

## RWorksheet\_Freires#4a

2024-10-23

1. The table below shows the data about shoe size and height. Create a data frame.

```
shoe_size <- c(6.5, 9.0, 8.5, 8.5, 10.5, 7.0, 9.5, 9.0, 13.0, 7.5, 10.5, 8.5, 12.0, 10.5, 13.0, 11.5, 8.5)
height <- c(66.0, 68.0, 64.5, 65.0, 70.0, 64.0, 70.0, 71.0, 72.0, 64.0, 74.5, 67.0, 71.0, 71.0, 77.0, 70.0, 65.0)
gender <- c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F")

data_table <- data.frame(Shoe_size = shoe_size, Height = height, Gender = gender)
print(data_table)
```

##	Shoe_size	Height	Gender
## 1	6.5	66.0	F
## 2	9.0	68.0	F
## 3	8.5	64.5	F
## 4	8.5	65.0	F
## 5	10.5	70.0	M
## 6	7.0	64.0	F
## 7	9.5	70.0	F
## 8	9.0	71.0	F
## 9	13.0	72.0	M
## 10	7.5	64.0	F
## 11	10.5	74.5	M
## 12	8.5	67.0	F
## 13	12.0	71.0	M
## 14	10.5	71.0	M
## 15	13.0	77.0	M
## 16	11.5	72.0	M
## 17	8.5	59.0	F
## 18	5.0	62.0	F
## 19	10.0	72.0	M
## 20	6.5	66.0	F
## 21	7.5	64.0	F
## 22	8.5	67.0	M
## 23	10.5	73.0	M
## 24	8.5	69.0	F
## 25	10.5	72.0	M
## 26	11.0	70.0	M
## 27	9.0	69.0	M
## 28	13.0	70.0	M

- a. Describe the data.
  - The data shows the table of Shoe sizes and Height of Male and Female genders.
- b. Create a subset by males and females with their corresponding shoe size and height. What its result? Show the R scripts.

```
males <- subset(data_table, Gender == "M", select = c(Shoe_size, Height))
print(males)
```

```
##      Shoe_size Height
## 5          10.5   70.0
## 9          13.0   72.0
## 11         10.5   74.5
## 13         12.0   71.0
## 14         10.5   71.0
## 15         13.0   77.0
## 16         11.5   72.0
## 19         10.0   72.0
## 22          8.5   67.0
## 23         10.5   73.0
## 25         10.5   72.0
## 26         11.0   70.0
## 27          9.0   69.0
## 28         13.0   70.0
```

```
females <- subset(data_table, Gender = "F", select = c(Shoe_size, Height))
```

```
## Warning: In subset.data.frame(data_table, Gender = "F", select = c(Shoe_size,
##      Height)) :
## extra argument 'Gender' will be disregarded
```

```
print(females)
```

```
##      Shoe_size Height
## 1           6.5   66.0
## 2           9.0   68.0
## 3           8.5   64.5
## 4           8.5   65.0
## 5          10.5   70.0
## 6           7.0   64.0
## 7           9.5   70.0
## 8           9.0   71.0
## 9          13.0   72.0
## 10          7.5   64.0
## 11         10.5   74.5
## 12          8.5   67.0
## 13         12.0   71.0
## 14         10.5   71.0
## 15         13.0   77.0
## 16         11.5   72.0
## 17          8.5   59.0
## 18          5.0   62.0
## 19         10.0   72.0
## 20          6.5   66.0
## 21          7.5   64.0
## 22          8.5   67.0
## 23         10.5   73.0
## 24          8.5   69.0
## 25         10.5   72.0
## 26         11.0   70.0
## 27          9.0   69.0
```

```
## 28      13.0    70.0
```

C. Find the mean of shoe size and height of the respondents. Write the R scripts and its result.

```
mean_shoesize <- mean(shoe_size)
print(mean_shoesize)
```

```
## [1] 9.410714
```

```
mean_height <- mean(height)
print(mean_height)
```

```
## [1] 68.57143
```

d. Is there a relationship between shoe size and height? Why?

