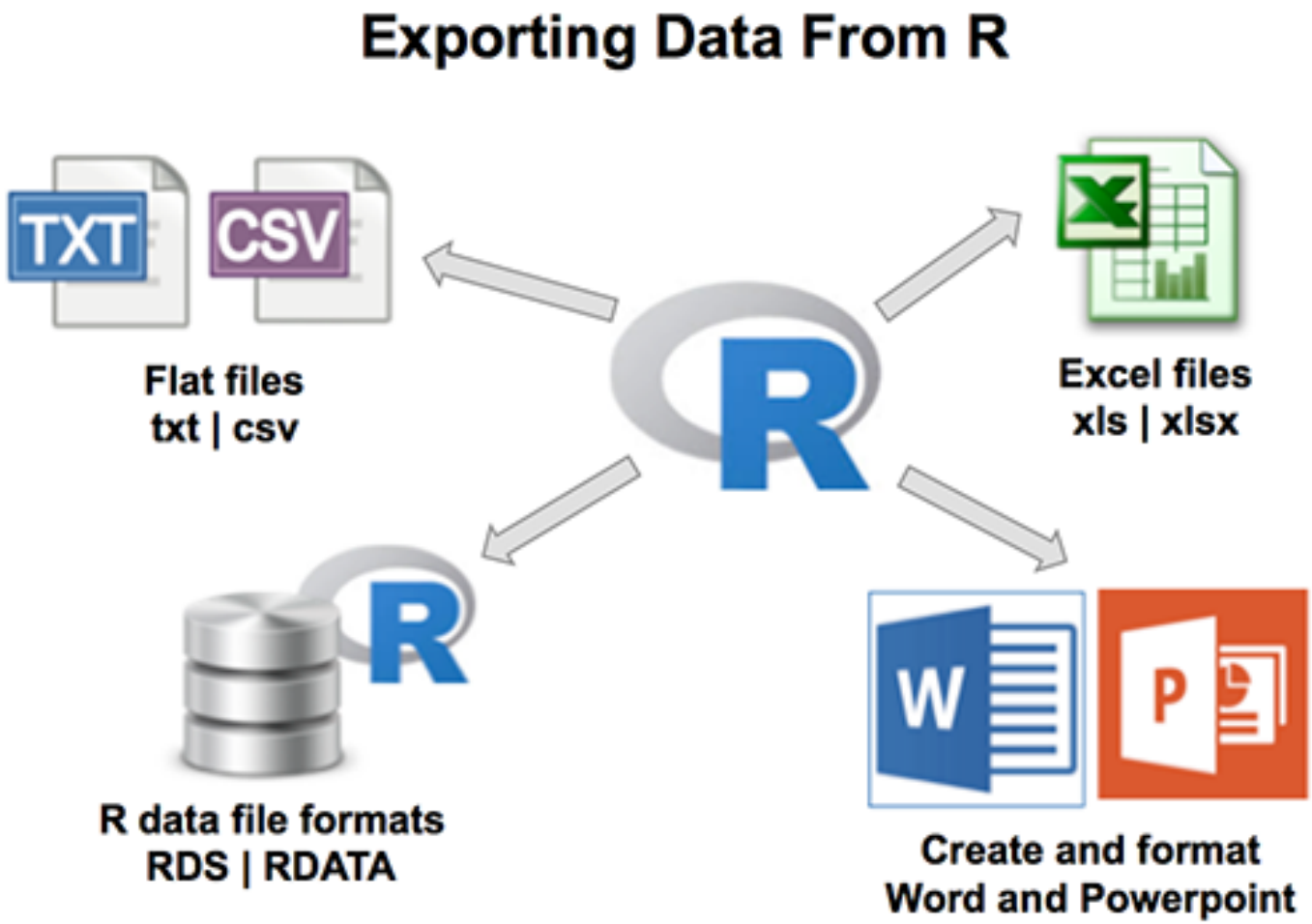
A4 Data Export

Jan-Philipp Kolb

21 Februar, 2020

Exporting data from R



Creating an example data record

```
A <- c(1,2,3,4)
B <- c("A","B","C","D")

mydata <- data.frame(A,B)
```

```
mydata
```

A	B
1	A
2	B
3	C
4	D

Overview data import/export

- if you continue working with R, `.RData` or **`rds`** format is the best choice:

```
save(mydata, file="mydata.RData")
saveRDS(mydata, "mydata.rds")
```

