R summary

2020年12月27日 星期日 10:20

Part 1 Intro R (R区分大小写)

只有数字不用加引号

1.Variable（变量）

Eg: my_var <- 4
x <- 42
可以直接加 variable

Eg: my_fruit = my_apples +my_oranges
输出形式：my_fruit

- Decimal values like 4.5 are called **numerics**.
- Whole numbers like 4 are called **integers**. Integers are also numerics.
- Boolean values (TRUE or FALSE) are called **logical**.
- Text (or string) values are called **characters**.

Eg: my_numeric <-42
my_character <-"university"
my_logical <- FALSE

Class(检查变量的数据类型）
Eg: Class(my_numeric)

2.Vector （向量）

Defination：A vector is a simple tool to store data.

Create a vector with the combine function c(),place the vector elements separated by a comma between parentheses.(在R中，使

用合并函数c（ ）创建一个向量。将向量元素放在括号之间，并以逗号分隔）

Eg: numeric_vector <- c(1,10,49)
character_vector <-c("a","b")
boolean_vector <- c(TRUE,FALSE,TRUE)

- Naming a vector(Names function)

Eg: some_vector <- c("John Doe","Poker player")
names(some_vector) <-c("Name","Profession")

Eg:

```
# Poker winnings from Monday to Friday
poker_vector <- c(140, -50, 20, -120, 240)

# Roulette winnings from Monday to Friday
roulette_vector <- c(-24, -50, 100, -350, 10)

# The variable days_vector
days_vector <- c("Monday", "Tuesday", "Wednesday", "Thursday",
"Friday")

# Assign the names of the day to roulette_vector and poker_vector
names(poker_vector) <-  c("Monday", "Tuesday", "Wednesday",
"Thursday", "Friday")
names(roulette_vector) <-c("Monday", "Tuesday", "Wednesday",
"Thursday", "Friday")
```

```
a <- c(1, 2, 3)
b <- c(4, 5, 6)
c <- a + b
```

```
A_vector <- c(1, 2, 3)
B_vector <- c(4, 5, 6)
total_vector <- A_vector + B_vector
```

Total:

```
Eg: total_poker <-sum(poker_vector)
total_roulette <-  sum(roulette_vector)

total_poker > total_roulette
```

- Vector selection

- select specific elements of the vector，use square brackets[ ]
  To select the second element of the vector, you type poker_vector[2], etc. Notice that the first element in a vector has index 1, not 0 as in many other programming languages.
- select multiple elements
  suppose you want to select the first and the fifth day of the week
  use the vector c(1, 5)
  Eg: Poker_vector[c(1,5)]
- c(2, 3, 4) can be abbreviated to2:4, which generates a vector with all natural numbers from 2 up to 4.
- Another way to tackle the previous exercise is by using the names of the vector elements (Monday, Tuesday, …) instead of their numeric positions.
  Eg: poker_vector["Monday"],
  poker_vector[c("Monday","Tuesday")]

  Selection by comparison
- c(4, 5, 6) > 5
  [1] FALSE FALSE TRUE
-
- # Which days did you make money on poker?
  selection_vector <- poker_vector > 0
  # Select from poker_vector these days
  poker_winning_days <- poker_vector[selection_vector]

3.Matrix（矩阵）

- 定义：矩阵是排列成固定数量的行和列的相同数据类型（数字，字符或逻辑）元素的集合。由于仅使用行和列，因此矩阵称为二维。
  Matrix(1:9,byrow=TRUE ,nrow =3)
  byrow 表示：矩阵由行填充，byrow=FALSE 矩阵由列填充
  nrow=3 表示：矩阵有三行

- Similar to vectors, you can add names for the rows and the columns of a matrix
  Eg:rownames(my_matrix) <- row_names_vector
  colnames(my_matrix) <- col_names_vector

  ```
  # Vectors region and titles, used for naming
  region <- c("US", "non-US")
  titles <- c("A New Hope", "The Empire Strikes Back", "Return o
  f the Jedi")
  # Name the columns with region
  colnames(star_wars_matrix)<-region
  # Name the rows with titles
  rownames(star_wars_matrix)<-titles
  ```

  Blind
  cbind（）函数将一列或多列添加到矩阵，该函数按列将矩阵和/或向量合并在一起。

  Eg: big_matrix <-cbind（matrix1，matrix2，vector1 ...）

  Add a row - rbind
  就像每个动作都有反应一样，每个cbind（）都有一个rbind ()

  Just like **cbind()** has **rbind()**,
  colSums() has **rowSums()**

  Selection of matrix elements
  use the square brackets [ ] to select one or multiple elements from a matrix.
  Eg:
  My_matrix[1,2] select the element at the first row and second row
  My_matrix[1:3,2:4] 包括1，2，3行 和2，3，4列数据

