National Czech Programme | eosc

# R pro analýzu vědeckých dat

Mgr. Lucie Hošková

Spolufinancováno Evropskou unií

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

MUNI ICS

**Registrační číslo IPs EOSC-CZ**

CZ.02.01.01/00/22_004/0007682

# Obsah

- Instalace R a Rstudio
- R vs. Rstudio: jaký je rozdíl?
- Stáhnutí dat
- Nastavení vizuálu
- Vytvoření projektu
- Popis oken
- Klávesové zkratky

National Czech Programme | eosc

# Instalace R a RStudio

## R

https://mirrors.nic.cz/R/

dostupné pro Windows, macOS i Linux

## Rstudio

Desktop verze: odkaz

dostupné pro Windows, macOS i Linux

# R vs. Rstudio: jaký je rozdíl?

**R**

- open source jazyk pro zpracovávání dat

**RStudio**

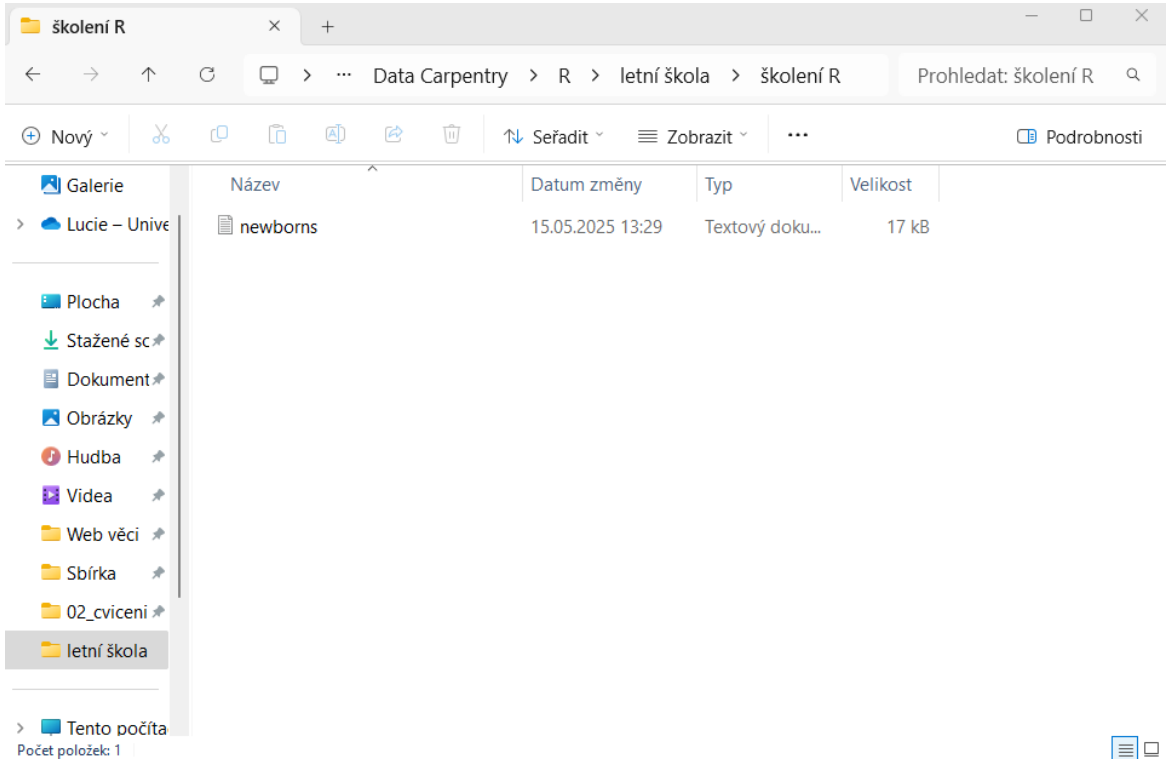
- open source vývojové prostředí pro práci s jazykem R
- zobrazuje kromě zdrojového řádku navíc oddělenou konzoli či globální prostředí, celkovo uživatelsky přívětivější

**Budeme pracovat pouze přímo uvnitř prostředí programu RStudio!**

National Czech Programme | eosc

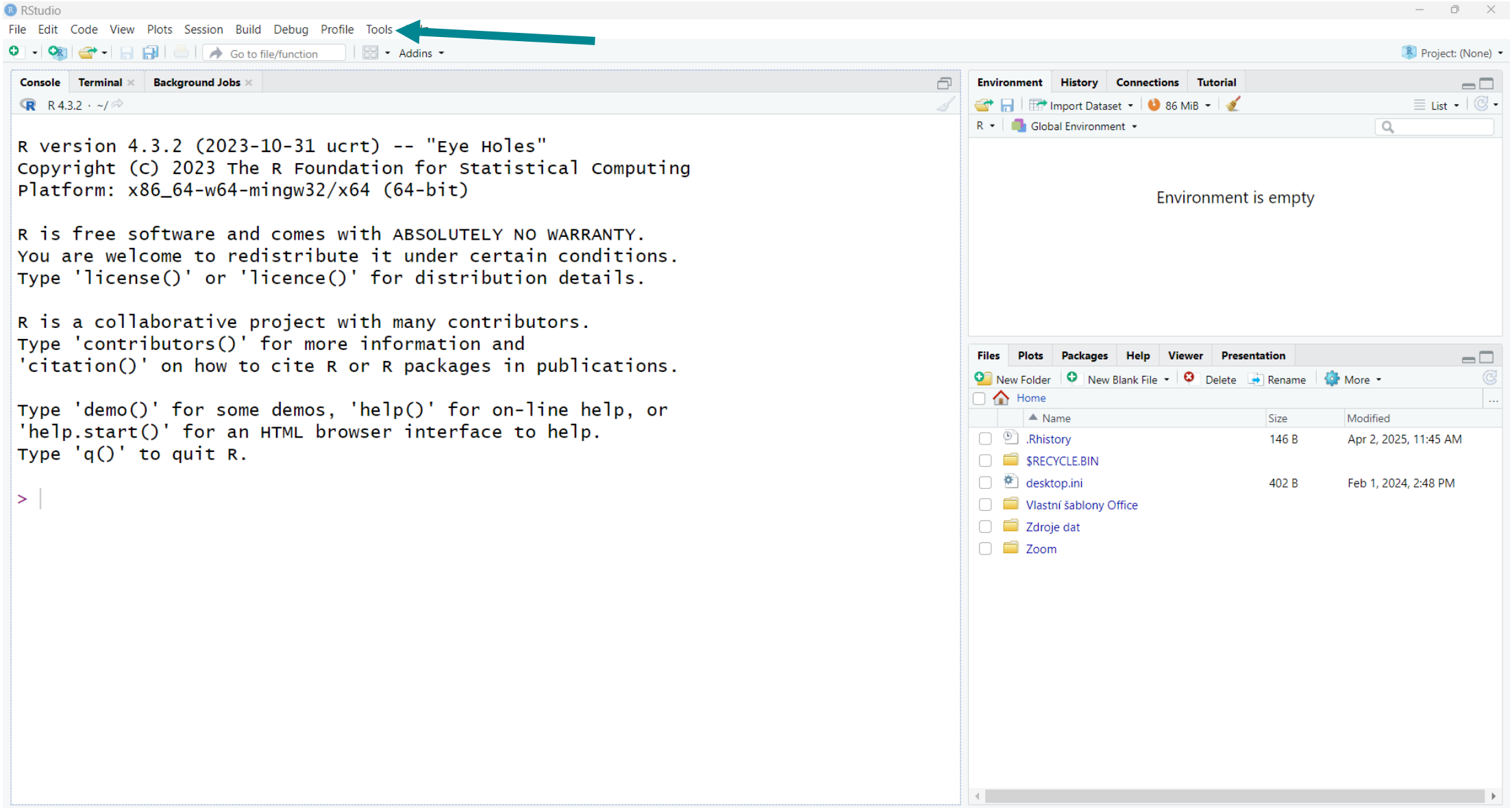
# Stáhnutí dat

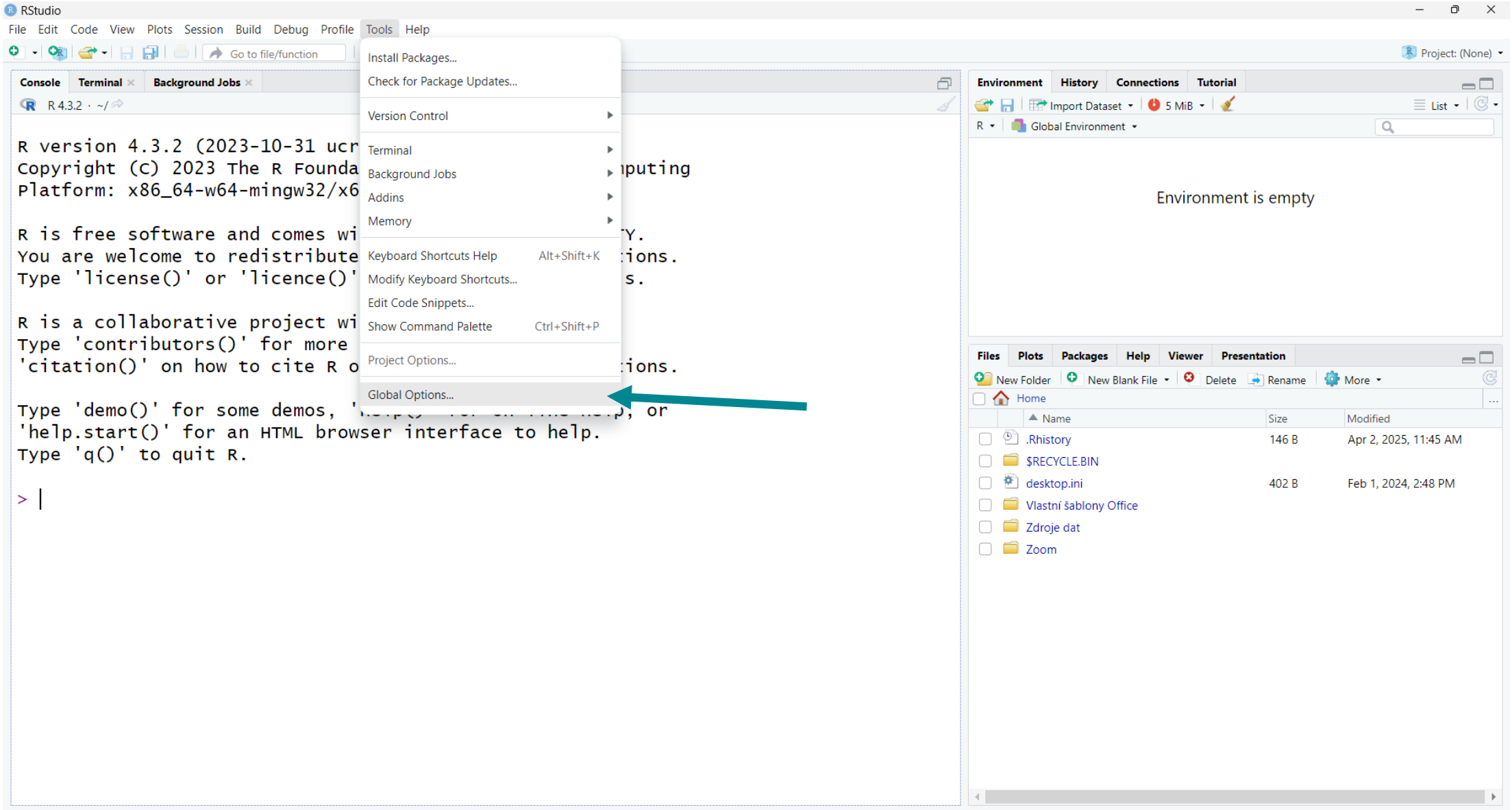
- Během školení se bude pracovat na předem připravených datech. **Stáhněte si proto prosím následující soubor „newborns.txt“**, a uložte si jej do nové složky, která bude sloužit pouze pro soubory k tomuto školení. Neukládejte do ní prosím v tuto chvíli žádné další soubory a uložte si ji na svém počítači na takovém místě, kde ji budete schopni opět snadno najít.

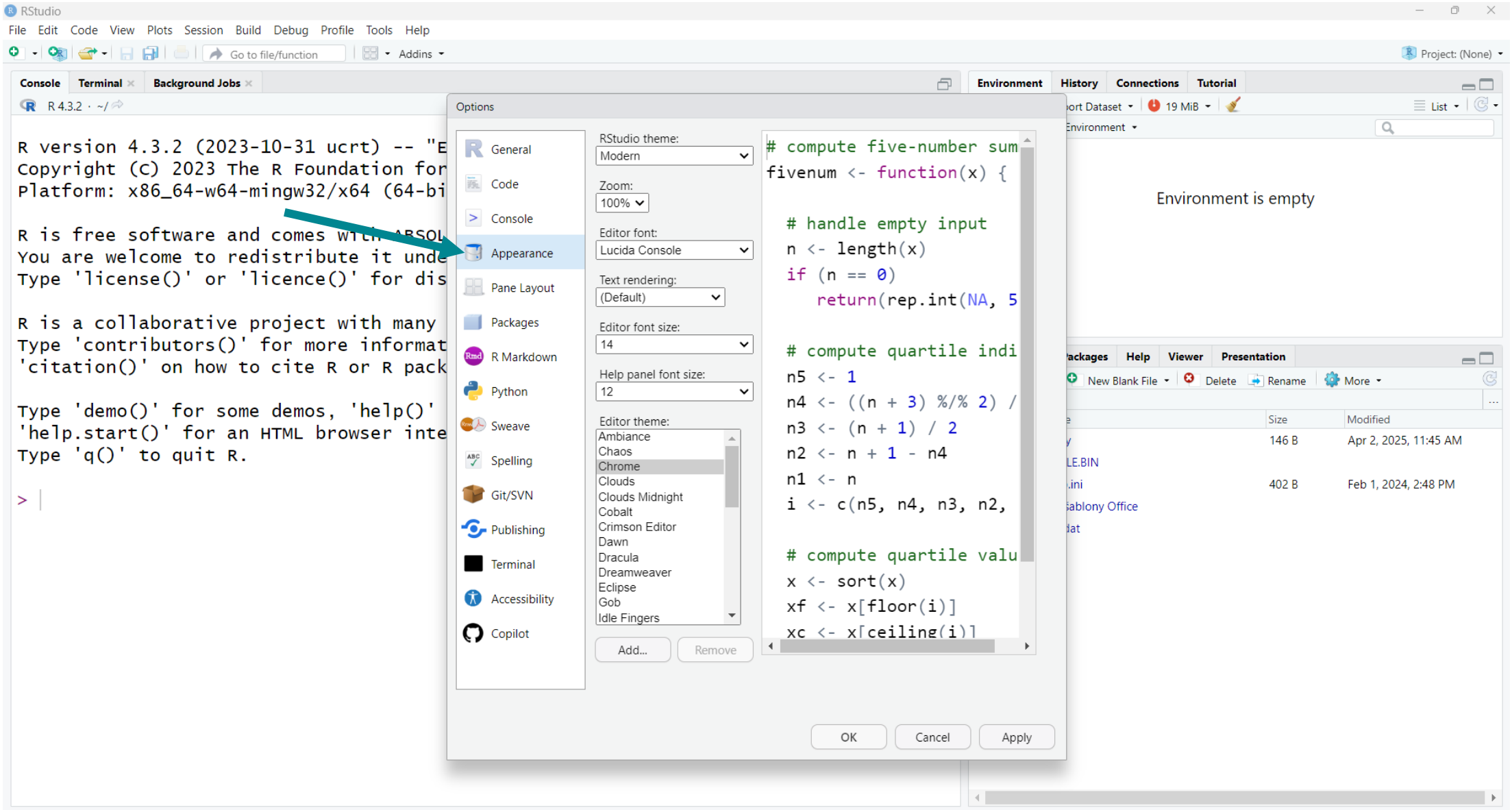


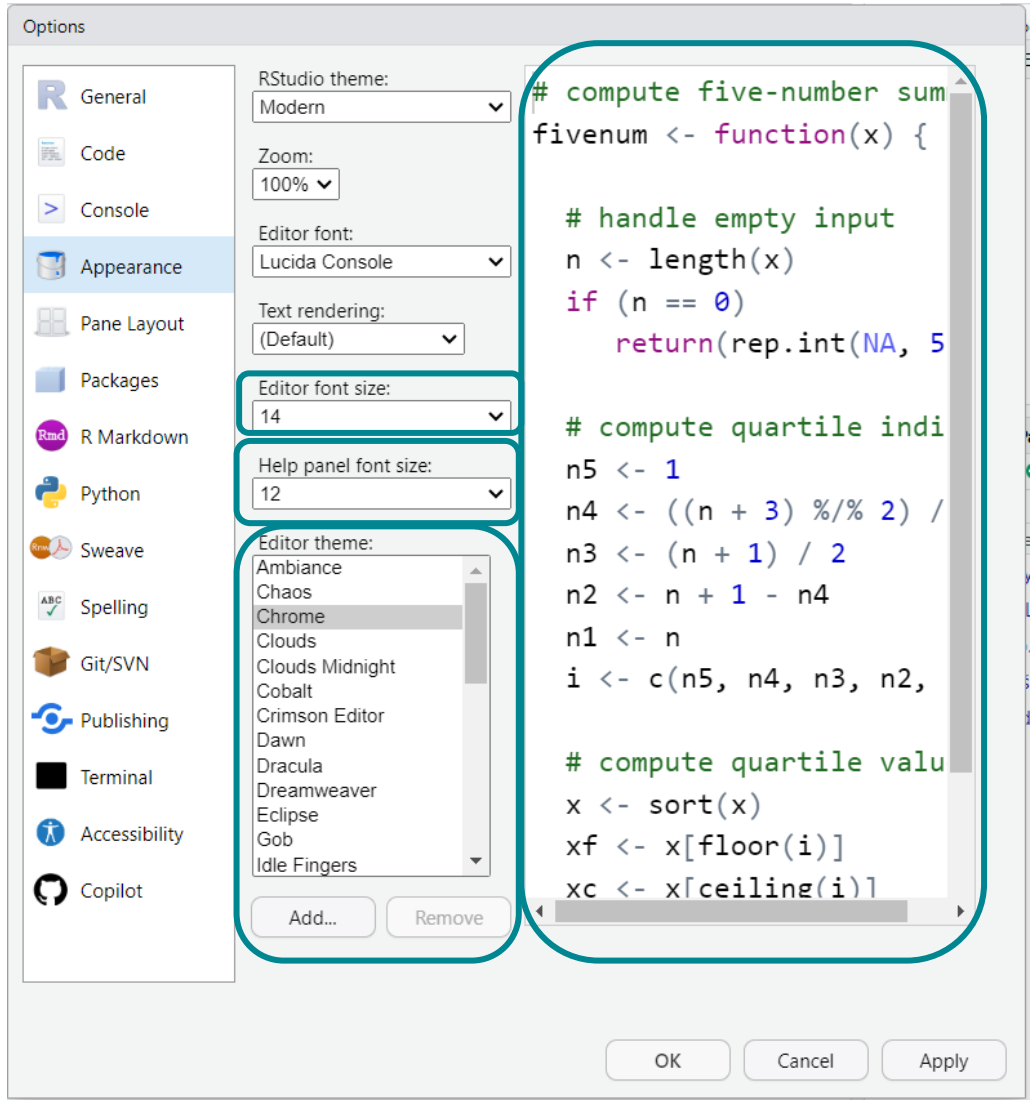
# Nastavení vizuálu RStudia

- Předtím než začnete v Rstudiu pracovat, je dobré nastavit si vizuál rozhraní.
- Můžete samozřejmě pracovat i se základním nastavením, ale doporučuji si i tak ale nastavení alespoň prohlédnout.
- Na následujících snímcích máte návod krok za krokem, jak tuto změnu vizuálu provést.


