**MSDS 6306: Doing Data Science**

# Live session Unit 01 assignment

**Due: Monday 11:59pm Jan 14th .**

Calculation Questions (using R) - Use R to calculate the following:

**NOTE:** Submit a word file with the R code in text format and a screenshot that shows the result.

1. (20 points) Basic Math – each question has 5% credits
   1. The log of a positive number.
   2. What is the default base for the log function? Calculate the log of your previous number with a different base.
   3. The log of a negative number. (explain the answer)
   4. The square-root of a positive number.
2. (15 points) Random number generation.
   1. Create a vector of 15 standard normal random variables. Calculate its mean and SD (Standard Deviation).
   2. Change the mean to 10 and the SD to 2 and recalculate the vector of 15 random normal variables. Calculate its mean and SD.
   3. Why are the means and SD not exactly the same as the means and SDs specified in the function?
3. (40 points) Vector Operations
   1. The weights of 6 individuals in kg are 60, 72, 57, 90, 95, 72.

**b.** Their heights (in m) are 1.80, 1.85, 1.72, 1.90, 1.74, 1.91.

1. Enter these vectors into R.
2. Create a scatterplot of weight vs. height. Interpret the scatterplot.
3. Calculate the BMI for each individual. (BMI = weight in kg divided by the square of the height in m)
4. Calculate the mean for weight.
5. Subtract the mean from each value of weight.
6. Sum the result. Now you know why we square the deviations from the mean to calculate a standard deviation!
7. (25 points) Your data science profile. Enter your data science profile into R as a data frame with two columns. Call it by your first name. The categories are computer programming, math, statistics, machine learning, domain expertise, communication and presentation skills, and data visualization. Your ranking for each category 1-5, with 5 as best. Create a bar graph of your data science profile. When you submit your work, please submit your code, including the data entry piece.

Results

1----A- > log(10)

[1] 2.302585

B-Log function in R –log() computes the natural logarithms (Ln) for a number or vector. Apart from log() function, R also has log10 and log2 functions.

log10 function –log10(), computes common logarithms (i.e. base 10)

log2 function – log2(), computes binary logarithms (i.e. base 2)

log(10, base=3)

[1] 2.095903

C----> log(-10)

[1] NaN

Warning message:

In log(-10) : NaNs produced

logarithm is only defined for positive numbers, you can't take the logarithm of negative values.

D--- > sqrt(10)

[1] 3.162278

2------A > x <- sample(15, replace = TRUE)

> x

[1] 11 3 2 7 10 10 2 1 13 8 3 9 13 7 13

> mean(x)

[1] 7.466667

> sd(x)

[1] 4.323799

B x <- rnorm(15, mean = 10, sd = 2)

> x

[1] 11.523512 7.279507 11.853291 8.237610 9.944753 8.288014 6.481113

[8] 8.309909 10.838592 7.911740 11.286956 9.070645 10.707820 9.652100

[15] 12.064920

> mean(x)

[1] 9.563365

> sd(x)

[1] 1.773737

C  This is illustrating the difference between sample and population statistics. I wouldn't expect a sample to have the same mean and sd as the population

3 A, B, C--> wt<- c(60, 72, 57, 90, 95, 72)

> ht<- c(1.80, 1.85, 1.72, 1.90, 1.74, 1.91)

> plot(ht, wt)

> plot(wt, ht)

> plot(ht, wt)

D--The graphic below show that there is no correlation between the Hight and the Weight

A screenshot of a cell phone

Description automatically generated

E-- > bmi<-wt/ht^2

> bmi

[1] 18.51852 21.03725 19.26717 24.93075 31.37799 19.73630

F--> mean(wt)

[1] 74.33333

G--- mean<-mean(wt)

> mean

[1] 74.33333

> v<- mean - wt

> v

[1] 14.333333 2.333333 17.333333 -15.666667 -20.666667 2.333333

H--> sum(v)

[1] -2.842171e-14

4- > matrix<-matrix(data = c(2,2,3,2,4,2,2), nrow = 7, ncol = 1, byrow = TRUE,dimnames = NULL)

> matrix

[,1]

[1,] 2

[2,] 2

[3,] 3

[4,] 2

[5,] 4

[6,] 2

[7,] 2

> Fabio<-c("computer programming", "math", "statistics", "machine learning", "domain expertise", "communication and presentation skills", "data visualization")

> Fabio

[1] "computer programming"

[2] "math"

[3] "statistics"

[4] "machine learning"

[5] "domain expertise"

[6] "communication and presentation skills"

[7] "data visualization"

> matrix2<-cbind(matrix, Fabio)

> matrix2

Fabio

[1,] "2" "computer programming"

[2,] "2" "math"

[3,] "3" "statistics"

[4,] "2" "machine learning"

[5,] "4" "domain expertise"

[6,] "2" "communication and presentation skills"

[7,] "2" "data visualization"

> matrix2<-cbind(Fabio, matrix)

> matrix2

Fabio

[1,] "computer programming" "2"

[2,] "math" "2"

[3,] "statistics" "3"

[4,] "machine learning" "2"

[5,] "domain expertise" "4"

[6,] "communication and presentation skills" "2"

[7,] "data visualization" "2"

> df<-data.frame(matrix2)

> df

Fabio V2

1 computer programming 2

2 math 2

3 statistics 3

4 machine learning 2

5 domain expertise 4

6 communication and presentation skills 2

7 data visualization 2

> colnames(df)<-c("Fabio", "Score")

> df

Fabio Score

1 computer programming 2

2 math 2

3 statistics 3

4 machine learning 2

5 domain expertise 4

6 communication and presentation skills 2

7 data visualization 2

> Fabio<-df

> barplot(height = c(2,2,3,2,4,2,2), width = 1, space = NULL, names.arg = c("computer programming", "math", "statistics", "machine learning", "domain expertise", "communication and presentation skills", "data visualization"), legend.text = "Fabio Profile")

> barplot(height = c(2,2,3,2,4,2,2), width = 1, space = NULL, names.arg = c("CP", "Math", "Stat", "ML", "DE", "Com", "DV"), xlab = "Categories", ylab = "Score", legend.text = "Fabio Profile")

A screenshot of a cell phone

Description automatically generated