STAT605: Data Science Computing Project

Homework 3: Distributed Computing via Slurm and the Statistics High Performance Computing (HPC) Cluster

1. Login to a suitable HPC computer.

   - Login to `lunchbox.stat.wisc.edu`.
   - Run `srun --pty /bin/bash` to get an interactive job on a compute node. (There are many compute nodes that can handle many students. There is only one `lunchbox` and it cannot handle many people programming and running computations.)
   - Run `cd /workspace/STATuser` to work in a directory the compute nodes can read. (They cannot read your home directory.)

2. Solve the `mtcars` exercise at
   `www.stat.wisc.edu/~jgillett/605/HPC/examples/5mtcarsPractice/instructions.txt`.

   Hint: I recommend that you now go to step (4) and turn in an incomplete but working version of your work. (We will grade your last submission before the deadline.)

   Since this exercise (2) started as group work, so it is ok for your solution to look like the solution of members of your group. For exercise (3), below, you should do independent work, so your solution should not look like other students' solutions.

3. Read `http://stat-computing.org/dataexpo/2009/the-data.html`, which links to and describes data on all U.S. flights in the period 1987-2008. Find out, for departures from Madison:

   - How far you can get in one flight?
   - What is the average departure delay for each day of the week?

   To do this, write a program `submit.sh` and supporting scripts to:

   (a) Run 22 parallel jobs, one for each year from 1987 to 2008. The first job should:
      i. download the 1987 data via
         `wget http://pages.stat.wisc.edu/~jgillett/605/HPC/airlines/1987.csv.bz2`
      ii. unzip the 1987 data via `bzip2 -d 1987.csv.bz2`
      iii. use a short bash pipeline to extract from `1987.csv` the columns `DayOfWeek`, `DepDelay`, `Origin`, `Dest`, and `Distance`; and retain only the rows whose `Origin` is `MSN` (Madison's airport code); and write a much smaller file, `MSN1987.csv`.

   The other 21 jobs should handle the other years analogously.

   (b) Collect the Madison data from your 22 `MSN*.csv` files into a single `allMSN.csv` file, and write a set of jobs to answer the following two questions:

   - How far can you get from Madison in one flight? Write a line like `MSN,ORD,109` to answer. This line says, "You can fly 109 miles from Madison (MSN) to Chicago (ORD)." But 109 isn't the farthest you can get from Madison in one flight; write the correct line. (Hint: I used a `bash` pipeline to do this.) Save the result in `farthest.txt`.

- What is the average departure delay for each day of the week? Write a pair of lines like these to a file `delays.txt`:

  ```
  Mo  Tu  We  Th  Fr  Sa  Su
  8.3 5.0 4.3 5.5 9.5 2.1 3.5
  ```

  (These are not the correct numbers.)
  Hint: I used R's `tapply()` to do this.

4. Organize files to turn in your solution. (See "Copying files with `scp`," below.)

   (a) On your VM, make a directory `NetID_hw3`, where `NetID` is your NetID.

   (b) Make a subdirectory `NetID_hw3/mtcars`. Copy the following files there:

   - `getData.sh`
   - `jobArray.sh`
   - `findLightest.sh`
   - `submit.sh`
   - `out`

   We should be able to recreate `out` by running `./submit.sh`.

   (c) Make a subdirectory `NetID_hw3/airlines`. Copy the following files there:

   - `submit.sh`
   - `farthest.txt`
   - `delays.txt`
   - any supporting files required by your `submit.sh`

   We should be able to recreate `farthest.txt` and `delays.txt` by running `./submit.sh`.

   (d) Make a file `README` in the directory `NetID_hw3` with a line of the form `NetID,LastName,FirstName`. If you collaborated with any other students on this homework, add additional lines of this form, one for each of your collaborators. So, for example, if George Box with NetID `gepbox` worked with John Bardeen with NetID `jbardeen`, George's `README` file should look like

   ```
   gepbox,Box,George
   jbardeen,Bardeen,John
   ```

   (e) From the parent directory of `NetID_hw3`, run `tar cvf NetID_hw3.tar NetID_hw3` and then upload `NetID_hw3.tar` as your HW3 submission on Canvas.

   You can verify your submission by downloading it from Canvas, and then:

   i. Make a directory to test in, e.g. `mkdir test_HW3`.
   ii. Move your downloaded `.tar` file there `cd` there.
   iii. Extract the `.tar` file with `tar xvf NetID_hw3.tar`. This will make a new directory, which should be called `NetID_hw3`.
   iv. Check that all your files are there.

Copying files with `scp`

`scp` ("secure copy") copies files between a local machine and a remote machine. (`rsync` is another commonly-used program for this task.) Our linux handout mentions this form:
`scp [[user@]host:]file1 [[user@]host2:]file2]`, where the things in square brackets `[]` are optional. For example, from your VM:

- copy `file.txt` from `lunchbox.stat.wisc.edu` to your local machine via
  `scp username@lunchbox.stat.wisc.edu:/workspace/STATuser/file.txt file.txt`
- copy `file.txt` from your local machine to `lunchbox.stat.wisc.edu` via
  `scp file.txt username@lunchbox.stat.wisc.edu:/workspace/STATuser/file.txt`

See `man scp` for more information.