RStudio
File Edit Code View Plots Session Build Debug Profile Tools
Go to file/function
Addins
Project: (None)
Console
Terminal
Background Jobs
R 4.3.2 · ~/
R version 4.3.2 (2023-10-31 ucrt) -- "Eye Holes"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
Environment
History
Connections
Tutorial
Import Dataset
86 MiB
List
R
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
Presentation
New Folder
New Blank File
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
146 B
Apr 2, 2025, 11:45 AM
$RECYCLE.BIN
desktop.ini
402 B
Feb 1, 2024, 2:48 PM
Vlastní šablony Office
Zdroje dat
Zoom


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Project: (None)
Console Terminal Background Jobs
R 4.3.2 · ~/
R version 4.3.2 (2023-10-31 ucr
Copyright (C) 2023 The R Founda
Platform: x86_64-w64-mingw32/x6
R is free software and comes wi
You are welcome to redistribute
Type 'license()' or 'licence()'
R is a collaborative project wi
Type 'contributors()' for more
'citation()' on how to cite R o
Type 'demo()' for some demos,
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
Install Packages...
Check for Package Updates...
Version Control
Terminal
Background Jobs
Addins
Memory
Keyboard Shortcuts Help Alt+Shift+K
Modify Keyboard Shortcuts...
Edit Code Snippets...
Show Command Palette Ctrl+Shift+P
Project Options...
Global Options...
Environment History Connections Tutorial
Import Dataset
5 MiB
List
R
Global Environment
Environment is empty
Files Plots Packages Help Viewer Presentation
New Folder
New Blank File
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory 146 B Apr 2, 2025, 11:45 AM
$RECYCLE.BIN
desktop.ini 402 B Feb 1, 2024, 2:48 PM
Vlastní šablony Office
Zdroje dat
Zoom

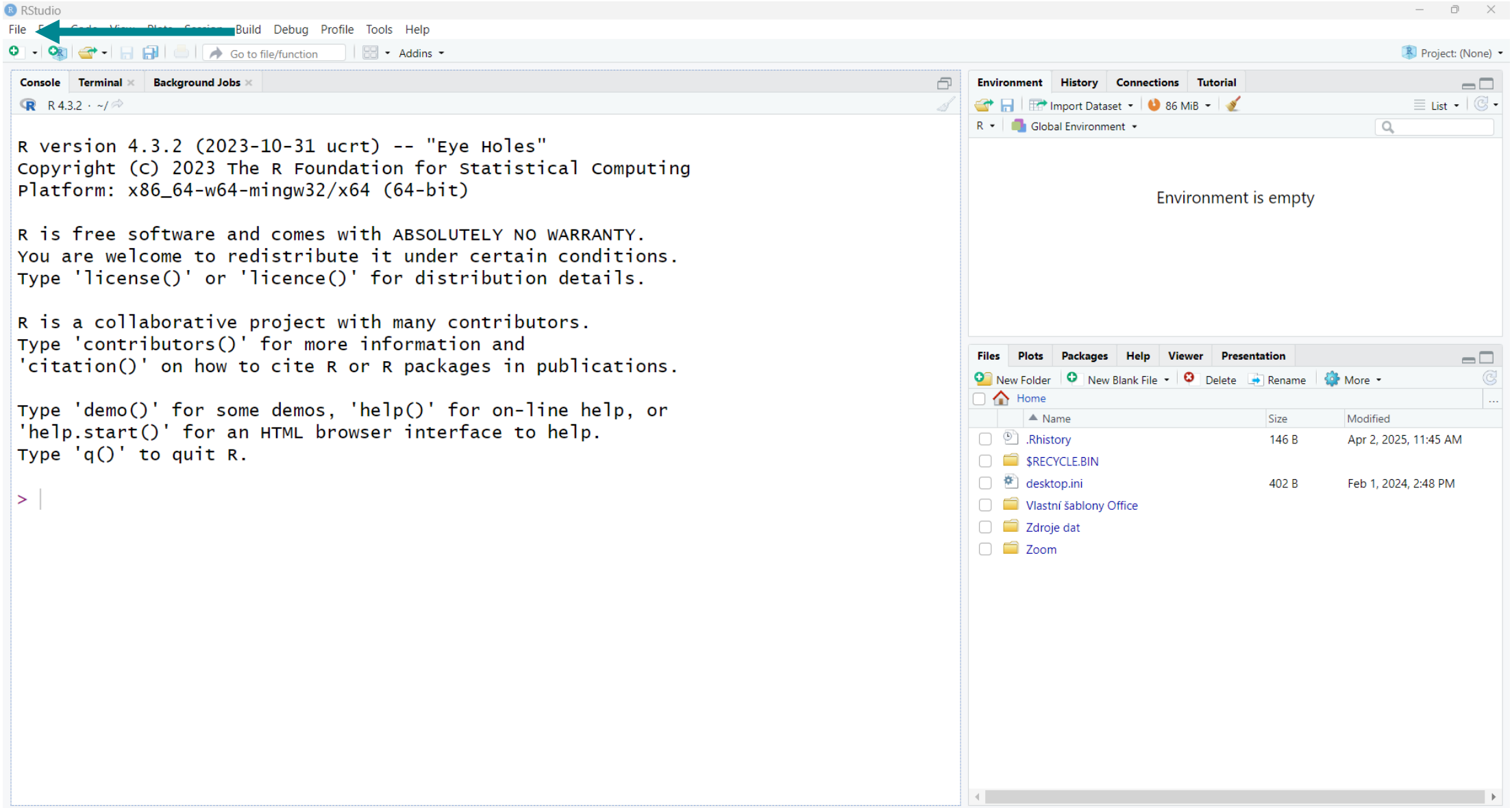

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Console
Terminal
Background Jobs
R 4.3.2 · ~/
R version 4.3.2 (2023-10-31 ucrt) -- "E
Copyright (C) 2023 The R Foundation for
Platform: x86_64-w64-mingw32/x64 (64-bi
R is free software and comes with ABSOL
You are welcome to redistribute it unde
Type 'license()' or 'licence()' for dis
R is a collaborative project with many
Type 'contributors()' for more informat
'citation()' on how to cite R or R pack
Type 'demo()' for some demos, 'help()'
'help.start()' for an HTML browser inte
Type 'q()' to quit R.
>
Environment
History
Connections
Tutorial
Import Dataset
19 MiB
List
Environment
Environment is empty
Packages
Help
Viewer
Presentation
New Blank File
Delete
Rename
More
Size
Modified
146 B
Apr 2, 2025, 11:45 AM
402 B
Feb 1, 2024, 2:48 PM
Options
General
Code
Console
Appearance
Pane Layout
Packages
R Markdown
Python
Sweave
Spelling
Git/SVN
Publishing
Terminal
Accessibility
Copilot
RStudio theme:
Modern
Zoom:
100%
Editor font:
Lucida Console
Text rendering:
(Default)
Editor font size:
14
Help panel font size:
12
Editor theme:
Ambiance
Chaos
Chrome
Clouds
Clouds Midnight
Cobalt
Crimson Editor
Dawn
Dracula
Dreamweaver
Eclipse
Gob
Idle Fingers
Add...
Remove
# compute five-number sum
fivenum <- function(x) {
# handle empty input
n <- length(x)
if (n == 0)
return(rep.int(NA, 5
# compute quartile indi
n5 <- 1
n4 <- ((n + 3) %/% 2) /
n3 <- (n + 1) / 2
n2 <- n + 1 - n4
n1 <- n
i <- c(n5, n4, n3, n2,
# compute quartile valu
x <- sort(x)
xf <- x[floor(i)]
xc <- x[ceiling(i)]
OK
Cancel
Apply

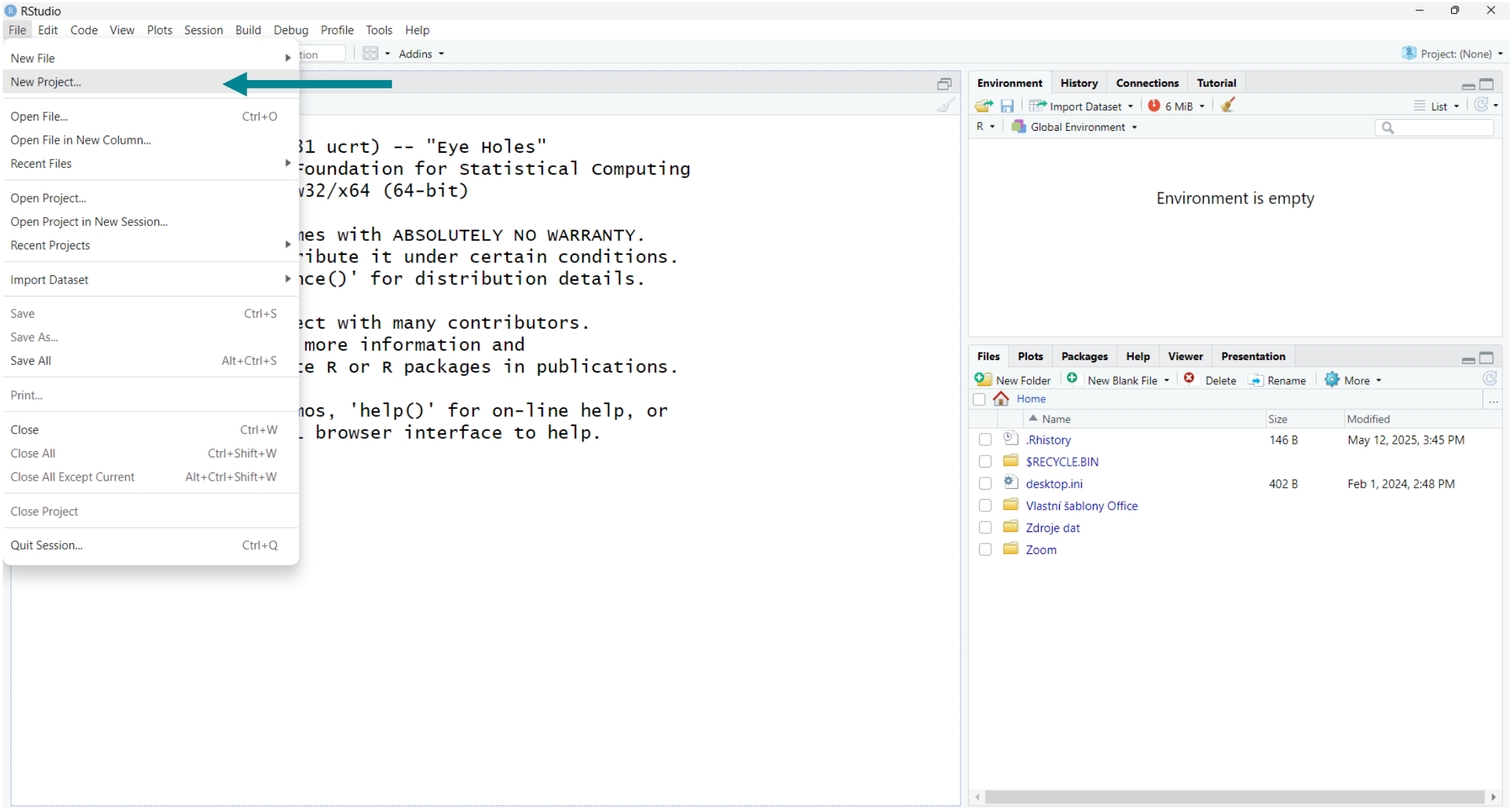



- Všechny změny které provádíte můžete vidět na příkladu kódu v pravé části okna.
- **Editor font size**: změní velikost textu kódu
- **Help panel size**: změní velikost textu nápovědy
- **Editor theme**: určuje styl a jeho barevné rozložení kódu
  - Doporučuji vybrat styl, který jasně barevně odlišuje čísla, poznámky, funkce a text (např. Cobalt).

