

HW2_jaeyounglee

Jaeyoung Lee

September 14, 2020

Problem 3

First of all, it is to handle a mistake. Also, one can handle various versions of a code and see the history of a code. Furthermore, using version control, it is easy to collaborate with others. This is because one can share a code and work on the cloud such as GitHub.

Problem 4

For each dataset, you should perform the cleaning 2x: first with base R functions (ie no dplyr, piping, etc), second using tidyverse function. Make sure you weave your code and text into a complete description of the process and end by creating a tidy dataset describing the variables, create a summary table of the data (summary, NOT full listing), note issues with the data, and include an informative plot.

- a. Sensory data from five operators. <http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat>

```
##### Sensory data #####
# Getting "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
url_sensory <- "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
sensory_rawdata <- fread(url_sensory, fill = TRUE, skip = 2, data.table = FALSE)
saveRDS(sensory_rawdata, 'sensory_rawdata.RDS')
sensory_rawdata <- readRDS('sensory_rawdata.RDS')
```

The raw data has missing values and the categories “Items” are in the data. We need to remove them from the data.

```
# Tidy data with base R function
matrix_sensory <- t(as.matrix(sensory_rawdata)) # Transpose the raw data
na <- which(is.na(matrix_sensory==TRUE))      # Find the indexes of Missing value

# The indexes where Item numbers are in the data
x <- 1
item <- x
for (i in 1:9){
  x <- x+18
  item <- c(item, x)
}

# Remove missing values and Item numbers from the data
# To focus on items, transpose the data table and rename the column names
sensory_data <- t(matrix(matrix_sensory[-c(na,item)], byrow = T, nrow = 10))
sensory_data <- data.table(sensory_data)
colnames(sensory_data) <- paste('Item', 1:10)
```

```
Opr <- rep(paste('Opr', 1:5), 3)
sensory_data <- cbind(Opr,sensory_data)
sensory_data <- sensory_data[order(sensory_data$Opr)]
```

Above is the converted tidy data frames using the base R functions. A summary of the data is as follows:

Opr	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	Item 7	Item 8	Item 9	Item 10
Length:15	Min. :3.300	Min. :4.200	Min. :1.300	Min. :5.90	Min. :4.90	Min. :1.100	Min. :0.700	Min. :3.000	Min. :6.700	Min. :2.80
Class	1st	1st	1st	1st	1st	1st	1st	1st	1st	1st
:character	Qu.:4.050	Qu.:4.700	Qu.:2.350	Qu.:6.40	Qu.:5.70	Qu.:1.750	Qu.:1.000	Qu.:4.400	Qu.:7.950	Qu.:3.90
Mode	Median	Median	Median	Median	Median	Median	Median	Median	Median	Median
:character	:4.400	:5.300	:2.600	:6.90	:5.90	:2.100	:1.200	:4.600	:8.800	:4.80
NA	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean
	:4.467	:5.313	:2.773	:6.88	:5.92	:2.393	:1.407	:4.427	:8.467	:4.52
NA	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd
	Qu.:5.100	Qu.:5.950	Qu.:3.050	Qu.:7.20	Qu.:6.15	Qu.:3.150	Qu.:1.550	Qu.:4.800	Qu.:9.000	Qu.:5.10
NA	Max. :5.700	Max. :6.300	Max. :4.600	Max. :8.20	Max. :7.00	Max. :4.000	Max. :3.100	Max. :4.900	Max. :9.400	Max. :5.50

```
# Sensory data with tidyverse package
matrix_sensory <- sensory_rawdata %>% as.matrix() %>% t() # Transpose the raw data
na <- which(is.na(matrix_sensory==TRUE)) # Find the indexes of Missing value

# The indexes where Item numbers are in the data
x <- 1
item <- x
for (i in 1:9){
  x <- x+18
  item <- c(item, x)
}

# Remove missing values and Item numbers from the data
# To focus on items, transpose the data table and rename the column names
sensory_data <- matrix_sensory[-c(na,item)] %>% matrix(byrow = T, nrow = 10) %>% t()
sensory_data <- data.table(sensory_data)
Opr <- rep(paste('Opr', 1:5), 3)
sensory_data <- bind_cols(Opr,sensory_data)

## New names:
## * NA -> ...1

colnames(sensory_data) <- c('Opr',paste('Item', 1:10))
sensory_data <- sensory_data[order(sensory_data$Opr)]
```

The summary of the data converted by tidyverse is as follows.

