

Homework 01

Christina Lee

09/10/2021

1.

- a. Create two vectors named v1 and v2, where v1 is the sequence of integers from 2 to 6, and v2 is the sequence of integers from 5 to 9.

To answer this, I created vectors v1 and v2 and placed integers 2 to 6 in v1 and 5 to 9 for v2.

```
v1 <-2:6  
v1
```

```
## [1] 2 3 4 5 6
```

```
v2 <- 5:9  
v2
```

```
## [1] 5 6 7 8 9
```

- b. What is v2 minus v1?

To answer this, I subtracted v2 and v1 in R.

```
v2-v1
```

```
## [1] 3 3 3 3 3
```

- c. What is the inner product of v1 and v2?

To answer this, I used %*% to get the inner product of v1 and v2.

```
v1 %*% v2
```

```
##      [,1]  
## [1,] 150
```

- d. Replace the elements in v1+v2 that are greater than 10 with the number 0. Show that vector.

To answer this, I first added v1 and v2 and assigned the output of that to v3 which I got 7,9,11,13,15. As a result, 11,13,15 are all numbers greater than 10, therefore, I then reassigned [3:5] of v3 to 0.

```
v1 + v2
```

```
## [1] 7 9 11 13 15
```

```
v3 <- v1+v2  
v3
```

```
## [1] 7 9 11 13 15
```

```
v3[3:5] <- 0  
v3
```

```
## [1] 7 9 0 0 0
```

2.

a. Create a 5 by 5 matrix with the numbers 1 to 25 as its elements, and call it m1.

To answer this, I created a 5 x 5 matrix with `matrix()` and filled it in with numbers 1 to 25 and added the following arguments: `nrow=5` and `ncol=5` so it will give me a matrix with 5 rows and 5 columns = 5 x 5 and I named this matrix m1.

```
m1 <- matrix(1:25,nrow=5, ncol=5)
m1
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    6   11   16   21
## [2,]    2    7   12   17   22
## [3,]    3    8   13   18   23
## [4,]    4    9   14   19   24
## [5,]    5   10   15   20   25
```

b. What is m1 times v1?

To answer this, I used `%*%` to get the inner product of m1 and v1.

```
m1 %*% v1
```

```
##      [,1]
## [1,]  270
## [2,]  290
## [3,]  310
## [4,]  330
## [5,]  350
```

c. What is v1 times m1?

To answer this, I used `%*%` to get the inner product of v1 and m1.

```
v1 %*% m1
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   70  170  270  370  470
```

d. What is m1 times the transpose of m1?

To answer this, I used `%*%` to get the inner product of m1 and the transpose of m1. To get the transpose of m1, you simply use the `t()` function and place m1 in it.

```
m1 %*% t(m1)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]  855  910  965 1020 1075
## [2,]  910  970 1030 1090 1150
## [3,]  965 1030 1095 1160 1225
## [4,] 1020 1090 1160 1230 1300
## [5,] 1075 1150 1225 1300 1375
```

3.

a. Create a date frame with at least five rows and three columns. The first variable (column) should be dates, the second variable should be strings (characters), and the third variable should be numbers. Name each variable something appropriate and short.

To answer this, I created a data frame using the `data.frame()` function and called it “df”. Inside of this data frame, I put dates, States and COVID Cases of 2021-09-09 as my data. I also used `c()` to create the individual vectors. To change the type of data of dates from “character” to “date”, I used `as.Date()` function and added in the argument ‘%Y-%m-%d’ so the dates are properly formatted.

```
df <- data.frame(Dates = as.Date(c('2021-09-09', '2021-09-09', '2021-09-09', '2021-09-09', '2021-09-09'),
                                '%Y-%m-%d'),
                States = c("Massachusetts", "New York", "Maine", "New Jersey", "Florida"),
                COVID.Cases = c(2407, 6454, 620, 2210, 12386))
```

```
df
```

```
##      Dates      States COVID.Cases
## 1 2021-09-09 Massachusetts      2407
## 2 2021-09-09      New York      6454
## 3 2021-09-09        Maine        620
## 4 2021-09-09    New Jersey      2210
## 5 2021-09-09      Florida     12386
```

b. Use `str()` to show that that your data frame is appropriately structured.

