RWorksheet_Gener#3a.Rmd

Kent Gener

2023-10-22

R Markdown

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

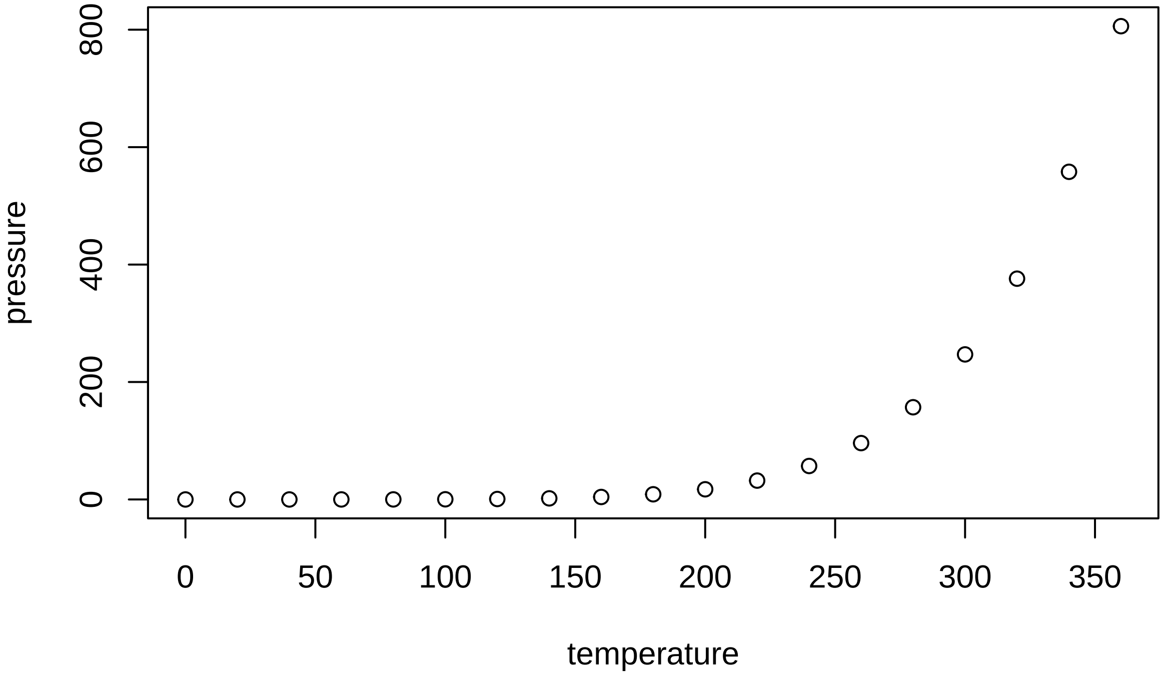
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
my_vector <- 1:10
my_vector
```

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

```
mean_value <- mean(my_vector)
mean(my_vector)
```

```
## [1] 5.5
```

Task 1 Using Vectors 1. There is a built-in vector LETTERS contains the uppercase letters of the alphabet and letters which contains the lowercase letters of the alphabet.

LETTERS [1] “A” “B” “C” “D” “E” “F” “G” “H” “I” “J” “K” “L” “M” “N” “O” “P” “Q” “R” “S” [20] “T” “U” “V” “W” “X” “Y” “Z”

letters [1] “a” “b” “c” “d” “e” “f” “g” “h” “i” “j” “k” “l” “m” “n” “o” “p” “q” “r” “s” [20] “t” “u” “v” “w” “x” “y” “z”

based on the above vector LETTERS: a. You need to produce a vector that contains the first 11 letters.

```
first_11_letters <- LETTERS[1:11]
first_11_letters
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K"
```

b. Produce a vector containing the odd-numbered letters.

```
odd_letters <- LETTERS
odd_letters
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S"
## [20] "T" "U" "V" "W" "X" "Y" "Z"
```

c. Produce a vector that contains the vowels

```
vowels <- c("A", "E", "I", "O", "U")
vowels
```

```
## [1] "A" "E" "I" "O" "U"
```

d. Produce a vector that contains the last 5 lowercase letters.

```
 last_5_lowercase_letters <- letters[22:26]
```

e. Produce a vector that contains letters between 15 to 24 in lowercase.