Opr	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	Item 7	Item 8	Item 9	Item 10
Length:15	Min. :3.300	Min. :4.200	Min. :1.300	Min. :5.90	Min. :4.90	Min. :1.100	Min. :0.700	Min. :3.000	Min. :6.700	Min. :2.80
Class	1st	1st	1st	1st	1st	1st	1st	1st	1st	1st
:character	Qu.:4.050	Qu.:4.700	Qu.:2.350	Qu.:6.40	Qu.:5.70	Qu.:1.750	Qu.:1.000	Qu.:4.400	Qu.:7.950	Qu.:3.90

Opr	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	Item 7	Item 8	Item 9	Item 10
Mode	Median	Median	Median	Median	Median	Median	Median	Median	Median	Median
:character	:4.400	:5.300	:2.600	:6.90	:5.90	:2.100	:1.200	:4.600	:8.800	:4.80
NA	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean
	:4.467	:5.313	:2.773	:6.88	:5.92	:2.393	:1.407	:4.427	:8.467	:4.52
NA	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd
	Qu.:5.100	Qu.:5.950	Qu.:3.050	Qu.:7.20	Qu.:6.15	Qu.:3.150	Qu.:1.550	Qu.:4.800	Qu.:9.000	Qu.:5.10
NA	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.
	:5.700	:6.300	:4.600	:8.20	:7.00	:4.000	:3.100	:4.900	:9.400	:5.50

b. Gold Medal performance for Olympic Men's Long Jump, year is coded as 1900=0.

<http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat>

```
##### Long Jump data #####
```

```
# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
```

```
url_medal <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
```

```
medal_rawdata <- fread(url_medal)
```

```
## Warning in fread(url_medal): Detected 12 column names but the data has 8
```

```
## columns. Filling rows automatically. Set fill=TRUE explicitly to avoid this
```

```
## warning.
```

```
saveRDS(medal_rawdata, 'medal_rawdata.RDS')
```

```
medal_rawdata <- readRDS('medal_rawdata.RDS')
```

The raw data has missing values and wide type data. It is better to covert into long type data. Also, we need to vectors : 'Year' and 'Long Jump'.

```
# Tidy data by base R function
```

```
# Year is coded as 1900 = 0
```

```
# Combine year and long jump into two vectors
```

```
year <- c(medal_rawdata[[1]], medal_rawdata[[3]], medal_rawdata[[5]], medal_rawdata[[7]]) + 1900
```

```
longjump <- c(medal_rawdata[[2]], medal_rawdata[[4]], medal_rawdata[[6]], medal_rawdata[[8]])
```

```
# Bind the vectors as a data table and rename the categories
```

```
medal_data <- data.table(year[1:(length(year)-2)], longjump[1:(length(longjump)-2)])
```

```
colnames(medal_data) <- c('Year', 'Long Jump')
```

Above is the converted tidy data frames using the base R functions. A summary of the data is as follows:

Year	Long Jump
Min. :1896	Min. :249.8
1st Qu.:1921	1st Qu.:295.4
Median :1950	Median :308.1
Mean :1945	Mean :310.3
3rd Qu.:1971	3rd Qu.:327.5
Max. :1992	Max. :350.5

```
# Using tidyverse
```

```
# Year is coded as 1900 = 0
```

```
# Combine year and long jump into two vectors
```

```
medal_data <- medal_rawdata[,1:8]
```

```
colnames(medal_data) <- paste(rep(c('Year', 'Jump'),4), rep(1:4,each = 2))
```

```

year <- medal_data[,c(1,3,5,7)] %>% gather(key = 'name1', value = 'Year', 1,2,3,4) %>% filter(Year != na)

## Warning in Year != na: longer object length is not a multiple of shorter object
## length
year[,2] <- year[,2] + 1900
jump <- medal_data[,c(2,4,6,8)] %>% gather(key = 'name2', value = 'LongJump', 1,2,3,4) %>% filter(LongJ

## Warning in LongJump != na: longer object length is not a multiple of shorter
## object length
# Bind the vectors as a data table and rename the categories
medal_data <- bind_cols(year[,2], jump[,2])

## New names:
## * NA -> ...1
## * NA -> ...2
colnames(medal_data) <- c('Year', 'Long Jump')

```

The summary of the data converted by tidyverse is as follows.