- my_matrix[,1]selects all elements of the first column.
  （选择第一列所有元素）
- my_matrix[1,] selects all elements of the first row.
  （选择第一行所有元素）

+,-,*,/　　2*My_matrix

4.Factor

两种类型：

1. Nominal categorical variable (名义类变量）：无顺序
   Eg:大象 长颈鹿 驴

```
factor_animals_vector ← factor(animals_vector)

factor_animals_vector

[1] Elephant Giraffe  Donkey   Horse
Levels: Donkey Elephant Giraffe Horse
```

2. Ordinal categorical variable （序别类变量）：有自然顺序 eg：高 中 低

3.

script.R　　Light Mode

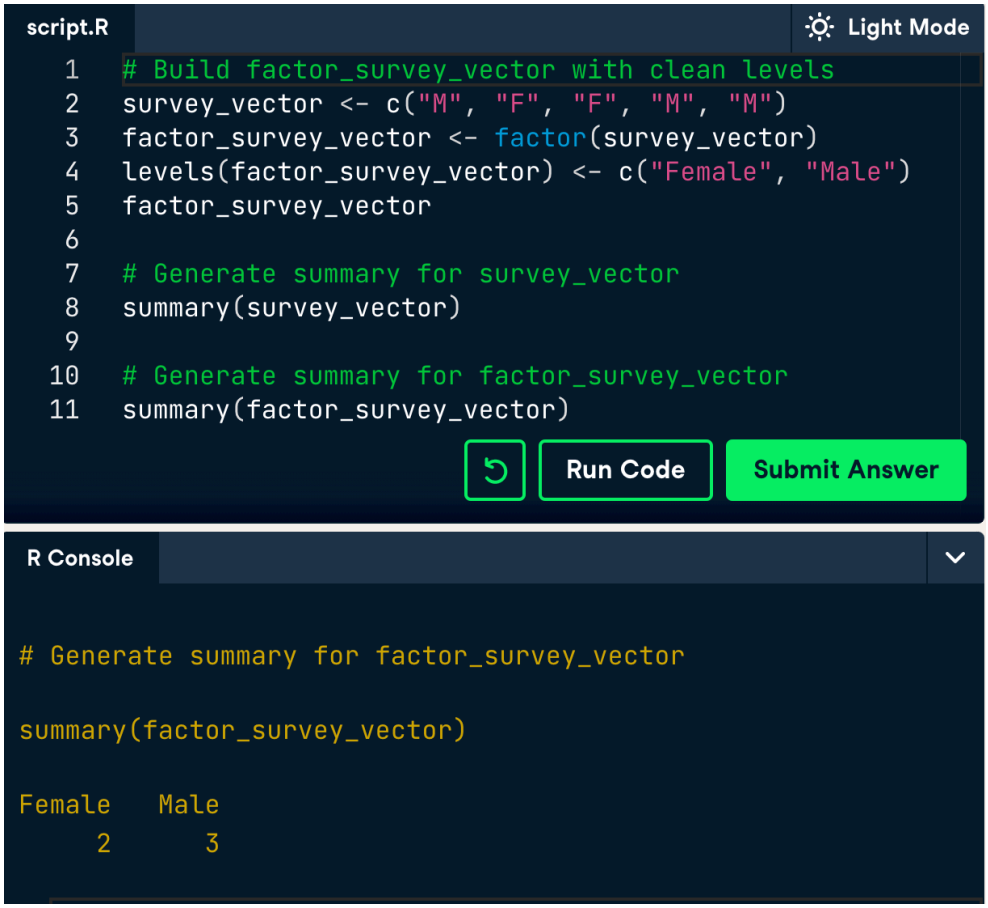
```r
# Animals
animals_vector <- c("Elephant", "Giraffe", "Donkey", "Horse")
factor_animals_vector <- factor(animals_vector)
factor_animals_vector

# Temperature
temperature_vector <- c("High", "Low", "High","Low", "Medium")
factor_temperature_vector <- factor(temperature_vector, order = TRUE, levels = c("Low", "Medium", "High"))
factor_temperature_vector
```

```
[1] High   Low    High   Low    Medium
Levels: Low < Medium < High
```

3. Factor levels
Level(factor_ vector)<- c("name1","name2",...)

Summary (总结 总数量)
Summary(my_var)



Order factors

```
factor(some_vector,
       ordered = TRUE,
       levels = c("lev1", "lev2" ...))
```

Eg:
#Create speed_ vector

Speed_ vector <-c("medium", "slow", " slow" ,
"medium", "fast")

"medium" , "fast")

