VKLÁDÁNÍ A ČIŠTĚNÍ DAT, ZJIŠŤOVÁNÍ ZÁKLADNÍCH INFORMACÍ O DATOVÉM SOUBORU

Vít Gabrhel

vit.gabrhel@mail.muni.cz

FSS MU,

2. 10. 2017

Harmonogram

0. Rekapitulace předchozí hodiny

1. Importování dat do R

2. Čištění dat

3. Popisné statistiky

Rekapitulace

Balíčky (dle Quick-R, n.d.)

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")
```


R PACKAGES BE LIKE

Import dat

Obecně

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

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

nebo

setwd("C:...****Data")

Import dat

Flat Files (= Prostý dabázový soubor)

= *Jednoduchá databáze* (většinou **tabulka**) uložená v **textovém souboru** ve formě **prostého textu** (Prostý databázový soubor, n.d.)

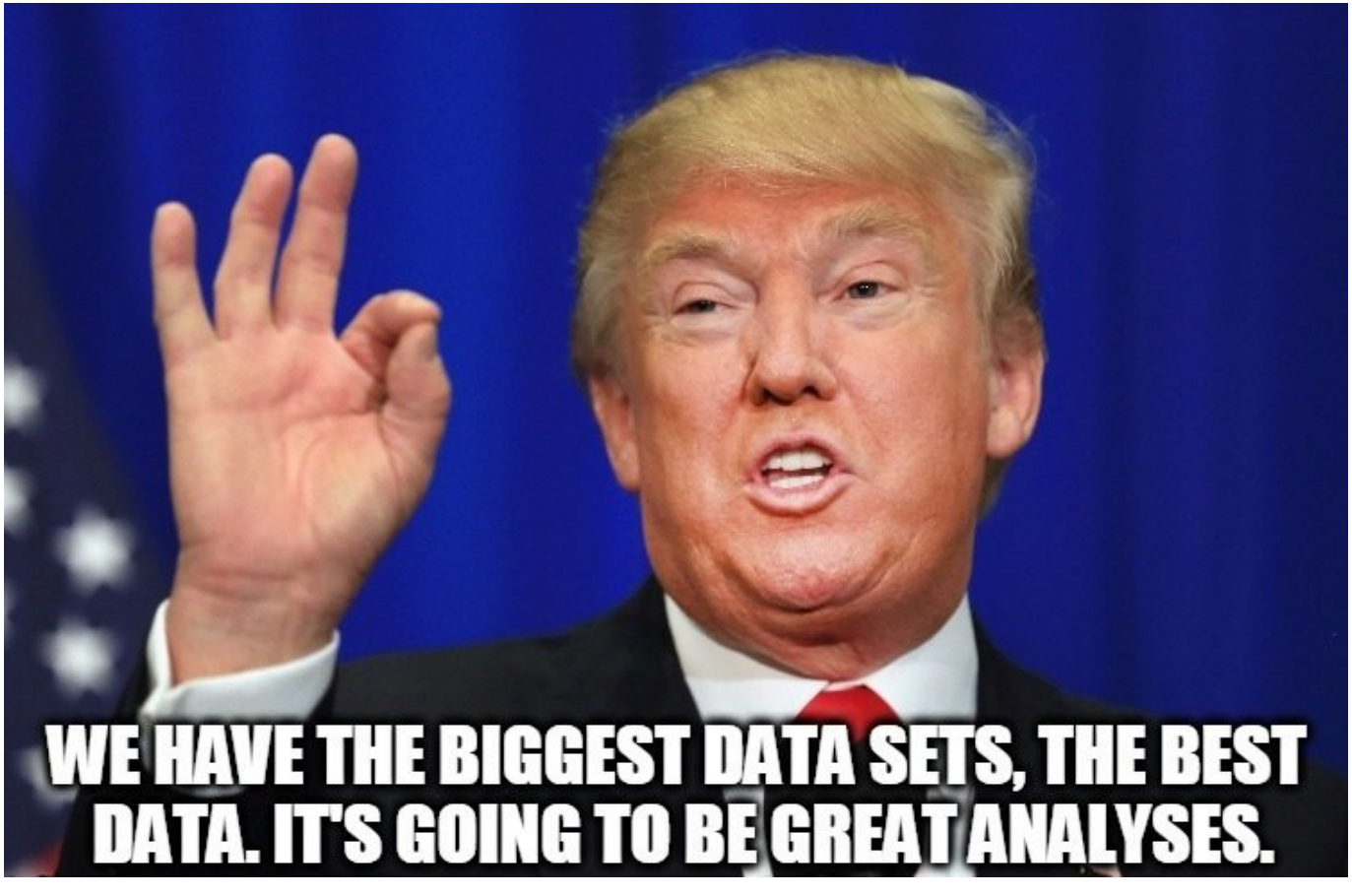
- .csv (comma-separated values)
- .txt

Existence celé řady balíčků odlišených podle preferovaného formátu (.csv, .txt) a míry automatizace (resp. počtu argumentů, které je třeba specifikovat).

Součástí R je balíček "**utils**":

- read.table(sep = "")
- read.csv(sep =)
- read.csv2(sep = ";")
- read.delim(sep = "\t")

?read.table


WE HAVE THE BIGGEST DATA SETS, THE BEST DATA. IT'S GOING TO BE GREAT ANALYSES.

2017 Military Strength Ranking

The complete Global Firepower list for 2017 puts the military powers of the world into full perspective.

The finalized Global Firepower ranking relies on over 50 factors to determine a given nation's PowerIndex ('PwrIndx') score. Our formula allows smaller, though more technologically-advanced, nations to compete with larger, lesser-developed ones. Modifiers (in the form of bonuses and penalties) are added to further refine the list. Some items to observe in regards to the finalized ranking:

Import dat

Flat Files - Utils - .csv