Year	Long Jump
Min. :1896	Min. :249.8
1st Qu.:1921	1st Qu.:295.4
Median :1950	Median :308.1
Mean :1945	Mean :310.3
3rd Qu.:1971	3rd Qu.:327.5
Max. :1992	Max. :350.5

c. Brain weight (g) and body weight (kg) for 62 species.

<http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat>

```

##### Brain weight data #####
# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
url_brain <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
brain_rawdata <- fread(url_brain)

## Warning in fread(url_brain): Detected 12 column names but the data has 6
## columns. Filling rows automatically. Set fill=TRUE explicitly to avoid this
## warning.

saveRDS(brain_rawdata, 'brain_rawdata.RDS')
brain_rawdata <- readRDS('brain_rawdata.RDS')

```

The data needs two columns which are 'Body Wt' and 'Brain Wt'.

```

# Tidy data with base R function
bodywt <- c(brain_rawdata[[1]], brain_rawdata[[3]], brain_rawdata[[5]])
brainwt <- c(brain_rawdata[[2]], brain_rawdata[[4]], brain_rawdata[[6]])

brain_data <- data.table(bodywt[-length(bodywt)], brainwt[-length(brainwt)])
colnames(brain_data) <- c('Body Wt', 'Brain Wt')

```

Above is the converted tidy data frames using the base R functions. A summary of the data is as follows:

Body Wt	Brain Wt
Min. : 0.005	Min. : 0.10
1st Qu.: 0.600	1st Qu.: 4.25
Median : 3.342	Median : 17.25
Mean : 198.790	Mean : 283.13
3rd Qu.: 48.202	3rd Qu.: 166.00
Max. :6654.000	Max. :5712.00

```
# Tidy data with tidyverse
brain_data <- brain_rawdata[,1:6]
colnames(brain_data) <- paste(rep(c('bw', 'brw'),3), rep(1:3,each = 2))
bw <- brain_data[,c(1,3,5)] %>% gather(key = 'name1', value = 'BW', 1,2,3)
brw <- brain_data[,c(2,4,6)] %>% gather(key = 'name2', value = 'BRW', 1,2,3)

# Bind the vectors as a data table and rename the categories
brain_data <- bind_cols(bw[,2], brw[,2])

## New names:
## * NA -> ...1
## * NA -> ...2

colnames(brain_data) <- c('Body Wt', 'Brain Wt')
```

The summary of the data converted by tidyverse is as follows.

Body Wt	Brain Wt
Min. : 0.005	Min. : 0.10
1st Qu.: 0.600	1st Qu.: 4.25
Median : 3.342	Median : 17.25
Mean : 198.790	Mean : 283.13
3rd Qu.: 48.202	3rd Qu.: 166.00
Max. :6654.000	Max. :5712.00
NA's :1	NA's :1

- d. Triplicate measurements of tomato yield for two varieties of tomatos at three planting densities.

<http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat>

```
##### Tomato data #####
# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
url_tomato <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
tomato_rawdata <- fread(url_tomato, skip = 1)
```

```
## Warning in fread(url_tomato, skip = 1): Detected 3 column names but the data has
## 4 columns (i.e. invalid file). Added 1 extra default column name for the first
## column which is guessed to be row names or an index. Use setnames() afterwards
## if this guess is not correct, or fix the file write command that created the
## file to create a valid file.
```

```
saveRDS(tomato_rawdata, 'tomato_rawdata.RDS')
tomato_rawdata <- readRDS('tomato_rawdata.RDS')
```

The data above should split the values.

```
# Tidy data with base R function
# Need to split the values
cells <- strsplit(unlist(tomato_rawdata), split = ',', fixed = T) # split the data
categories <- unlist(c(cells[1],cells[2])) # two categories
values <- as.numeric(unlist(c(cells[3:8]))) # numerical data

# Combine into data frame
tomato_matrix <- matrix(values, byrow = T, ncol = 3)
tomato_matrix <- t(cbind(tomato_matrix[1:2,], tomato_matrix[3:4,], tomato_matrix[5:6,]))
tomato_data <- data.frame(tomato_matrix, as.character(rep(c(10000,20000,30000)), each=3))
colnames(tomato_data) <- c(categories, 'Density')
```

Above is the converted tidy data frames using the base R functions. A summary of the data is as follows:

Ife#1	PusaEarlyDwarf	Density
Min. :15.30	Min. : 8.10	Length:9
1st Qu.:16.60	1st Qu.:10.10	Class :character
Median :18.00	Median :12.70	Mode :character
Mean :18.11	Mean :12.02	NA
3rd Qu.:19.20	3rd Qu.:13.70	NA
Max. :21.00	Max. :15.40	NA

```
# Using tidyverse
tomato_data <- tomato_rawdata[, -1] %>%
  separate(col = '10000', into = c("1", "2", "3"), sep = ",", convert = T) %>%
  separate(col = '20000', into = c("4", "5", "6"), sep = ",", convert = T) %>%
  separate(col = '30000', into = c("7", "8", "9"), sep = ",", convert = T) %>%
  as.matrix() %>% t()
```

```
## Warning: Expected 3 pieces. Additional pieces discarded in 1 rows [2].
```

```
dens <- rep(c(10000,20000,30000), each = 3)

tomato_data <- tomato_data %>% cbind(dens) %>% as.data.table()
colnames(tomato_data) <- tomato_rawdata[,1] %>% unlist() %>% c("Density")
```

Ife#1	PusaEarlyDwarf	Density
Min. :15.30	Min. : 8.10	Min. :10000
1st Qu.:16.60	1st Qu.:10.10	1st Qu.:10000
Median :18.00	Median :12.70	Median :20000
Mean :18.11	Mean :12.02	Mean :20000
3rd Qu.:19.20	3rd Qu.:13.70	3rd Qu.:30000
Max. :21.00	Max. :15.40	Max. :30000

Problem 5

Finish this homework by pushing your changes to your repo. In general, your workflow for this should be:

1. git pull – to make sure you have the most recent repo

2. In R: do some work
3. git add – this tells git to track new files
4. git commit – make message INFORMATIVE and USEFUL
5. git push – this pushes your local changes to the repo

If you have difficulty with steps 1-5, git is not correctly or completely setup. See me for help.

Only submit the .Rmd and .pdf solution files. Names should be formatted HW2__lastname.Rmd and HW2__lastname.pdf

Optional preperation for next class:

TBD

Appendix

```
knitr::opts_chunk$set(echo = TRUE)
library(data.table)
library(tidyverse)
##### Sensory data #####
# Getting "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
url_sensory <- "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
sensory_rawdata <- fread(url_sensory, fill = TRUE, skip = 2, data.table = FALSE)
saveRDS(sensory_rawdata, 'sensory_rawdata.RDS')
sensory_rawdata <- readRDS('sensory_rawdata.RDS')

# Tidy data with base R function
matrix_sensory <- t(as.matrix(sensory_rawdata)) # Transpose the raw data
na <- which(is.na(matrix_sensory==TRUE))        # Find the indexes of Missing value

# The indexes where Item numbers are in the data
x <- 1
item <- x
for (i in 1:9){
  x <- x+18
  item <- c(item, x)
}

# Remove missing values and Item numbers from the data
# To focus on items, transpose the data table and rename the column names
sensory_data <- t(matrix(matrix_sensory[-c(na,item)], byrow = T, nrow = 10))
sensory_data <- data.table(sensory_data)
colnames(sensory_data) <- paste('Item', 1:10)
Opr <- rep(paste('Opr', 1:5), 3)
sensory_data <- cbind(Opr,sensory_data)
sensory_data <- sensory_data[order(sensory_data$Opr)]

knitr::kable(summary(sensory_data))
# Sensory data with tidyverse package
matrix_sensory <- sensory_rawdata %>% as.matrix() %>% t() # Transpose the raw data
```

```

na <- which(is.na(matrix_sensory==TRUE))      # Find the indexes of Missing value

# The indexes where Item numbers are in the data
x <- 1
item <- x
for (i in 1:9){
  x <- x+18
  item <- c(item, x)
}

# Remove missing values and Item numbers from the data
# To focus on items, transpose the data table and rename the column names
sensory_data <- matrix_sensory[-c(na,item)] %>% matrix(byrow = T, nrow = 10) %>% t()
sensory_data <- data.table(sensory_data)
Opr <- rep(paste('Opr', 1:5), 3)
sensory_data <- bind_cols(Opr,sensory_data)
colnames(sensory_data) <- c('Opr',paste('Item', 1:10))
sensory_data <- sensory_data[order(sensory_data$Opr)]

knitr::kable(summary(sensory_data))
##### Long Jump data #####
# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
url_medal <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
medal_rawdata <- fread(url_medal)
saveRDS(medal_rawdata, 'medal_rawdata.RDS')
medal_rawdata <- readRDS('medal_rawdata.RDS')

# Tidy data by base R function
# Year is coded as 1900 = 0
# Combine year and long jump into two vectors
year <- c(medal_rawdata[[1]], medal_rawdata[[3]], medal_rawdata[[5]], medal_rawdata[[7]]) + 1900
longjump <- c(medal_rawdata[[2]], medal_rawdata[[4]], medal_rawdata[[6]], medal_rawdata[[8]])

# Bind the vectors as a data table and rename the categories
medal_data <- data.table(year[1:(length(year)-2)], longjump[1:(length(longjump)-2)])
colnames(medal_data) <- c('Year', 'Long Jump')

knitr::kable(summary(medal_data))
# Using tidyverse
# Year is coded as 1900 = 0
# Combine year and long jump into two vectors
medal_data <- medal_rawdata[,1:8]
colnames(medal_data) <- paste(rep(c('Year', 'Jump'),4), rep(1:4,each = 2))
year <- medal_data[,c(1,3,5,7)] %>% gather(key = 'name1', value = 'Year', 1,2,3,4) %>% filter(Year != na)
year[,2] <- year[,2] + 1900
jump <- medal_data[,c(2,4,6,8)] %>% gather(key = 'name2', value = 'LongJump', 1,2,3,4) %>% filter(LongJ

# Bind the vectors as a data table and rename the categories
medal_data <- bind_cols(year[,2], jump[,2])
colnames(medal_data) <- c('Year', 'Long Jump')

##### Brain weight data #####