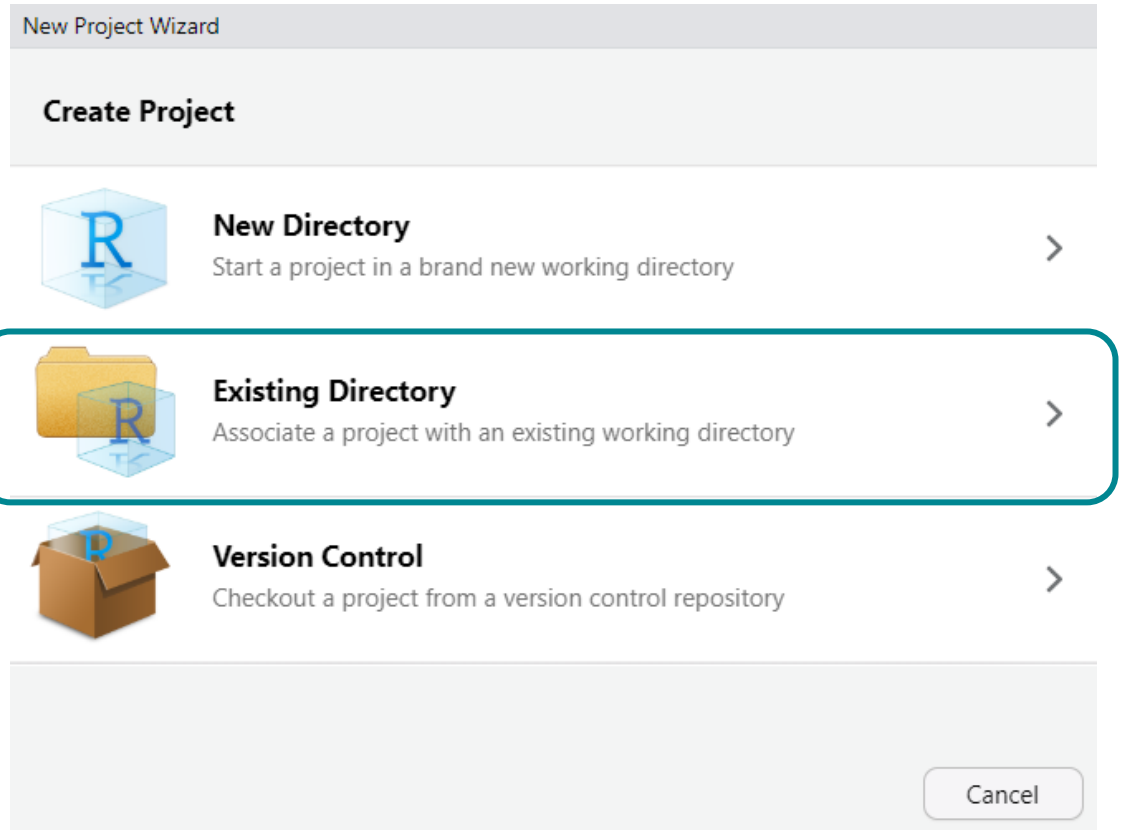
# Vytvoření projektu

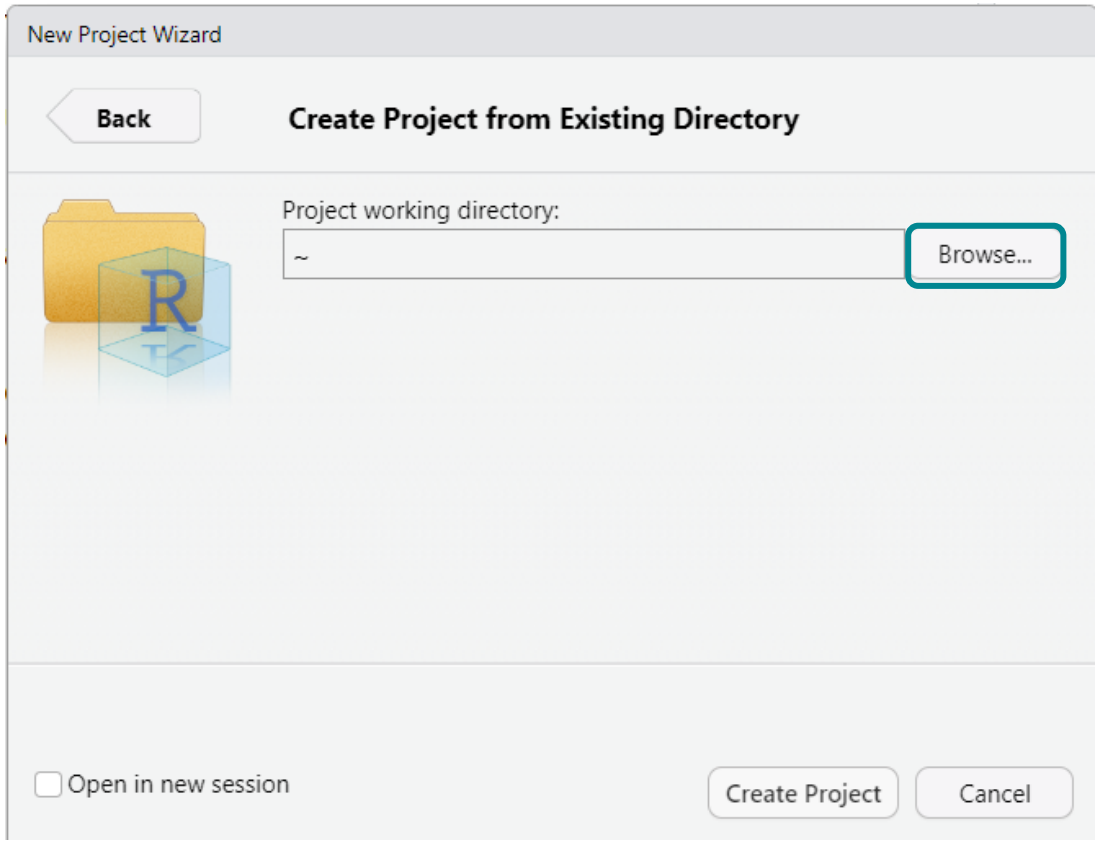
- Jako první krok pro přípravu prostředí si vytvoříme **Projekt**.
- Projekt je „složka“, do které se nám budou ukládat veškeré naše skripty, objekty, načtená data a historie naší práce.
- RStudio můžete samozřejmě používat i bez projektů, zvlášť když si jen rychle potřebujete ve skriptu projet nějaký výpočet. **Pro zpracovávání výzkumných dat v jejich celku ovšem doporučuji projekty používat**.
- Vytvořením projektu vás provedou následujících několik snímků.


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Console
Terminal
Background Jobs
R 4.3.2 · ~/
R version 4.3.2 (2023-10-31 ucrt) -- "Eye Holes"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
Environment
History
Connections
Tutorial
Import Dataset
86 MiB
List
R
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
Presentation
New Folder
New Blank File
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
146 B
Apr 2, 2025, 11:45 AM
$RECYCLE.BIN
desktop.ini
402 B
Feb 1, 2024, 2:48 PM
Vlastní šablony Office
Zdroje dat
Zoom

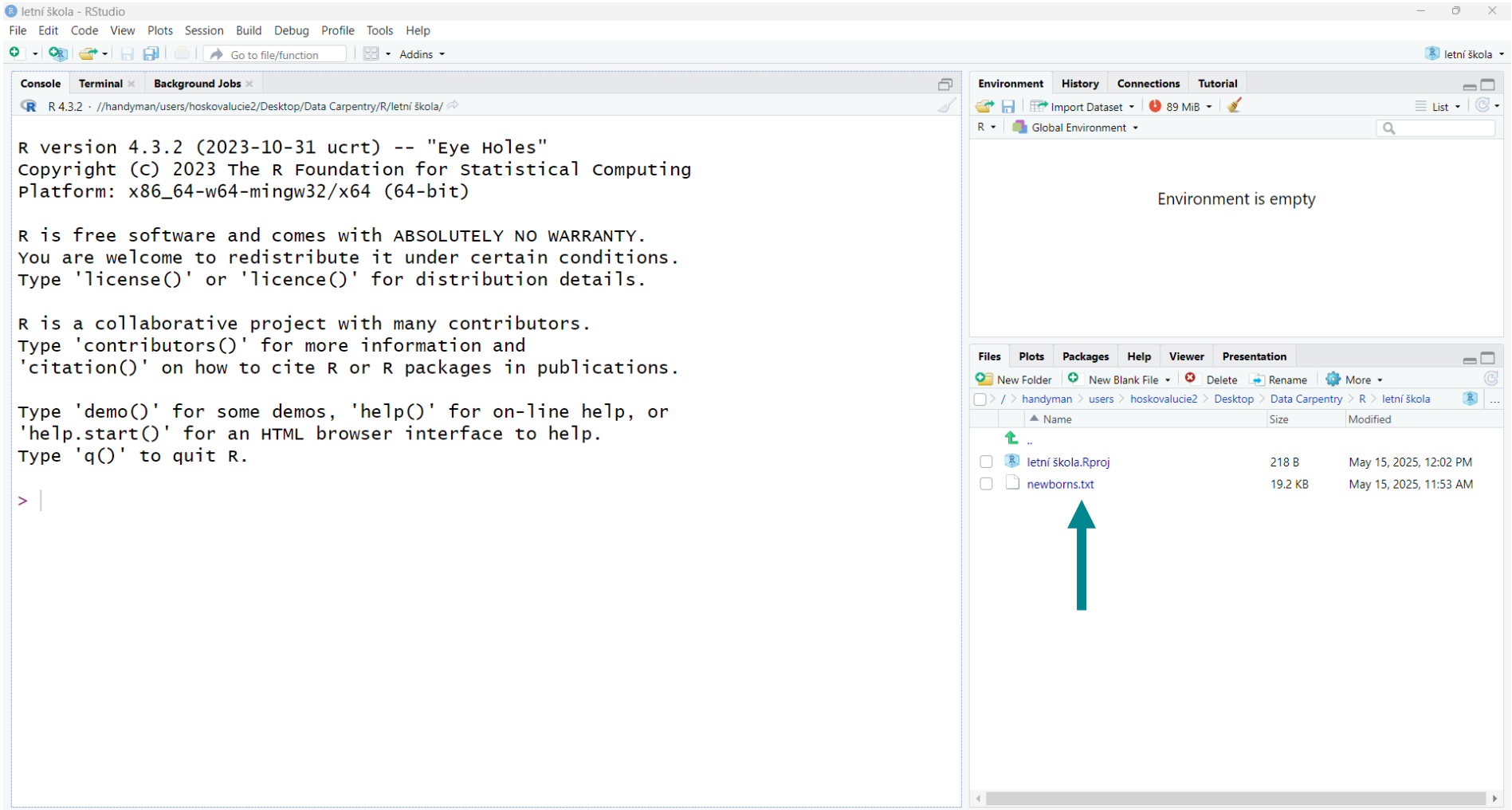

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
New File
New Project...
Open File... Ctrl+O
Open File in New Column...
Recent Files
Open Project...
Open Project in New Session...
Recent Projects
Import Dataset
Save Ctrl+S
Save As...
Save All Alt+Ctrl+S
Print...
Close Ctrl+W
Close All Ctrl+Shift+W
Close All Except Current Alt+Ctrl+Shift+W
Close Project
Quit Session... Ctrl+Q
Addins
Project: (None)
Environment History Connections Tutorial
Import Dataset
6 MiB
List
R
Global Environment
Environment is empty
Files Plots Packages Help Viewer Presentation
New Folder
New Blank File
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory 146 B May 12, 2025, 3:45 PM
$RECYCLE.BIN
desktop.ini 402 B Feb 1, 2024, 2:48 PM
Vlastní šablony Office
Zdroje dat
Zoom

- Vyberte složku, do které jste si uložili data, se kterými budeme pracovat.



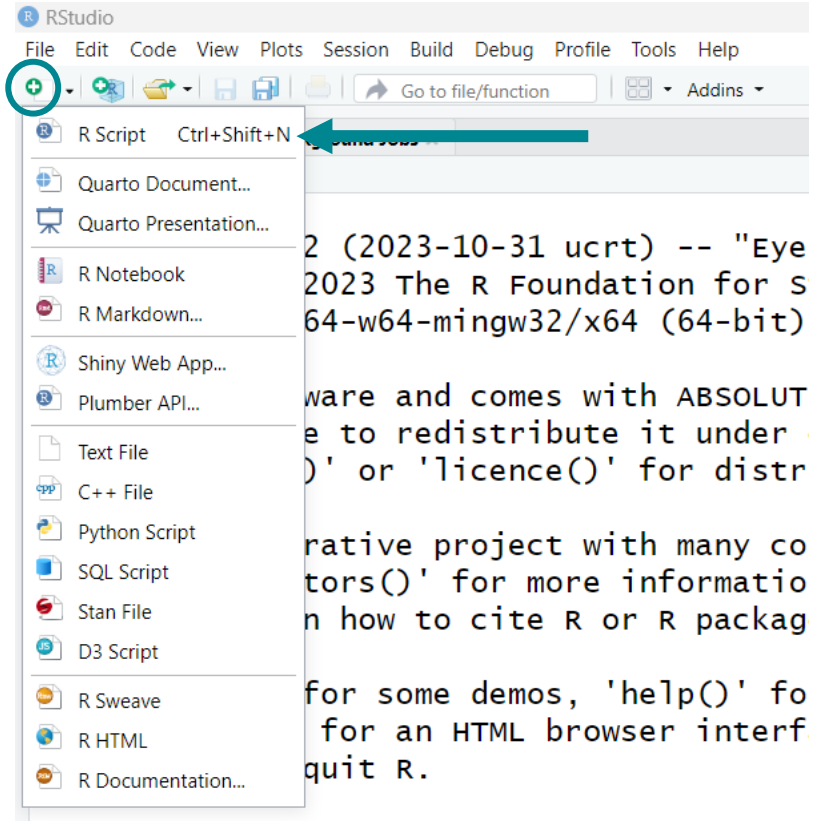



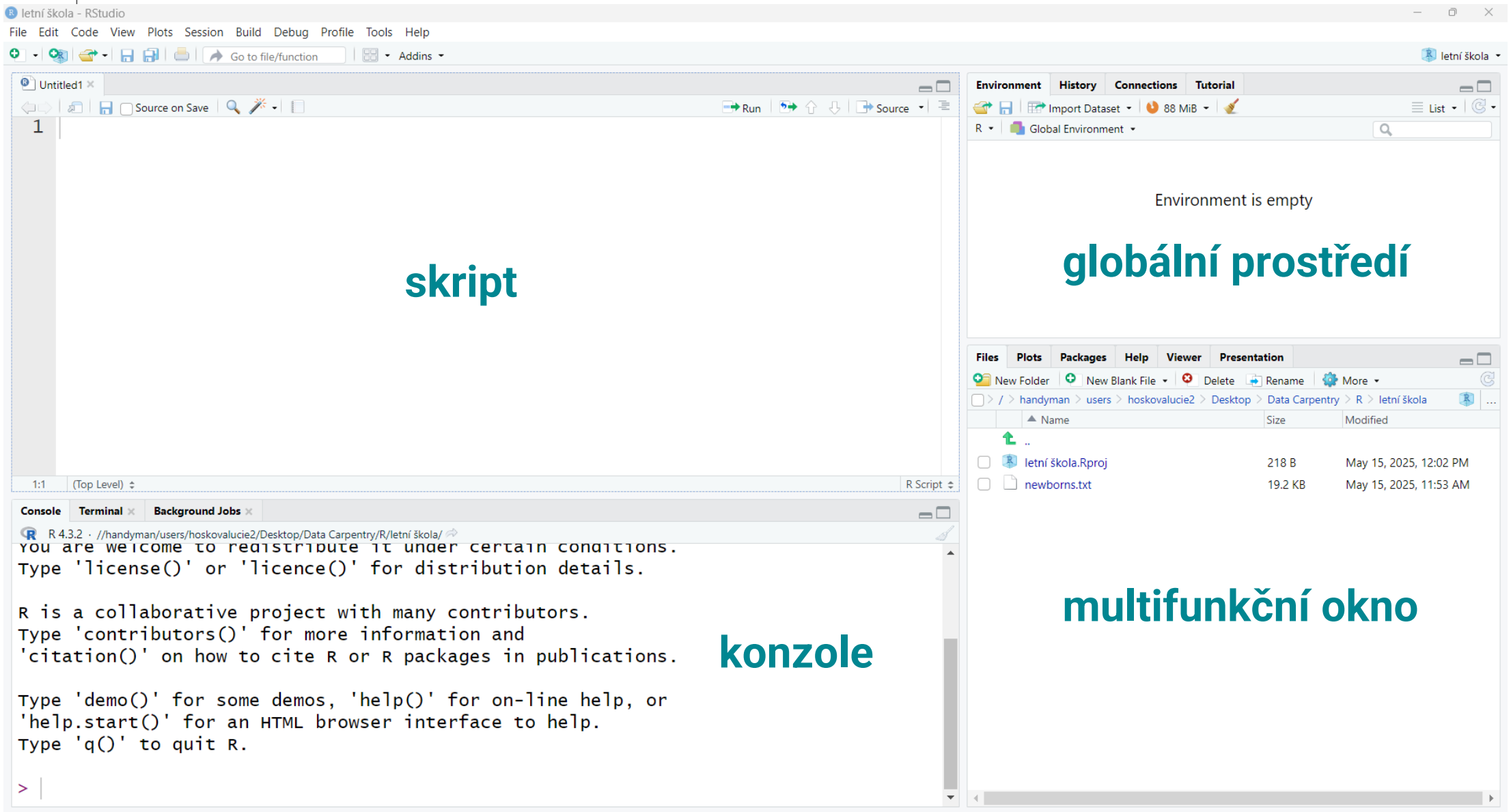
- Vaše okna by měla po vytvoření projektu vypadat následovně:



# Popis oken

- Rstudio je v základním nastavení rozloženo do čtyř oken.
- Předtím, než si jednotlivá okna popíšeme, je potřeba abyste si vytvořili nový prázdný R skript.
- Skript můžete vytvořit pomocí klávesové zkratky či přímo výběrem, viz. obrázek vpravo.




letní škola - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
letní škola
Untitled1
Source on Save
Run
Source
1
skript
1:1
(Top Level)
R Script
Environment History Connections Tutorial
Import Dataset
88 MiB
List
R
Global Environment
Environment is empty
globální prostředí
Files Plots Packages Help Viewer Presentation
New Folder
New Blank File
Delete
Rename
More
/ handyman users hoskovalucie2 Desktop Data Carpentry R letní škola
Name
Size
Modified
..
letní škola.Rproj
218 B
May 15, 2025, 12:02 PM
newborns.txt
19.2 KB
May 15, 2025, 11:53 AM
multifunkční okno
Console
Terminal
Background Jobs
R 4.3.2 · //handyman/users/hoskovalucie2/Desktop/Data Carpentry/R/letní škola/
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
konzole

- **Skript**
  - zde píšete kód
- **Konzole**
  - zde vidíte výsledky kódu a programové oznámení (např. Error)
- **Globální prostředí**
  - seznam načtených dat a přiřazených objektů
- **Multifunkční okno**
  - Files: seznam souborů v directory ve kterém pracujete
  - Plots: vytvořené grafy
  - Packages: balíčky (pluginy)
  - Help: nápověda

# Klávesové zkratky

- K psaní kódu budete potřebovat umět napsat několik specifických symbolů.
- Zde můžete najít jejich seznam společně s klávesovými zkratkami pro Windows s českou klávesnicí.
- Doporučuji si psaní symbolů vyzkoušet a během školení mít tento seznam po ruce.
- Pokud používáte jinou klávesnici a symboly proto mají jiné zkratky, doporučuji vám vytvořit si vlastní verzi seznamu zkratek.

