Lab 01: Equations and -Basics

**Handout date:** Wednesday, August 26, 2020

**Due date:** Wednesday, September 2, 2020 submitted as Word document to eLearning’s ***SumitLab01*** link.

*This lab counts 4 % toward your total grade*

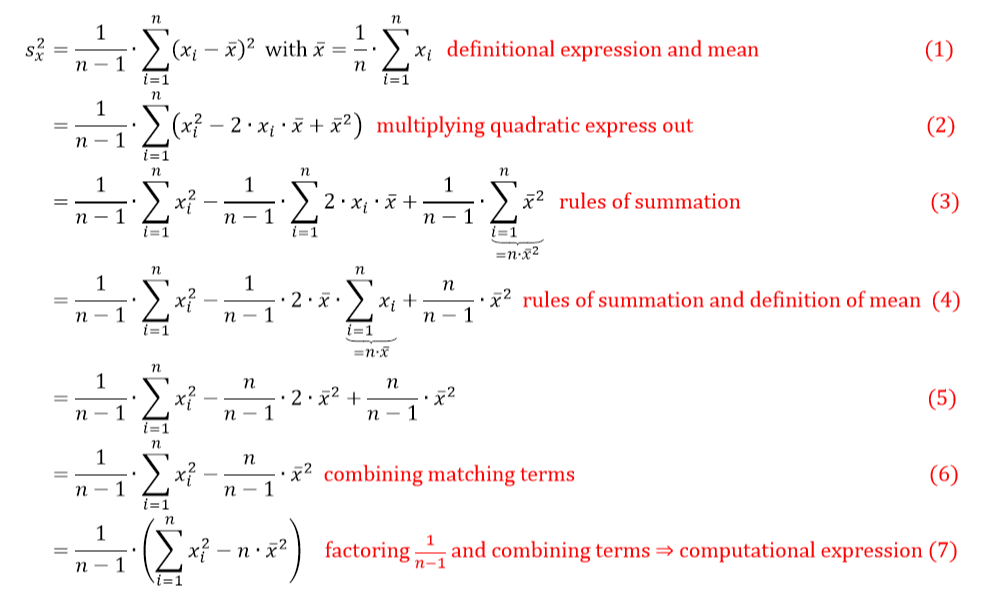
**Objectives:** In this lab   
[a] you will practice your understanding of the structure of mathematical equations and type-setting equations with Word,   
[b] you will practice your  skills.

**Format of answer:** Your answers (graphs and verbal description) should be handed in as Word document. Add a running title into the header of the document with the following information: ***your name***, ***GISC6301-Lab01*** and ***page numbers***. Label each answer properly starting with its task number. Maintain the sequence of questions. Format any code and computer output properly before inserting it into the document with your answer. -code and text output need to be in a ***monospaced*** font (i.e., fixed-pitch font) such as Courier New so proper spacing and alignments are preserved. Excessive and irrelevant output will lead to a deduction of your accumulated points.

# Task 1: Equation Editor Exercise (1 pt)

Typeset the derivation of the definitional equation of the variance equation in line (1) from its computational counterpart in line (7). That is, your sequence of equations needs to be in the opposite order starting with equation (7) and ending at equation (1)

Make sure that your equation formatting follows exactly the one given in the screen shot below. In particular, focus on [a] the change from the *italics* equation mode to the upright text mode now in your first equation, [b] proper nesting of the employed mathematical templates, [c] the alignment of all equation at the “”-sign, and [d] the use of underbraces in equations (3) and (4). You do not need to replicate the *explanations* and *line numbers* in red. Deviations from the formatting in the screen shot below will lead to a loss of partial points.



**Figure 1: Definitional (1) and computational (7) expressions of the empirical variance estimators**

# Task 2: Importing Data (0.5 pts)

Setup an  working directory and save the files **MyPower.RData** , **Concord1.sav** and **CPS1985.dbf**. into this directory.

1. Explore the **load( )** function and import **MyPower.RData** .
2. Use a function from the library **foreign** to import **Concord1.sav** and save it under the name **Concord**.
3. Use a function from the library **foreign** to import **CPS1985.dbf** and save it under the name **CPS1985**.
4. Explore the documentation of the data-frame **Mroz** in the library **carData** and link the data-frame to your  session with the **data( )** function.
5. To demonstrate that everything worked as intended show a screenshot of **Global Environment**, which displays all 4 data-frames.

# Task 3: Data-frame Basics (1.5 pts)

1. For the data-frame **MyPower** calculate the average daily power consumption by using the variables **kWhBill** and **DaysBill** and add the new variable to the data-frame **MyPower** with the variable name **DailykWh**. Show your  code for this calculation. (0.2 pts)
2. Apply the statements   
    **MyPowerNames <- names(MyPower)  
    length(MyPowerNames)  
    MyPowerNames[4:6]**  
   What are these statements doing? (0.2 pts)
3. Apply the statement **sapply(***data-frame***, is.factor)** on the data-frame **MyPower** to evaluate the which variables are factors. What is this statement doing? Show a copy of the Console with the output of this investigation. (0.1 pts)
4. Apply the **str( )** on the data-frame **MyPower**. What information about the data-frame does the **str( )** provide to you? (0.1 pts)
5. What are the following statements doing? Show the plot and elaborate on the syntax of the statements. Is the power consumption over time decreasing? (0.3 pts)  
    **plot(DailykWh~SeqID, data=MyPower)  
    abline( lm(DailykWh~SeqID, data=MyPower) )**
6. What is the following statement doing? Show the plot and elaborate on the syntax of the statements. Why does the power consumption fluctuate over the seasons? (0.2 pts)  
    **plot(DailykWh~Month, data=MyPower)**
7. Use the syntax **MyPower[***rows***,** *cols***]** to select all records with the three variables **c("MinTemp","AveTemp","MaxTemp")** (alternatively you could use **MyPowerNames[4:6]**) in the month of **MyPower$Month=="JAN"**. Show the code and its output. (0.2 pts)
8. Look and show at the header and the tail of the data-frame **MyPower** with the functions **head( )** and **tail( )**. (0.1 pts)
9. What class is the output **MetricPower** of the operation below? (0.1 pts)  
   **MetricPower <- MyPower[ , c("MinTemp","AveTemp","MaxTemp", "DailykWh")**

# Task 4: R Basics (1 pt)

1. Depending on the input object class type  functions behave differently. To see the different class-specific implementations of the generic **summary( )** function try the command **methods(summary)**.  
   Discuss the difference in the behavior of the **summary( )** function when applied to a **data.frame** or a **lm** object. (0.4 pts) (0.2 pts)
2. Explore the online help for  
    **?summary.lm  
    ?summary.data.frame**and discuss the optional parameters (0.2 pts)
3. What are the following statements below doing when processing the vector x of length 6? (0.3 pts)  
    **x <- c(1,3,5,7,9,NA)**  
    **x \* 2; x + 2**  
    **y <- seq(0,2, by=1); x \* y**  
    **z <- rep(c(1,2),3); x \* z**Note, the semicolon **;** allows to place several independent  commands into one line. Look the online help up for the functions **seq( )** and **rep( )**.
4. Define the function **myMean( )** and apply it on **x**  
    **myMean <- function(x){  
    x <- na.omit(x)  
    sum(x)/length(x)  
    }  
    myMean(x)**Compare its output to that of the standard  function **base::mean( )**. Look up the help for the functions **na.omit( )** and **mean( )**. (0.2 pts)
5. How does the statement **x[c(T,T,T,T,T,F)]** work? (0.1 pts)