```
# Import Manpower.csv:
Manpower = read.csv("Manpower.csv")

# Print the structure of Manpower
str(Manpower)

# Import Manpower.csv correctly: Manpower2
Manpower2 = read.csv("Manpower.csv", stringsAsFactors = FALSE)

# Check the structure of pools
str(Manpower2)
```

Import dat

Flat Files - Utils - .txt

```
Naval_1 = read.delim("Naval_1.txt", header = TRUE)
Naval_2 = read.delim("Naval_2.txt", header = FALSE, col.names = c("Country", "ISO3", "Rank",
"Total Naval Assets", "Aircraft Carriers", "Frigates", "Destroyers", "Corvettes",
"Submarines", "Patrol Craft", "Mine Warfare Vessels"))


summary(Naval_1)
str(Naval_1)


# Select the country with the least naval assets: Total_Naval_Assets
Total_Naval_Assets <- Naval_1[which.min(Naval_1$Total.Naval.Assets), ]


# Select the country with the most submarines: Submarines
Submarines = Naval_1[which.max(Naval_1$Submarines), 9]
```

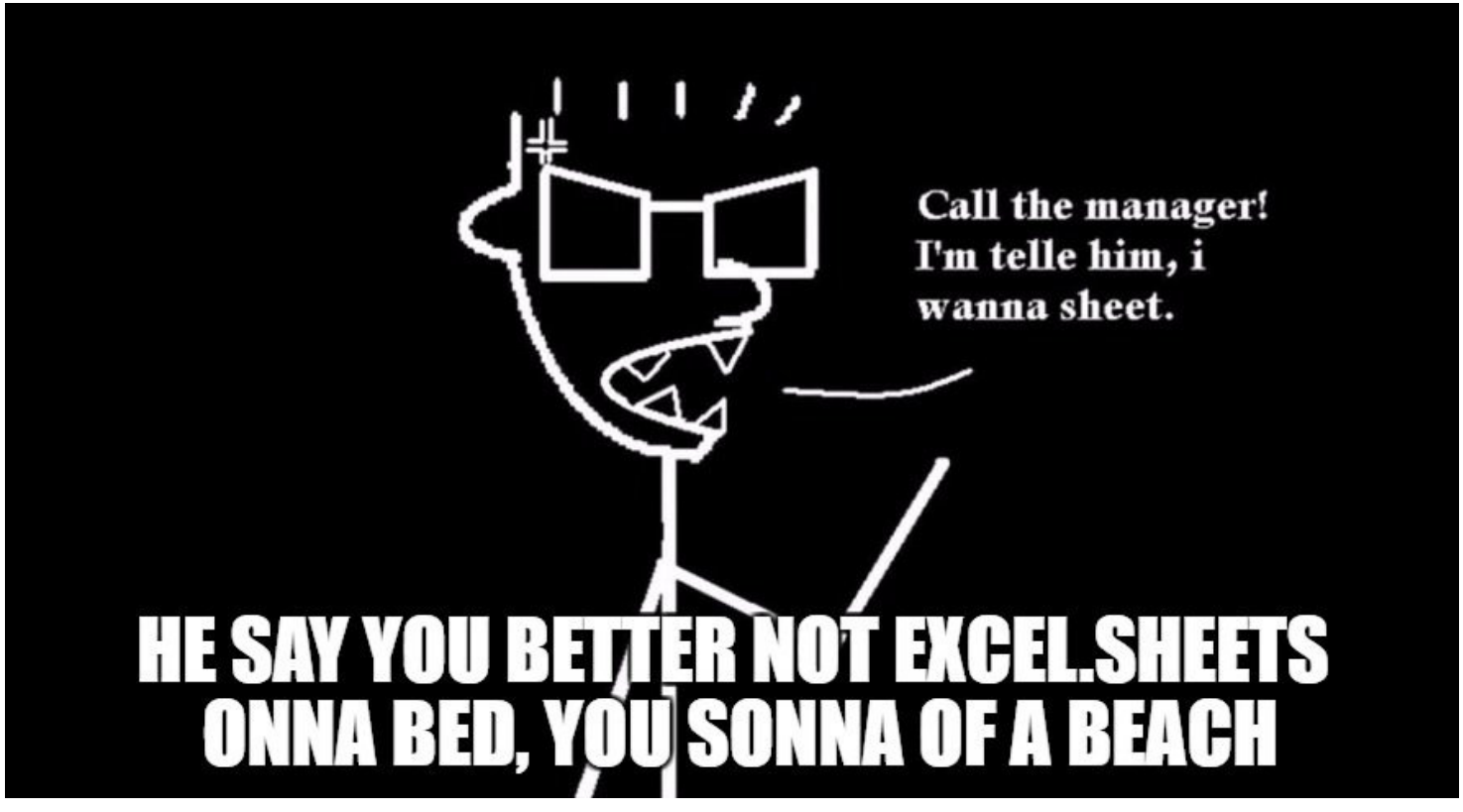
Import dat

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("Resources.xlsx")
```


Call the manager!
I'm telle him, i
wanna sheet.
HE SAY YOU BETTER NOT EXCEL.SHEETS
ONNA BED, YOU SONNA OF A BEACH

Import dat

Excel - readxl

```
# Read the first sheet of Resources.xlsx:
Population = read_excel("Resources.xlsx", sheet = "Population")
View(Population)

# Read the second sheet of Resources.xlsx:
Airports = read_excel("Resources.xlsx", sheet = 2)
View(Airports)

# Put Population and Airports in a list:
Resources_List = list(Population, Airports)
View(Resources_List)
```

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 third Excel sheet of Resources.xlsx (R gives names):
Population2 = read_excel("Resources.xlsx", sheet = 3, col_names = FALSE)
View(Population2)

# Import the the third Excel sheet of Resources.xlsx (specify col_names):
Population3= read_excel("Resources.xlsx", sheet = 3, col_names = c("Country", "ISO3", "Rank",
"Population"))
View(Population3)

# Print the summary of Population2
summary(Population2)

# Print the summary of Population3
summary(Population3)
```

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:

Airports2 = read_excel("Resources.xlsx", sheet = 4, **skip = 15**)

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

Pozor na posunutí matice!

Airports3 = read_excel("Resources.xlsx", sheet = 4, skip = 15, **col_names = FALSE**)

Import dat

Excel - readxl - slučování listů do jedné matice a chybějící hodnoty

```
Resources_all <- cbind(Population, Airports[-1:-3])
View(Resources_all)

# Argument [-1:-3] se týká prvních tří sloupců v rámci dané matice

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

# Print out a summary of Population
summary(Population_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 jako "factors" u dílčích proměnných v R?

```
# 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 u "factors" v R u dílčích proměnných?