- In my conclusion, there is a relationship because the data has shown that most tall people have bigger shoe sizes.

2. Construct character vector months to a factor with factor() and assign the result to factor\_months\_vector. Print out factor\_months\_vector and assert that R prints out the factor levels below the actual values. Consider data consisting of the names of months:

“March”, “April”, “January”, “November”, “January”, “September”, “October”, “September”, “November”, “August”, “January”, “November”, “November”, “February”, “May”, “August”, “July”, “December”, “August”, “August”, “September”, “November”, “April”

```
months_vector <- c("March", "April", "January", "November", "January",
"September", "October", "September", "November", "August", "January", "November", "November", "February", "May")
print(months_vector)
```

```
## [1] "March"      "April"      "January"    "November"   "January"    "September"
## [7] "October"    "September"  "November"   "August"     "January"    "November"
## [13] "November"   "February"   "May"        "August"     "July"       "December"
## [19] "August"     "August"     "September"  "November"   "February"   "April"
```

```
factor_months_vector <- factor(months_vector)
print(factor_months_vector)
```

```
## [1] March      April       January     November    January     September   October
## [8] September   November    August      January     November    November    February
## [15] May         August      July        December    August      August      September
## [22] November    February    April
## 11 Levels: April August December February January July March May ... September
```

3. Then check the summary() of the months\_vector and factor\_months\_vector. | Interpret the results of both vectors. Are they both equally useful in this case?

```
summary(months_vector)
```

```
##      Length      Class      Mode
##         24 character character
```

```
summary(factor_months_vector)
```

```
##      April      August  December  February   January      July      March      May
##         2         4         1         2         3         1         1         1
## November    October  September
##         5         1         3
```

- My interpretation of the results of months vector is that it shows the length, class, and mode functions. wherein the length is 24, the class is character and the mode is also a character. The results of the

factor months vector is showing each month and their levels. Both summary are useful in this case as it has its own function.

4. Create a vector and factor for the table below.

- Note: Apply the factor function with required order of the level. `new_order_data <- factor(factor_data, levels = c("East", "West", "North"))` `print(new_order_data)`

```
direction <- c("East", "West", "North")
```

```
frequency <- c(1,4,3)
```

```
data <- data.frame(direction, frequency)
print(data)
```

```
## direction frequency
## 1      East         1
## 2      West         4
## 3     North         3
```

```
new_data <- factor(direction, levels = c("East", "West", "North"))
print(new_data)
```

```
## [1] East West North
## Levels: East West North
```

5. Enter the data below in Excel with file name = import\_march.csv

a. Import the excel file into the Environment Pane using `read.table()` function. Write the code.

```
library(readxl)
excelData <- read.table("/cloud/project/Worksheet#4/import_march.csv", header = TRUE, sep = ",")
```

b. View the dataset. Write the R scripts and its result.

```
print(excelData)
```

```
## Students Strategy.1 Strategy.2 Strategy.3
## 1      Male         8         10         8
## 2      Male         4          8         6
## 3      Male         0          6         4
## 4    Female        14          4        15
## 5    Female        10          2        12
## 6    Female         6          0         9
```

Using Conditional Statements (IF-ELSE)

6. Full Search

Exhaustive search is a methodology for finding an answer by exploring all possible cases. When trying to find a desired number in a set of given numbers, the method of finding the corresponding number by checking all elements in the set one by one can be called an exhaustive search. Implement an exhaustive search function that meets the input/output conditions below.

- a. Create an R Program that allows the User to randomly select numbers from 1 to 50. Then display the chosen number. If the number is beyond the range of the selected choice, it will have to display a string "The number selected is beyond the range of 1 to 50". If number 20 is inputted by the User, it will have to display "TRUE", otherwise display the input number.

```
num <- readline(prompt = "Enter a number from 1 to 50: ")
```

```
## Enter a number from 1 to 50:
```

```
if(num < 1 || num > 50) {  
  paste("The number selected is beyond the range of 1 to 50")  
}else if(num == 20) {  
  print("TRUE")  
} else{  
  print(num)  
}
```

```
## [1] "The number selected is beyond the range of 1 to 50"
```