# Klávesové zkratky

| Symbol | Zkratka | Symbol | Zkratka |
|---|---|---|---|
| # | Alt + x | > | Alt + . |
| $ | Alt + ů | < | Alt + , |
| & | Alt + c | { } | Alt + b/n |
| ~ | Alt + + | [ ] | Alt + f/g |
| \| | Alt + w | ^ | Alt + 94 |

# R pro analýzu vědeckých dat

Lucie Hošková

# Odkaz na prezentaci

https://tinyurl.com/EOSC-RproDS

# Organizace

- bloky 4 x 45 minut
  1. Úvod práce s R a Rstudiem
  2. Nahrávání dat a manipulace s dataframe
  3. Statistická analýza a testování hypotéz
  4. Vizualizace
- kombinace code-along v RStudiu a prezentace
- mock-up reálných situací práce s daty
- doplňující otázky na konci bloku

# 1. blok

- základní typy hodnot v R
- operátory
- spuštění kódu
- komentáře a pravidla psaní kódu
- vytvoření objektu
- funkce, balíčky
- typy objektů

# Typy objektů

- **Skalár**
  - objekt obsahující pouze a právě jednu hodnotu

```
1 sk1 <- 5; sk1
```

```
[1] 5
```

```
1 sk2 <- "ID"; sk2
```

```
[1] "ID"
```

- **Vektor**
  - objekt složený z x skalárů stejného typu hodnot
  - jednorozměrný objekt

Funkce **c()** vytvoří vektor z vložených hodnot¨.

```
1 cisla <- c(5, 8, 11); cisla
```

```
[1]  5  8 11
```

```
1 popisky <- c("ID", "No.", "Age"); popisky
```

```
[1] "ID"  "No." "Age"
```

## Více typů hodnot ve vektoru vede k nucené konverzi na **character**

```
1 mix <- c("ID", 2, TRUE); mix
```

```
[1] "ID"   "2"    "TRUE"
```

```
1 is.character(mix)
```

```
[1] TRUE
```

```
1 c(sk1, sk2) #muzeme kombinovat i objekty
```

```
[1] "5"  "ID"
```

- **Data frame**
  - dvojrozměrný objekt složený z vektorů stejné délky

```
1 df <- data.frame(cisla, popisky, mix); df
```

```
  cisla popisky  mix
1     5       ID   ID
2     8      No.    2
3    11      Age TRUE
```

# Jaká je struktura dataframe? Z jakých vektorů se skládá?

```
1 str(df)
```

```
'data.frame':   3 obs. of  3 variables:
 $ cisla  : num  5 8 11
 $ popisky: chr  "ID" "No." "Age"
 $ mix    : chr  "ID" "2" "TRUE"
```

# CVIČENÍ

1. Překrývají se nějaká čísla v sekvencích 8:20 a 5:12?
2. Vytvořte alespoň dvěmi způsoby číselnou řadu 1 1 2 2 3 3 4.

# ŘEŠENÍ

## 1. Překrývají se nějaká čísla v sekvencích 8:20 a 5:12?

```
1  5:12 %in% 8:20
```

```
[1] FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
```

## 2. Vytvořte alespoň dvěmi způsoby číselnou řadu 1 1 2 2 3 3 4.

```
1  rep(1:4, each = 2, length.out = 7)
```

```
[1] 1 1 2 2 3 3 4
```

```
1  c(1,1,2,2,3,3,4)
```

```
[1] 1 1 2 2 3 3 4
```

```
1  c(rep(1:3, each = 2), 4)
```

```
[1] 1 1 2 2 3 3 4
```

# 2. blok

- načtení dat
- práce s dataframe
- faktory
- výběr dat
- balíček dplyr
- úprava dataframe

# Načtení dat

Načtení tabulkového datasetu do R se provádí pomocí **read.table** funkcí.

- read.delim() pro tabulátorem oddělené hodnoty
- read.csv() pro CSV soubory

Načteme si zkušební dataset **newborns.txt**.

```
1 data <- read.delim("newborns.txt", header = TRUE, sep = "\t", dec =
```

Nešlo by to jednodušeji?

```
1 data2 <- read.delim("newborns.txt")
```

Načetli se datasety stejně? Proč?

```
1 identical(data, data2)
```

```
[1] TRUE
```

# Popis a struktura datasetu

```
1 str(data)
```

```
'data.frame':   1402 obs. of  4 variables:
 $ edu.M   : int  2 2 2 1 3 2 1 2 2 1 ...
 $ prch.N  : int  0 0 0 0 0 1 2 0 0 0 ...
 $ sex.C   : chr  "m" "m" "f" "m" ...
 $ weight.C: int  3470 3240 2980 3280 3030 3650 4080 3040 3070 3110 ...
```

- **edu.M:** vzdělání matky
  - 1: základní
  - 2: středoškolské bez maturity
  - 3: středoškolské s maturitou
  - 4: vysokoškolské
- **prch.N:** počet biologických starších sourozenců
- **sex.C:** pohlaví dítěte
  - m: muž
  - f: žena
- **weight.C:** porodní hmotnost dítěte v gramech

# Faktory

## Kategoriální data lze v R zapsat ve specifickém data typu - **faktoru.**

```
data$sex.C <- as.factor(data$sex.C)
levels(data$sex.C)
```

```
[1] "f" "m"
```

```
str(data)
```

```
'data.frame':   1402 obs. of  4 variables:
 $ edu.M   : int  2 2 2 1 3 2 1 2 2 1 ...
 $ prch.N  : int  0 0 0 0 0 1 2 0 0 0 ...
 $ sex.C   : Factor w/ 2 levels "f","m": 2 2 1 2 2 2 1 1 1 2 ...
 $ weight.C: int  3470 3240 2980 3280 3030 3650 4080 3040 3070 3110 ...
```

## R umožňuje konverzi dat na faktory a zároveň **přejmenování úrovní.**

```
data$edu.M <- factor(data$edu.M, labels = c("ZS", "SS", "SSm", "VS"
levels(data$edu.M)
```

```
[1] "ZS"  "SS"  "SSm" "VS"
```

```
head(data)
```

| | edu.M | prch.N | sex.C | weight.C |
|---|---|---|---|---|
| 1 | SS | 0 | m | 3470 |
| 2 | SS | 0 | m | 3240 |
| 3 | SS | 0 | f | 2980 |
| 4 | ZS | 0 | m | 3280 |
| 5 | SSm | 0 | m | 3030 |
| 6 | SS | 1 | m | 3650 |

# Výběr dat z objektů pomocí base R

V base R jsou dva hlavní způsoby výběru dat.

- **výběr pomocí** []
  - objekt[určení výběru]

```
1 vektor <- 2:30; vektor
```

```
 [1]  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
25 26
[26] 27 28 29 30
```

```
1 vektor[3]
```

```
[1] 4
```

Při použití [ ] je potřeba specifikovat jakou část chcete vybrat u **každé dimenze**. Dataframe je například dvojrozměrný objekt (má řádky a sloupce), je proto potřeba definovat výběr u obojího.

```
data[1, 4] #1. radek, 4. sloupec
```

```
[1] 3470
```

```
data[c(1, 5, 7),4]
```

```
[1] 3470 3030 4080
```

```
data[1:3, ] #1.- 3. radek, vsechny sloupce
```

```
  edu.M prch.N sex.C weight.C
1    SS       0     m     3470
2    SS       0     m     3240
3    SS       0     f     2980
```

```
data[2, -1] #2. radek, vsechny sloupce krome 1.
```

```
  prch.N sex.C weight.C
2      0     m     3240
```

## Lze použít i **jména sloupců**.

```
data[1:3, c("edu.M", "weight.C")]
```

```
  edu.M weight.C
1    SS     3470
2    SS     3240
3    SS     2980
```

```
data["edu.M"] #subset pouze s jednou dimenzi u dataframe vzdy vra
```

```
    edu.M
1      SS
2      SS
3      SS
```

```
4        ZS
5       SSm
6        SS
7        ZS
8        SS
9        SS
10       ZS
11      SSm
12      SSm
13       ZS
14       ZS
```

- **výběr pomocí $**
  - objekt$název sloupce

```
1 data$edu.M
```

```
  [1] SS   SS   SS   ZS   SSm  SS   ZS   SS   SS   ZS   SSm  SSm  ZS   ZS
 [15] SS   ZS   ZS   ZS   SS   SS   SSm  ZS   ZS   SSm  SSm  ZS   ZS   ZS
 [29] SS   SSm  ZS   SS   SSm  ZS   ZS   SSm  ZS   ZS   ZS   ZS   SSm  SS
 [43] ZS   ZS   ZS   ZS   SS   SS   SS   ZS   SS   ZS   SS   SS   SS   ZS
 [57] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   SS   ZS   ZS   ZS
 [71] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   SS   ZS   SS   SSm  SS   ZS
 [85] ZS   ZS   ZS   ZS   SS   SS   ZS   ZS   SS   ZS   ZS   SS   ZS   SS
 [99] ZS   SS   ZS   ZS   SSm  SS   ZS   SS   SS   ZS   ZS   ZS   SS   SS
[113] ZS   ZS   SS   ZS   ZS   SS   ZS   SS   ZS   ZS   SS   ZS   ZS   ZS
[127] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS
[141] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   SS
[155] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS
```

```
 [169] ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   ZS   SSm  ZS
 [183] ZS   ZS   SS   ZS   ZS   ZS   SS   ZS   ZS   SS   ZS   ZS   ZS   ZS
 [197] ZS   ZS   SSm  SS   ZS   ZS   ZS   SSm  ZS   ZS   ZS   SS   SS   ZS
```

## Rozdíl mezi výsledky výběru

- **$** vrátí *vektor* obsahující hodnoty vybraného sloupce
  - sloupec musí mít jasně určené jméno
  - nelze vybírat řádky
- [ ] vrátí část *objektu* obsahující vybrané hodnoty
  - zachovává si přitom vlastnosti objektu

```
identical(data["edu.M"], data$edu.M)
```

```
[1] FALSE
```

```
mode(data$edu.M)
```

```
[1] "numeric"
```

```
mode(data["edu.M"])
```

```
[1] "list"
```

# Tyto znalosti pak lze použít k výběru **specifických hodnot.**

```
1 data[data$edu.M == "ZS", ]
```

| | edu.M | prch.N | sex.C | weight.C |
|---|---|---|---|---|
| 4 | ZS | 0 | m | 3280 |
| 7 | ZS | 2 | f | 4080 |
| 10 | ZS | 0 | m | 3110 |
| 13 | ZS | 1 | m | 2940 |
| 14 | ZS | 0 | f | 1550 |
| 16 | ZS | 1 | f | 1960 |
| 17 | ZS | 0 | f | 2910 |
| 18 | ZS | 0 | f | 3670 |
| 22 | ZS | 1 | f | 2660 |
| 23 | ZS | 2 | f | 2550 |
| 26 | ZS | 1 | f | 2540 |
| 27 | ZS | 0 | m | 3170 |
| 28 | ZS | 0 | m | 2970 |
| 31 | ZS | 1 | f | 2160 |

```
1 data[data$edu.M == "ZS" & data$prch.N >= 2, c("sex.C", "weight.C"
```

| | sex.C | weight.C |
|---|---|---|
| 7 | f | 4080 |
| 23 | f | 2550 |
| 35 | m | 2480 |
| 87 | f | 3140 |
| 91 | f | 3040 |
| 182 | m | 2710 |
| 186 | m | 2780 |
| 187 | m | 2190 |

| | | |
|---|---|---|
| 205 | m | 3670 |
| 207 | m | 2390 |
| 210 | f | 2540 |
| 220 | f | 2850 |
| 235 | f | 3430 |
| 239 | f | 1590 |

# Výběr dat z objektů pomocí balíčku {dplyr}

Balíček dplyr je součástí balíčku **tidyverse**, balíček dplyr jsme si nainstalovali během prvního bloku.

```
1 library(dplyr)
```

Umožňuje výběr pomocí jednodušších příkazů než těch, které požaduje base R.

```
1 select(data, edu.M, sex.C)
```

```
     edu.M sex.C
1       SS     m
2       SS     m
3       SS     f
4       ZS     m
5      SSm     m
6       SS     m
7       ZS     f
8       SS     f
9       SS     f
```

```
10       ZS      m
11      SSm      f
12      SSm      m
13       ZS      m
14       ZS      f
```

# Dokáže také provést výběr dle specifických podmínek.

```
1 filter(data, edu.M == "SS", weight.C > 1500)
```

```
   edu.M prch.N sex.C weight.C
1     SS      0     m     3470
2     SS      0     m     3240
3     SS      0     f     2980
4     SS      1     m     3650
5     SS      0     f     3040
6     SS      0     f     3070
7     SS      0     m     2040
8     SS      0     m     2790
9     SS      0     m     3490
10    SS      1     m     2460
11    SS      1     m     2350
12    SS      0     f     3340
13    SS      0     f     4060
14    SS      0     m     3770
```

# Co všechno balíček zvládne?

```
1 help(package = dplyr)
```

# Úprava datasetu

K dataframe je také možno údaje přidávat. Vytvořme se sloupec weight.K, který bude obsahovat zda se porodní váha novorozence nacházela v normě, či zda byla vysoká či nízká.

- **nízká:** weight. C < 2500
- **norma:** 2500 <= weight.C <= 4200
- **vysoká:** weight.C > 4200

```
data$weight.K[data$weight.C < 2500] <- "nizka"
data$weight.K[data$weight.C >= 2500 & data$weight.C <= 4200] <- "no
data$weight.K[data$weight.C > 4200] <- "vysoka"

head(data)
```

```
  edu.M prch.N sex.C weight.C weight.K
1    SS      0     m     3470    norma
2    SS      0     m     3240    norma
3    SS      0     f     2980    norma
4    ZS      0     m     3280    norma
```

```
5    SSm       0      m      3030     norma
6     SS       1      m      3650     norma
```

## Hodnoty si převedeme na faktor.

```
1 data$weight.K <- as.factor(data$weight.K)
2
3 summary(data) #shrn mi objekt data
```

```
 edu.M          prch.N        sex.C      weight.C       weight.K
ZS  :420   Min.   :0.0000   f:673   Min.   : 580   nizka : 272
SS  :451   1st Qu.:0.0000   m:729   1st Qu.:2670   norma :1080
SSm :437   Median :1.0000           Median :3170   vysoka:  44
VS  : 81   Mean   :0.9492           Mean   :3071   NA's  :   6
NA's: 13   3rd Qu.:1.0000           3rd Qu.:3560
           Max.   :9.0000           Max.   :4970
           NA's   :5                NA's   :6
```

## V summary vidíme, že dataframe má v sobě **jedince s chybějícími hodnotami**. Vytvoříme si proto soubor čistě s úplnými údaji.

```
1 data <- na.omit(data)
2 summary(data)
```

```
edu.M          prch.N        sex.C      weight.C       weight.K
ZS :417    Min.   :0.0000   f:662   Min.   : 580   nizka : 266
SS :448    1st Qu.:0.0000   m:719   1st Qu.:2680   norma :1071
SSm:435    Median :1.0000           Median :3170   vysoka:  44
```

```
VS : 81    Mean   :0.9428          Mean   :3078
           3rd Qu.:1.0000          3rd Qu.:3570
           Max.   :9.0000          Max.   :4970
```

# CVIČENÍ

Vyberte data pro všechny chlapce, kteří se narodili s porodní váhou mezi 2000 a 2200 gramy (včetně) matkám prvorodičkám s nižším stupněm vzdělání než je vysokoškolský titul.

# ŘEŠENÍ

## První možnost

```r
data[data$weight.C %in% 2000:2200 & #vaha v intervalu 2000:2200
       data$sex.C == "m" & #chlapci
       data$edu.M != "VS" & #vzdelani krome VS
       data$prch.N == 0,] #0 starsich sourozencu
```

```
      edu.M prch.N sex.C weight.C weight.K
15       SS      0     m     2040    nizka
290      SS      0     m     2120    nizka
299     SSm      0     m     2180    nizka
398      SS      0     m     2010    nizka
403     SSm      0     m     2190    nizka
832     SSm      0     m     2150    nizka
842      ZS      0     m     2160    nizka
850      ZS      0     m     2020    nizka
961     SSm      0     m     2090    nizka
1290    SSm      0     m     2170    nizka
```

## Druhá možnost

```r
data[data$weight.C >= 2000 & #vaha >= 2000
            data$weight.C <= 2200 & #vaha <= 2200
            data$sex.C == "m" &
```

```
4               data$edu.M %in% c("ZS", "SS", "SSm") & #vycet vsech vzd
5               data$prch.N == 0,]
```

```
     edu.M prch.N sex.C weight.C weight.K
15      SS      0     m     2040    nizka
290     SS      0     m     2120    nizka
299    SSm      0     m     2180    nizka
398     SS      0     m     2010    nizka
403    SSm      0     m     2190    nizka
832    SSm      0     m     2150    nizka
842     ZS      0     m     2160    nizka
850     ZS      0     m     2020    nizka
961    SSm      0     m     2090    nizka
1290   SSm      0     m     2170    nizka
```

# Třetí možnost

```
1  filter(data, weight.C %in% 2000:2200,
2         sex.C == "m",
3         edu.M != "VS",
4         prch.N == 0)
```

```
   edu.M prch.N sex.C weight.C weight.K
1     SS      0     m     2040    nizka
2     SS      0     m     2120    nizka
3    SSm      0     m     2180    nizka
4     SS      0     m     2010    nizka
5    SSm      0     m     2190    nizka
6    SSm      0     m     2150    nizka
7     ZS      0     m     2160    nizka
8     ZS      0     m     2020    nizka
```

```
9     SSm       0       m       2090     nizka
10    SSm       0       m       2170     nizka
```

Jaké jsou mezi výsledky rozdíly? Kdy je lepší použít který přístup?