```
letters_15_to_24 <- letters[15:24]
letters_15_to_24
```

```
##  [1] "o" "p" "q" "r" "s" "t" "u" "v" "w" "x"
```

For task 2: a. Create a character vector for the cities: Tuguegarao City, Manila, Iloilo City, Tacloban, Samal Island, and Davao City, named “city.”

```
city <- c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban", "Samal Island", "Davao City")
city
```

```
## [1] "Tuguegarao City" "Manila"          "Iloilo City"     "Tacloban"       
## [5] "Samal Island"    "Davao City"
```

b. Create a numeric vector for the average temperatures, named “temp.”

```r
temp <- c(42, 39, 34, 34, 30, 27)
temp
```

```
## [1] 42 39 34 34 30 27
```

c. Create a data frame to combine the “city” and “temp.”

```r
weather_data <- data.frame(City = city, Temperature = temp)
weather_data
```

```
##                City Temperature
## 1 Tuguegarao City          42
## 2          Manila          39
## 3     Iloilo City          34
## 4        Tacloban          34
## 5    Samal Island          30
## 6      Davao City          27
```

d. Rename the columns using the names() function.

```r
names(weather_data) <- c("City", "Temperature")
weather_data
```

```
##                City Temperature
## 1 Tuguegarao City          42
## 2          Manila          39
## 3     Iloilo City          34
## 4        Tacloban          34
## 5    Samal Island          30
## 6      Davao City          27
```

e. Print the structure by using str() function. Describe the output.

```r
str(weather_data)
```

```
## 'data.frame':    6 obs. of  2 variables:
##  $ City       : chr  "Tuguegarao City" "Manila" "Iloilo City" "Tacloban" ...
##  $ Temperature: num  42 39 34 34 30 27
```

f. From the answer in d, what is the content of row 3 and row 4 What is its R code and its output?

g. From the answer in d, display the city with highest temperature and the city with the lowest temperature. What is its R code and its output?

```r
city_highest_temp <- weather_data[which.max(weather_data$Temperature), "City"]
city_lowest_temp <- weather_data[which.min(weather_data$Temperature), "City"]
cat("City with the highest temperature:", city_highest_temp, "\n")
```

```
## City with the highest temperature: Tuguegarao City
```

```r
cat("City with the lowest temperature:", city_lowest_temp, "\n")
```

```
## City with the lowest temperature: Davao City
```

Using Matrices • Matrix can be created by specifying the rows and columns. row = 2 matrix(c(5,6,7,4,3,2,1,2,3,7,8,9),nrow = 2) [,1] [,2] [,3] [,4] [,5] [,6] [1,] 5 7 3 1 3 8 [2,] 6 4 2 2 7 9 row = 3 and column = 2 atrix(data = c(3,4,5,6,7,8),3,2) [,1] [,2] [1,] 3 6 [2,] 4 7 [3,] 5 8

4

creating a diagonal matrix where x value will always be 1 diag(1,nrow = 6,ncol = 5) [,1] [,2] [,3] [,4] [,5] [1,] 1 0 0 0 0 [2,] 0 1 0 0 0 [3,] 0 0 1 0 0 [4,] 0 0 0 1 0 [5,] 0 0 0 0 1 [6,] 0 0 0 0 0 diag(6) [,1] [,2] [,3] [,4] [,5] [,6] [1,] 1 0 0 0 0 0 [2,] 0 1 0 0 0 0 [3,] 0 0 1 0 0 0 [4,] 0 0 0 1 0 0 [5,] 0 0 0 0 1 0 [6,] 0 0 0 0 0 1 2. Create a matrix of one to eight and eleven to fourteen with four columns and three rows.

a. What will be the R code for the #2 question and its result?

```
matrix_2a <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
matrix_2a
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    4    7   12
## [2,]    2    5    8   13
## [3,]    3    6   11   14
```

b. Multiply the matrix by two. What is its R code and its result?

```
matrix_2b <- matrix_2a * 2
matrix_2b
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    2    8   14   24
## [2,]    4   10   16   26
## [3,]    6   12   22   28
```

c. Content of row 2? What is its R code?

```
row_2_content <- matrix_2a[2, ]
row_2_content
```

```
## [1]  2  5  8 13
```

d. What will be the R code if you want to display the column 3 and column 4 in row 1 and row 2? What is its output?

```
cols_3_4_rows_1_2 <- matrix_2a[1:2, 3:4]
cols_3_4_rows_1_2
```

```
##      [,1] [,2]
## [1,]    7   12
## [2,]    8   13
```

e. What is the R code is you want to display only the columns in 2 and 3, row 3? What is its output?

