Applied Statistical Programming - Apply

2/16/2022

Write the R code to answer the following questions. Write the code, and then show what the computer returns when that code is run. Thoroughly comment your solutions.

You have until the beginning of class 2/21 at 10:00am to answer all of the questions below. You may use R, but not any online documentation. Submit the Rmarkdown and the knitted PDF to Canvas. Have one group member submit the activity with all group members listed at the top.

A simulation experiment using `apply` & `plyr`

For this assignment, **you cannot use looping structures**. You will also need to create *arrays* to work with this problem. If you can imagine matrices layered on top of each other, this is an array. Use the following example to familiarize yourself with making arrays and referencing their values.

```
# Create two vectors of different lengths
vector1 <- c(5, 9, 3)
vector2 <- c(10, 11, 12, 13, 14, 15)

# Put these vectors into an array of two 3x3 matrices.
result <- array(c(vector1, vector2), dim = c(3, 3, 2))
print(result)
```

```
## , , 1
##
##      [,1] [,2] [,3]
## [1,]    5   10   13
## [2,]    9   11   14
## [3,]    3   12   15
##
## , , 2
##
##      [,1] [,2] [,3]
## [1,]    5   10   13
## [2,]    9   11   14
## [3,]    3   12   15
```

```
# Reference the 3rd column in matrix 2
result[, 3, 2]
```

```
## [1] 13 14 15
```

1. Make a three dimensional array with `dim=c(20,5, 1000)` and fill it with random data. Think of this as 1000 random datasets with 20 observations and 5 covariates.

```
# Code
```

2. Use the provided vector of linear model coefficients `Beta`. Make a function to create Y values for a linear model. The Y values should be a linear combination of the X's plus some random noise. The output should be a 20 by 1000 array.

```
# Remove the eval=FALSE header from this code block before continuing
Beta <- matrix(c(1, 2, 0, 4, 0), ncol = 1)
X <- matrix(rnorm(100), ncol = 5, nrow = 20)
X[1, ]
Beta
X %*% Beta  # No noise has been included yet
```

3. Run 1,000 regressions across all of this simulated data. Have as the output a 1000 by 6 matrix of estimated regression coefficients.

```
# Code
```

4. Create a density plot for each of the 6 coefficients, each of which should have been estimated 1,000 times in the previous step. Describe what the density plot represents.

```
# Code
```

5. Re-run that code in parallel. Calculate the differences in run time.

```
# Code
```