HDS 5230 High Performance Computing

Homework Week 4 - PARALLEL CODE, BOOTSTRAPPING, AND PROFILING

Miao Cai[*]

2019-02-12

You will use these datasets to answer some questions listed below. You must be careful to think about what the appropriate denominator is for each question. As you code the answers, be mindful to use the 'high performance' coding approaches in data.table.

1) Load up the dataset and convert to a data.table

```
pacman::p_load(gapminder, data.table, parallel, doParallel, foreach, ggplot2)
data(gapminder)
gapminder = data.table(gapminder)
```

2) We will focus on answering the following question: 'Is the median life expectancy in 2007 for Asia, Americas, Europe, and Africa significantly different from each other? Estimate the median life expectancy in 2007 for Asia, Americas, Europe, and Africa, and bootstrap confidence intervals for each continent.'

- a. First calculate the median life expectancy for Asia, Americas, Europe, and Africa for 2007. This is your 'point estimate' of the median for each continent.

```
gapminder2007 = gapminder[year == 2007 & continent != 'Oceania']
gapminder2007[, .(median_lifeExp = median(lifeExp,na.rm=T)), keyby = continent]
```

```
##    continent median_lifeExp
## 1:    Africa        52.9265
## 2:  Americas        72.8990
## 3:      Asia        72.3960
## 4:    Europe        78.6085
```

- b. For the following bootstrapped approach, implement both a parallel version using foreach(), and a non-parallel version using foreach(). Time the difference in the approaches. You can simply use Sys.time() as I did in the lecture.
- c. Now bootstrap a distribution of 100,000 possible medians for each continent. You should end up with 4 separate distributions of medians (you must do this separately for each continent – in class, we only did a single bootstrapped distribution).

```
detectCores()
```

```
## [1] 8
```

[*]Department of Epidemiology and Biostatistics, Saint Louis University. Email address miao.cai@slu.edu

```
cores = 8
cl <- makeCluster(cores)
registerDoParallel(cl)
n_boot = 1e5
gc()
```

```
##           used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells  720442 38.5    1165667 62.3         NA  1165667 62.3
## Vcells 1359307 10.4    8388608 64.0      16384  2352545 18.0
```

```
# Not parallel
starttime1 = Sys.time()
full_result1 <-
  foreach(i=1:n_boot, .combine=function(x,y)rbindlist(list(x,y))) %do% {
  result1_a = gapminder2007[sample(.N, replace = T),
                                .(median_lifeExp = median(lifeExp, na.rm=T)),
                               continent]
  result1_a
}
endtime1 <- Sys.time()
endtime1 - starttime1
```

```
## Time difference of 24.65049 mins
```

```
#hist(result1)
full_result1[continent == 'Asia', quantile(median_lifeExp, c(0.025, 0.975))]
```

```
##    2.5%   97.5%
## 68.9565 74.1920
```

```
# Parallel version
rm(result1_a)
gc()
```

```
##           used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells 1032101 55.2    1766560 94.4         NA  1165667 62.3
## Vcells 2195752 16.8    8388608 64.0      16384  8388608 64.0
```

```
starttime2 <- Sys.time()
full_result2 <- foreach(i=1:cores,
          .combine='c',
          .packages = 'data.table') %dopar% {
    lapply(1:floor(n_boot/cores), function(x){
      gapminder2007[sample(.N, replace = T),
                    .(median_lifeExp = median(lifeExp, na.rm=T)),
                    keyby = continent]
    })
}
full_result2 = rbindlist(full_result2)
endtime2 <- Sys.time()
endtime2 - starttime2
```

```
## Time difference of 1.436212 mins
```

```
#hist(result2)
full_result2[continent == 'Asia', quantile(median_lifeExp, c(0.025, 0.975))]
```

```
##    2.5%   97.5%
## 68.9565 74.1920
```

```
stopCluster(cl)
```

- d. Summarize the 95% of these 4 distributions by calculating the 2.5% quantile and the 97.5% quantile. This is done separately for each continent (each distribution). You can just directly calculate this using the quantile function in R like so: quantile(sample, c(.975,.025))