```
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)
```


MISSING DATA?
MULTIPLE AMPUTATION!


HYNEK CÍGLER MŮŽE ZA VAŠE MEMY.

Čištění dat

Explorace hrubých dat - base

```
# Matice
bmi_1 = read_excel("bmi.xlsx", sheet = 2)

# Check the class of bmi
class(bmi_1)

# Check the dimensions of bmi
dim(bmi_1)

# View the column names of bmi
colnames(bmi_1)

# Struktura dat
str(bmi_1)

# Sumarizace
summary(bmi_1)

# Prvních 10 a posledních 10 řádků
head(bmi_1, n = 10)
tail(bmi_1, n = 10)
```

Čištění dat

Explorace hrubých dat - psych

```
# Load psych
install.packages("psych")
library("psych")

# Check the structure of bmi, the psych way
describe(bmi_1)
```

Čištění dat

Explorace hrubých dat - grafy

```
# Matice
bmi_2 = read_excel("bmi.xlsx", sheet = 3)
bmi_all = cbind(bmi_1, bmi_2[-1])

# Histogram
hist(bmi_1$BMI_1980)

# Scatterplot
plot(bmi_all$BMI_1980, bmi_all$BMI_2000)
```

Čištění dat

Příprava dat pro analýzu

```
Infrastructure= read.csv2("Infrastructure.csv")

# Preview Infrastructure with str()
str(Infrastructure)

# Coerce Country to character
Infrastructure$Country <- as.character(Infrastructure$Country)

# Coerce Rank to factor
Infrastructure$Rank <- as.character(Infrastructure$Rank)

# Look at Infrastructure once more with str()
str(Infrastructure)
```

Čištění dat

Příprava dat pro analýzu - dílčí manipulace se strings

```
# Load the stringr package
install.packages("stringr")
library("stringr")

# Trim all leading and trailing whitespace
name = c("    Filip ", "Nick  ", " Jonathan")
str_trim(name)

# Pad these strings with leading zeros
pad = c("23485W", "8823453Q", "994Z")
str_pad(pad, width = 9, side = "left", pad = "0")

# Print state abbreviations
Population$country

# Make states all uppercase and save result to states_upper
states_upper = toupper(Population$Country)
states_upper

# Make states_upper all lowercase again
states_lower = tolower(Population$Country)
states_lower
```

Čištění dat

Příprava dat pro analýzu - dílčí manipulace se strings

```
# Look at the head of Infrastructure
head(Infrastructure)

# Detect all "Republic" in Country
str_detect(Infrastructure$Country,
"Republic")
```

```
# In the Country column, replace
"Republic" with "R"...

Infrastructure$Country <-
str_replace(Infrastructure$Country,
"Republic", "R")
```

Čištění dat

Příprava dat pro analýzu - missing values

```
name = c("Jerry", "Beth", "Rick", "Morty")
n_friends = c(0, NA, NA, 2)
status = c("Listening to human music", "Happy Family", "Garage", "")
social_df = data.frame(cbind(name, n_friends, status))
```

```
# Call is.na() on the full social_df to spot all NAs
is.na(social_df)
```

```
# Use the any() function to ask whether there are any NAs in the data
any(is.na(social_df))
```

```
# View a summary() of the dataset
summary(social_df)
```

```
# Call table() on the status column
table(social_df$status)
```

```
# Replace all empty strings in status with NA
social_df$status[social_df$status == ""] <- NA
```

```
# Print social_df to the console
social_df
```

```
# Use complete.cases() to see which rows have no missing values
complete.cases(social_df)
```

```
# Use na.omit() to remove all rows with any missing values
na.omit(social_df)
```

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