CIND 123 - Data Analytics: Basic Methods

Karnaz Obaidullah

Assignment 1 (10%)

Karnaz Obaidullah

CIND123 Section DHD 501000900

# Instructions

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. Review this website for more details on using R Markdown <http://rmarkdown.rstudio.com>.

Use RStudio for this assignment. Complete the assignment by inserting your code wherever you see the string “#INSERT YOUR ANSWER HERE”.

When you click the **Knit** button, a document (PDF, Word, or HTML format) will be generated that includes both the assignment content as well as the output of any embedded R code chunks.

**NOTE**: YOU SHOULD NEVER HAVE install.packages IN YOUR CODE; OTHERWISE, THE Knit OPTION WILL GIVE AN ERROR. COMMENT OUT ALL PACKAGE INSTALLATIONS.

Submit **both** the rmd and generated output files. Failing to submit both files will be subject to mark deduction. PDF or HTML is preferred.

## Sample Question and Solution

Use seq() to create the vector .

seq(3, 30, 2)

## [1] 3 5 7 9 11 13 15 17 19 21 23 25 27 29

seq(3, 29, 2)

## [1] 3 5 7 9 11 13 15 17 19 21 23 25 27 29

## Question 1 (32 points)

## Q1a (8 points)

Create and print a vector x with all integers from 15 to 100 and a vector y containing multiples of 5 in the same range. Hint: use seq()function. Calculate the difference in lengths of the vectors x and y. Hint: use length()

x <- seq(15, 100)  
y <- seq(15, 100, 5)  
length(x) - length(y)

## [1] 68

## Q1b (8 points)

Create a new vector, x\_square, with the square of elements at indices 1, 11, 21, 31, 41, 51, 61, and 71 from the variable x. Hint: Use indexing rather than a for loop. Calculate the mean and median of the FIRST five values from x\_square.

x\_square <- c(x[1],x[11], x[21], x[31], x[41], x[51], x[61], x[71])^2  
mean(x\_square[1:5])

## [1] 1425

median(x\_square[1:5])

## [1] 1225

## Q1c (8 points)

For a given factor variable of factorVar <- factor(c(10.8, 2.7, 5.0, 3.5)). To convert the factor to number, you need to either: 1) use level() to extract the level labels, then use as.numeric() to convert the labels to numbers, or 2) use as.charactor() to convert the values in the factorVar, then use as.numeric() to convert the values to numbers

Please provide both solutions

factorVar <- factor(c(10.8, 2.7, 5.0, 3.5))  
  
as.numeric(levels(factorVar))

## [1] 2.7 3.5 5.0 10.8

as.numeric(as.character(factorVar))

## [1] 10.8 2.7 5.0 3.5

## Q1d (8 points)

A comma-separated values file dataset.csv consists of missing values represented by Not A Number (null) and question mark (?). How can you read this type of files in R? NOTE: Please make sure you have saved the dataset.csv file at your current working directory.

read.csv("dataset.csv", na.strings=c("?", "null"))

## X1 X2 X3 X4 X5 X6 X7 X8 X9 X10  
## 1 11 12 13 14 15 16 17 18 19 20  
## 2 21 22 23 24 25 26 27 28 29 30  
## 3 31 32 33 34 35 36 37 38 39 40  
## 4 41 42 43 44 45 NA 47 48 49 50  
## 5 51 52 53 NA 55 56 57 NA 59 60  
## 6 61 62 63 64 65 66 67 68 69 70  
## 7 71 72 NA 74 75 76 77 78 79 80  
## 8 81 82 83 84 85 86 87 88 89 NA  
## 9 91 92 93 94 95 96 97 98 99 100  
## 10 NA 102 103 104 105 106 107 108 109 110  
## 11 111 112 113 114 115 116 117 118 119 120  
## 12 121 122 123 124 125 126 127 128 129 130  
## 13 131 132 133 134 135 136 137 138 139 NA  
## 14 141 142 143 144 145 146 147 148 149 150  
## 15 151 152 153 154 155 156 157 158 159 160  
## 16 161 162 163 164 NA 166 167 168 169 170

# Question 2 (32 points)

## Q2a (8 points)

Compute:

Hint: Use factorial(n) to compute .

sample\_func <- function(n){  
 3 ^ n  
}  
n\_max <- 40  
n\_min <- 10  
  
sequence\_of\_n <- c(n\_min: n\_max)  
  
(1/factorial(4)) \* sum(sample\_func(sequence\_of\_n))

## [1] 7.598541e+17

## Q2b (8 points)

Compute:

NOTE: The symbol represents multiplication.

