

Lecture 7: Learning Slurm and NLOpt

Su, Danyang

Goal:

- Introduce parallel computing
- Set up Slurm
- Learn how to bootstrap

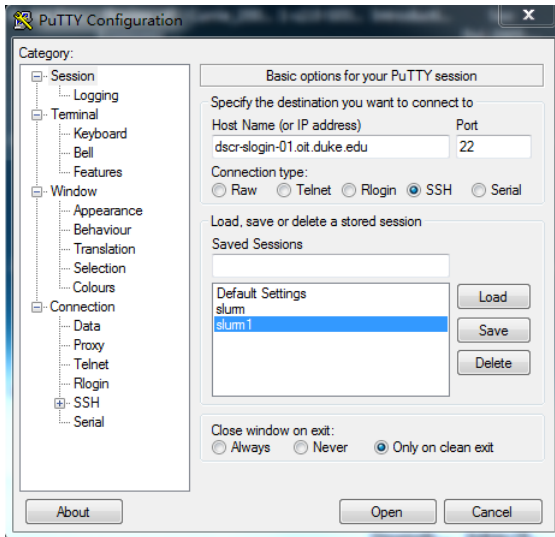
- Matlab functions:
 - parfor: not really relevant
 - GPU computing: a bit complicated (but might make a difference)
 - Refer to gpu computing page.
- Too expensive

- Duke university-wise clustering service.
- IMHO: better than econ department service
- Reference:
 - <https://wiki.duke.edu/display/SCSC/SLURM+Queueing+System>
 - <https://wiki.duke.edu/display/SCSC/Training>
 - Linux: <http://swc-osg-workshop.github.io/2015-10-27-duke/contents.html>

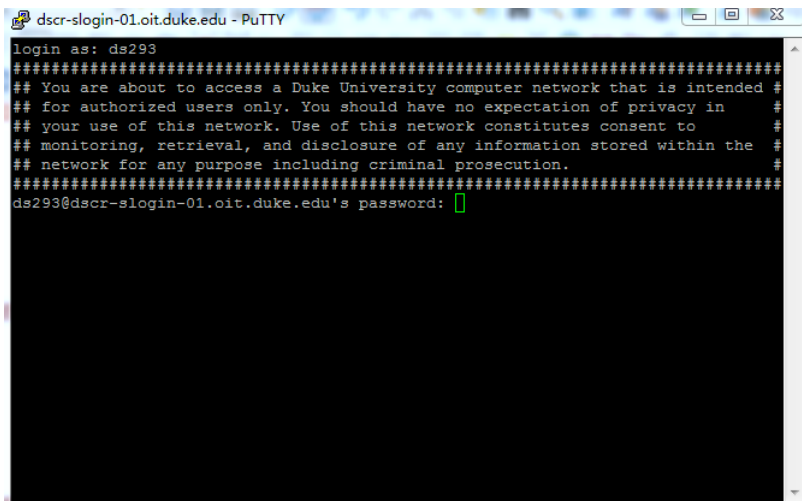
Prerequisite:

- Account
- PuTTY
- FileZilla
- GUI software: Cygwin64 Terminal
- Duke VPN if off campus

Setup PuTTY:



PuTTY Interface:



The screenshot shows a PuTTY terminal window titled "dscr-slogin-01.oit.duke.edu - PuTTY". The terminal output is as follows:

```
login as: ds293
#####
## You are about to access a Duke University computer network that is intended #
## for authorized users only. You should have no expectation of privacy in   #
## your use of this network. Use of this network constitutes consent to      #
## monitoring, retrieval, and disclosure of any information stored within the  #
## network for any purpose including criminal prosecution.                    #
#####
ds293@dscr-slogin-01.oit.duke.edu's password: [ ]
```

The password prompt is followed by a green cursor character.

PuTTY Interface:

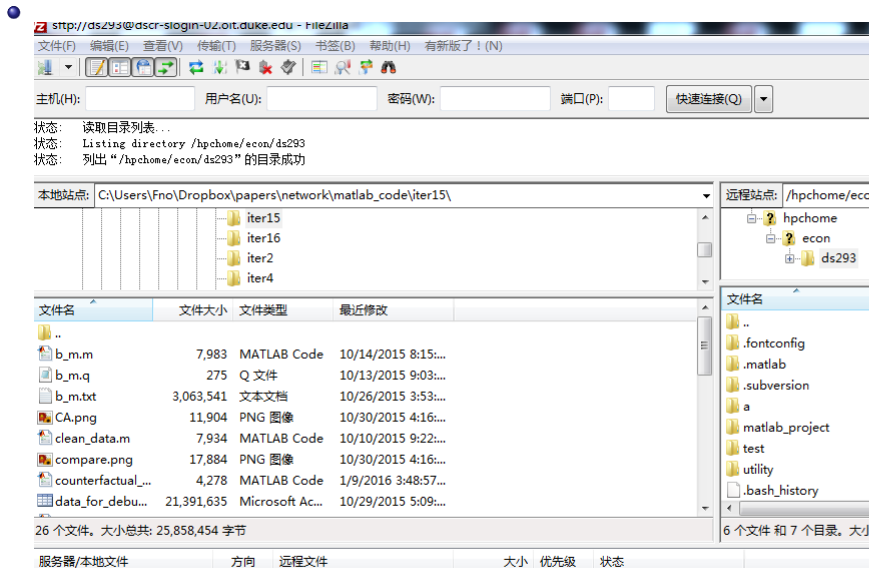
```
login as: ds293
#####
## You are about to access a Duke University computer network that is intended #
## for authorized users only. You should have no expectation of privacy in #
## your use of this network. Use of this network constitutes consent to #
## monitoring, retrieval, and disclosure of any information stored within the #
## network for any purpose including criminal prosecution. #
#####
ds293@dscr-slogin-01.oit.duke.edu's password:
Last login: Thu May 26 11:03:38 2016 from 10.182.19.14
#####
#
#          ***** ATTENTION ***** ATTENTION *****          #
#
# This is a Duke University computer system. This computer system, #
# including all related equipment, networks and network devices #
# (includes internet access) are provided only for authorized Duke #
# University use. Duke University computer systems may be monitored for #
# all lawful purposes, including to ensure that their use is authorized, #
# for management of the system, to facilitate protection against #
# unauthorized access, and to verify security procedures, survivability, #
# and operational security. During monitoring, information may be #
# examined, recorded, copied, and used for authorized purposes. All #
# information, including personal information, placed on or sent over #
# this system may be monitored. use of this Duke University computer #
# system, authorized or unauthorized, constitutes consent to #
# monitoring. Unauthorized use of this Duke University computer system #
# may subject you to criminal prosecution. Evidence of unauthorized use #
# collected during monitoring may be used for administrative, criminal, #
# or other adverse action. Use of this system constitutes consent to #
# monitoring for all lawful purposes. #
#####
#####
# This server is managed by PUPPET. If you make changes, please do so #
# through coordination with your systems administrator. Otherwise, we #
# cannot ensure your changes will persist #
#####
Last login by user ds293: Thu May 26 11:03 - 11:03 (00:00) from: 10.182.19.14
```


Prelim on Linux:

- Linux is a tree structure: root→sub dir...
 - no use of extension (unlike windows)
- ls: listing of files
- cd dir: go to the directory “dir”
- cd ..: go one up in the tree
- pwd: show current dir
- cd \$HOME go to home dir (Stata also borrows some syntax)

Transfer Files to/from Server:

- FileZilla (but you can use other alternatives as well)



Linux Command with Slurm:

- `srun`: execute a command
 - <http://www.schedmd.com/slurmdocs/srun.html>
- `-o`: short options (e.g. `-o slurm.out`)
- `--`: long options (e.g. `--output=slurm.out`)
- `srun /opt/apps/matlabR2015a/bin/matlab`
`-nodisplay -nodesktop -nosplash`
`-singleCompThread -nojvm -r`
`"rank=$SLURM_ARRAY_TASK_ID;t_m;quit"`
- Matlab related options: see
http://www.mathworks.com/help/matlab/ref/matlablinux.html?s_tid

Set up NLOpt in Linux:

- `chmod a+x configure`
- `./configure --prefix=$HOME/utility/nlopt_compile --enable-shared`
- `make distclean`
- `make`
- `make install`
- `export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:$HOME/utility/nlopt_compile/lib`

Set up NLOpt in Linux:

- run `"/opt/apps/matlabR2015a/bin/matlab -nosplash -nodisplay -nodesktop"`
- in matlab, run `"mex -setup c++"`
- rename `nlopt_optimize-mex.c` as `nlopt_optimize.c` in the octave folder under nlopt unzipped folder
- in the unzipped folder that store the `nlopt_optimize.c` file, run
`"mex nlopt_optimize.c
$HOME/utility/nlopt_compile/lib/libnlopt.so
-I$HOME/utility/nlopt_compile/include/"`
- run
`"addpath('/hpchome/econ/ds293/utility/nlopt-2.4.2/octave')"`
in matlab

Run nlopt in command mode:

- First you need to tell machine where the library is stored. Run:

- export

`LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:$HOME/utility/nlopt`

- `$HOME/utility/nlopt_compile` is the target directory where you install the nlopt

- run `"/opt/apps/matlabR2015a/bin/matlab -nosplash -nodisplay -nodesktop"`
- in matlab, run `"local_machine = 1;rank_id = 1;"`
 - If you take a look at test.m file, you know why do I have to do it.
- run `"test;"`
 - now you should be able to see the optimized result if you run `"theta_opt"`
- Remember after you are finished, run `"quit"` to exit Matlab environment

My Generic Script for Parallel:

- `#!/bin/bash`
- `#SBATCH --array=1-20`
- `#SBATCH --output=slurm_tm.out`
- `export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:$HOME/utility/nlopt_compile/lib`
- `srun /opt/apps/matlabR2015a/bin/matlab -nodisplay -nodesktop -nosplash -singleCompThread -nojvm -r "rank=$SLURM_ARRAY_TASK_ID;t_m;quit"`
- In workspace type "sbatch test.q"

Bootstrap:

- If you are not econometrician, then bootstrap.
- Idea: resampling and repeated estimation
- use rank as a feed for rng

- Run Script: `sbatch test.q`
- check status: `squeue -u (NetID)`
- cancel job: `scancel (job id or NetID)`