Data I/O

The R Bootcamp

www.therbootcamp.com

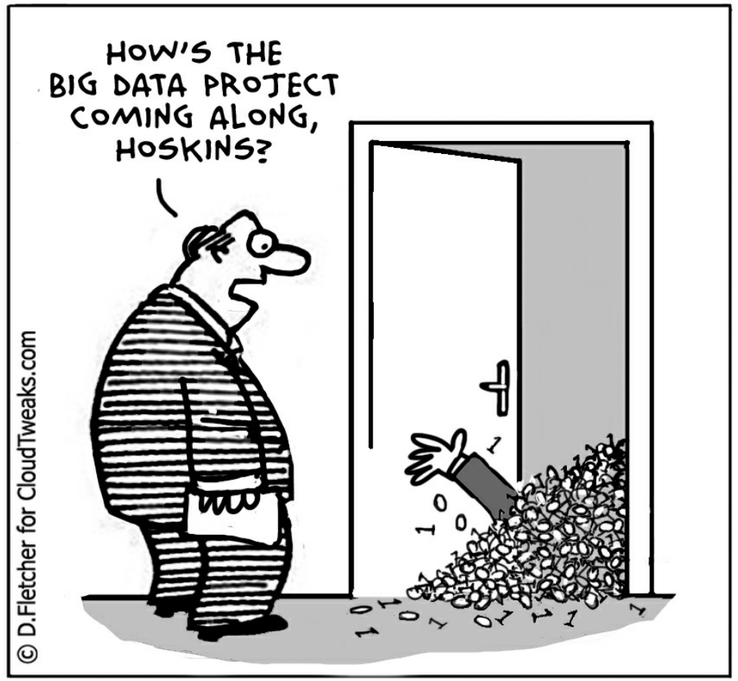
@therbootcamp

July 2018

Data Input/Output

Raw data can come in many shapes and sizes, but **R's got you covered**.

Package	Description
`readr`	`.csv`, `.txt`, etc.
`haven`	`.sav`, `.sas7bdat`, `.dta`
`readxl`	`.xls`, `.xlsx`
`R.matlab`	`.mat`
`jsonlite`	`.json`
`rvest`	`.html`
`XML`, `xml2`	`.xml`



Raw (structured) Data

delim-separated data

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

markup data

```
<!doctype html>
<html lang="en" class="gr__therbootcamp_github_io">
▶<head>…</head>
▼<body data-gr-c-s-loaded="true">
   <script async src="https://www.google-analytics.com/analytics
   <script type="text/javascript" async src="https://snap.licdn.
   insight.min.js"></script>
   <script type="text/javascript">
         _linkedin_data_partner_id = "111419";
      </script>
  ▶<script type="text/javascript">…</script>
  ▼<div id="particles-js">
    ▼<div class="content">
      ▼<h1>
         <span class="site-title">TheRBootcamp</span>
         <span class="site-description">Learn Data Science in R</
        ▼<a class="link" href="#upcoming" data-scroll>
           <font size="6" color="#FF3A2A">Basel July 21 22 28 29
         </a>
       </h1>
```

Delim-separated data

1 - Most typical file format.

2 - Requires **delimiter** to separate entries.

delim-separated data

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

readr

`readr` is a `tidyverse` package that provides convenient functions to **read in** (non-nested) data files into data frames (`tibbles` to be precise):

