Exercise 1: Data import

Philipp Kling

15.06.2020

Contents

Various ways to import data **1**

R Data 2

Comma separated files 2

Excel sheets 2

Basic overiew **2**

Rmarkdown **5**

Git and assignment **6**

```
library(knitr)

## Global options
options(max.print="75")
opts_chunk$set(echo=FALSE,
                 cache=FALSE,
               prompt=FALSE,
               tidy=TRUE,
               comment=NA,
               message=FALSE,
               warning=FALSE)
opts_knit$set(width=75)
rm(list = ls())
```

Various ways to import data

Here we import the same dataset in 3 common fileformats: an R-data file, a comma separated file, and an Microsoft excel sheet. The first lines of the data look as follows. It has 3 columns, and about 2.500 rows. It contains the links and sections of articles from the Guardian.

```
rm(list = ls())
load("/home/philipp/Documents/fds-2020-exercise/data/ex1/testdata.Rda")
testdata$id <- paste(substr(testdata$id, start = 1, stop = 25), "...", sep = "")
testdata$link <- paste(substr(testdata$link, start = 1, stop = 25), "...", sep = "")
head(testdata)
```

```
                            id                         link    sectionId
1 artanddesign/2020/apr/06/... https://www.theguardian.c... artanddesign
2 artanddesign/2020/apr/06/... https://www.theguardian.c... artanddesign
3 artanddesign/2020/apr/06/... https://www.theguardian.c... artanddesign
4 artanddesign/2020/apr/10/... https://www.theguardian.c... artanddesign
5 artanddesign/2020/apr/10/... https://www.theguardian.c... artanddesign
6 artanddesign/2020/apr/11/... https://www.theguardian.c... artanddesign
```

R Data

Load with **load()**.

```
rm(list = ls())
load("/home/philipp/Documents/fds-2020-exercise/data/ex1/testdata.Rda")
dim(testdata)
```

```
[1] 2498    3
```

Comma separated files

Use **read.csv()**.

```
rm(list = ls())
testdata <- read.csv("/home/philipp/Documents/fds-2020-exercise/data/ex1/testdata.csv")
```

Attention: check the dimensions: only 1 column, but the dataset included 4 columns.

```
dim(testdata)
```

```
[1] 2498    1
```

```
head(testdata)
```

```

1                   artanddesign/2020/apr/06/andy-warhol-take-a-virtual-tour-around-the-tate-modern-exhi
2                                   artanddesign/2020/apr/06/bathtime-and-black-paint-tracey-emin-posts-
3                   artanddesign/2020/apr/06/how-i-became-the-duke-of-urbino-getty-museum-recreate-maste
4                                                                                 artanddesign/2020/apr/
5                                                             artanddesign/2020/apr/10/virtual-design-fe
6 artanddesign/2020/apr/11/mick-rock-releases-unseen-photographs-of-1970s-rock-royalty-to-support-nhs;ht
```

Inspect it with a text editor of your choice: you will see that values are not separated by commas, but by semicolons.

```
rm(list = ls())
testdata <- read.csv("/home/philipp/Documents/fds-2020-exercise/data/ex1/testdata.csv",
    sep = ";")
dim(testdata)
```

```
[1] 2498    3
```

Excel sheets

Install and use the **readxl** package and use the **read_xlsx()** command.

```
rm(list = ls())
combined_excel <- readxl::read_xlsx("/home/philipp/Documents/fds-2020-exercise/data/ex1/testdata.xlsx")
```

Basic overiew

To get a basic overview of a dataset, we might use **str()**

```
str(combined_excel)
```

```
Classes 'tbl_df', 'tbl' and 'data.frame':   2498 obs. of  3 variables:
 $ id       : chr  "artanddesign/2020/apr/06/andy-warhol-take-a-virtual-tour-around-the-tate-modern-exhi
 $ link     : chr  "https://www.theguardian.com/artanddesign/2020/apr/06/andy-warhol-take-a-virtual-tour
 $ sectionId: chr  "artanddesign" "artanddesign" "artanddesign" "artanddesign" ...
```

As mentioned above, **dim()** provides us with a basic overview of how many rows and columns are included in the dataset.

```
dim(combined_excel)
```

```
[1] 2498    3
```

The **table()** command provides us with an easy overview of the distribution of a dichotomous or categorical variable.