# 3. blok

- popisná statistika
- dplyr::summarise
- dplyr::group_by()
- pipes
- analýza nominálních dat
- analýza ordinálních dat
- analýza intervalových dat

# Popisná statistika

Jakými různými způsoby můžeme získat základní statistické údaje o objektu? Známe již funkci **summary()**.

```
1 summary(data)
```

```
 edu.M          prch.N        sex.C      weight.C       weight.K    
 ZS :417   Min.   :0.0000   f:662   Min.   : 580   nizka : 266  
 SS :448   1st Qu.:0.0000   m:719   1st Qu.:2680   norma :1071  
 SSm:435   Median :1.0000           Median :3170   vysoka:  44  
 VS : 81   Mean   :0.9428           Mean   :3078                
           3rd Qu.:1.0000           3rd Qu.:3570                
           Max.   :9.0000           Max.   :4970                
```

Jak můžeme tyto údaje získat **individuálně**? Ukažme si to pomocí sloupce weight.C.

```
1 mean(data$weight.C) #prumer
```

```
[1] 3078.027
```

```
1 min(data$weight.C) #minimum
```

```
[1] 580
```

```
1 max(data$weight.C) #maximum
```

```
[1] 4970
```

```
median(data$weight.C) #median
```

```
[1] 3170
```

```
quantile(data$weight.C, type = 2) #kvartily
```

```
  0%  25%  50%  75% 100% 
 580 2680 3170 3570 4970 
```

```
quantile(data$weight.C, probs = 0.25, type = 2) #pouze prvni kvarti
```

```
 25% 
2680 
```

```
IQR(data$weight.C, type = 2) #interkvartilove rozpeti
```

```
[1] 890
```

Funkce summary nám ukáže i zastoupení jednotlivých faktorů u slupců s kategorickými hodnotami. Stejný výsledek dostaneme i pomocí funkce **table()**.

```
table(data$edu.M)
```

```
 ZS  SS SSm  VS 
417 448 435  81 
```

```
table(data$edu.M, data$sex.C)
```

```
     f   m
```

```
   ZS  190 227
   SS  212 236
   SSm 212 223
   VS   48  33
```

Tato funkce vytvoří *kontingenční tabulku* absolutních četností faktorů vybraných hodnot.

Výpočet počtu zastoupení můžeme provést i jednotlivě, a to funkcí **sum()**.

```r
sum(data$edu.M == "ZS")
```

```
[1] 417
```

Funkce sum() spočítá *kolikrát se objeví* x v daném datovém objektu.

Podobnou tabulku jako dostaneme z table() si poté můžeme vytvořit pomocí funkce **data.frame()**.

```r
tabulka_cetnosti_edu.M <- data.frame(sum(data$edu.M == "ZS"),
            sum(data$edu.M == "SS"),
            sum(data$edu.M == "SSm"),
            sum(data$edu.M == "VS")); tabulka_cetnosti_edu.M
```

```
  sum.data.edu.M.....ZS.. sum.data.edu.M.....SS.. sum.data.edu.M.....SSm..
1                     417                     448                      435
  sum.data.edu.M.....VS..
1                      81
```

Tato funkce vytvoří *dataframe*, kdy po sloupcích spojí vložené objekty. Pokud bychom chtěli spojit objekty po řádcích, můžeme použít funkci **rbind.data.frame()**.

Vidíme, že R **automaticky vygenerovalo názvy sloupců** na základě jejich vlastností. Pro lepší přehled si proto sloupce přejmenujeme, a to pomocí funkce **colnames()**.

```r
colnames(tabulka_cetnosti_edu.M) <- c("ZS", "SS", "SSm", "VS")
```

Podobnou funkci má u řádků funkce row.names().

Této potřebě přejmenovávání se dá vyhnout tak, že do funkce data.frame() rovnou vložíme **pojmenované objekty**.

```r
ZS <- sum(data$edu.M == "ZS")
SS <- sum(data$edu.M == "SS")
SSm <- sum(data$edu.M == "SSm")
VS <- sum(data$edu.M == "VS")
```

```r

data.frame(ZS, SS, SSm, VS)
```

```
  ZS  SS SSm VS
1 417 448 435 81
```

# dplyr::summarise

K vytváření popisných tabulek lze také použít funkci z balíčku dplyr, **summarise()**. Pro použití funkce musíte mít buď aktivovanou knihovnu balíčku (jako již máme), nebo funkci můžete zavolat přímo bez aktivace knihovny pomocí **dplyr::summarise()**.

```
dplyr::summarise(data, prumer_vahy = mean(weight.C),
                       max_sourozencu = max(prch.N))
```

```
  prumer_vahy max_sourozencu
1    3078.027              9
```

Funkce umožňuje vytvářet dataframe pomocí pojmenování statistických funkcí aplikovaných na hodnoty definovaného objektu.

Pro více informací o tom, co vše funkce dokáže, si můžete

# dplyr::group_by()

Při práci s funkcí summarise se můžete setkat se situací, kdy byste rádi rozdělili výsledný dataframe podle určitých kategorií. K tomu slouží funkce **group_by()**.

```
m_vzdelani <- group_by(data, edu.M)
```

**Funkce sama objekt nezmění**, ale při kombinaci s funkcí summarise umožní rozdělení statistických analýz dle kategorií definovaných v group_by.

```
summarise(m_vzdelani, prumer_vahy = mean(weight.C),
                      max_sourozencu = max(prch.N))
```

```
# A tibble: 4 × 3
  edu.M prumer_vahy max_sourozencu
  <fct>       <dbl>          <int>
1 ZS          2931.              9
2 SS          3128.              6
3 SSm         3153.              4
4 VS          3157.              3
```

# Pipes

Je možné, že budete provádět analýzu na výběru z výběru z výběru datasetu, jako se stalo v přechozích kapitolách. Jak si práci usnadníte, abyste nemuseli stále ukládat další a další objekty? Použitím takzvaného **operátoru pipe %>%**. Jeho funkce bud ukázána na následujícím kódu, jednou napsaném bez použití operátoru a jednou s ním:

```
matka_ZS <- filter(data, edu.M == "ZS")
vaha_srovnana <- arrange(matka_ZS, weight.C)
head(vaha_srovnana, 5)
```

```
  edu.M prch.N sex.C weight.C weight.K
1    ZS      0     f      750    nizka
2    ZS      2     m      970    nizka
3    ZS      0     f      970    nizka
4    ZS      0     m     1190    nizka
5    ZS      0     m     1210    nizka
```

V tomto příkladě se setkáváme s novou funkcí **dplyr::arrange()**, která seřadí data v X podle údajů v Xa od nejnižší hodnoty po nejvyšší.

```
data %>%
  filter(edu.M == "ZS") %>%
  arrange(weight.C) %>%
  head(5)
```

```
  edu.M prch.N sex.C weight.C weight.K
1    ZS      0     f      750    nizka
2    ZS      2     m      970    nizka
3    ZS      0     f      970    nizka
4    ZS      0     m     1190    nizka
5    ZS      0     m     1210    nizka
```

# CVIČENÍ

Spočítejte, jaký je medián a IQR váhy novorozenců, jejichž matky mají už alespoň jednoho potomka a dosáhly nanejvýše středního vzdělání s maturitou. Berte ohled na pohlaví. Výsledky uložte do tabulky.

# Řešení - první možnost

```r
chlapci <- filter(data, prch.N >= 1,
                  edu.M != "VS", sex.C == "m")

divky <- filter(data, prch.N >= 1,
                   edu.M != "VS", sex.C == "f")

median_chlapci <- median(chlapci$weight.C)
IQR_chlapci <- IQR(chlapci$weight.C, type = 2)

median_divky <- median(divky$weight.C)
IQR_divky <- IQR(divky$weight.C, type = 2)

data.frame(median_chlapci,
                   IQR_chlapci,
                   median_divky,
                   IQR_divky)
```

```
  median_chlapci IQR_chlapci median_divky IQR_divky
1           3290         860         3155       830
```

# Řešení - druhá možnost

```r
data %>% group_by(sex.C) %>%
  filter(prch.N >= 1, edu.M != "VS") %>%
  summarise(median = median(weight.C),
                    IQR = IQR(weight.C, type = 2))
```

```
# A tibble: 2 × 3
  sex.C median   IQR
  <fct>  <dbl> <dbl>
1 f       3155   830
2 m       3290   860
```

# Analýza dat

Poté co jste si upravili dataset a seznámili jste se s ním, tak na něm můžete provádět různé analýzy.

**Nominální data**

- kvalitativní data
- barva očí, typ publikace
- nelze je hierarchicky porovnávat

**Ordinální**

- kvalitativní data
- pořadí narození jedinců, známky ve škole
- nelze určit míru rozdílu mezi stupni

**Intervalová**

- kvantitativní data
- po sobě jdoucí data s jasnými intervaly
- výška, počet dětí

# Analýza nominálních dat

## Cramerův koeficient

## <0:1>

- 0: nízká závislost
- 1: vysoká závislost

```
1 porovnani <- table(data$sex.C, data$edu.M)
2 lsr::cramersV(porovnani) #crameruv koeficient
```

```
[1] 0.06183871
```

```
1 chisq.test(porovnani) #ziskani p-value
```

```
    Pearson's Chi-squared test

data:  porovnani
X-squared = 5.281, df = 3, p-value = 0.1523
```

Pozor! Funkce cramersV je určená specificky k **porovnávání kontingenčních tabulek**!

Mezi vzděláním matky pohlavím dítěte je pouze velmi nízká statisticky nevýznamná závislost.

# Analýza ordinálních dat

## Spearmanův korelační koeficient

## <-1:1>

- -1: nízká přímá závislost
- 1: vysoká přímá závislost

Lze jej použít i při **porovnávání ordinálních a intervalových dat**, protože je postavený na provnávání hodnot s jasnými rozestupy - například první stupeň je přesně o jednu hodnotu níže než druhý a 99.8 je přesně o dvě hodnoty níže než 100.

```
1 cor.test(data$prch.N, data$weight.C, method = "spearman")
```

```
    Spearman's rank correlation rho

data:  data$prch.N and data$weight.C
```

```
S = 418267338, p-value = 0.07985
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
0.04715031
```

Mezi počtem starších sourozenců a porodní hmotností je jen velmi nízká přímá závislost.

Vzhledem k p-value vyšší jak 0.05 můžeme prohlásit, že velmi nízká závislost mezi počtem starších sourozenců a porodní hmotností je statisticky nevýznamná.

# Analýza intervalových dat

## Pearsonův korelační koeficient

## <0:1>

- 0: nízká přímá závislost
- 1: vysoká přímá závislost

Jelikož náš dataset neobsahuje dvoje intervalová data k porovnání, použijeme testovací dataset trees který je v balíčku base R.

```
1 head(trees)
```

```
  Girth Height Volume
1   8.3     70   10.3
2   8.6     65   10.3
3   8.8     63   10.2
4  10.5     72   16.4
```

```
5   10.7      81    18.8
6   10.8      83    19.7
```

```r
cor.test(trees$Girth, trees$Height, method = "pearson")
```

```
    Pearson's product-moment correlation

data:  trees$Girth and trees$Height
t = 3.2722, df = 29, p-value = 0.002758
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2021327 0.7378538
sample estimates:
      cor
0.5192801
```

## Lze použít i bez určení argumentu method, jelikož Pearson se bere jako baseline.

```r
cor.test(trees$Girth, trees$Height)
```

```
    Pearson's product-moment correlation

data:  trees$Girth and trees$Height
t = 3.2722, df = 29, p-value = 0.002758
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2021327 0.7378538
sample estimates:
```

```
      cor
0.5192801
```

Mezi obvodem a výškou kmene stromů v datasetu je středně pozitivní závislost.

Vzhledem k tomu že p-value < 0.05 můžeme říct že závislost mezi hodnotami je statisticky významná.

# 4. blok

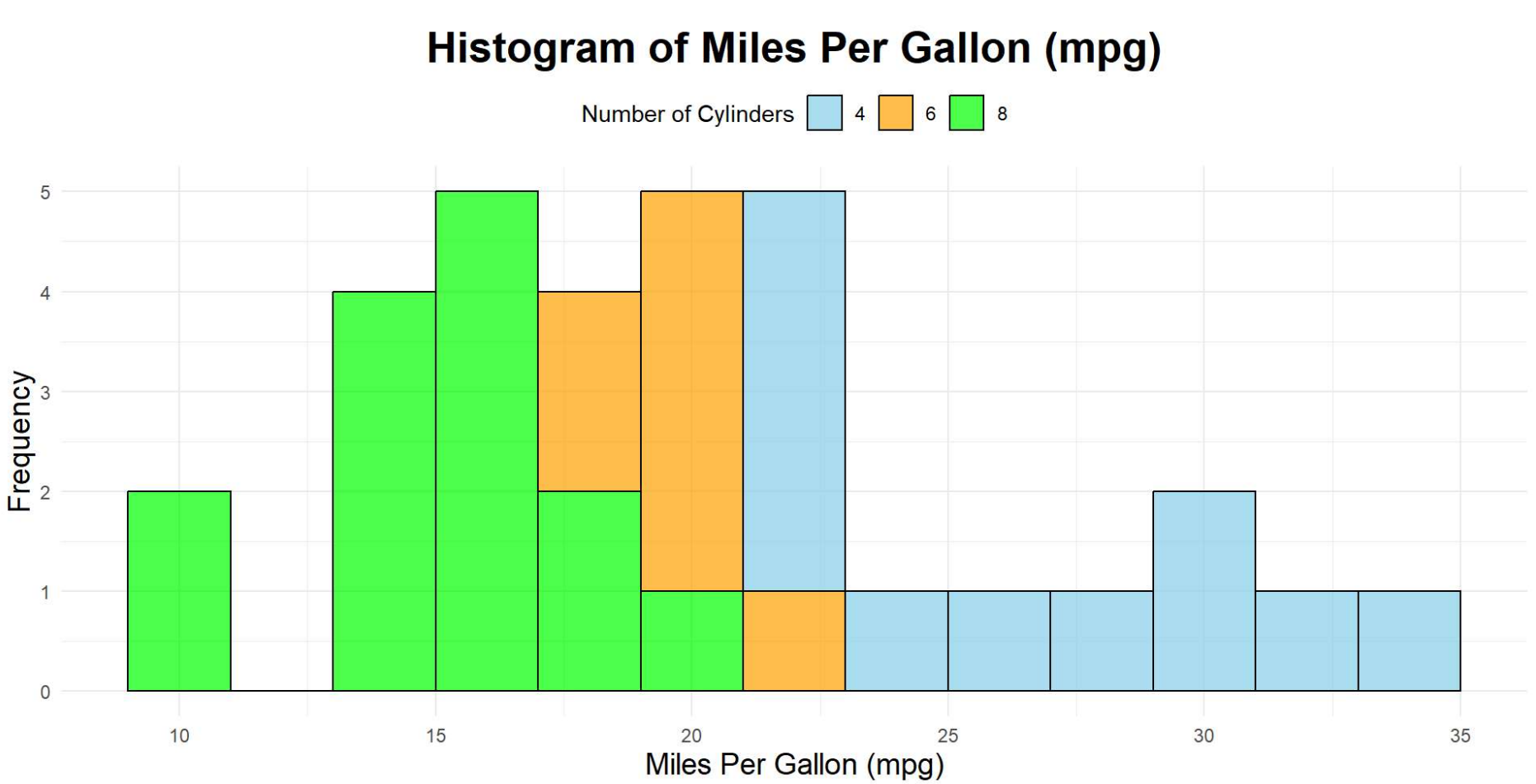
- úvod do vizualizace v R
- balíček ggplot2
- barplot
- histogram
- boxplot
- scatterplot

# Vizualizace v R

R je schopné provádět rozličné typy datových vizualizací, od jendoduchých histogramů

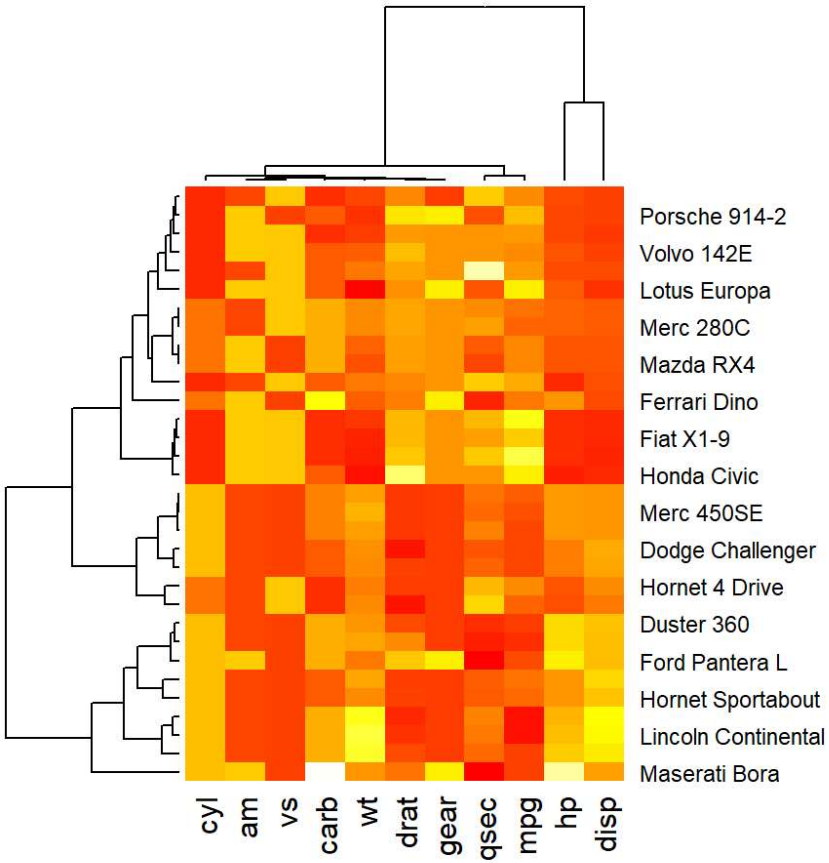
```r
library(ggplot2)

ggplot(mtcars, aes(x = mpg, fill = factor(cyl))) +
  geom_histogram(binwidth = 2, color = "black", alpha = 0.7) +
  scale_fill_manual(values = c("4" = "skyblue", "6" = "orange", "
  labs(title = "Histogram of Miles Per Gallon (mpg)",
       x = "Miles Per Gallon (mpg)",
       y = "Frequency",
       fill = "Number of Cylinders") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5, size = 20, face =
        axis.title = element_text(size = 14),
        legend.position = "top")
```




# heatmap

```
1 heatmap(as.matrix(mtcars), scale = "column", col = heat.colors(25
```




# po 3D grafy.

```
1 library(rgl)
2
3 x <- rnorm(100)
```

# {ggplot2}

Zápis kódu v ggplot2 je rozděleno do tří základních částí.

