Interacting with Remote Systems + MPI

Advanced Statistical Programming Camp
Jonathan Olmsted (Q-APS)

Day 2: May 28th, 2014
PM Session
Getting the Setup Script

1. Log in to Adroit.
2. In your home directory, run:
   git clone https://github.com/olmjo/tigress-scripts.git
3. This will create a directory called tigress-scripts.
4. Navigate into it.
Create a Personal R Library

1. Start R at the Adroit prompt with `R`.
2. Install a package to your local library with `install.packages("foreach")`. This could be any package.
3. After answering “yes” and selecting a CRAN repository, quit R.
4. Now HPC packages can be installed in your user directory.
Running the Setup Script

1. From within the tigress-scripts directory run ./setup.sh.
2. Answer “yes” to all of the questions.
3. This will set up all of the R packages we need on Adroit.
Single-Node Parallelization

```r
## pseudo-code --- do not run
library(doParallel)
library(doRNG)

cl <- makeCluster(4, "PSOCK")
registerDoParallel(cl)
registerDoRNG()

out <- foreach(it = 1:10) %dopar% {
  ret <- getSomeResult()
  return(ret)
}
```

The parallelization we have seen so far will only work on one node.
## Multi-Node Parallelization

### pseudo-code --- do not run

```r
library(doMPI)
library(doRNG)

cl <- startMPICluster()
registerDoMPI(cl)
registerDoRNG()

out <- foreach(it = 1:10) %dopar% {
  ret <- getSomeResult()
  return(ret)
}

closeCluster(cl)
mpi.quit()
```

We have to use MPI parallelization to use more than one node at a time. This uses the doMPI package.
Portable Parallel Execution

- If you try to install or load the R package for doMPI on your laptop it won’t work.

- This means our script isn’t portable.

- The solution is to add a conditional expression into your R script.
## Portable Parallel Execution

```r
## Read Env Variable

inSLURM <- (Sys.getenv("SLURM_JOB_ID") != "")
## true only if a SLURM job

Are we in a SLURM job?
```
Portable Parallel Execution

If SLURM, use MPI.

```r
if (inSLURM) {
  library("doMPI")
  cl <- startMPIcluster()
  ## will auto detect number
  ## of workers from SLURM
  registerDoMPI(cl)
} else {
  library("doParallel")
  cl <- makeCluster(4, "PSOCK")
  registerDoParallel(cl)
}
```

If not, use a socket cluster.
Pre-Calculation Workers

We can let this dummy function stand in for a computational job.

doSome <- function(number) {
    return(rnorm(number))
}

nDraws <- 8000

Ultimately, we want 8,000 draws.
Pre-Calculating Workers

Whether we pre-calculate or not, we need the parallel backend.

```r
library("foreach")
library("doParallel")

## Loading required package: iterators
## Loading required package: parallel

cl <- makeCluster(spec = 4, type = "PSOCK")
registerDoParallel(cl)
```
Pre-Calculting Workers

We can run the job with 1-by-1 “assignment”.

```r
system.time({
  vDraws1 <- foreach(it = 1:nDraws,
      .export = c("doSome"),
      .combine = c,
      .packages = "foreach"
  ) %dopar% {
    return(doSome(1))
  }
})
```

## Warning: already exporting variable(s): doSome

```
##    user  system elapsed
##   3.928   0.190   4.209
```
Pre-Calculating Workers

Or, we can give each worker there share of the job upfront.

```r
nWorker <- getDoParWorkers()
system.time({
  vDraws2 <- foreach(it = 1:nWorker,
    .export = c("doSome", "nDraws"),
    .combine = rbind,
    .packages = "foreach"
  ) %dopar% {
    return(doSome(nDraws / nWorker))
  }
})
```

## Warning: already exporting variable(s): doSome, nDraws

## user  system elapsed
##  0.010  0.000  0.011
Getting the Code

- Download `allinone.R` from Blackboard.
- Download `allinone.slurm` from Blackboard.

- Place them in a convenient ASPC subdirectory on your local machine.
  - For example, `~/Desktop/aspc/`.
Moving the Code

- If you are on Windows, you are stuck with SFTP software like FileZilla.

- If you are on Linux or Mac, you can use SFTP software.
- You can also use rsync.
  - To copy the directory `aspc` and its contents into your home directory on Adroit, use

    ```
    rsync -rv ~/Desktop/aspc/ jolmsted@adroit.princeton.edu:~
    ```

    from your local machine.
Submitting the Job

1. Navigate to your SLURM script on the Adroit system.

2. Submit the job with `sbatch allinone.slurm`.

3. Output will write to `ovb-logit.csv`. 
Recovering the Output

• Again, if you are on Windows, you are stuck with SFTP software like FileZilla.

• If you are on Linux or Mac, you can use SFTP software.
• You can also use rsync.
  • To copy the directory aspc and its contents down from Adroit, use

    rsync -rv jolmsted@adroit.princeton.edu:~/aspc ~/Desktop/

    from your local machine.