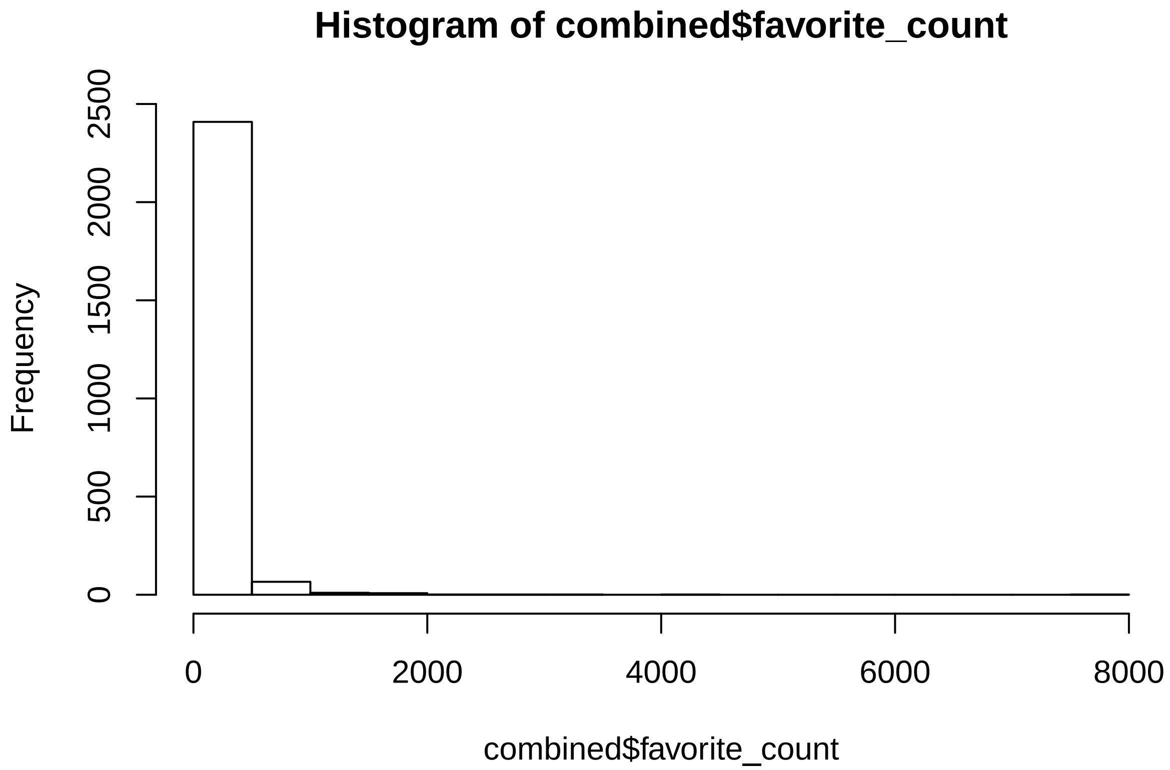
```
table(combined_excel$sectionId)
```

```

      artanddesign     australia-news              books 
                21                 87                 52 
          business      commentisfree          community 
               175                213                 19 
           culture          education        environment 
                25                 37                 46 
           fashion               film              focus 
                13                 34                  1 
              food           football              games 
                16                155                  4 
            global global-development         inequality 
                 1                 17                  1 
               law       lifeandstyle              media 
                 4                 71                 25 
        membership              money              music 
                 1                 20                 53 
              news           politics            science 
                18                127                 21 
           society              sport              stage 
                90                104                 13 
        technology        theobserver             travel 
                33                  1                 19 
      tv-and-radio            uk-news            us-news 
                41                 84                139 
             world 
               717 
```