result <- 1  
  
for (n in 1:20){  
 y <- 3\*n + 1/n  
   
 result <- result \* y  
}  
  
result

## [1] 1.373708e+28

## Q2c (8 points)

Describe what the following R command does: c(0:5)[NA]

c(0:5)[NA]

## [1] NA NA NA NA NA NA

Answer: It creates a vector or NAs of length 5

## Q2d (8 points)

Describe the purpose of is.vector(), is.character(), is.numeric(), and is.na() functions? Please use x <- c("a","b",NA,2) to explain your description.

x <- c("a","b",NA,2)  
  
is.vector(x)

## [1] TRUE

is.character(x)

## [1] TRUE

is.numeric(x)

## [1] FALSE

is.na(x)

## [1] FALSE FALSE TRUE FALSE

is.vector() determines whether object is vector is.character() determines whether object is of character data type is.numeric() determines whether object is of numeric data type is.na() determines whether object has null values

# Question 3 (36 points)

The airquality dataset contains daily air quality measurements in New York from May to September 1973. The variables include Ozone level, Solar radiation, wind speed, temperature in Fahrenheit, month, and day. Please see the detailed description using help("airquality").

help("airquality")

Install the airquality data set on your computer using the command install.packages("datasets"). Then load the datasets package into your session.

#library(datasets)

## Q3a (4 points)

Display the first 6 rows of the airquality data set.

head(airquality)

## Ozone Solar.R Wind Temp Month Day  
## 1 41 190 7.4 67 5 1  
## 2 36 118 8.0 72 5 2  
## 3 12 149 12.6 74 5 3  
## 4 18 313 11.5 62 5 4  
## 5 NA NA 14.3 56 5 5  
## 6 28 NA 14.9 66 5 6

## Q3b (8 points)

Compute the average of the first four variables (Ozone, Solar.R, Wind and Temp) for the fifth month using the sapply() function. Hint: You might need to consider removing the NA values; otherwise, the average will not be computed.

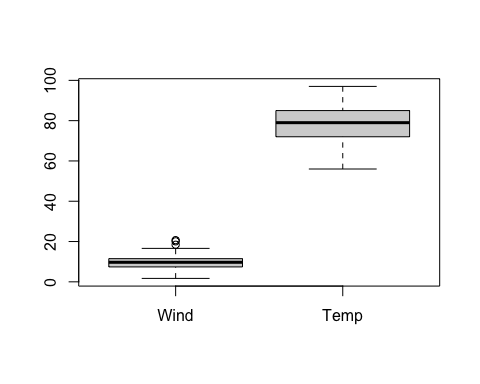
filtered <- na.omit(airquality[airquality$Month == 5, ])  
  
average <- sapply(filtered[, 1:4], mean)  
  
average

## Ozone Solar.R Wind Temp   
## 24.12500 182.04167 11.50417 66.45833

## Q3c (8 points)

Construct a boxplot for the all Wind and Temp variables, then display the values of all the outliers which lie beyond the whiskers.

boxplot(airquality$Wind, airquality$Temp, names = c("Wind", "Temp"))  
  
outliers <- boxplot(airquality$Wind, airquality$Temp, plot = FALSE)$out  
  
identify(outliers, labels = outliers)



## integer(0)

## Q3d (8 points)

Compute the upper quartile of the Wind variable with two different methods. HINT: Only show the upper quartile using indexing. For the type of quartile, please see <https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/quantile>.

# Method 1  
upper\_quartile1 <- quantile(airquality$Wind, probs = 0.75)  
upper\_quartile1

## 75%   
## 11.5

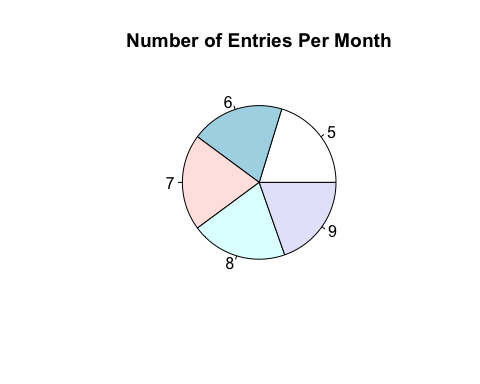
# Method 2  
sort\_wind <- sort(airquality$Wind)  
n <- length(sort\_wind)  
index <- ceiling(0.75 \* n)  
upper\_quartile2 <- sort\_wind[index]  
upper\_quartile2

## [1] 11.5

## Q3e (8 points)

Construct a pie chart to describe the number of entries by Month. HINT: use the table() function to count and tabulate the number of entries within a Month.

entries\_per\_month <- table(airquality$Month)  
  
pie(entries\_per\_month, labels = names(entries\_per\_month), main="Number of Entries Per Month")



END of Assignment #1.