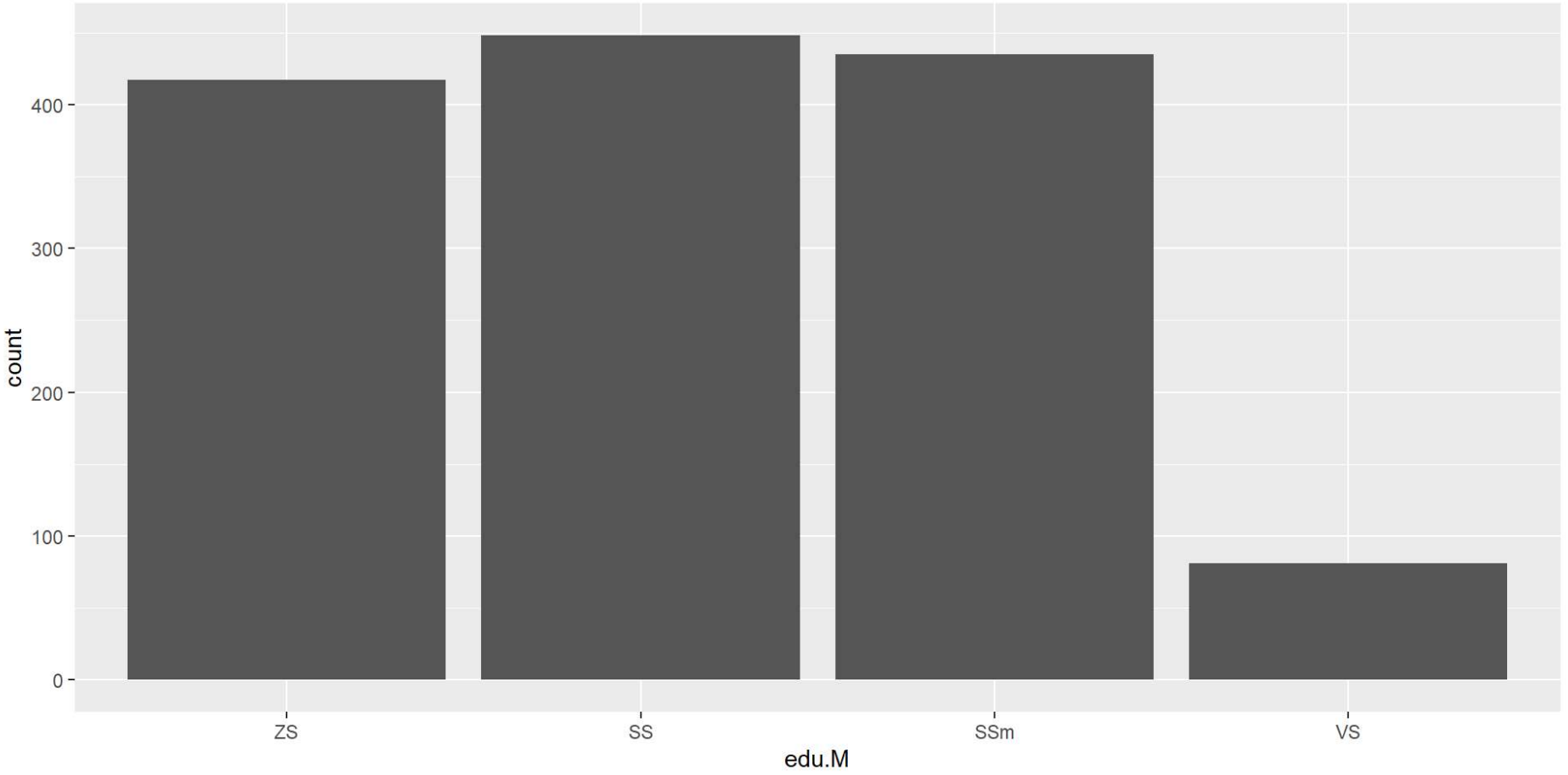
1. **data**: definice dat které jsou vizualizovány
2. **(aes):** mapování mezi hodnotami v datsetu a vizuální styl
3. **geom()**: část definující jaký typ vizualizace má být vytvořena.

Tyto části jsou ty **základní, které jsou k fungování ggplot nutné**, ale **je možné k nim přidat** nepřeberné množství dalších podfunkcí, které tak mohou dále vizualizaci vyšperkovat.

# Barplot

Určený k **porovnávání jednotlivých kategorií** a jejich variant.

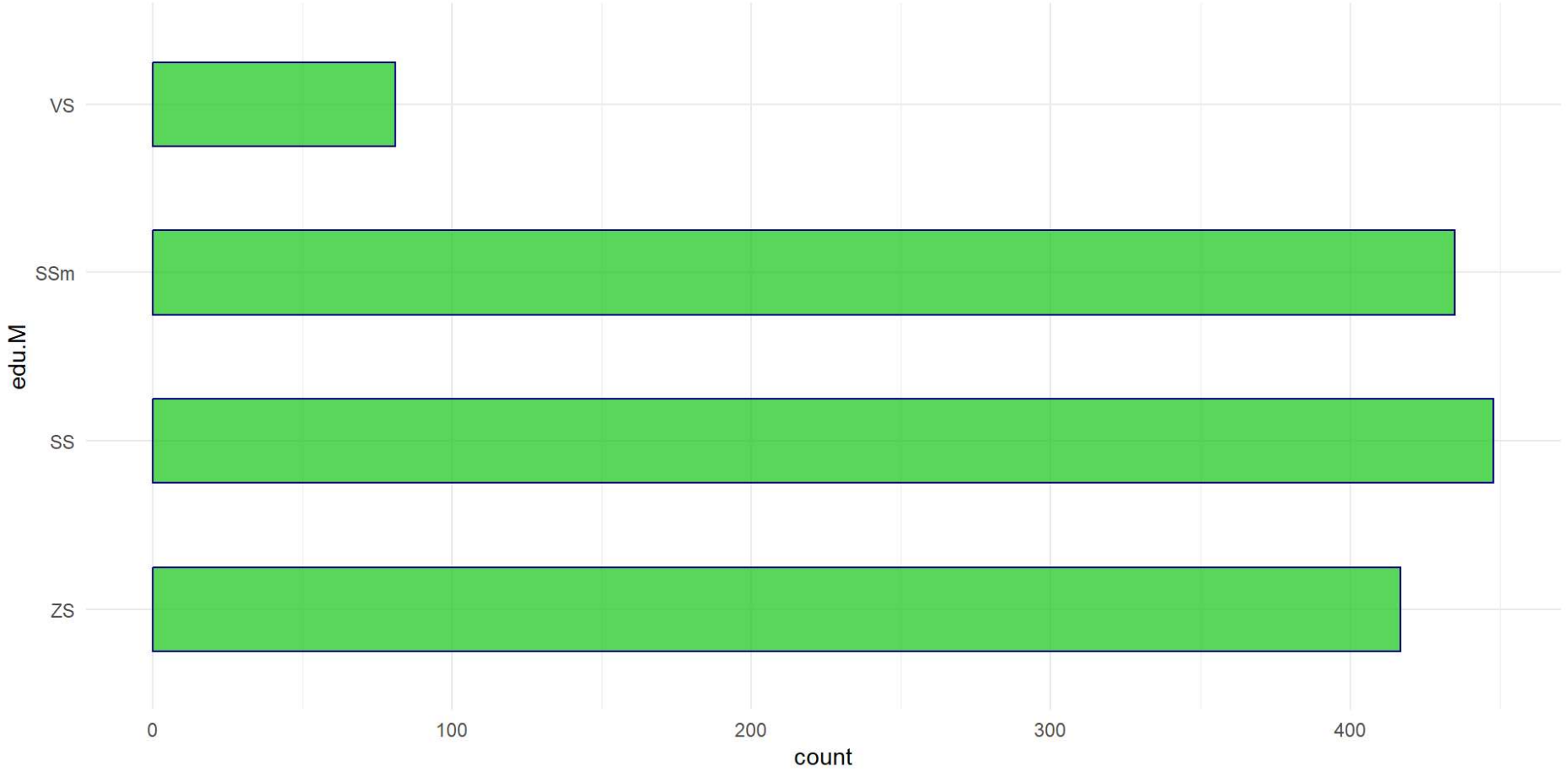
```
1 library(ggplot2)
2 ggplot(data = data) +
3   aes(x = edu.M) + #y neni definovano tim padem R secte jednotlive
4   geom_bar() #vykresli jako barplot
```




# ggplot umožňuje i další vizuální upřesnění grafu.

```
ggplot(data = data) +
  aes(x = edu.M) +
  geom_bar(width = 0.5, #sire sloupcu
           color="darkblue", #obrys sloupcu
           fill=rgb(0.1,0.8,0.1,0.7)) + #vypln sloupcu
  #rgb funkce k urceni specificke barvy
```

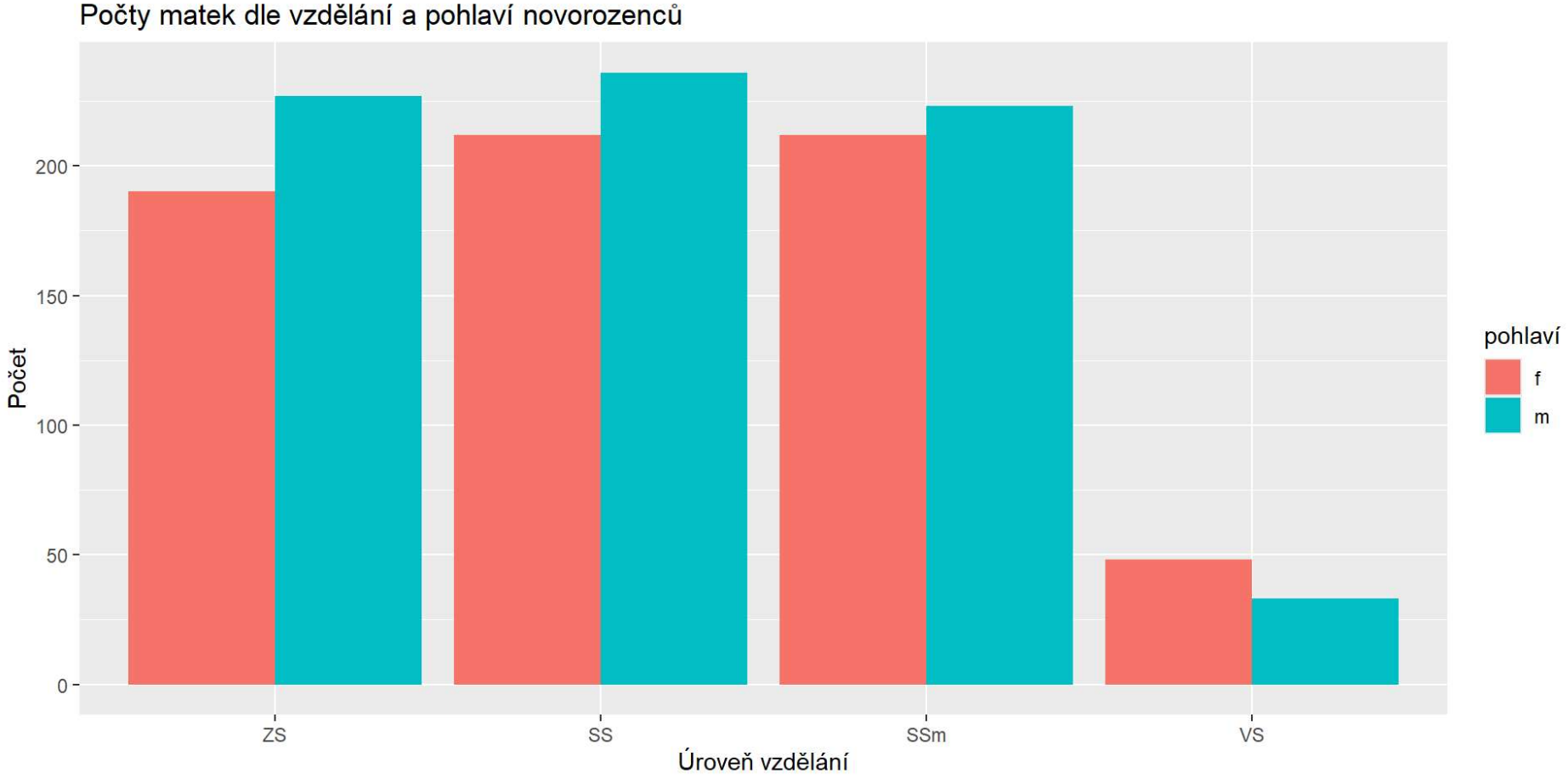
```
7    coord_flip() + #otoceni na sirku
8    theme_minimal() #zmena vizualu pozadi
```




# Barplot je možné taky r**ozdělit podle jednotlivých skupin.**

```
1  ggplot(data = data,
2         aes(x = edu.M, #vizualizuj edu.M
3             fill = sex.C)) + #barevna vypln bude podle sex.C
4  geom_bar(position = position_dodge()) + #at jsou skupiny vedle sebe
5    #bez position na sobe
```

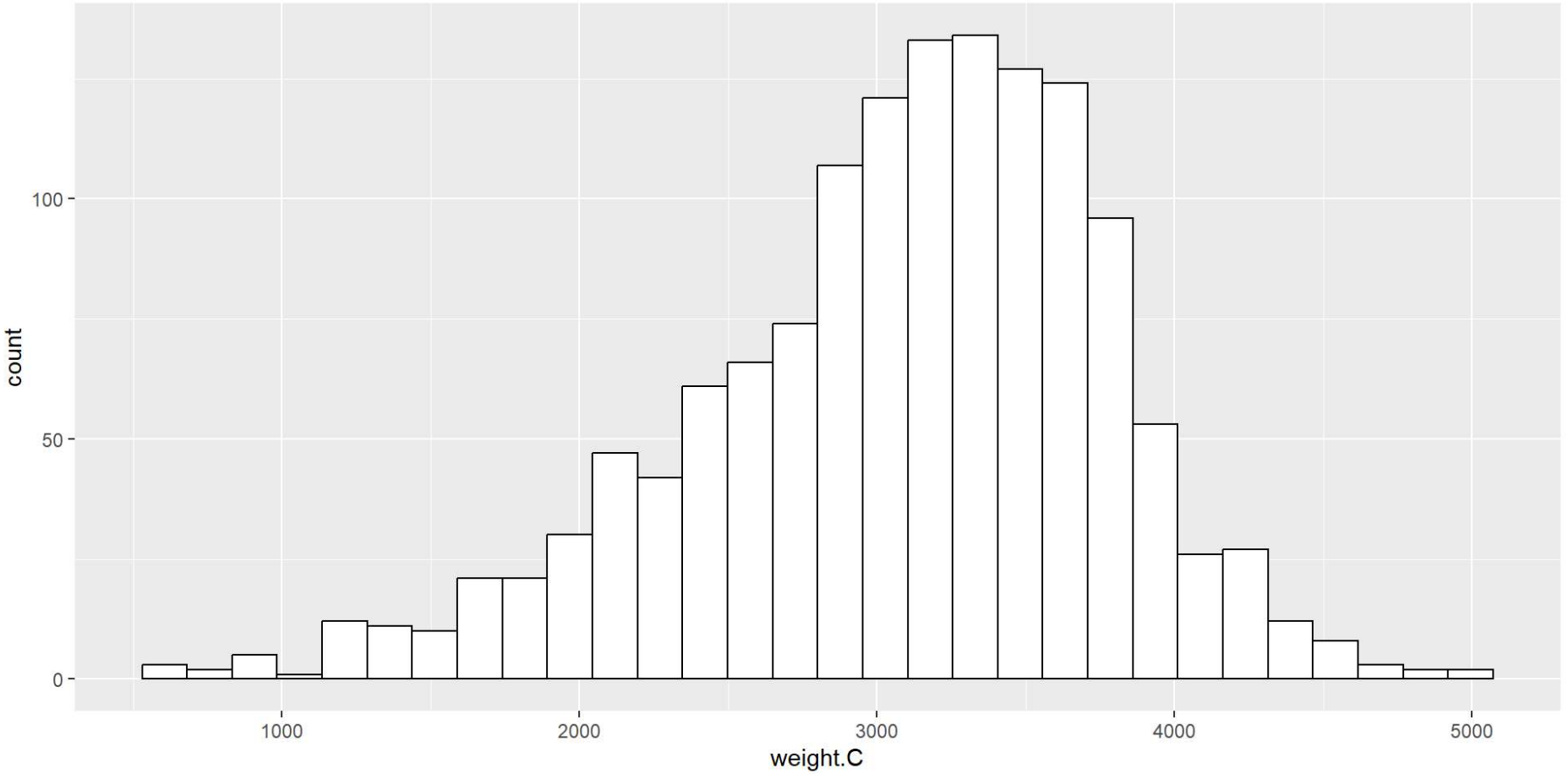
```
6    labs(title="Počty matek dle vzdělání a pohlaví novorozenců",
7         x="Úroveň vzdělání", y = "Počet", fill = "pohlaví") #popis
```



# Histogram

Určený k vizualizaci **rozložení četnosti** veličiny.

```
1 ggplot(data)+
2   aes(x = weight.C)+
3   geom_histogram(color = "black", fill = "white")
```



R automaticky vybere počet sloupců. Ten si ovšem můžeme upravit, a to pomocí **Sturgessova pravidla**.

