

Work sheet #4b

Kenan Jake Jimenez

2025-10-13

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

```
shoe <- 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,
       13.0,11.5,8.5,5.0,10.0,6.5,8.5,10.5,6.5,10.5,11.0,9.0,13.0)

height <- c(66.0,68.0,64.5,65.0,70.0,64.0,70.0,71.0,72.0,
          64.0,74.5,67.0,71.0,71.0,77.0,72.0,59.0,62.0,72.0,66.0,64.0,
          73.0,69.0,72.0,71.0,69.0,70.0)

gender <- c("F","F","F","M","M","F","F","F","M","M","F","M",
           "M","F","F","M","M","F","F","M","M","M","M","M")

df <- data.frame(shoe, height, gender)
df
```

```
##      shoe 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     M
## 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     F
## 10    7.5    64.0     M
## 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     F
## 17    8.5    59.0     F
## 18    5.0    62.0     M
## 19   10.0    72.0     M
## 20    6.5    66.0     F
## 21    8.5    64.0     F
## 22   10.5    73.0     M
## 23    6.5    69.0     M
## 24   10.5    72.0     M
## 25   11.0    71.0     M
```

```
## 26 9.0 69.0 M
## 27 13.0 70.0 M
```

#a. Describe the data.

#The dataset contains 27 respondents, and each person has three variables #recorded: shoe size, height, and gender. Shoe size and height are numerical #variables, while gender is a categorical factor with two levels #(Male and Female). The data reflects basic anthropometric measurements #commonly used to compare physical characteristics across groups.

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

```
male_data <- subset(df, gender == "M", select = c(shoe, height, gender))
female_data <- subset(df, gender == "F", select = c(shoe, height, gender))

male_data
```

```
##      shoe height gender
## 4     8.5   65.0    M
## 5    10.5   70.0    M
## 10   7.5   64.0    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
## 18   5.0    62.0    M
## 19   10.0   72.0    M
## 22   10.5   73.0    M
## 23   6.5    69.0    M
## 24   10.5   72.0    M
## 25   11.0   71.0    M
## 26   9.0    69.0    M
## 27  13.0   70.0    M
```

```
female_data
```

```
##      shoe height gender
## 1     6.5   66.0    F
## 2     9.0   68.0    F
## 3     8.5   64.5    F
## 6     7.0   64.0    F
## 7     9.5   70.0    F
## 8     9.0   71.0    F
## 9    13.0   72.0    F
## 12   8.5   67.0    F
## 16   11.5   72.0    F
## 17   8.5   59.0    F
## 20   6.5   66.0    F
## 21   8.5   64.0    F
```

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

```

mean_shoe <- mean(df$Shoe_size)

## Warning in mean.default(df$Shoe_size): argument is not numeric or logical:
## returning NA

mean_height <- mean(df$Height)

## Warning in mean.default(df$Height): argument is not numeric or logical:
## returning NA

mean_shoe

## [1] NA

mean_height

## [1] NA

```

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

Yes. In general, larger shoe sizes tend to appear with greater heights in the dataset. This relationship is expected because shoe size is partly correlated with body proportions, and taller individuals typically have larger feet. While this dataset shows a pattern, a statistical method such as correlation analysis would confirm the strength of the relationship more rigorously.

#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_vector <- 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_vector)
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
```

#No. The character vector simply stores text and cannot summarize categories. #The factor version groups identical values and provides counts, which is more useful for analysis especially for categorical data like months. #Therefore, the factor is much more useful when studying frequency, patterns, or performing statistical analysis.

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

```
factor_data <- c("East", rep("West", 4), rep("North", 3))
factor_data
```

```
## [1] "East"   "West"   "West"   "West"   "West"   "North"  "North"  "North"
```

```
new_order_data <- factor(factor_data, levels = c("East", "West", "North"))
print(new_order_data)
```

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

#5. Enter the data below in Excel with file name = import_march.csv

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

```
##   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
```

#Using Conditional Statements (IF-ELSE) #6. Full Search

```
num <- as.integer(readline("Select a number from 1 to 50: "))
```

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

```
cat("Chosen number:", num, "\n")
```

```
## Chosen number: NA
```

```

if (is.na(num)) {
  cat("Invalid input. Please enter a number.\n")

} else if (num < 1 || num > 50) {
  cat("The number selected is beyond the range of 1 to 50\n")

} else if (num == 20) {
  print(TRUE)

} else {
  print(num)
}

```

Invalid input. Please enter a number.

#7. Change

```

min_bills <- function(price) {
  bills <- c(1000, 500, 200, 100, 50)

  remaining <- price
  count <- 0

  for (b in bills) {
    if (remaining >= b) {
      count <- count + (remaining %/% b)
      remaining <- remaining %% b
    }
  }

  return(count)
}

```

#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.

```

students <- 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)
)

students

```

| | 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.
```

```
for (i in 1:nrow(students)) {  
  avg <- (students$Grade1[i] + students$Grade2[i] +  
           students$Grade3[i] + students$Grade4[i]) / 4  
  
  if (avg > 90) {  
    cat(students>Name[i], "'s average grade this semester is ", avg, ".\n", sep="")  
  }  
}
```

```
#c.
```

```
tests <- students[, 2:5]  
  
for (j in 1:4) {  
  total <- sum(tests[, j])  
  avg <- total / nrow(tests)  
  
  if (avg < 80) {  
    cat("The", j, "th test was difficult.\n")  
  }  
}
```

```
## The 2 th test was difficult.
```

```
#d
```

```
for (i in 1:nrow(students)) {  
  grades <- students[i, 2:5]  
  highest <- grades[1]  
  
  for (g in grades) {  
    if (g > highest) highest <- g  
  }  
  
  if (highest > 90) {  
    cat(students>Name[i], "'s highest grade this semester is ", highest, ".\n", sep="")  
  }  
}
```

```
## Annie's highest grade this semester is 100.  
## Hanna's highest grade this semester is 100.
```