Importing and exporting data in R

Joddy

2024-02-14

Contents

R Open Lab 2: Reading, writing, and managing multiple datasets

Part 1: Importing and exporting data in R

Getting and setting your path, and using paths in R

Get the working directory

```
getwd()
```

```
## [1] "/Users/jimarche/R/r-openlab/r_open_lab-2-read_write_join"
```

```
# Set the working directory (Linux and Mac)
# setwd("/Users/jimarche/R/r open labs/r_open_lab-1-read_write_join/data")

# What if you are on Windows?
# setwd("C:/Users/jimarche/R/r open labs/r_open_lab-1-read_write_join/data")
```

1. Importing data

Text file read.table() is the most basic importing function. You can specify tons of different arguments in this function.

```
stocks <- read.table("data/sample-5.tsv", sep="\t", skip=26, skipNul=TRUE, header=TRUE)
stocks <- stocks[,-1] # Get rid of the first column, which is blank

library(readr)

stocks <- read_tsv("data/sample-5.tsv", skip=20, col_select=-1) # col_select lets us delete column 1 in
```

```
## New names:
## Rows: 252 Columns: 6
## -- Column specification
## -------------------------------------------------------- Delimiter: "\t" chr
## (1): Date dbl (4): Open, High, Low, Close num (1): Volume
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

Comma separated variable (csv)

```
library(readr)

birds <- read_csv("data/birds.csv")
```

```
## Rows: 4 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): county
## dbl (5): year, eagleNest, eagleAir, warblerNest, warblerAir
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
birds <- read_csv("data/birds.csv", col_names=TRUE, skip=0)
```

```
## Rows: 4 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): county
## dbl (5): year, eagleNest, eagleAir, warblerNest, warblerAir
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Excel (xls, xlsx)

```
library(readxl)

ufo <- read_excel("data/ufo.xlsx")
```

```r
# using range= to: take the first 200 rows from ufo.xlsx, dropping the last 3 columns
ufo <- read_excel("data/ufo.xlsx", col_names=TRUE, range="B1:H201")

# Loading a sheet from an excel file with multiple sheets
ufo <- read_excel("data/ufo.xlsx", sheet="ufo")

# read_xls for older excel files (.xls)
participants <- read_xls("data/participants.xls")
```

Other data formats using Haven

```r
library(haven)

#  **SAS**: `read_sas()` reads .sas7bdat and .sas7bcat files and `read_xpt()` reads SAS transport files
# `write_sas()` writes .sas7bdat files.
airline <- read_sas("data/airline.sas7bdat")

# **SPSS**: `read_sav()` reads .sav files and `read_por()` reads the older .por files. `write_sav()` wr
bankloan <- read_sav("data/bankloan.sav")

# **Stata**: `read_dta()` reads .dta files (up to version 15).
# `write_dta()` writes .dta files (versions 8-15).
auto <- read_dta("data/auto.dta")
```

2. Writing datasets from data.frame or tibble

Writing .txt and .csv files

Loading mtcars data

```r
# Write data to txt file: tab separated values
write.table(mtcars, file = "data/mtcars.txt", sep = "\t",
            row.names = TRUE, col.names = NA)

# Write data to csv files:
# decimal point = "." and value separators = comma (",")
write.csv(mtcars, file = "data/mtcars.csv")
```

Using the readr package (tidyverse)

```r
# Loading mtcars data
data(mtcars)
library(readr)

# Writing mtcars data to a tsv file
write_tsv(mtcars, file = "data/mtcars.txt")
# Writing mtcars data to a csv file
write_csv(mtcars, file = "data/mtcars.csv")
```

Writing .xlsx files

```
library(writexl)
```

```
## Warning: package 'writexl' was built under R version 4.3.1
```

```
df <- data.frame(
  name = c("UCLA", "Berkeley", "Jeroen"),
  founded = c(1919, 1868, 2030),
  website = xl_hyperlink(c("http://www.ucla.edu", "http://www.berkeley.edu", NA), "homepage") )
df$age <- xl_formula('=(YEAR(TODAY()) - INDIRECT("B" & ROW()))')
write_xlsx(df, 'data/universities.xlsx')
```

Exporting R data files

```
# Saving and restoring one single R object (save a single object to a file):
saveRDS(mtcars, "data/mtcars.rds")
# Restore it under a different name
my_data <- readRDS("data/mtcars.rds")
```

Saving and restoring one or more R objects

```
ufo1 <- ufo[1:100,]
ufo2 <- ufo[101:200,]

# Save multiple objects
save(ufo1, ufo2, file = "data/ufo_first200.RData")
# To load the data again
load("data/ufo_first200.RData")
```

What other file types have you worked with? Let's try to import them into R!