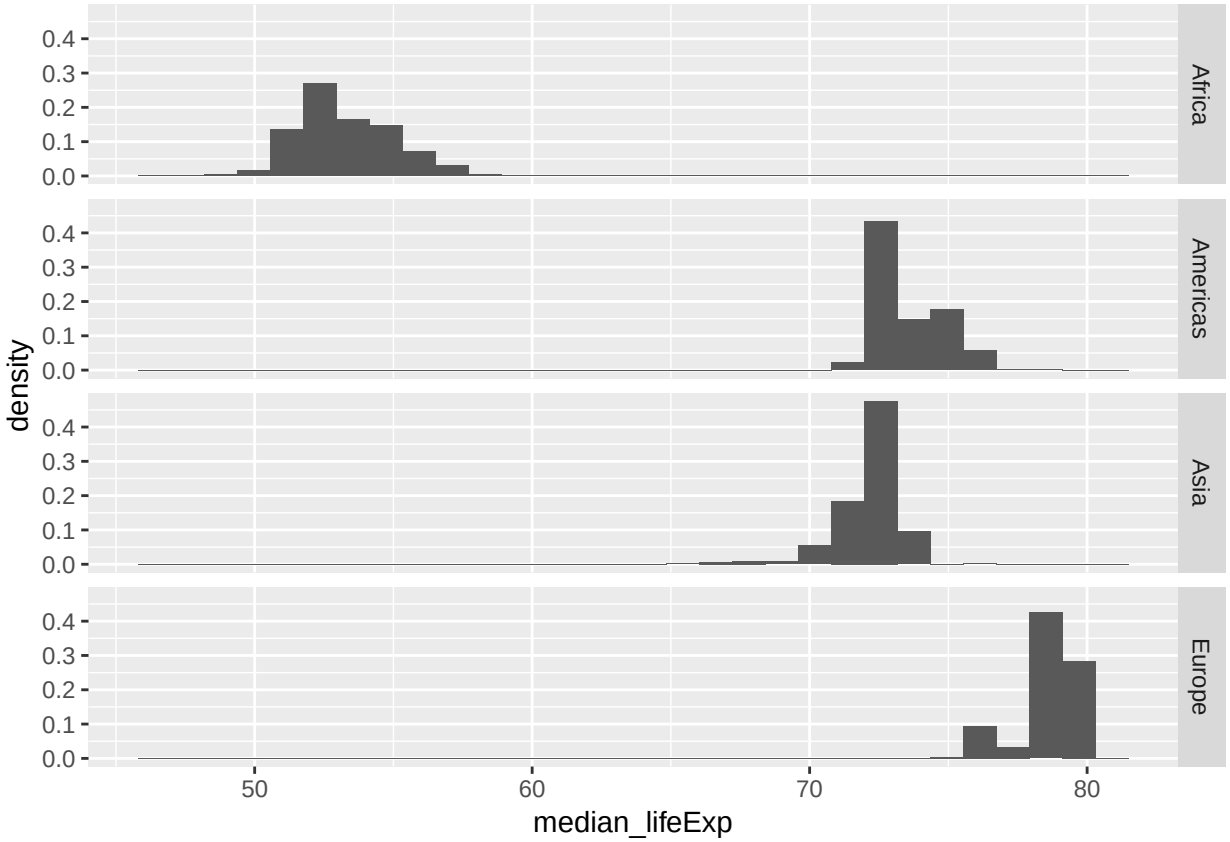
```
full_result2[continent != 'Oceania', .(CI_left = quantile(median_lifeExp, 0.025),
                CI_right = quantile(median_lifeExp, 0.975)),
            by = continent]
```

```
##    continent CI_left CI_right
## 1:    Africa 50.5610  56.7315
## 2:  Americas 71.8780  76.1950
## 3:      Asia 68.9565  74.1920
## 4:    Europe 75.7480  79.4830
```

- e. Create a separate histogram for each of the 4 distributions. Stack these distributions on top of each other. This is easily done with ggplot2 and facet_grid if the data is all in one dataframe in a long format with a group indicator. Here is sample code (without the bootstrap part!):

```
ggplot(full_result2, aes(x=median_lifeExp)) +
  geom_histogram(aes(y=..density..)) +
  facet_grid(continent~.)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


Africa
Americas
Asia
Europe
density
0.0
0.1
0.2
0.3
0.4
50
60
70
80
median_lifeExp