Business Analytics – Homework 2

Due 9 am, Friday September 13

In this assignment you will improve your R coding skills. You will be asked to write short code snippets or code-chunks in R. You will be graded both on your code, and the written answers you provide. When evaluating the code, the grader will take on the role of a co-worker. Code will evaluated both in terms of how correct and how clear it is. By correctness, I mean that the code fulfills the requirements of the question. By clarity, I mean that the grader should be able to understand what your code does within 30 seconds of reading it. As discussed in class, this is aided by clear comments, good variable names, proper indentation, and short lines.

1. Create the matrix `X_data`. Its dimensions must be 1000 rows and 12 columns. The content of each entry must be a randomly chosen number between 1 and 15 with no more than 3 decimal points. Columns should be named `X_data_`1, `X_data_`2… `X_data_`12. For the random number generation use the command `rnunif()`

   Create one column vector named `betas` with the same number of column as `X_data`. Each component must be an integer random number between 1 and 5 both included.

2. Create two column vectors with the same number of rows as `X_data`. One must be named `Noise` and the other must be named `Y_data`. Each content of noise must be a random draw of a random normal distribution with mean zero and sigma = 4. Hint, Use the command `rnorm().` The Y vector must contain the result of the following expression.

   $$Y_{data} = X_{data}\beta + Noise$$

   Note that to multiply to matrices in R use the operator `%*%`

We are going to recover the value of `beta` vector using only the information contained in `Y_data` and `X_data`. In statistics this is called estimating `beta`, in machine learning this is called learning `beta`. This simplest way to do so running a (multivariate regression). Go to any book in Econometrics and look the expression for such estimator. You will find that it is given by $(X'X)^{-1}\,(X'Y)$

3. Create a column vector that is called `beta_hat` which contains the result of $(X'X)^{-1}\,(X'Y)$

4. Finally create a matrix called `estimation_error`. `estimation_error` has two columns and the same number of rows as vectors `beta_hat` and `beta_data`. Make sure

that the columns have the correct name. Are they similar? What is the biggest difference in absolute value?

5. Do you think we will get a smaller error if instead of 1000 rows we have 20000? Why? Let's find out.

    a. Create the vector `max_error_obtained` with components named `n_rows_1000`, `n_rows_2000`, `n_rows_20000`. That is the length of the `max_error_obtained` is 20. The first component of `max_error_obtained` should store the solution of question 4. For the others 19 components you have to find it out replacing the 1000 with 2000, …, 20000

    b. Make a barplot representing the vector `error_obtained`

6. (completely optional) More generally if you copy-paste the following code in your R-studio console

```
set.seed(123546)
X = matrix(runif(100*10),100,10)
Y = X%*%cbind(1:10) + runif(100)
summary(lm(Y~0+X))
```

Then you will get like:

```
Call:
lm(formula = Y ~ 0 + X)
Residuals:
     Min       1Q   Median       3Q      Max 
-0.70734 -0.18780  0.03039  0.22660  0.53028 

Coefficients:
    Estimate Std. Error t value Pr(>|t|)    
X1   1.09930    0.11106   9.898 4.65e-16 ***
X2   2.21425    0.10229  21.647  < 2e-16 ***
X3   3.04822    0.10944  27.853  < 2e-16 ***
X4   4.07262    0.10418  39.091  < 2e-16 ***
X5   5.01407    0.10663  47.022  < 2e-16 ***
X6   6.02976    0.11488  52.489  < 2e-16 ***
X7   7.17835    0.11545  62.176  < 2e-16 ***
X8   8.12032    0.11266  72.077  < 2e-16 ***
X9   9.17265    0.09627  95.279  < 2e-16 ***
X10 10.01759    0.10037  99.805  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 0.3091 on 90 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9999 
F-statistic: 9.678e+04 on 10 and 90 DF,  p-value: < 2.2e-16
```

Try replicating every single number in the table… Good luck Brandeisians