```
cols_2_3_row_3 <- matrix_2a[3, 2:3]
cols_2_3_row_3
```

```
## [1]  6 11
```

f. What is the R code is you want to display only the columns 4? What is its output?

```
col_4 <- matrix_2a[, 4]
col_4
```

```
## [1] 12 13 14
```

g. Name the rows as isa, dalawa, tatlo and columns as uno, dos, tres, quatro for the matrix that was created in b.‘. What is its R code and corresponding output?

```
rownames(matrix_2a) <- c("isa", "dalawa", "tatlo")
colnames(matrix_2a) <- c("uno", "dos", "tres", "quatro")
matrix_2a
```

```
##         uno dos tres quatro
## isa       1   4    7     12
## dalawa    2   5    8     13
## tatlo     3   6   11     14
```

h. From the original matrix you have created in a, reshape the matrix by assigning a new dimension with dim(). New dimensions should have 2 columns and 6 rows. What will be the R code and its output?

```
new_dim <- matrix_2a
dim(new_dim) <- c(6, 2)
new_dim
```

```
##      [,1] [,2]
## [1,]    1    7
## [2,]    2    8
## [3,]    3   11
## [4,]    4   12
## [5,]    5   13
## [6,]    6   14
```

Using Arrays • Array can have more than two dimensions by using the array() function and dim() to specify the dimensions

6 creates a two-dimensional array containing numbers from 1 to 24 that have 3 rows and 4 columns array_dta <- array(c(1:24), c(3,4,2)) array_dta

, , 1

[,1] [,2] [,3] [,4] [1,] 1 4 7 10 [2,] 2 5 8 11 [3,] 3 6 9 12

, , 2

[,1] [,2] [,3] [,4] [1,] 13 16 19 22 [2,] 14 17 20 23 [3,] 15 18 21 24 checking for the dimensions row, column, dimension dim(array_dta)

[1] 3 4 2 checking for the number of elements length(array_dta)

[1] 24

- Another way to create arrays vectorA <- c(1:24) creating an array an_Array <- array(vectorA, dim = c(3,4,2)) an_Array

7

, , 1 [,1] [,2] [,3] [,4] [1,] 1 4 7 10 [2,] 2 5 8 11 [3,] 3 6 9 12

, , 2

[,1] [,2] [,3] [,4] [1,] 13 16 19 22 [2,] 14 17 20 23 [3,] 15 18 21 24

a. Create an array for the above numeric values. Each values will be repeated twice What will be the R code if you are to create a three-dimensional array with 4 columns and 2 rows. What will be its output?

```
array_values <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)
repeated_values <- rep(array_values, each = 2)
my_array <- array(repeated_values, dim = c(2, 4, 3))
my_array
```

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

```
##
## , , 2
##
##      [,1] [,2] [,3] [,4]
## [1,]    7    8    9    0
## [2,]    7    8    9    0
##
## , , 3
##
##      [,1] [,2] [,3] [,4]
## [1,]    3    4    5    1
## [2,]    3    4    5    1
```

#b. How many dimensions do your array have?

```
num_dimensions <- length(dim(my_array))
num_dimensions
```

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

#c. Name the rows as lowercase letters and columns as uppercase letters starting from the A. The array names should be “1st-Dimensional Array”, “2nd-Dimensional Array”, and “3rd-Dimensional Array”. What will be the R codes and its output?

```
dimnames(my_array) <- list(
  letters[1:2],
  LETTERS[1:4],
  c("Layer1", "Layer2", "Layer3")
)

dimnames(my_array) <- list("1st-Dimensional Array" = rownames(my_array),
                             "2nd-Dimensional Array" = colnames(my_array),
                             "3rd-Dimensional Array" = NULL)
my_array
```

```
## , , 1
##
##                      2nd-Dimensional Array
## 1st-Dimensional Array A B C D
##                     a 1 2 3 6
##                     b 1 2 3 6
##
## , , 2
##
##                      2nd-Dimensional Array
## 1st-Dimensional Array A B C D
##                     a 7 8 9 0
##                     b 7 8 9 0
##
## , , 3
##
##                      2nd-Dimensional Array
## 1st-Dimensional Array A B C D
##                     a 3 4 5 1
##                     b 3 4 5 1
```