Worksheet 4a

Elijah June G. Tuares

2024-10-22

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

a. Describe the data.

```
household <- 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
  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, 7
  Gender = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F", "
)

print(household)
```

```
##    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
```

b. Create a subset by males and females with their corresponding shoe size and height. What its result? Show the R scripts.

```
males <- household[household$Gender == "M",]
females <- household[household$Gender == "F",]

print(males)
```

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

```
print(females)
```

```
##    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
## 6        7.0   64.0      F
## 7        9.5   70.0      F
## 8        9.0   71.0      F
## 10       7.5   64.0      F
## 12       8.5   67.0      F
## 17       8.5   59.0      F
## 18       5.0   62.0      F
## 20       6.5   66.0      F
## 21       7.5   64.0      F
## 24       8.5   69.0      F
```

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

```
mean(household$Shoe_size)
```

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

```
mean(household$Height)
```

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

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

Yes

```
cor(household$Shoe_size, household$Height)
```

```
## [1] 0.7766089
```

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.

```
months <- c("March", "April", "January", "November", "January",
            "September", "October", "September", "November",
            "August", "January", "November", "November",
            "February", "May", "August", "July",
            "December", "August", "August", "September",
            "November", "February", "April")

factor_months_vector <- factor(months)

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
```

```
levels(factor_months_vector)
```

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

```
direction <- c("East", "West", "North", "West", "North")
frequency <- c(1, 4, 3, 2, 1)
factor_direction <- factor(direction, levels = c("East", "West", "North"))
print(factor_direction)
```

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

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

```
factor_data <- c("East", "West", "North")
frequency_vector <- c(1, 4, 3)
new_order_data <- factor(factor_data,levels = c("East","West","North"))
print(new_order_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.

```
data <- read.table("import_march.csv", header = TRUE, sep = ",")
```

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

```
data
```

```
##   Students Strategy.1 Strategy.2 Strategy.3
## 1     Male          8         10          8
## 2                   4          8          6
## 3                   0          6          4
```

```
## 4   Female         14          4         15
## 5                  10          2         12
## 6                   6          0          9
```

6. Full Search

```
exhaustive_search <- function() {
  # Get input from the user and check if it's a valid number
  number <- suppressWarnings(as.integer(readline(prompt = "Please select a number between 1 and 50: ")))
  if (is.na(number)) {
    print("Invalid input. Please enter a number.")
  } else if (number < 1 || number > 50) {
    print("The number selected is beyond the range of 1 to 50")
  } else if (number == 20) {
    print(TRUE)
  } else {
    print(number)
  }
}
exhaustive_search()
```

```
## Please select a number between 1 and 50:
## [1] "Invalid input. Please enter a number."
```

7. Change

```
min_bills <- function(price) {
  bills <- c(1000, 500, 200, 100, 50)
  count <- 0
  for (bill in bills) {

    while (price >= bill) {
      price <- price - bill
      count <- count + 1
    }
  }
  return(count)
}
snack_price <- 2700
cat("Minimum number of bills needed:", min_bills(snack_price), "\n")
```

```
## Minimum number of bills needed: 4
```

8. The following is each student's math score for one semester. Based on this, answer the following questions.

a. Create a dataframe from the above table. Write the R codes and its output.

```
grades <- data.frame(
  Name = c("Annie", "Thea", "Steve", "Hanna"),
  Grade1 = c(85, 65, 75, 95),
  Grade2 = c(65, 75, 55, 75),
  Grade3 = c(85, 90, 80, 100),
  Grade4 = c(100, 90, 85, 90)
)
print(grades)
```

```
##    Name Grade1 Grade2 Grade3 Grade4
```

```
## 1 Annie     85     65     85    100
## 2  Thea     65     75     90     90
## 3 Steve     75     55     80     85
## 4 Hanna     95     75    100     90
```

b. Without using the rowMean function, output the average score of students whose average math score over 90 points during the semester. write R code and its output.

```
for (i in 1:nrow(grades)) {
  avg <- sum(grades[i, 2:5]) / 4
  print(paste(grades$Name[i], "'s average grade this semester is", avg))
}
```

```
## [1] "Annie 's average grade this semester is 83.75"
## [1] "Thea 's average grade this semester is 80"
## [1] "Steve 's average grade this semester is 73.75"
## [1] "Hanna 's average grade this semester is 90"
```

c. Without using the mean function, output as follows for the tests in which the average score was less than 80 out of 4 tests.

```
for (j in 2:5) {
  avg_test <- mean(grades[,j])
  if (avg_test < 80) {
    print(paste("The", j-1, "test was difficult with an average score of", avg_test))
  }
}
```

```
## [1] "The 2 test was difficult with an average score of 67.5"
```

d. Without using the max function, output as follows for students whose highest score for a semester exceeds 90 points.

```
for (i in 1:nrow(grades)) {
  highest_score <- grades[i, 2]
  for (j in 3:5) {
    if (grades[i, j] > highest_score) {
      highest_score <- grades[i, j]
    }
  }
  if (highest_score > 90) {
    print(paste(grades$Name[i], "'s highest grade this semester is", highest_score))
  }
}
```

```
## [1] "Annie 's highest grade this semester is 100"
## [1] "Hanna 's highest grade this semester is 100"
```