03. Vkládání dat

Harmonogram

- 01. Rekapitulace

- 02. Import flat files (.txt, .csv)

- 03. Import souborů z MS Excel (.xlsx)

- 04. Import souborů z IBM SPSS (.sav)

Rekapitulace

Packages are collections of **R functions**, **data**, and **compiled code** in a well-defined format.
- The directory where packages are stored is called the library.
- **R** comes with a standard set of packages.
- Others are available for download and installation.
  - Once installed, they have to be loaded into the session to be used.

```
# get library location
.libPaths()

# see all packages installed
library()

# see packages currently loaded
search()
```

```
# nainstaluje konkrétní balíček
install.packages("psych")

# načte konkrétní balíček
library(psych)
```

If statistics programs/languages were cars...
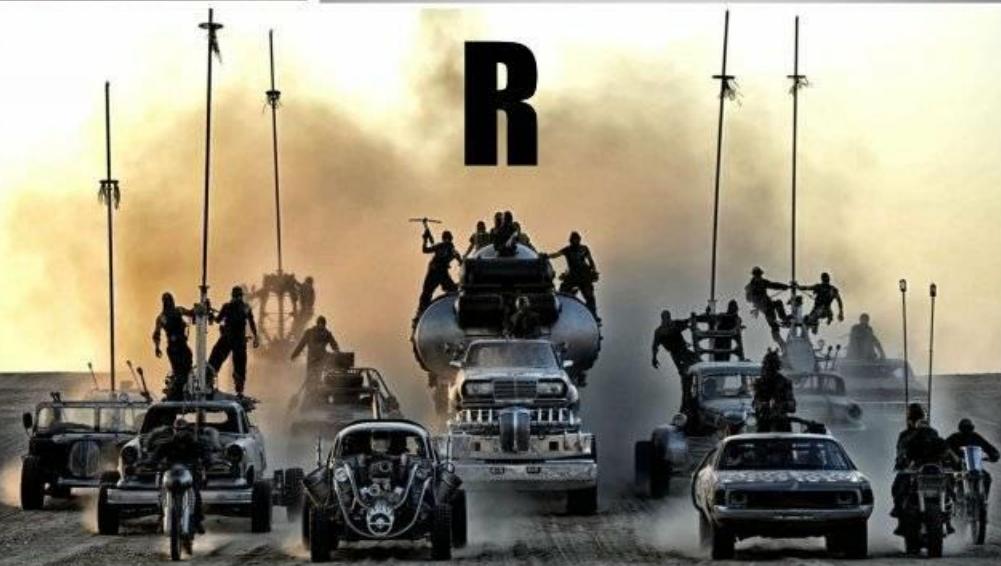
Import dat

Obecně

```
# Zjištění pracovní složky (get working directory)
getwd()

# Nastavení pracovní složky (set working directory)
setwd("…/Data")

nebo

setwd("…\\Data")
```

Import dat
Flat Files – Utils - .csv

```
# Import swimming_pools.csv:
pools = read.csv("swimming_pools.csv")

# Print the structure of pools
str(pools)

# Import swimming_pools.csv correctly: pools
pools = read.csv("swimming_pools.csv", stringsAsFactors = FALSE)

# Check the structure of pools
str(pools)
```

Import dat

Flat Files – Utils - .txt

```
hotdogs_1 = read.delim("hotdogs_1.txt", header = TRUE)
hotdogs_2 = read.delim("hotdogs_2.txt", header = FALSE, col.names = c("type", "calories",
"sodium"))

summary(hotdogs_1)
str(hotdogs_1)

# Select the hot dog with the least calories: Cal
Cal <- hotdogs_1[which.min(hotdogs_1$Calories), ]

# Select the observation with the most sodium: Sod
Sod = hotdogs_1[which.max(hotdogs_1$Sodium), ]

str(hotdogs_1)
```

Import dat

Excel – readxl

```
# Instalace a nahrání balíčku
install.packages("readxl")
library(readxl)

# Dva základní příkazy:
excel_sheets() # Výčet listů v daném excelovském (.xls, .xlsx) souboru
read_excel() # Načtení souboru excelovského formátu

excel_sheets("latitude.xlsx")
```

