lab\_01

lisa liubovich

2024-01-25

DS 413/613  CLASSWORK/LAB   **Functions**

Instructions:  Email your code and work (coding and output) by generating an Rmardown File and the default Word File.

1. In three or four sentences, explain why constructing a Function . in order to execute tasks, is beneficial or advantageous

Constructing a function to execute tasks is beneficial because it makes your code more efficient. Functions also make code more readable by providing structure and naming conventions. Functions are also easier to identify and fix specific issues while debugging.

Write a function (using r code and structure demonstrated in class) to calculate a z score for a given observed value, a mean, and a standard deviation value. And then use your function to find a z score for the following problem. (Research the internet to find the formula used to calculate a z score)

Observed value = 25.77,  mean =  23.54,  standard deviation = 2.442

calculate\_z\_score <- function(observed\_value, mean, standard\_deviation) {  
 z\_score <- (observed\_value - mean) / standard\_deviation  
 return(z\_score)  
 }  
   
observed\_value <- 25.77  
mean\_value <- 23.54  
std\_deviation <- 2.442  
  
z\_score\_result <- calculate\_z\_score(observed\_value, mean\_value, std\_deviation)  
print(paste("Z-Score:", round(z\_score\_result, 4)))

## [1] "Z-Score: 0.9132"

Write a function (using r code and the structure demonstrated in class) to calculate the natural log of a number multiplied by the common log of the same number divided by the cube root of a given prime number.

Use your function to find the answer if the number to be used for both log expressions is 32 and the given prime number is 11.

Also use R code to round your answer to the nearest tenth

calculate\_log\_expression <- function(number, prime\_number) {  
 result <- log(number) \* log10(number) / (prime\_number^(1/3))  
 return(result)  
}  
  
  
number\_value <- 32  
prime\_number <- 11  
  
log\_expression\_result <- calculate\_log\_expression(number\_value, prime\_number)  
rounded\_result <- round(log\_expression\_result, 1)  
  
print(paste("Result of Logarithmic Expression:", rounded\_result))

## [1] "Result of Logarithmic Expression: 2.3"

Use and show R coding to calculate the standard deviation for each variable of the data table **mtcars** using the **"Special For Loop Method"** demonstrated in the class notes.

for (col in colnames(mtcars)) {  
 if (is.numeric(mtcars[[col]])) {  
 std\_dev <- sd(mtcars[[col]])  
 print(paste("Standard Deviation for", col, ":", round(std\_dev, 2)))  
 }  
}

## [1] "Standard Deviation for mpg : 6.03"  
## [1] "Standard Deviation for cyl : 1.79"  
## [1] "Standard Deviation for disp : 123.94"  
## [1] "Standard Deviation for hp : 68.56"  
## [1] "Standard Deviation for drat : 0.53"  
## [1] "Standard Deviation for wt : 0.98"  
## [1] "Standard Deviation for qsec : 1.79"  
## [1] "Standard Deviation for vs : 0.5"  
## [1] "Standard Deviation for am : 0.5"  
## [1] "Standard Deviation for gear : 0.74"  
## [1] "Standard Deviation for carb : 1.62"