CIND 123 - Data Analytics: Basic Methods

Ajay Herod

Assignment 1 (10%)

Ajay Herod

CIND123 D10 500764104

## 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 R 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.

Submit **both** the rmd and generated output files. Failing to submit both files will be subject to mark deduction.

## Sample Question and Solution

Use seq() to create the vector .

seq(1,10)

## [1] 1 2 3 4 5 6 7 8 9 10

## Question 1

1. Use the seq() function to create the vector . Note that each term in this sequence is of the form where .

seq(1, 61, by = 6)

## [1] 1 7 13 19 25 31 37 43 49 55 61

1. Use rep() to create the vector in which the sequence is repeated 5 times.

rep(2:4, 5)

## [1] 2 3 4 2 3 4 2 3 4 2 3 4 2 3 4

1. To convert factor to number, would it be correct to use the following commands? Explain your answer.

factorVar <- factor(c(1, 6, 5.4, 3.2));as.numeric(factorVar)

cat("No, factorvar is used to convert categorical variables into leveled integers. In our case we simply would like the double numbers to be converted to integers. Expected Output (1, 6, 5, 3) Actual Output (1, 4, 3, 2)")

## No, factorvar is used to convert categorical variables into leveled integers. In our case we simply would like the double numbers to be converted to integers. Expected Output (1, 6, 5, 3) Actual Output (1, 4, 3, 2)

1. A comma-separated values file dataset.csv consists of missing values represented by question marks (?) and exclamation mark (!). How can you read this type of files in R?

Dataframe <- read.csv("Dataframe.csv", sep = ",", na.strings = c("?","!"))

## Question 2

1. Compute:

sum(seq(10:100)^3)[1]

## [1] 17522596

1. Compute:

n <- 1:10  
sum((2^n/n^2)+(n^4/4^n))

## [1] 35.80589

1. Compute:

n <- 0:10  
sum(1/factorial(n+1))

## [1] 1.718282

1. Compute:

n <- 3:33  
prod(3\*n+(3/n^(1/3)))

## [1] 9.054339e+51

1. Explain the output of this R-command: c(0:5)[NA]

cat("c(0:5) by it self outputs 0, 1, 2, 3, 4, 5  
c(0:5)[n] gives us the nth number in the vector  
c(0:5)[NA] gives us NA NA NA NA NA NA  
This is because we are asking R to find the NAth number in the sequence. This makes no sense and instead R returns NA for each number in the vector")

## c(0:5) by it self outputs 0, 1, 2, 3, 4, 5  
## c(0:5)[n] gives us the nth number in the vector  
## c(0:5)[NA] gives us NA NA NA NA NA NA  
## This is because we are asking R to find the NAth number in the sequence. This makes no sense and instead R returns NA for each number in the vector

1. What is the difference between is.vector() and is.numeric() functions?

cat("is.vector tests if the argument is a vector or not and returns TRUE or False. I.E. Numeric, Character or Logical  
is.numeric tests if the argument is numeric or not and returns True or False. I.E. double or integer")

## is.vector tests if the argument is a vector or not and returns TRUE or False. I.E. Numeric, Character or Logical  
## is.numeric tests if the argument is numeric or not and returns True or False. I.E. double or integer

1. List at least three advantages and three disadvantages of using RShiny package?

cat("Rshiny package provides a web framework for building web applications using R  
The pros are Data Visualization, Open Source and Users just need to learn the basic R knowledge  
The cons are Poor support for high concurrency, Not so flexible for professional web developer, Less support for distributed computing")

## Rshiny package provides a web framework for building web applications using R  
## The pros are Data Visualization, Open Source and Users just need to learn the basic R knowledge  
## The cons are Poor support for high concurrency, Not so flexible for professional web developer, Less support for distributed computing

## Question 3

iris dataset gives the measurements in centimeters of the variables sepal length, sepal width, petal length and petal width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

Install the iris dataset on your computer using the command install.packages("datasets"). Then, load the datasets package into your session using the following command.

library(datasets)

1. Display the first six rows of the iris data set.

head(iris)

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 1 5.1 3.5 1.4 0.2 setosa  
## 2 4.9 3.0 1.4 0.2 setosa  
## 3 4.7 3.2 1.3 0.2 setosa  
## 4 4.6 3.1 1.5 0.2 setosa  
## 5 5.0 3.6 1.4 0.2 setosa  
## 6 5.4 3.9 1.7 0.4 setosa

1. Compute the average of the first four variables (Sepal.Length, Sepal.Width, Petal.Length and Petal.Width) using sapply() function.