To answer this, I used the `str()` function to see if everything in my data frame is structured correctly. All seems to be correct, in which, I have 5 obs. of 3 variables and each variable are in correct data type. Dates is “Date”, States is “chr”, and COVID.Cases is “num”.

```
str(df)
```

```
## 'data.frame':    5 obs. of  3 variables:
## $ Dates      : Date, format: "2021-09-09" "2021-09-09" ...
## $ States     : chr  "Massachusetts" "New York" "Maine" "New Jersey" ...
## $ COVID.Cases: num  2407 6454 620 2210 12386
```

c. Save it as a csv file, and then reload the data from the csv file.

To answer this, I used `write.csv()` function to save my data frame “df” to “HW01-COVID19-df.csv” so it will be saved as a csv file. I also added `row.names = FALSE` as an argument to get rid of the additional row # that R has generated beside the Dates so, the csv file will only have my three variables (Dates, States, COVID.Cases) with 5 rows of data that I put into it to make it look pretty and clean.

To reopen the saved csv file I used the `read.csv()` function and assigned a new name called “DF” to it in order to call & open the file.

```
write.csv(df, "HW01-COVID19-df.csv", row.names = FALSE)
DF <- read.csv("HW01-COVID19-df.csv")
DF
```

```
##      Dates      States COVID.Cases
## 1 2021-09-09 Massachusetts      2407
## 2 2021-09-09      New York      6454
## 3 2021-09-09        Maine        620
## 4 2021-09-09    New Jersey      2210
## 5 2021-09-09      Florida     12386
```

d. Create a new data frame that is just a subset of your data: the first, third, and last rows, and the first two variables.

To answer this, I created a subset of my data and created another data frame and called it “newdf”. To only extract the first, third, and last rows, and the first two variables from my original data frame, I used `c()` to get the 1,3,5 rows of data and 1:2 to get the first two columns/variables (Dates and States).

```
newdf <- data.frame(df[c(1,3,5), 1:2])
newdf
```

```
##      Dates      States
```

```
## 1 2021-09-09 Massachusetts
## 3 2021-09-09      Maine
## 5 2021-09-09      Florida
```

- e. Replace all the even numbers in the original data frame with 0.

To answer this, all numbers under COVID.Cases are even numbers except for the first number. So, all numbers except the 2407 should be 0. To reassign all even numbers, I used “\$” to extract the “COVID.Cases” column of my original data frame “df” and specified numbers that are in 2:5 to be 0.

```
df$COVID.Cases[2:5] <-0
df
```

```
##      Dates      States COVID.Cases
## 1 2021-09-09 Massachusetts      2407
## 2 2021-09-09      New York         0
## 3 2021-09-09      Maine           0
## 4 2021-09-09    New Jersey         0
## 5 2021-09-09      Florida         0
```

- f. Create a list with v1,v2,m1, and your data frame. Give all the items in that list a name. Now pick out the third item’s second item.

To answer this, I created a list with list() function that includes v1,v2,m1, and my original data frame “DF” that does not have its even numbers replaced with 0 and called it “hw01”. I gave all items in the list “hw01” a name with the names() function and called them V1,V2,M1,COVID19. I then ran hw01 and the output that I got was V1 as my first item, V2 as the second, M1 as the third, and COVID19 as the fourth item.

```
hw01 <- list(v1,v2,m1,DF)
names(hw01) <- c("V1","V2","M1", "COVID19")
hw01
```

```
## $V1
## [1] 2 3 4 5 6
##
## $V2
## [1] 5 6 7 8 9
##
## $M1
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    6   11   16   21
## [2,]    2    7   12   17   22
## [3,]    3    8   13   18   23
## [4,]    4    9   14   19   24
## [5,]    5   10   15   20   25
##
## $COVID19
##      Dates      States COVID.Cases
## 1 2021-09-09 Massachusetts      2407
## 2 2021-09-09      New York      6454
## 3 2021-09-09      Maine        620
## 4 2021-09-09    New Jersey      2210
## 5 2021-09-09      Florida     12386
```

To pick out the third item’s second item, I picked out M1’s column 2, which included numbers 6,7,8,9,10.

```
hw01[[3]][,2]
```

```
## [1] 6 7 8 9 10
```

4. Using latex equation notation in your .Rmd file, write out the quadratic formula, so that in your html file it looks pretty and like the version we all learned in high school. (Eg, see the box in the top right of this Wikipedia page: http://en.wikipedia.org/wiki/Quadratic_equation.)

To answer this, I followed the example given on Introduction to R 1.4: RMarkdown and Latex in module 1 and typed in the quadratic formula.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$