

# BUAN 6357 (Johnston)

## Homework 6

Code Due: 13 April 2019 (6PM)

Part B Due: 14 April 2019 (11:59PM)

Points available: 120.

This assignment is about generating and evaluating OLS coefficient error estimates. Use the “Concrete\_Data\_wj.csv” file from UTDbox>demo as a starting point. Part A involves data import, modeling, error calculation, and observation tracking across several Bootstrapping techniques and scenarios. Part B will require the summary of parameter estimate variability for each Bootstrap technique. Be prepared to perform both parametric and non-parametric summary operations on these calculated parameter estimate values.

For this assignment you will need the packages “tidyverse”, “broom”, and “data.table”, in that order. You will not need any additional packages. You should use only the “require()” or “library()” statement in your code. Any use of the install.packages() function in submitted code will result in a score of 0 for that submission.

The first commands of your code submitted for grading to eLearning MUST be:

```
setwd(“c:/data/BUAN6357/HW_6”); source(“prep.txt”, echo=T)
```

and the last command of your code MUST be:

```
source(“validate.txt”, echo=T)
```

Be careful with the quote characters as they must ALL be the same at the beginning and end of a string. (Use the single or double quote character from the key next to “Enter”.) Inclusion of these lines is required BEFORE your code will be tested.

Submit the code to eLearning as an ASCII file which can be copied directly into R.

You may submit this assignment as many times as needed until you get full credit.

The model being evaluated for this assignment is an OLS using the variable “strength” as outcome.

Each deliverable is to be a data.table object generated using tidy() on OLS models. After reading the data, you should not add any variables to the data table.

Based on previous code submission practices, any time multiple versions of a submission are listed as pending by eLearning, only the most recent one will be evaluated. (E.g: if eLearning displays information on an ungraded code submit as, say, 2 of 5, that submit (#2 of 5) will be graded as failed (score: 0). )

Reset the RNG seed before each Bootstrap deliverable. Use seed values of 899796546, 544546653, 518959292, and 212055872 for deliverables #3 through #6, respectively. Use the sample() function for sampling and randomization. You will want to collect run-times for deliverables #2 through #6 using system.time() .

Deliverables (each as a data table of all applicable models):

- |         |  |
|---------|--|
| 1. raw  | original data, as read                       |
| 2. m1   | baseline (non-Bootstrap) model               |
| 3. d2f  | classic (non-balanced) Bootstrap using for() |
| 4. d3dt | classic (non-balanced) Bootstrap using dt[]  |
| 5. d4f  | balanced Bootstrap using for()               |
| 6. d5dt | balanced Bootstrap using dt[]                |

Part B of HW 6 will direct you to explore the deliverables from Part A and answer questions about each and/or using each of them. You may submit answers to HW 6 part B as many times as you wish but only the score for the last submitted code will be retained.