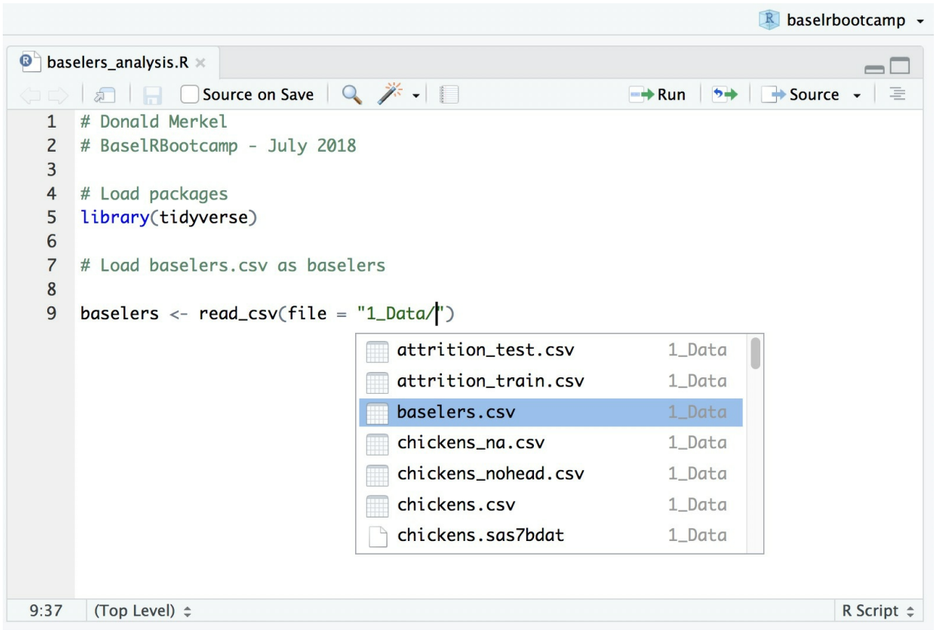

```
# Importing data from a file

data <- read_csv(file, ...)   # comma-delimited
data <- read_csv2(file, ...)  # semicolon-delimeted
data <- read_delim(file, ...) # arbitrary-delimited

# Writing a data frame to a file

write_csv(data_object, file, ...)    # comma-delimited
write_delim(data_object, file, ...)  # arbitrary-delimited
```

Finding the file path

1 - Identify the file path using the **auto-complete**.

2 - Initiate auto-complete and browse through the folder structure by placing the cursor between two quotation marks and using the **tab key**.

3 - Auto-complete begins with the project folder - **place your data inside your project folder!**



Identifying the delimiter

1 - **Find the file** on your hard drive. Should be in your data folder inside your project.

2 - **Open the file** in RStudio (right-click on the file in the pane) a text viewer, e.g., (Mac), (Mac), (Windows).

baselers.csv

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

Identifying the delimiter

1 - **Find the file** on your hard drive. Should be in your data folder inside your project.

2 - **Open the file** in RStudio (right-click on the file in the pane) a text viewer, e.g., (Mac), (Mac), (Windows).

```
# Read with explicit column names
baselers <- read_delim(file = ".../baselers.csv",
                       delim = c(",")
```

baselers.csv

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

Handling headers

1 - `readr`- functions typically expect the **column names** in the first line.

2 - If no column names are available, use the `col_names`-**argument** to provide them.

```
# Read with explicit column names
baselers <- read_csv(file = ".../baselers.csv",
                     col_names = c("id",
                                   "age",
                                    ...))
```

baselers.csv

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

Handling data types

Reading in data, `readr` **infers the type of data** for each column.

```
# Read baselers
read_csv(file = "1_Data/baselers.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   sex = col_character(),
##   height = col_double(),
##   weight = col_double(),
##   income = col_double(),
##   education = col_character(),
##   confession = col_character(),
##   food = col_double(),
##   fasnacht = col_character(),
##   eyecor = col_character()
## )

## See spec(...) for full column specifications.
```

baselers.csv

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

Handling data types

Incorrect data types can be fixed. Typically this involves:

1 - **removing character elements** from otherwise numeric variables.

2 - Setting **explicit NA strings** using the *na*-argument.

3 - Re-running `type_convert`.

```
# Read baselers
baseslers <- read_csv(file = ".../baselers.csv",
                      na = c('NA'))

# Try to fix incorrect data types
baselers <- type_convert(baselers)
```