#Convert speed_ vector to ordered factor vector

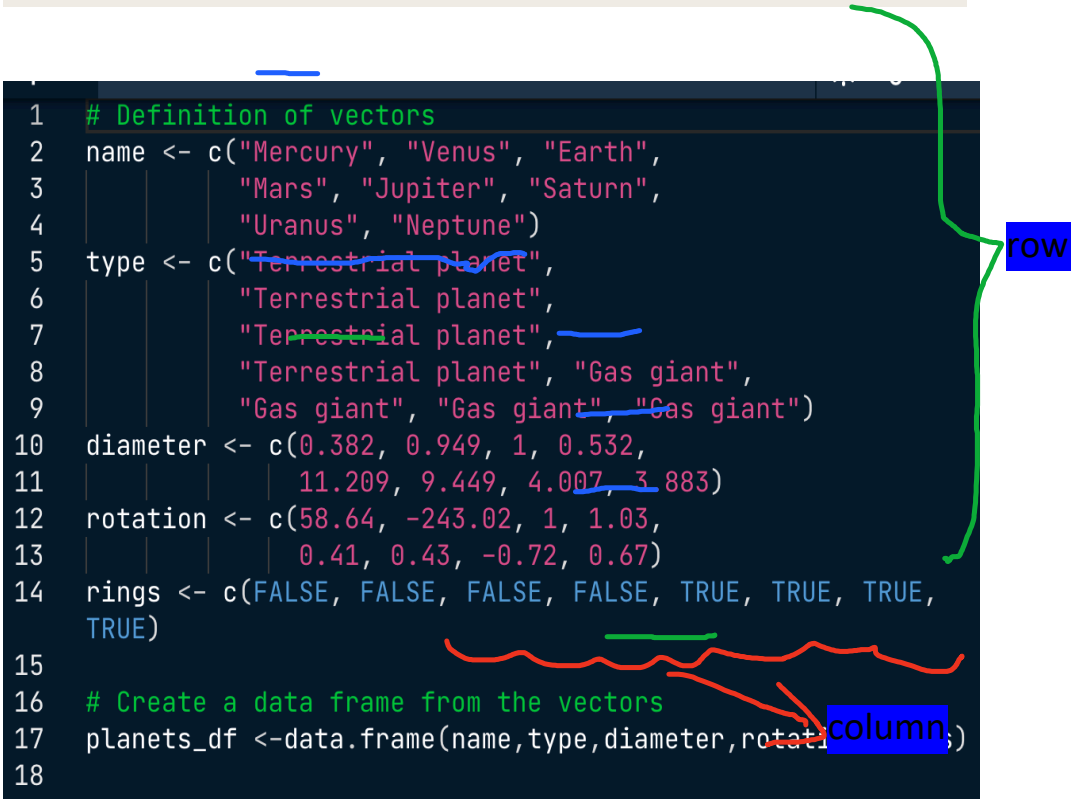
Factor_speed_vector <- factor (speed_vector ,
Ordered = TRUE ,levels =c("slow" ,"medium" ,"fast"))

#print factor_speed_vector

Factor_speed_vector
Summary (factor_speed_vector)

5. Data frames （数据帧）

Mtcars ：内置的演示用的数据集
head() first observations head(mtcars)
tail () last observation tail(mtcars)
str() 用于快速概览数据，用于展示数据集的结构




```r
# Definition of vectors
name <- c("Mercury", "Venus", "Earth",
          "Mars", "Jupiter", "Saturn",
          "Uranus", "Neptune")
type <- c("Terrestrial planet",
          "Terrestrial planet",
          "Terrestrial planet",
          "Terrestrial planet", "Gas giant",
          "Gas giant", "Gas giant", "Gas giant")
diameter <- c(0.382, 0.949, 1, 0.532,
              11.209, 9.449, 4.007, 3.883)
rotation <- c(58.64, -243.02, 1, 1.03,
              0.41, 0.43, -0.72, 0.67)
rings <- c(FALSE, FALSE, FALSE, FALSE, TRUE, TRUE, TRUE,
TRUE)

# Create a data frame from the vectors
planets_df <-data.frame(name,type,diameter,rotati
```

Use [ ] 选择行和列
Planets_ df [1,3] row1 ,column 3

```
# Print out diameter of Mercury (row 1, column 3)

planets_df[1,3]

[1] 0.382
```

Planets_ df [4,] entire fourth row

```
planets_df[4,]

  name                type diameter rotation rings
4 Mars Terrestrial planet    0.532     1.03 FALSE
```

Select first 5 values of diameter column
Planets_ df [1:5 ," diameter"]

```
# Select first 5 values of diameter column

planets_df[1:5,"diameter"]

[1]  0.382  0.949  1.000  0.532 11.209
```

如果column有名字的话 用$
Eg plannets_df$diameter

```
# Select the rings variable from planets_df
rings_vector <- planets_df$rings

# Print out rings_vector
rings_vector
```

```
# Select the rings variable from planets_df
rings_vector <- planets_df$rings
```

```
# Print out rings_vector
rings_vector
```

