Parallel Computing in R

Joseph S. Koopmeiners

Division of Biostatistics
University of Minnesota

PubH 8400
September 18, 2013
Simulation Studies will play an important role in your dissertation.

Why?
- Evaluate the small sample properties of an asymptotic result.
- Evaluate the properties of a complicated estimator for which it is very difficult or even impossible to derive analytical results.
Simulation Studies: Problem

- Simulation studies can be extremely time consuming
  - Evaluating the properties of an estimator that utilizes computationally intensive methods, such as the bootstrap or MCMC
  - Evaluating the operating characteristics of an adaptive clinical trial design
- Simulation studies that take days or even weeks can limit the number of simulations or the number of scenarios considered
Parallel computing refers to completing a large number of computations simultaneously using multiple computers or multiple cores within a single computer.

For example: It would be quicker for ten computers to calculate 100 integrals each than for one computer to calculate 1000 integrals.
A simulation study is essentially a one large computation that needs to be completed many times.

In standard computing, these computations are completed sequentially.

When utilizing parallel computing, a small number of simulations are completed on each machine/core and the results are combined for summarizing the results.

For example: consider a simulation study to evaluate the operating characteristics of a novel clinical trial design, 100 simulated trials could be completed on 10 computers and the results combined for a total of 1000 simulated trials.

This results in a substantial reduction in the time needed to complete the simulation study.
The Division of Biostatistics recently purchased several new servers:
- Student servers: potassium, cesium, carbon, chromium
- Faculty servers: silicon, rubidium

Each server has 24 cores.

Running a simulation study without parallel computing will only use a single core and the time will be no different than if the simulation study were run on a desktop/laptop.

You have to use parallel computing to fully utilize the Division’s computing resources.

How?
Parallel computing in R

- multicore is an R package for completing parallel computing in R
- multicore is user friendly and most simulation studies can be adapted to parallel computing by only changing a single line!
- multicore is installed on all of the new machines
Recall Julian’s example of a simulation study for multivariate logistic regression

- Julian utilized the ‘sapply’ function for running his simulation study
  
  ```r
  sim.results <- t(sapply(1:nsim,doSim,N=N, gamma=gamma))
  ```

- Julian’s code can be updated to utilize parallel computing by replacing `sapply` with `mclapply`
  
  ```r
  sim.results <- t(unlist(mclapply(1:nsim,doSim,N=N, gamma=gamma)))
  ```
• mclapply generalizes the lapply function to parallel computing
  • mclapply identifies the number of cores available
  • mclapply automatically splits your simulation into a collection of smaller computations and distributes them across available cores
  • mclapply recombines results to an identical form of the standard lapply function
The only argument worth remembering is the mc.cores argument

- This tells mclapply how many cores to use for parallel computing
- The default is to distribute parallel processing over all available cores
- I would use the default unless the servers are particularly busy
multicore does have its disadvantages

- mclapply returns results in a list (hence the use of the ‘unlist’ function)... this really isn’t that big of a deal
- You can’t set the seed. This is not ideal but it is worth it for the substantial savings in computation time