Import dat

Excel – readxl

```
# Read the first sheet of latitude.xlsx:
latitude_1 = read_excel("latitude.xlsx", sheet = "1700")
latitude_1


# Read the second sheet of latitude.xlsx:
latitude_2 = read_excel("latitude.xlsx", sheet = 2)
latitude_2


# Put latitude_1 and latitude_2 in a list:
lat_list = list(latitude_1, latitude_2)
```

Import dat

Excel – readxl – col_names

Apart from path and sheet, there are several other arguments you can specify in read_excel(). One of these arguments is called col_names.

```
# Import the the first Excel sheet of latitude_nonames.xlsx (R gives names):
latitude_3 = read_excel("latitude.xlsx", sheet = 3, col_names = FALSE)
latitude_3

# Import the the first Excel sheet of latitude_nonames.xlsx (specify col_names):
latitude_4 = read_excel("latitude.xlsx", sheet = 3, col_names = c("country", "latitude"))
latitude_4

# Print the summary of latitude_3
summary(latitude_3)

# Print the summary of latitude_4
summary(latitude_4)
```

Import dat

Excel – [readxl](#) – skip

Another argument that can be very useful when reading in Excel files that are less tidy, is skip.
With skip, you can tell R to ignore a specified number of rows inside the Excel sheets you're trying to pull data from.

Have a look at this example:

read_excel("latitude.xlsx", skip = 15)

In this case, the first 15 rows in the first sheet of "data.xlsx" are ignored.

Pozor na posunutí matice!

read_excel("latitude.xlsx", skip = 15, col_names = FALSE)

Import dat

Excel – [readxl](#) – slučování listů do jedné matice a chybějící hodnoty

```
latitude_all <- cbind(latitude_1, latitude_2[-1])
latitude_all

# Argument [-1] se týká prvního sloupce v rámci dané matice

# Remove all rows with NAs from latitude_all
latitude_all_clean = na.omit(latitude_all)

# Print out a summary of latitude_all
summary(latitude_all_clean)
```

Import dat
SPSS - foreign

```
# Balíček foreign (základní součást R)
library(foreign)

# K načtení dat z SPSS (.sav, .por) slouží příkaz read.spss()
```
- Aby měla nahraná data povahu data frame, je nutné uvnitř příkazu read.spss() jako argument zadat "to.data.frame = TRUE"

```
# Načtení dat
demo_1 = read.spss(".../international.sav", to.data.frame = TRUE)

# Načtení několika prvních řádků
head(demo_1)
```

Import dat
SPSS - foreign

Jak nastavit 'value labels' z SPSS jako 'factors' v R?

Skrze argument "**se.value.labels**" v rámci příkazu "**read.spss()**". Tento
argument upřesňuje, zda mají být "**value labels**" konvertovány do R jako
"**factors**".
- Argument je "TRUE by default", výchozím stavem je tedy provedení výše uvedené konverze

```
# Načtení dat
demo_2 = read.spss("…/international.sav", to.data.frame = TRUE, use.value.labels =FALSE)

# Načtení několika prvních řádků
head(demo_2)
```

Import dat

SPSS - foreign

Jak nastavit 'value labels' z SPSS u 'factors' v R u dílčích proměnných?

```
# Summary demo_2$contint
summary(demo_2$contint)
class(demo_2$contint)

# Konverze demo_2$contint na faktor
demo_2$contint = as.factor(demo_2$contint)

# Summary demo_2$contint znovu
summary(demo_2$contint)
class(demo_2$contint)
```

Jak nastavit 'value labels' z SPSS jako 'factors' u dílčích proměnných v R?

```
continents = c("Africa", "Americas", "Asia", "Europe")
demo_2$contint = factor(demo_2$contint, levels = c(1, 2, 3, 4), labels = continents)
summary(demo_2$contint)
```

Zdroje

Packages (n.d.) Packages. In Quick-R. Staženo dne 2. 10. 2016 z
http://www.statmethods.net/interface/packages.html

Prostý databázový soubor. (n.d.). In Wikipedia. Staženo dne 2. 10. 2016 z
https://cs.wikipedia.org/wiki/Prost%C3%BD_datab%C3%A1zov%C3%BD_soubor