

RWorksheet_Benedicto#4a.R

Miguel F. Josue Jr

2024-10-14

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

```
data <- 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, 72.0),
  Gender = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F", "F")
)
data
```

##	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 table provides a summary of data about individuals' shoe size, height, and gender. Whereas shoe size and height are in a numeric data types since they are measurements while the gender(F and M) correspond to the initial characters of male and female. The table implies that there is a correlation between these variables.

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

- The R scripts displays the gender-based shoe size and height data, through the “select” function we are able to only view only the Shoe Size and Height, this is used in order to avoid redundancy.

```
male_data <- subset(data, Gender == "M", select = c(Shoe_Size, Height))
```

```
male_data
```

```
##      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
```

```
female_data <- subset(data, Gender == "F", select = c(Shoe_Size, Height))
```

```
female_data
```

```
##      Shoe_Size Height
## 1           6.5   66.0
## 2           9.0   68.0
## 3           8.5   64.5
## 4           8.5   65.0
## 6           7.0   64.0
## 7           9.5   70.0
## 8           9.0   71.0
## 10          7.5   64.0
## 12          8.5   67.0
## 17          8.5   59.0
## 18          5.0   62.0
## 20          6.5   66.0
## 21          7.5   64.0
## 24          8.5   69.0
```

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

```
mean_shoe_size <- mean(data$Shoe_Size)
```

```
mean_shoe_size
```

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

```
mean_height <- mean(data$Height)
mean_height
```

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

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

- There is indeed a relationship between shoe size and height, as the height increases the shoe size also increase relatively. It is also observed in the data that men have a significant gap in both said variables compared to women.

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)
```

```
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? The summary(month_vector) presents the length and other generic properties of the vector like Class and mode, while the summary(factor_months_vector) counts the occurrences of each month. Both these summaries provide significant data, one shows the structure of the vector while the other is more on classifying categorical data.

```
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
```

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

```
direction_vector <- rep(c("East", "West", "North"),c(1,4,3))

factor_data <- factor(direction_vector)

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

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

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

- Import the excel file into the Environment Pane using read.table() function. Write the code.
- View the dataset. Write the R scripts and its result.

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

```
## Students Strategy.1 Strategy.2 Strategy3
## 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
```