Hint: You might need to consider removing the NA values, otherwise the average will not be computed.

sapply(iris[,-5] ,mean ,rm = NA)

## Sepal.Length Sepal.Width Petal.Length Petal.Width   
## 5.843333 3.057333 3.758000 1.199333

1. Show how to use R to replace the missing values in this dataset with plausible ones.

iris[is.na(iris)] <- 0

1. Compute the standard deviation for only the first and the third variables (Sepal.Length and Petal.Length)

sd(iris$Sepal.Length)

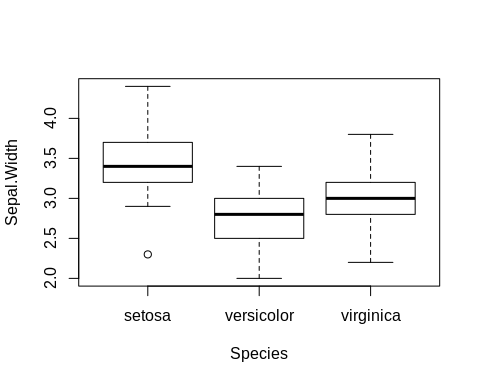
## [1] 0.8280661

sd(iris$Petal.Length)

## [1] 1.765298

1. Construct a boxplot for Sepal.Width variable, then display the values of all the outliers. Explain how these outliers have been calculated.

boxplot(Sepal.Width~Species,data = iris)



out <- boxplot.stats(iris$Sepal.Width)$out  
out\_ind <- which(iris$Sepal.Width %in% c(out))  
iris[out\_ind, ]

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 16 5.7 4.4 1.5 0.4 setosa  
## 33 5.2 4.1 1.5 0.1 setosa  
## 34 5.5 4.2 1.4 0.2 setosa  
## 61 5.0 2.0 3.5 1.0 versicolor

cat("The outliers are calcutlated using the interquartile range. The Sepal.Width above the q0.75+1.5\*IQR or below the q0.25-1.5\*IQR are considered outliers. THey can be identified by using the boxplot.stats(iris$Sepal.Width)$out.  
Then found using which(iris$Sepal.Width %in% c(out)) and displayed by using iris[out\_ind, ].")

## The outliers are calcutlated using the interquartile range. The Sepal.Width above the q0.75+1.5\*IQR or below the q0.25-1.5\*IQR are considered outliers. THey can be identified by using the boxplot.stats(iris$Sepal.Width)$out.  
## Then found using which(iris$Sepal.Width %in% c(out)) and displayed by using iris[out\_ind, ].

1. Compute the upper quartile of the Sepal.Width variable with two different methods.

summary(iris$Sepal.Width)

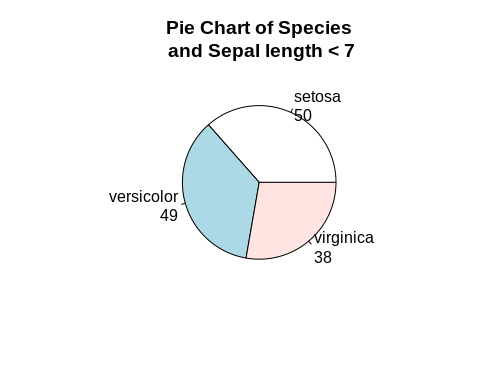
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 2.800 3.000 3.057 3.300 4.400

quantile(iris$Sepal.Width, .75)

## 75%   
## 3.3

1. Construct a pie chart to describe the species with ‘Sepal.Length’ less than 7 centimeters.

sepalLength <- subset(iris, iris$Sepal.Length < 7)  
mytable <- table(sepalLength$Species)  
lbls <- paste(names(mytable), "\n", mytable, sep="")  
pie(mytable, labels = lbls,  
 main="Pie Chart of Species\n and Sepal length < 7")



END of Assignment #1.