```



```

# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
url_brain <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
brain_rawdata <- fread(url_brain)
saveRDS(brain_rawdata, 'brain_rawdata.RDS')
brain_rawdata <- readRDS('brain_rawdata.RDS')
# Tidy data with base R function
bodywt <- c(brain_rawdata[[1]], brain_rawdata[[3]], brain_rawdata[[5]])
brainwt <- c(brain_rawdata[[2]], brain_rawdata[[4]], brain_rawdata[[6]])

brain_data <- data.table(bodywt[-length(bodywt)], brainwt[-length(brainwt)])
colnames(brain_data) <- c('Body Wt', 'Brain Wt')
knitr::kable(summary(brain_data))
# Tidy data with tidyverse
brain_data <- brain_rawdata[,1:6]
colnames(brain_data) <- paste(rep(c('bw', 'brw'),3), rep(1:3,each = 2))
bw <- brain_data[,c(1,3,5)] %>% gather(key = 'name1', value = 'BW', 1,2,3)
brw <- brain_data[,c(2,4,6)] %>% gather(key = 'name2', value = 'BRW', 1,2,3)

# Bind the vectors as a data table and rename the categories
brain_data <- bind_cols(bw[,2], brw[,2])
colnames(brain_data) <- c('Body Wt', 'Brain Wt')

##### Tomato data #####
# Getting "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
url_tomato <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
tomato_rawdata <- fread(url_tomato, skip = 1)
saveRDS(tomato_rawdata, 'tomato_rawdata.RDS')
tomato_rawdata <- readRDS('tomato_rawdata.RDS')

# Tidy data with base R function
# Need to split the values
cells <- strsplit(unlist(tomato_rawdata), split = ',', fixed = T) # split the data
categories <- unlist(c(cells[1],cells[2])) # two categories
values <- as.numeric(unlist(c(cells[3:8]))) # numerical data

# Combine into data frame
tomato_matrix <- matrix(values, byrow = T, ncol = 3)
tomato_matrix <- t(cbind(tomato_matrix[1:2,], tomato_matrix[3:4,], tomato_matrix[5:6,]))
tomato_data <- data.frame(tomato_matrix, as.character(rep(c(10000,20000,30000)), each=3))
colnames(tomato_data) <- c(categories, 'Density')

knitr::kable(summary(tomato_data))
# Using tidyverse
tomato_data <- tomato_rawdata[,-1] %>%
  separate(col = '10000', into = c("1","2","3"), sep = ",", convert = T) %>%
  separate(col = '20000', into = c("4","5","6"), sep = ",", convert = T) %>%
  separate(col = '30000', into = c("7","8","9"), sep = ",", convert = T) %>%
  as.matrix() %>% t()
dens <- rep(c(10000,20000,30000), each = 3)

tomato_data <- tomato_data %>% cbind(dens) %>% as.data.table()

```

```
colnames(tomato_data) <- tomato_rawdata[,1] %>% unlist() %>% c("Density")
```