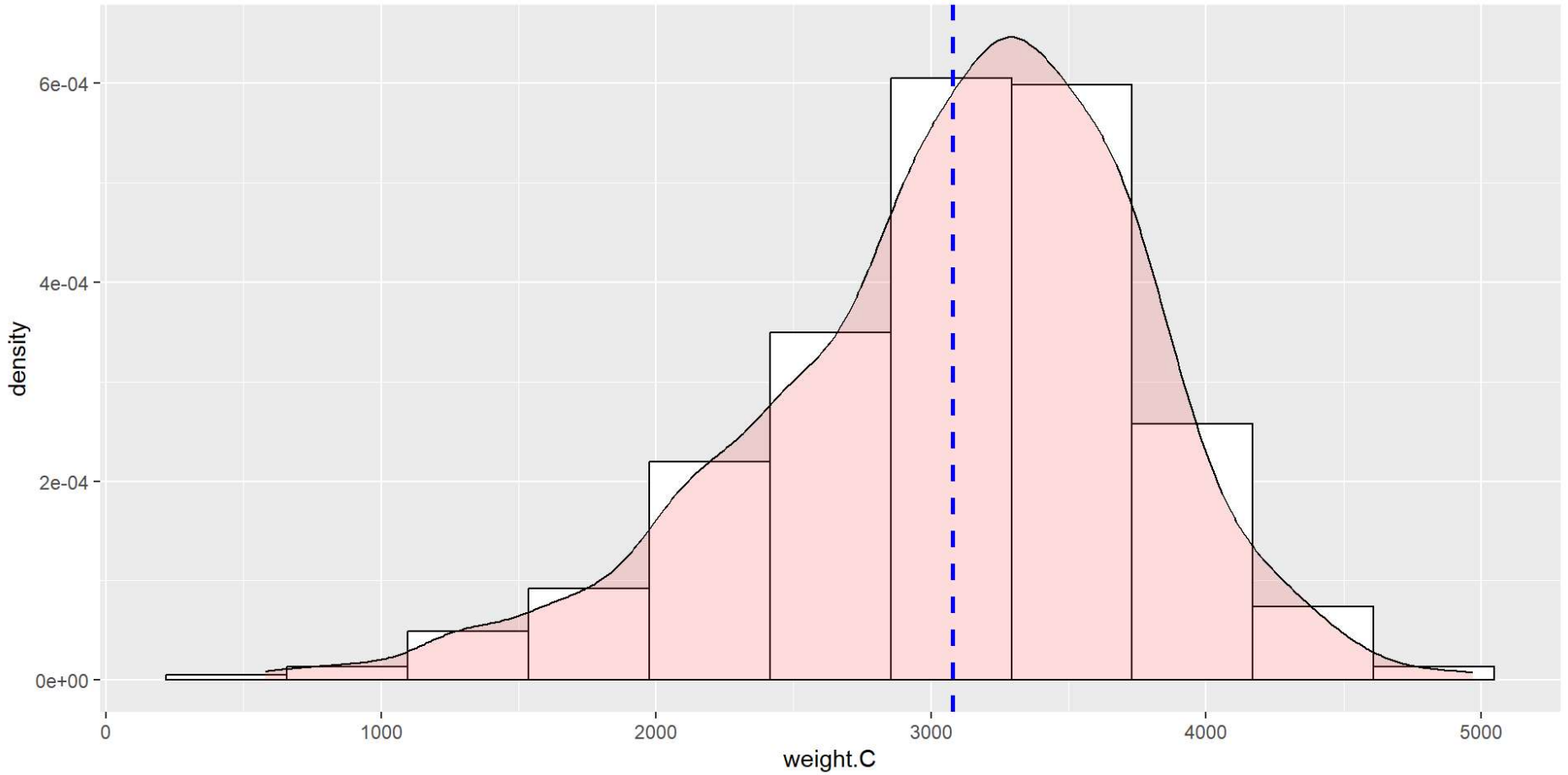
```
round(1 + 3.3 * log10(length(data$weight.C)))
```

```
[1] 11
```

```
#log10(pocet hodnot)
```

Vytvoříme si histogram znovu se správným počtem sloupců a rovnou si k němu přidáme **vizualizaci hustoty** a **průměrnou hodnotu.**

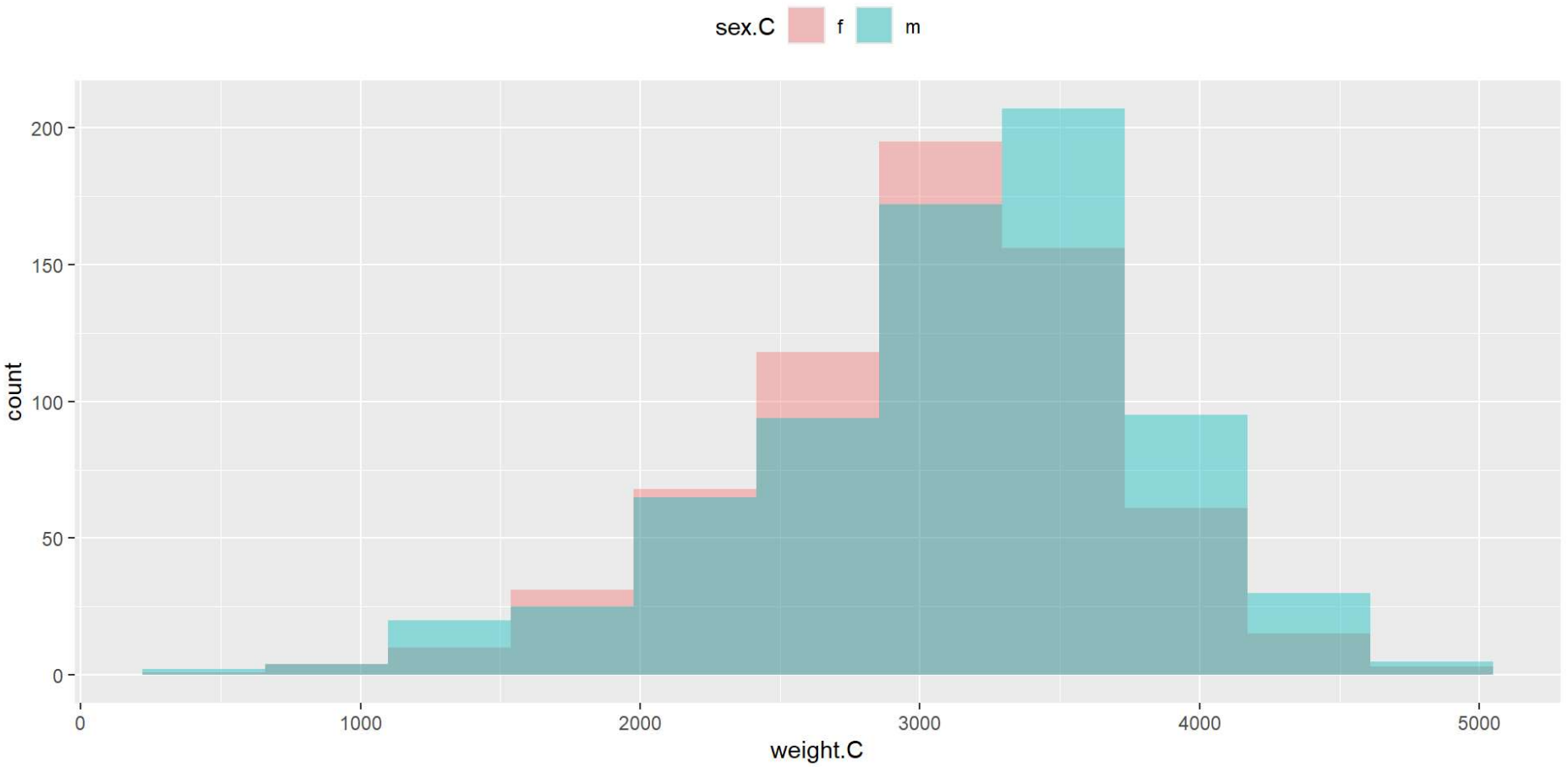
```
ggplot(data) +
  aes(x = weight.C) +
  geom_histogram(aes(y = after_stat(density)), #aes prvni cast ko
                 color = "black", fill = "white", bins = 11) + #b
  geom_density(alpha=.2, #prusvitnst
               fill="#FF6666") + #barva vyplne
  geom_vline(aes(xintercept = mean(weight.C)), #urceni prumeru
             color = "blue", #barva linie
             linetype = "dashed", #typ linie
             linewidth = 1) #velikost linie
```




Můžeme vytvářet i navzájem **překrývající se histogramy**. Všimněte si, že spousta argumentů se nám *opakuje* napříč grafy (například, color, fill, etc).

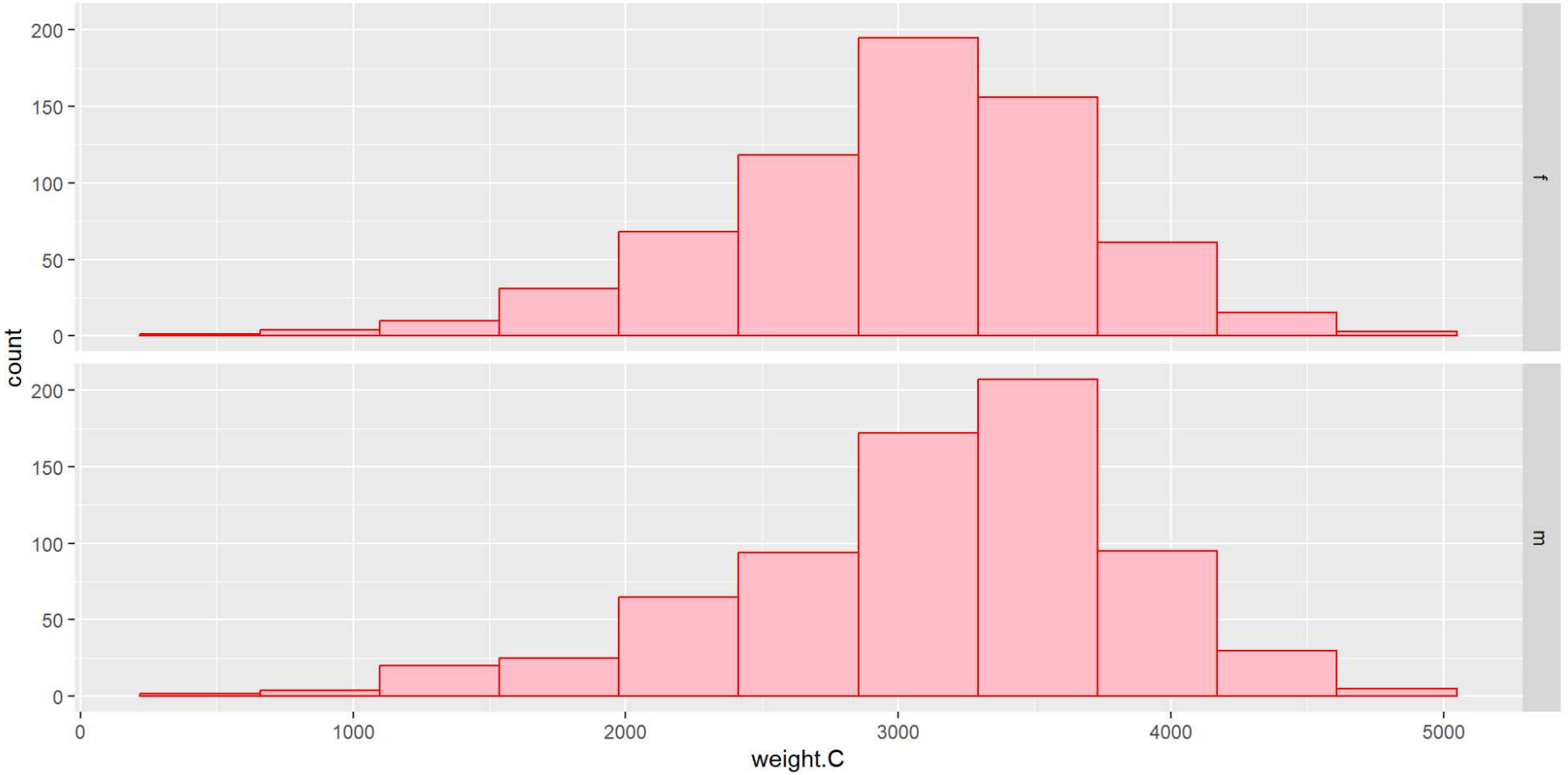
```
ggplot(data,
       aes(x = weight.C, fill = sex.C)) +
  geom_histogram(bins = 11,
                 alpha = 0.4,
```

```
5                     position = position_identity()) + #nemen pozici
6    theme(legend.position = "top") #chceme legendu nad grafem
```




# Skupiny ale také můžeme **rozložit na více grafů.**

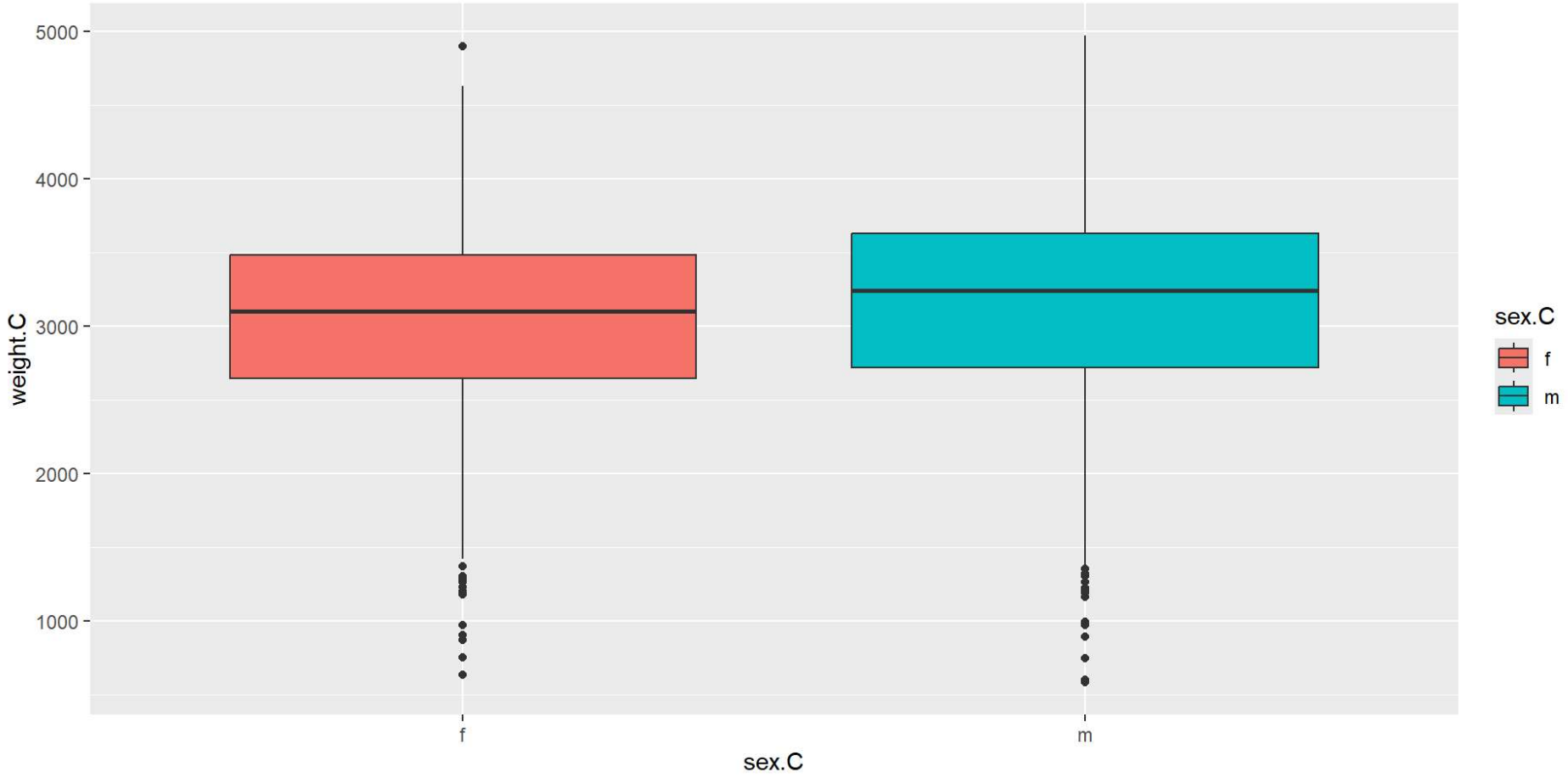
```
1  ggplot(data,
2         aes(x = weight.C)) +
3    geom_histogram(bins = 11, fill = "pink", color = "red") +
4    facet_grid(sex.C ~ .) #rozdel dle kategorie sex.C do dvou grafu
```


count
200
150
100
50
0
f
m
0
1000
2000
3000
4000
5000
weight.C

# Boxplot

Určený k vizualizaci rozložení kvantitativních dat. Specifikuje medián, IQR, minimum, maximum a odchylky nad intervalem spolehlivosti.

