Statistics 5014: Homework 7

Due In Class October 25, 9am

2017-10-18

For each assignment, turn in by the due date/time. Late assignments must be arranged prior to submission. In every case, assignments are to be typed neatly using proper English in Markdown.

This week, we spoke about parallelizing R. We looked at two methods, parallizing for loops and parallelizing the apply family of functions. In this homework, we will explore parallelization. A word of caution, this homework involves parallel functions. If you try to start a cluster when one is already started, you can easily lock up your computer and have to do a hard boot. A. Save (commit) your homework often. B. Remember to stop the cluster before starting another!! C. Figure out how many cores your computer has to play with and back off by 1. IE, mine has 8, I prefer to not exceed cluster sizes of 6 or 7.

Problem 1 (GitHub 2 pts + 2 Style pts)

This week we will change gears a little on GitHub. Today we start making homework something we can use to highlight our professionalism and abilities. You will continue to retrieve the lectures and homeworks from my GitHub (anyway you like), but you will:

1. create a new GitHub Repository (online)

2. invite me as a collaborator (settings –> collaborators)

3. setup an Rproject pointing to this new repository

4. create a new RNotebook file within the project folder

5. save the notebook HW7_lastname_firstname

Note that this homework will be graded for professionalism as well as your ability to solve the problems (passing is ≥ 7. Remember, reproducibility requires weaving text (appropriate for a reader), code (commented) and necessary output as a compendium for what was done.

Problem 2: Sums of Squares (2 pts)

Similar to the last homework, we will calculate sums of squares total using:

a. a for loop to iterate through all data points calculating the summed squared difference between the data points and mean of the data.
b. repeat part a, but use vector operations to effect the same computation
c. repeat part a, but use dopar
d. repeat part a, but use parSapply

In all cases, wrap the code in "system.time({})". You should report the final answer and timings in a nice table. Make note of any parameters you had to set. What observations do you have?

To generate the data, use:

```r
set.seed(12345)
y <- rnorm(n = 1e+07, mean = 1, sd = 1)
```

Problem 3: Gradient Descent (2 pts)

From the last homework, the algorithm is:

- $while(abs(\Theta_0^i - \Theta_0^{i-1})$ AND $abs(\Theta_1^i - \Theta_1^{i-1}) > tolerance)$ {

$$\begin{aligned}\Theta_0^i &= \Theta_0^{i-1} - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_0(x_i) - y_i) \\ \Theta_1^i &= \Theta_1^{i-1} - \alpha\frac{1}{m}\sum_{i=1}^{m}((h_0(x_i) - y_i)x_i)\end{aligned}$$

}

Where $h_0(x) = \Theta_0 + \Theta_1 x$.

What would you parallelize around here? Hint: what parameters do YOU need to specify.

Given $\mathbf{X}$ and $\vec{h}$ below, implement the above algorithm, parallelizing around the hint given above. Compare the values obtained and contrast those with the results given by lm(h~0+**X**).

```r
set.seed(1256)
theta <- as.matrix(c(1, 2), nrow = 2)
X <- cbind(1, rep(1:10, 10))
h <- X %*% theta + rnorm(100, 0, 0.2)
```

Problem 4: Bootstrapped regression (2 pts)

There are situations where you want another realization of the data from a population. If you have a random sample from a population, you can randomly draw from that sample (with replacement) to produce a new realization of the sample. If you do this many times calculating some sort of summary statistic for each bootstrapped sample, this is called bootstrapping.

The basic procedure for bootstrapping in the regression setting is:

For b ∈ {1,...,B}

- Sample $Z^{(b)} = (X, Y)^{(b)}$
- Calculate $\hat{\beta}^{(b)}$

A. Impliment this algorithm using the data generated below for B=10,000. Do not use the boot package, use the sample function in base R: sample(x=,size=,replace=T).
B. Create a table of the result with the appropriate summary statistics.
C. Create histograms of the distribution of $\hat{\beta}$'s.

Which parallelization method did you use? What impediments did you encounter? How long did it take?

```r
set.seed(1267)
n <- 200
X <- 1/cbind(1, rt(n, df = 1), rt(n, df = 1), rt(n, df = 1))
beta <- c(1, 2, 3, 0)
Y <- X %*% beta + rnorm(100, sd = 3)
```

Problem 5

Push your homework to YOUR GitHub.

This is YOUR GitHub Repository. Save what you want, there is a limit to how big a repo can be, so be a little sparing on what you put up there. Please save the file you would like me to read as: HW#_lastname_firstname.html. I am not going to go through your Markdown to try to figure out what you were doing.

Optional preperation for next class:

Read up on regular expressions.