High Performance Computing Cheat Sheet
Pablo Barberá

Access

Using PuTTY in Windows or Terminal.app in Mac. First, ssh into the bastion node:

```
ssh NETID@hpc.nyu.edu
```

Then log on to the login node of the cluster you want to use:

```
ssh NETID@bowery.es.its.nyu.edu
```
or
```
ssh NETID@usq.es.its.nyu.edu
```

Basic UNIX commands

`cd` *dirname*	Change directory	`cat` *file*	Print file in console
`mkdir` *dirname*	Create new directory	`head` *file*	Print first lines of file
`cp` *oldfile newfile*	Copy a file	`tail` *file*	Print last lines of file
`mv` *oldfile newfile*	Move a file	`wc -l` *file*	Count lines in file
`ls -lh`	List your files (with sizes)		

Managing Data

Three storage systems:

`/home/NETID/` - Program development space
`/scratch/NETID/` - Computational work space
`/archive/NETID/` - Long-term storage

Important:

`/scratch/` is not backed up and all inactive files older than 30 days are removed.
`/home/` has only 5GB of space allocated

Type `myquota` to check your space usage.

Moving files to/from the cluster.

If your files are on dropbox, easiest way is to save them in your public folder and download them to `scratch` using `wget`:

```
wget https://dl.dropboxusercontent.com/u/XXXX/filename.data  --no-check-certificate
```

Of course, you can also download them from anywhere else:

```
wget http://jackman.stanford.edu/ideal/currentSenate/rc.rda
```

To retrieve files, you need to use the `scp` command with some additional configuration. For more details, see the tutorials on the following URL:

`https://wikis.nyu.edu/display/NYUHPC/SCP+through+SSH+Tunneling`

Available Software

The Bowery cluster has the most updated software. As of December 2013: `R 3.0.1`, `Stata 13`, `JAGS`, `gcc` (C++ compiler, required for stan), `python`.

To check the list of available software, type `module avail -t`.

To load software, type `module load` *program*. For example: `module load stata/13`

Running Jobs
Submitting and running jobs involves the following three steps:

1) Setup
The first step is to create and submit a PBS script, which includes the commands to be run in the HPC as well as the characteristics of the job (number of nodes and processors, length of time, name of job, etc.). A typical PBS script looks like this:

```
#!/bin/bash

#PBS -l nodes=1:ppn=1,mem=10GB,walltime=01:00:00
#PBS -N IRTexample
#PBS -M pba220@nyu.edu
#PBS -m abe
#PBS -e localhost:/scratch/pba220/${PBS_JOBNAME}.e${PBS_JOBID}
#PBS -o localhost:/scratch/pba220/${PBS_JOBNAME}.o${PBS_JOBID}

source /etc/profile.d/env-modules.sh
module load r/intel/3.0.1
cd /scratch/pba220/
R CMD BATCH irt_example.R irt_example.log
cp irt_results.Rdata ~/

exit 0;
```

A PBS script is just a text file that you can create in your home folder. The easiest way of preparing the file is to type `vi` in the console; then `i` to edit; and paste it from a regular plain text editor. To close `VIM`, type `:wq`, which will save the file and return to the console.
If your script requires more than one core, you can change the `ppn` option (processors per node).

2) Launch
To submit the job, type: `qsub job.pbs`. This will assign your job to the queue. Depending on the resources available, it might not start immediately. You will receive an email confirming that the job has started, and once it has finished (or has been terminated).
If you want to cancel the job after it has started, use the command `qdel JOBID`.

3) Monitor
To check whether your job is running on the cluster, type one of the following commands:

```
showq | grep NEITD
showq -u pba220
```

You can also monitor the progress of your job by printing the files `JOBNAME.e7.crunch.local` (for error messages) and `JOBNAME.o7.crunch.local` (for output messages).
If you're using R with `R CMD BATCH`, you can also see how much of your code has run by looking at the log file.

More information: `http://wikis.nyu.edu/display/NYUHPC/`