- The data set can be imported with `load`.

```
load("mydata.RData")
mydata <- readRDS("mydata.rds")
```

- `saveRDS()` doesn't save the both the object and its name it just saves a representation of the object

Export as Excel

- Create an example **tibble**:

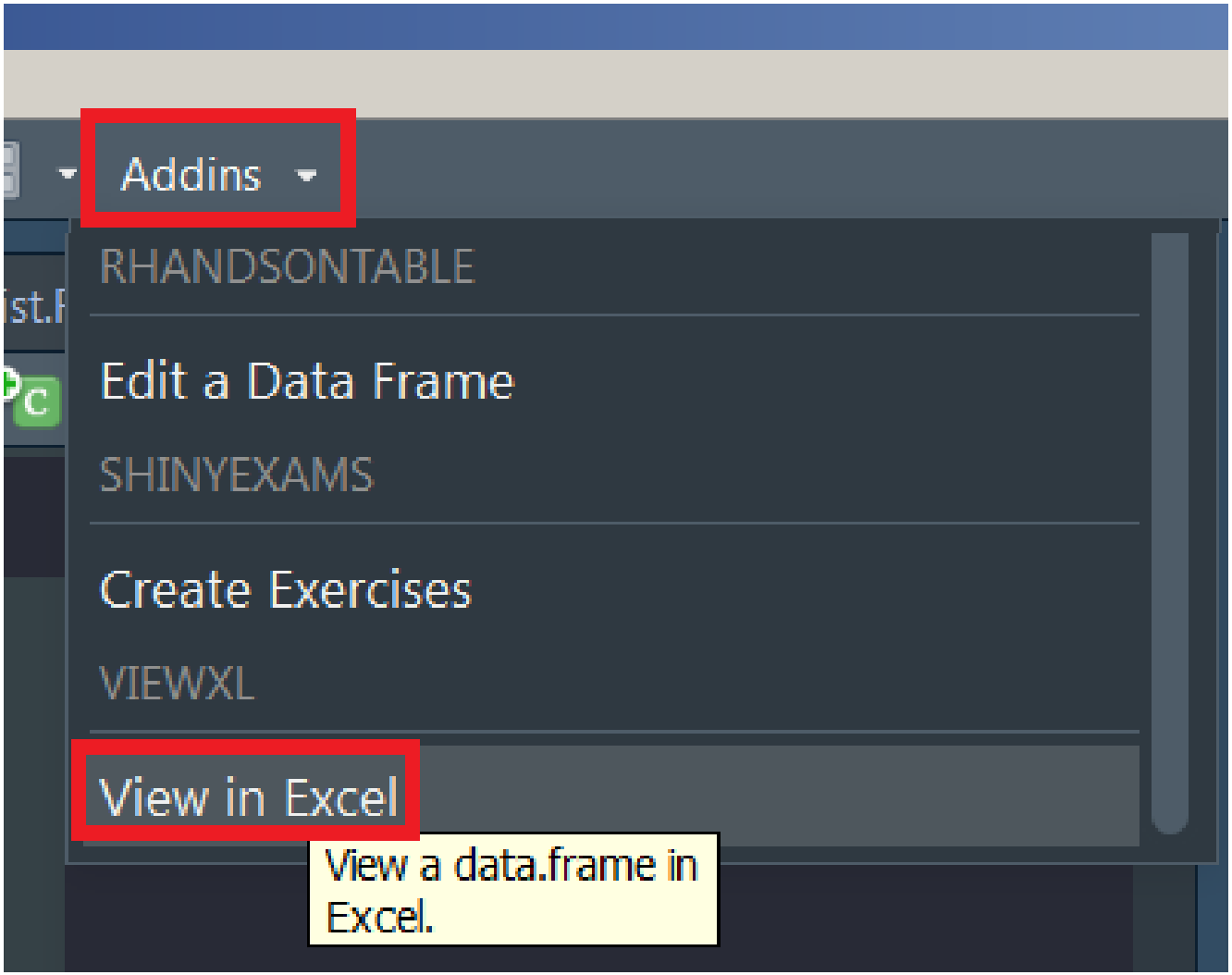
```
library(tibble)
ab <- tibble(a=1:4,b=4:1)
```

```
library(xlsx)
setwd("D:/Daten/GitLab/IntroDataAnalysis/data")
write.xlsx(ab,file="ab.xlsx")
```

Addin to open dataset in Excel

```
devtools::install_github("dreamRs/viewxl")
```

- select a `data.frame` in script -> it is opened in Excel.