如果只是输入rings_ vector 会得到[1] FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE 但是可以通过输入 planets_df [rings_vector,]得到只有带光环的行星

```
# Adapt the code to select all columns for planets with rings
planets_df[rings_vector, ]
```

```
planets_df[rings_vector, ]

     name      type diameter rotation rings
5 Jupiter Gas giant   11.209     0.41  TRUE
6  Saturn Gas giant    9.449     0.43  TRUE
7  Uranus Gas giant    4.007    -0.72  TRUE
8 Neptune Gas giant    3.883     0.67  TRUE
```

Subset()

You should see the **subset()** function as a short-cut（快捷方式） to do e

格式：subset(my_df ,subset = some_condition)

Eg : subset (planets_df ,subset= rings)

```
# Select planets with diameter < 1
subset(planets_df,subset=diameter<1)
```

```
subset(planets_df,subset=diameter<1)
```

exactly

```
     name               type diameter rotation rings
1 Mercury Terrestrial planet    0.382    58.64 FALSE
2   Venus Terrestrial planet    0.949  -243.02 FALSE
4    Mars Terrestrial planet    0.532     1.03 FALSE
```

Order()

Eg: a<- c(100,10,1000)
Order (a)
[1] 2 1 3
a [order (a)]
[1] 10 100 1000

Diameter 从小到大排序

```
# planets_df is pre-loaded in your workspace

# Use order() to create positions
positions <- order(planets_df$diameter )

# Use positions to sort planets_df
planets_df[positions,]
```

```
planets_df[positions,]

     name               type diameter rotation rings
1 Mercury Terrestrial planet    0.382    58.64 FALSE
4    Mars Terrestrial planet    0.532     1.03 FALSE
2   Venus Terrestrial planet    0.949  -243.02 FALSE
3   Earth Terrestrial planet    1.000     1.00 FALSE
8 Neptune          Gas giant    3.883     0.67  TRUE
7  Uranus          Gas giant    4.007    -0.72  TRUE
6  Saturn          Gas giant    9.449     0.43  TRUE
```

```
5 Jupiter          Gas giant   11.209     0.41  TRUE
```

前三章summary

Vectors (one dimensional array): can hold numeric, character or logical values. The elements in a vector all have the same data type.

（向量（一维数组）：可以容纳数字，字符串或逻辑值。向量中的元素具有相同的数据类型）

Matrices (two dimensional array): can hold numeric, character or logical values. The elements in a matrix all have the same data type.

（矩阵（二维数组）：可以容纳数字，字符或逻辑值。矩阵中的元素都具有相同的数据类型）

Data frames (two-dimensional objects): can hold numeric, character or logical values.Within a column all elements have the same data type, but different columns can be of different data type.

数据框（二维对象）：可以容纳数字，字符或逻辑值。列中的所有元素都具有相同的数据类型，但是不同的列可以具有不同的数据类型）

list（ ）

my_list <- list(comp1,comp2....)

```r
# Vector with numerics from 1 up to 10
my_vector <- 1:10

# Matrix with numerics from 1 up to 9
my_matrix <- matrix(1:9, ncol = 3)
```

```r
# First 10 elements of the built-in data frame
mtcars
my_df <- mtcars[1:10,]

# Construct list with these different elements:
my_list <- list( my_vector,my_matrix,my_df)
```

My_list <- list(name1 = your_comp 1,

name 2 = your_comp2)

```r
# Vector with numerics from 1 up to 10
my_vector <- 1:10

# Matrix with numerics from 1 up to 9
my_matrix <- matrix(1:9, ncol = 3)

# First 10 elements of the built-in data frame
mtcars
my_df <- mtcars[1:10,]

# Adapt list() call to give the components names
my_list <- list(vec=my_vector,mat=my_matrix,
df=my_df)

# Print out my_list
my_list
```

```
my_list

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

$mat
     [,1] [,2] [,3]
```

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

$df
                  mpg cyl  disp  hp drat    wt  qsec vs
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0
```

[[]] 和 $ 都是可以表示column的名字的

Eg:

```
shining_list[["actors"]]
shining_list $ actors
```

要加引号

```
shining_list[["actors"]][2]
shining_list$actors[2]

shining_list[["reviews"]]
shining_list$reviews
```

Creating a new list for another movie

```
# Use the table from the exercise to define the comments and scores vect
ors
scores <- c(4.6, 5, 4.8, 5, 4.2)
comments <- c("I would watch it again", "Amazing!", "I liked it", "One of t
he best movies", "Fascinating plot")
# Save the average of the scores vector as avg_review
avg_review <- mean(scores)
# Combine scores and comments into the reviews_df data frame
reviews_df <-data.frame(scores,comments)
# Create and print out a list, called departed_list
departed_list <- list (movie_title,movie_actors,reviews_df,avg_review)
departed_list
```