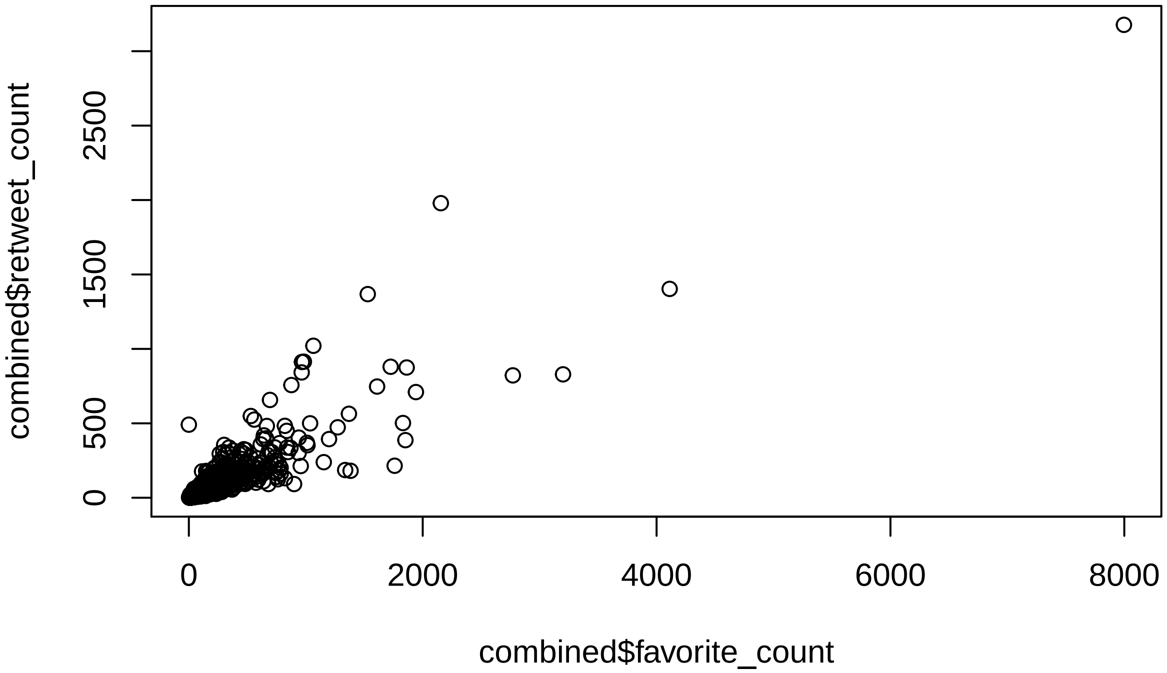
You can use **hist()** to plot a histogram of a numeric variable and get an overview.

```
load("/home/philipp/Documents/foundationsdatascience-2020/data/ex1/combined.Rda")
hist(combined$favorite_count)
```



You can use **plot()** to plot two variables against each other.

```
plot(combined$favorite_count, combined$retweet_count)
```



Rmarkdown

This file is or is produced by a R Markdown file. You will find a detailed introduction to RMarkdown here: https://bookdown.org/yihui/rmarkdown/ and a summary sheet here: https://github.com/rstudio/cheatsheets/raw/master/rmarkdown-2.0.pdf.

In our case, RMarkdown files consist of sections that include text in combination with code "chunks". Markdown files allow us to combine text with code. Markdown files keep the syntax simple and use comparable syntax to LaTeX and HTML. Often, the same commands that work in LaTeX work with Markdown as well. Basic text formatting is done with the following commands:

- you can make text italic by putting * or _ around it. E.g. *text* looks like *text*
- you can make text bold by putting ** it. E.g. **text** looks like **`text`**
- if you want to resemble code you need to put ‘ around it. E.g. ‘codetext‘ looks like `codetext`
- Sections are introduced by using #.
- Lists can be done with using a space and either *, -, or + and then again a space.

Additionally, you need the package `knitr` to compile or "knit" a Markdown File to an output format. In RStudio you can then chose if you want a PDF, HTML or Word Document.

Code chunks also allow for options. Code Chunks are introduced by "‘ and also closed like this. After the opening, we need to specify which type of programming language we want to insert (`r`) and then we e.g. can specify if we want to `echo` our code (`TRUE`) or whether we want to omit it (`FALSE`). Other options are e.g. `eval`, which indicates whether we want to evaluate the code in the chunk below or whether we just want to skip it (`eval=FALSE`).

Git and assignment

Next you may set up your own github account and download or clone the github repository accompanying the lecture and this exercise. You will find the assignment of the first exercise in the folder "ex1" under the name "ex1_assignment.Rmd" or its HTML and PDF version. You need to complete this assignment by adding the necessary code to the prepared RMarkdown file. Please change the name to "firstname_lastname_assignment_1.Rmd" and upload it in the Dropbox section on **OLAT**. Please keep the general name pattern throughout the next assignments as well.