Save data in .csv format

```
write.csv(mydata,file="mydata.csv")
```

- If you want to continue working with German Excel, it is better to use `write.csv2`

```
write.csv2(mydata,file="mydata.csv")
```

- Otherwise, the result looks like this:

	A
1	,"A","B"
2	1,1,"A"
3	2,2,"B"
4	3,3,"C"
5	4,4,"D"
6	

Argument row.names

Prevent row names to be written to file when using write.csv

```
write.csv(mydata,file="mydata.csv", row.names=FALSE)
```

- or for German data:

```
write.csv2(mydata,file="mydata.csv", row.names=FALSE)
```

```
install.packages("rio")
```

Import, Export, and Convert Data Files

The idea behind **rio** is to simplify the process of importing data into R and exporting data from R. This process is, probably unnecessarily, extremely complex for beginning R users. Indeed, R supplies an entire manual describing the process of data import/export. And, despite all of that text, most of the packages described are (to varying degrees) out-of-date. Faster, simpler, packages with fewer dependencies have been created for many of the file types described in that document. **rio** aims to unify data I/O (importing and exporting) into two simple functions: `import()` and `export()` so that beginners (and experienced R users) never have to think twice (or even once) about the best way to read and write R data.

Save data as .sav (SPSS)

```
library("rio")
# create file to convert

export(mtcars, "data/mtcars.sav")
```

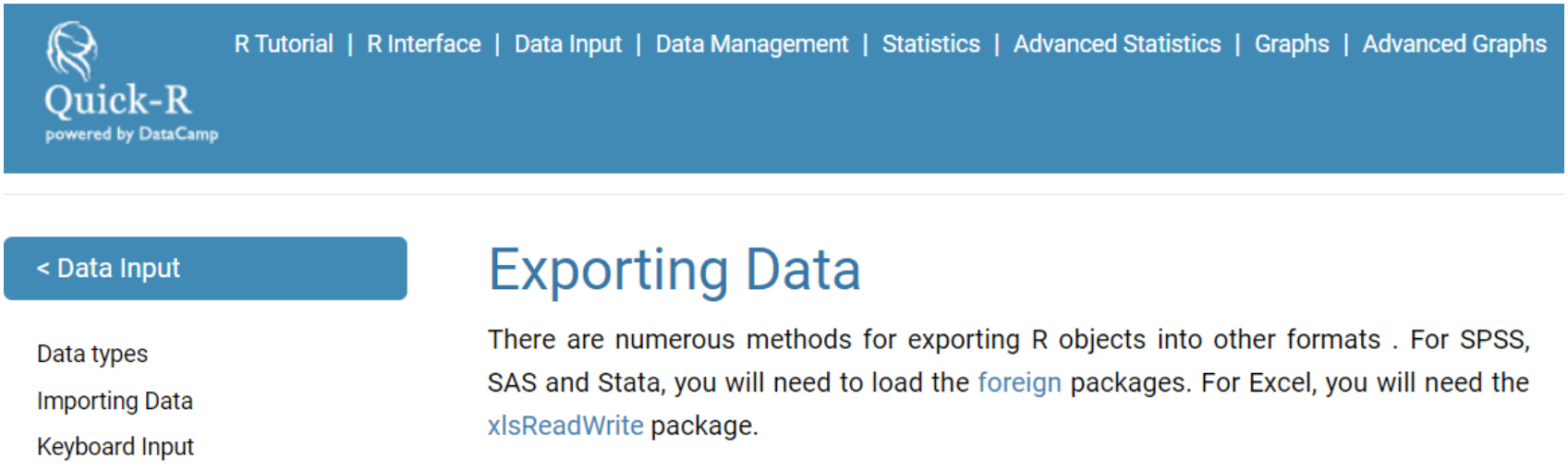
Convert file formats

```
export(mtcars, "data/mtcars.dta")

# convert Stata to SPSS
convert("data/mtcars.dta", "data/mtcars.sav")
```

Links Export

- **Quick R** for the export of data



- Help for exporting on the **CRAN Server**
- **Export data from R**
- Youtube video - **Export data from R**
- Quick R - **Exporting data**
- dummies - **How to Get Your Data Out of R**
- R Core Team - **R Data Import/Export**

Exercise: Export dataset

- Please load the `iris` built-in dataset
- Export the `iris` dataset to Excel