Datacamp_Importing & Cleaning Data in R: Case Studies___School Attendance Data

dizhen

2019/4/5

Importing the data

```
# Load the gdata package
# library(gdata)
# att <- read.xls("data/attendance.xls")

# Import the spreadsheet: att
library(readxl)
att <- read_excel("data/attendance.xls")
```

```
## New names:
## * `` -> ...2
## * `` -> ...3
## * `` -> ...4
## * `` -> ...5
## * `` -> ...6
## * ... and 11 more problems
```

Examining the data

```
# Print the column names
names(att)
```

```
##  [1] "Table 43. Average daily attendance (ADA) as a percentage of total enrollment, school day lengt
##  [2] "...2"
##  [3] "...3"
##  [4] "...4"
##  [5] "...5"
##  [6] "...6"
##  [7] "...7"
##  [8] "...8"
##  [9] "...9"
## [10] "...10"
## [11] "...11"
## [12] "...12"
## [13] "...13"
## [14] "...14"
## [15] "...15"
## [16] "...16"
## [17] "...17"
```

```
# Print the first 6 rows
head(att,n = 6)
```

```
## # A tibble: 6 x 17
##   `Table 43. Aver~ ...2    ...3 ...4     ...5 ...6    ...7 ...8   ...9
##   <chr>            <chr>  <dbl> <chr>   <dbl> <chr>  <dbl> <chr> <dbl>
## 1 <NA>             Tota~ NA     <NA>  NA      <NA>  NA     <NA>  NA
## 2 <NA>             ADA ~ NA     Aver~ NA      Aver~ NA     Aver~ NA
## 3 1                2     NA     3     NA      4     NA     5     NA
## 4 United States .~ 93.0~  0.219 6.64~  0.0176 180    0.143 1192~  3.09
## 5 Alabama .......~ 93.8~  1.24  7.02~  0.0656 180    0.755 1266~ 12.3
## 6 Alaska ........~ 89.9~  1.22  6.47~  0.0499 180    3.43  1162~ 22.9
## # ... with 8 more variables: ...10 <chr>, ...11 <dbl>, ...12 <chr>,
## #   ...13 <dbl>, ...14 <chr>, ...15 <chr>, ...16 <chr>, ...17 <chr>
```

```
# Print the last 6 rows
tail(att, n = 6)
```

```
## # A tibble: 6 x 17
##   `Table 43. Aver~ ...2    ...3 ...4     ...5 ...6    ...7 ...8   ...9
##   <chr>            <chr>  <dbl> <chr>   <dbl> <chr>  <dbl> <chr> <dbl>
## 1 Wisconsin .....~ 94.9~  0.566 6.91~  0.0427 180    0.736 1246~  8.63
## 2 Wyoming .......~ 92.3~  1.15  6.85~  0.0458 175    1.28  1200~  8.33
## 3 <U+2020>Not applicable. <NA>  NA     <NA>  NA      <NA>  NA     <NA>  NA
## 4 <U+2021>Reporting stan~ <NA>  NA     <NA>  NA      <NA>  NA     <NA>  NA
## 5 NOTE: Averages ~ <NA>  NA     <NA>  NA      <NA>  NA     <NA>  NA
## 6 "SOURCE: U.S. D~ <NA>  NA     <NA>  NA      <NA>  NA     <NA>  NA
## # ... with 8 more variables: ...10 <chr>, ...11 <dbl>, ...12 <chr>,
## #   ...13 <dbl>, ...14 <chr>, ...15 <chr>, ...16 <chr>, ...17 <chr>
```

```
# Print the structure
str(att)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    68 obs. of  17 variables:
##  $ Table 43. Average daily attendance (ADA) as a percentage of total enrollment, school day length, a
##  $ ...2
##  $ ...3
##  $ ...4
##  $ ...5
##  $ ...6
##  $ ...7
##  $ ...8
##  $ ...9
##  $ ...10
##  $ ...11
##  $ ...12
##  $ ...13
##  $ ...14
##  $ ...15
##  $ ...16
##  $ ...17
```

Removing unnecessary rows

```r
# Create remove
remove <- c(3,56:59)

# Create att2
att2 <- att[-remove,]
```

Removing useless columns

```r
# Create remove
remove <- c( 3, 5, 7, 9, 11, 13, 15, 17)

# Create att3
att3 <- att2[,-remove]
```

Splitting the data

```r
# Subset just elementary schools: att_elem
att_elem <- att3[,c(1,6,7)]

# Subset just secondary schools: att_sec
att_sec <- att3[,c(1,8,9)]

# Subset all schools: att4
att4 <- att3[,c(1:5)]
```

Replacing the names

```r
# Define cnames vector (don't change)
cnames <- c("state", "avg_attend_pct", "avg_hr_per_day",
            "avg_day_per_yr", "avg_hr_per_yr")

# Assign column names of att4
colnames(att4) <- cnames

# Remove first two rows of att4: att5
att5<- att4[-c(1:2),]

# View the names of att5
names(att5)
```

```
## [1] "state"          "avg_attend_pct" "avg_hr_per_day" "avg_day_per_yr"
## [5] "avg_hr_per_yr"
```

Cleaning up extra characters

```r
# Remove all periods in state column
library("stringr")
att5$state <- str_replace_all(att5$state,"\\.","")

# Remove white space around state names
att5$state <- str_trim(att5$state)

# View the head of att5
head(att5)
```

```
## # A tibble: 6 x 5
##   state     avg_attend_pct   avg_hr_per_day  avg_day_per_yr avg_hr_per_yr
##   <chr>     <chr>            <chr>           <chr>          <chr>
## 1 United S~ 93.078962000000~ 6.644700000000~ 180            1192.647200000~
## 2 Alabama   93.812370999999~ 7.028520000000~ 180            1266.6205
## 3 Alaska    89.917597000000~ 6.476880000000~ 180            1162.9084
## 4 Arizona   89.036961000000~ 6.433690000000~ 181            1159.114399999~
## 5 Arkansas  91.827111000000~ 6.885419999999~ 179            1228.888099999~
## 6 Californ~ 93.241016999999~ 6.24064         181            1128.769399999~
```

Some final type conversions

```
# Change columns to numeric using dplyr (don't change)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
example <- mutate_at(att5, vars(-state), funs(as.numeric))
```

```
## Warning: funs() is soft deprecated as of dplyr 0.8.0
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once per session.
```

```
# Define vector containing numerical columns: cols
cols <- -1

# Use sapply to coerce cols to numeric
att5[, cols] <- sapply(att5[, cols], as.numeric)
```