`baselers.csv`

```
id,sex,age,height,weight,income,education,confession,childre
1,male,44,174.3,113.4,6300,SEK_III,catholic,2,5,7,610,40,6,4
2,male,65,180.3,75.2,10900,obligatory_school,confessionless,
3,female,31,168.3,55.5,5100,SEK_III,NA,2,7,6,720,14,3,6,102,
4,male,27,209,93.8,4200,SEK_III,catholic,2,7,8,680,39,6,0,11
5,male,24,176.7,NA,4e3,SEK_III,catholic,1,5,4,260,19,0,1,82,
6,male,63,186.6,67.4,11400,SEK_III,evangelical-reformed,0,7,
7,male,71,151.6,83.3,12e3,SEK_III,evangelical-reformed,2,8,5
8,female,41,155.7,67.8,7600,SEK_III,confessionless,1,7,2,135
9,male,43,176.1,69.3,8500,apprenticeship,catholic,2,7,5,150,
10,female,31,166.1,66.3,6100,SEK_II,catholic,1,6,7,700,0,0,3
11,female,42,157.8,51.9,8e3,obligatory_school,catholic,2,9,7
12,male,31,165.9,66,5900,apprenticeship,evangelical-reformed
13,female,38,162.5,73.4,6200,apprenticeship,confessionless,2
14,female,49,182.8,46.9,NA,SEK_III,evangelical-reformed,1,6,
15,female,39,160,NA,5600,SEK_III,other,2,7,4,540,35,7,4,122,
16,female,54,139.7,50.3,10900,SEK_III,evangelical-reformed,3
17,female,78,153.1,64.1,11e3,SEK_III,confessionless,2,7,2,97
18,female,62,174.6,63.8,11500,SEK_III,confessionless,2,9,7,1
19,male,88,191.4,99.8,14200,SEK_III,confessionless,2,7,3,121
20,male,74,183.8,78.1,12100,apprenticeship,catholic,2,5,7,11
```

Other data

R provides **read and write functions** for practically all data file formats. See **rio**.

readr


```
# read fixed width files (can be fast)
data <- read_fwf(file, ...)

# read Apache style log files
data <- read_log(file, ...)
```

haven


```
# read SAS's .sas7bat and sas7bcat files
data <- read_sas(file, ...)

# read SPSS's .sav files
data <- read_sav(file, ...)

# etc
```

readxl

```
# read Excel's .xls and xlsx files
data <- read_excel(file, ...)
```

Other

```
# Read Matlab .mat files
data <- R.matlab::readMat(file, ...)

# Read and wrangle .xml and .html
data <- XML::xmlParseParse(file, ...)

# from package jsonlite: read .json files
data <- jsonlite::read_json(file, ...)
```

Websites

R provides **read and write functions** for practically all data file formats. See **rio**.

```
# load package
library(rvest)
library(xml2)

# get html page (abbreviated)
url <- '.../R_(programming_language)'
page <- read_html(u)

# get xpath (abbreviated)
xpath <- '.../div/table[2]'

# get table using XPath
table <- page %>%
  html_node(
    xpath = xpath) %>%
  html_table()
```

```
## # A tibble: 15 x 3
##    Release Date       Description
##    <chr>   <chr>      <chr>
##  1 0.16    ""         This is the last alpha version
##  2 0.49    1997-04-23 This is the oldest source rele
##  3 0.60    1997-12-05 "R becomes an official part of "
##  4 0.65.1  1999-10-07 First versions of update.packa
##  5 1.0     2000-02-29 Considered by its developers s
##  6 1.4     2001-12-19 "S4 methods are introduced and "
##  7 2.0     2004-10-04 Introduced lazy loading, which
##  8 2.1     2005-04-18 Support for UTF-8 encoding, an
##  9 2.11    2010-04-22 Support for Windows 64 bit sys
## 10 2.13    2011-04-14 Adding a new compiler function
## 11 2.14    2011-10-31 Added mandatory namespaces for
## 12 2.15    2012-03-30 "New load balancing functions. "
## 13 3.0     2013-04-03 Support for numeric index valu
## 14 3.4     2017-04-21 Just-in-time compilation (JIT)
## 15 3.5     2018-04-23 Packages byte-compiled on inst
```

Remote databases

R provides **all necessary tools to pull data from or directly work with** remote databases such as, e.g., a SQL database. Find out more at:

db.rstudio.com

R's data formats

R's own formats provide the possibility to store **data as R objects** as well as substantial **compression**.

.RData

1 - Bundles **several R objects**.
2 - Loads objects **directly into workspace**.

```
# save data as .RData
save(baselers, zuerichers, ...,
     file = "my_data.RData")

# load data from .RData
load("my_data.RData")
```

.RDS

1 - Stores **single R objects**.
2 - Import is **assigned to object**.

```
# save data as .RDS
saveRDS(baselers,
        file = "baselers.rds")

# load data from .RDS
baselers <- readRDS("baselers.rds")
```

Practical

Link to practical