performance Computing System

HPC refers to high-performance computing and will allow us to process our data and run our computations at high speeds.

jupyter.hpc.amherst.edu

- to open Jupyter notebook
- to access shell

In order to create a HPC Python kernel:

- Navigate to the  symbol on the left panel (the second one).
- Use the existing *Python 3.9 via SLURM* template.
- Name it something simple, like Seldonian1.
 - Kernel name, prefix of the job name, and display name should be the same (for easier organization).
- Add the '*amh-python-extra*' module (the first one that pops up when you search).
 - Feel free to add additional modules here, but any packages imported in your notebook should be available to the kernel as well.
- Modify the number of tasks to run, which refers to the number of nodes to allocate for parallel computing.
 - We've been using 16 for now.
 - Make sure this value is compatible with the code you're running.
- Leave the list of the generic consumable resources blank.
- Modify the home directory of the kernel to the folder where the Jupyter notebook is hosted.
 - For us, it's */home/dasienga24/StatisticsThesis*.
- Select '*cpu-q*' as the partition for resource allocation.
- Check this box: The job allocation can over-subscribe resources with other running jobs.
- Complete creation of the cluster.

In order to use the HPC kernel:

- Navigate to the top right of the notebook (right to the left of the circle icon)
- Click and select the HPC kernel created using the steps above

Useful Shell Commands

*we'll run these commands in the terminal provided by the interface.

module spider ray

- search for a specific module in your python environment

pip install ray***pip install -U ipywidgets***

- to install python modules/ packages/ libraries

python

- enter into a python kernel
- can be a way to check if a module has been installed/ if a line of code is now able to run with all dependencies

exit()

- to exit python

squeue-hpc --me

- to check overall job status

ssh node011

- to access a specific node securely

exit

- to close connection to the previous node i.e. go back to the parent node

ps ux

- for more details on job status (give more information about all processes)

ps u

- for some details on job status

top

- real-time view of running processes on the kernel

q

- quit display

ls

- to see what files are in the current directory/ folder

ls -a

- to see all files (any files starting with a '.' are "hidden" files)

cat slurm-112541.out

- to see specific file on screen ("concatenate")

cd

- change directory // get back to the parent directory

pwd

- print working directory

ls .local

ls .local/share

ls .local/share/jupyter

ls .local/share/jupyter/kernels

- view files located along a file path

.local/share/jupyter/kernels

- check whether this is a valid file path

cd .local/share/jupyter/kernels

- change directory to specific file path
- *note that this is the directory where we can access our HPC kernel

nano Seldonian1 (didn't have the desired effect)

- control-x to exit

cd Seldonian1

- navigate to a specific directory

ls

- list files in current directory

nano kernel.json

- use nano to edit the kernel files
- control-x to exit

history

- to see all commands you've used in shell

exit

- to close out the shell terminal

***shell commands do not work well with spaces between files! Get into a habit of naming files without any spaces!**