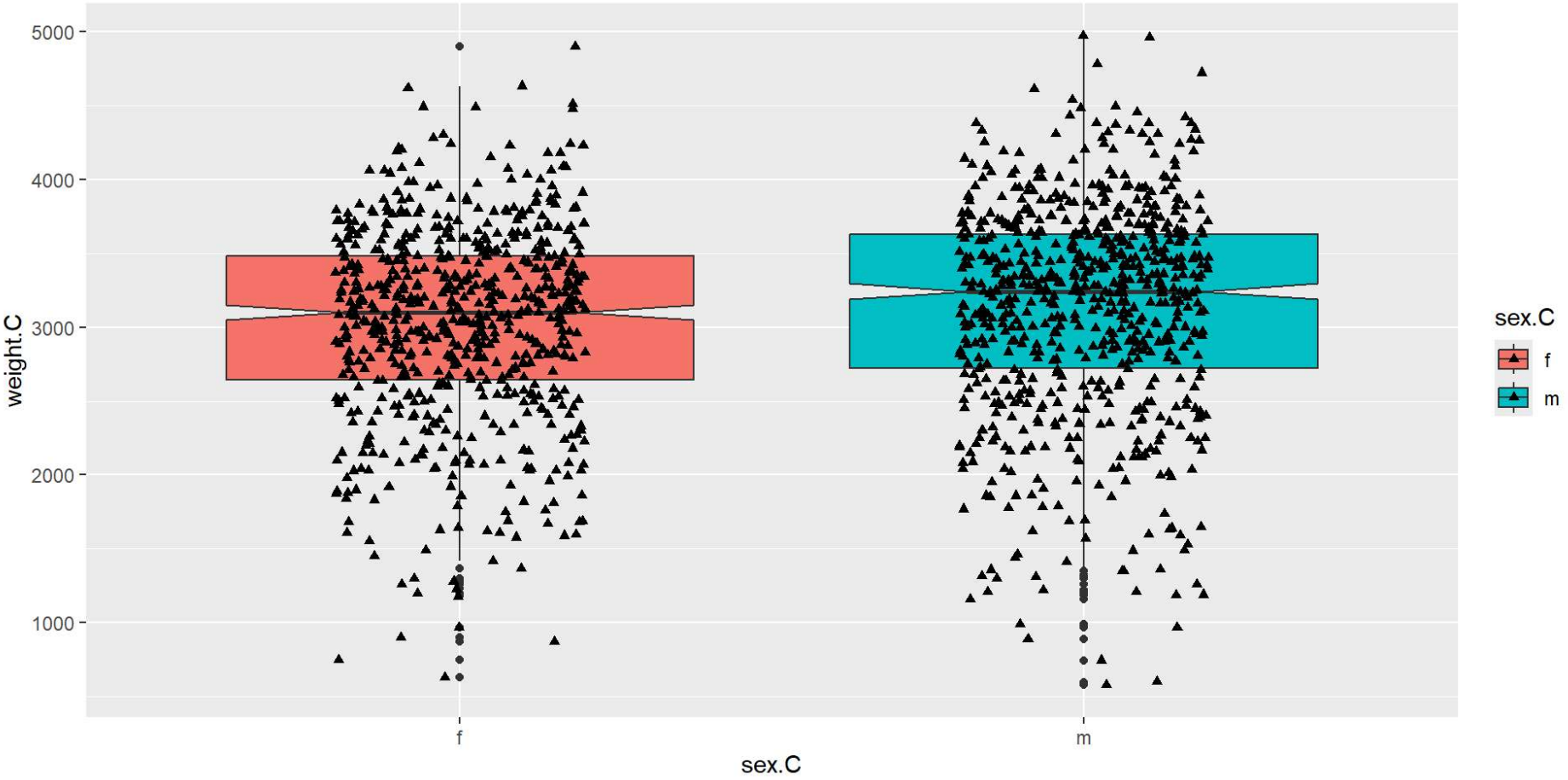
```
ggplot(data) +
  aes(x = sex.C, #jak bude dle sex.C
      y = weight.C, #vypadat rozlozeni weight.C?
      fill = sex.C) +
  geom_boxplot()
```




Můžeme si také **vizualizovat jednotlivé hodnoty** přímo na boxplotu.

```
ggplot(data) +
  aes(x = sex.C, #jak bude dle sex.C
      y = weight.C, #vypadat rozlozeni weight.C?
      fill = sex.C) +
  geom_boxplot(notch = TRUE) + #pridame zarez
```

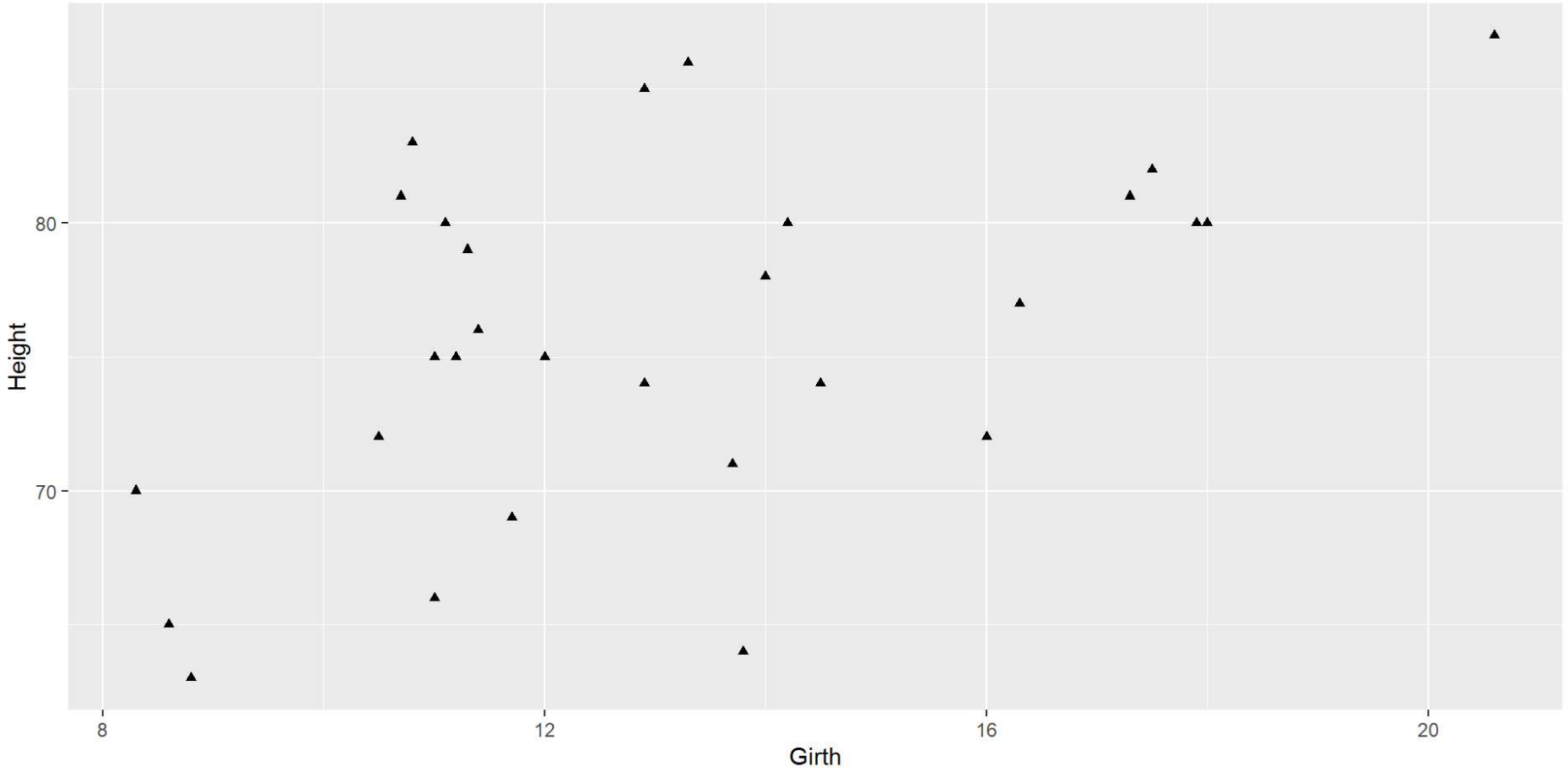
```
6    geom_jitter(shape = 17, #jaky ma byt tvar bodu
7                position = position_jitter(0.2)) #jak daleko od stredu
```



# Scatterplot

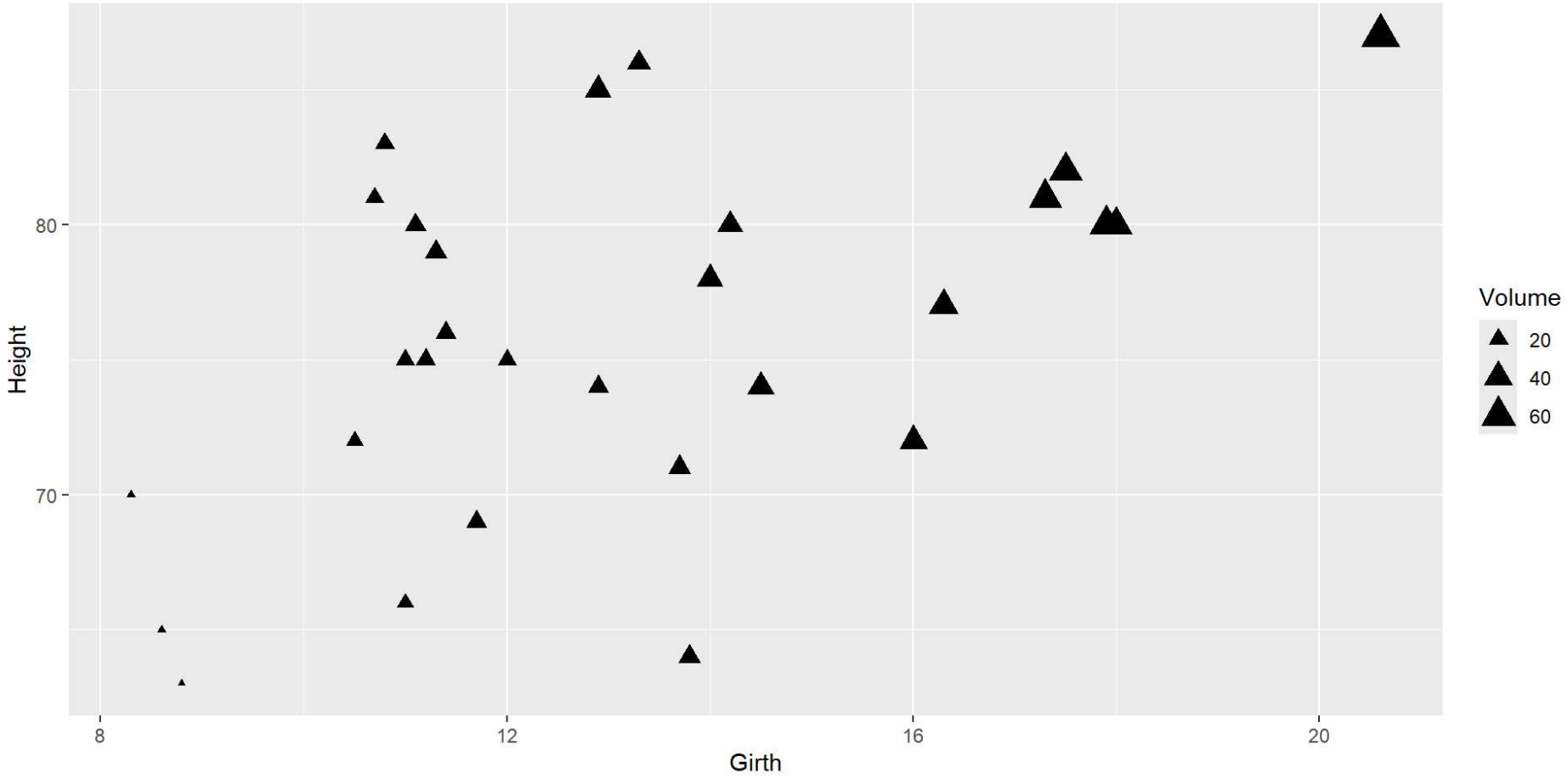
Slouží k porovnávání dvou **kvantitativních hodnot.**

```
1 ggplot(data = trees)+
2   aes(x = Girth, y = Height)+
3   geom_point(shape = 17)
```



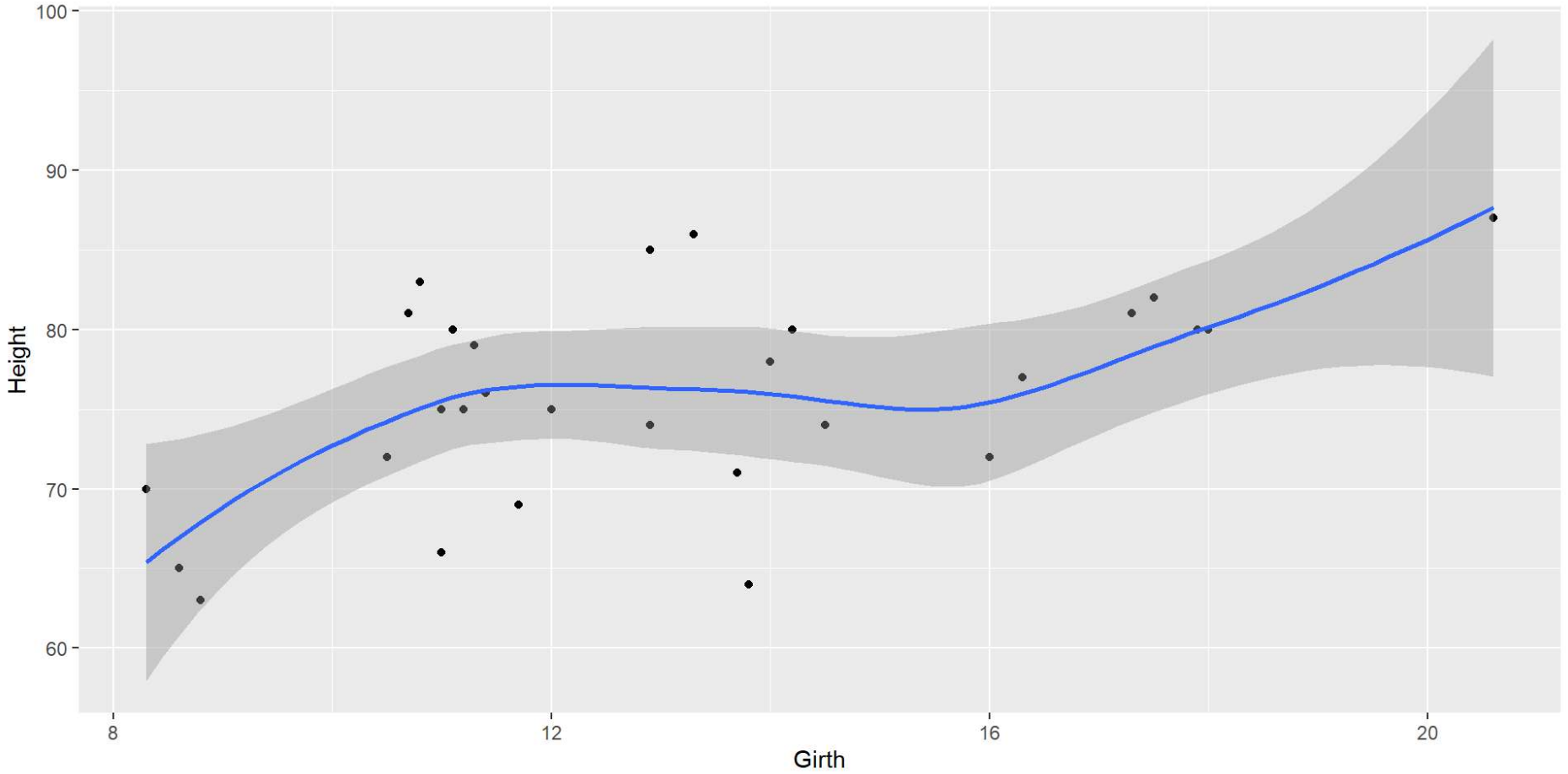
Můžeme si upřesnit, aby **velikost bodu** byla závislá na jiné proměnné.

```
ggplot(data = trees)+
  aes(x = Girth, y = Height)+
  geom_point(aes(size = Volume), #velikost je zavisla na hodnotac
    shape = 17)
```



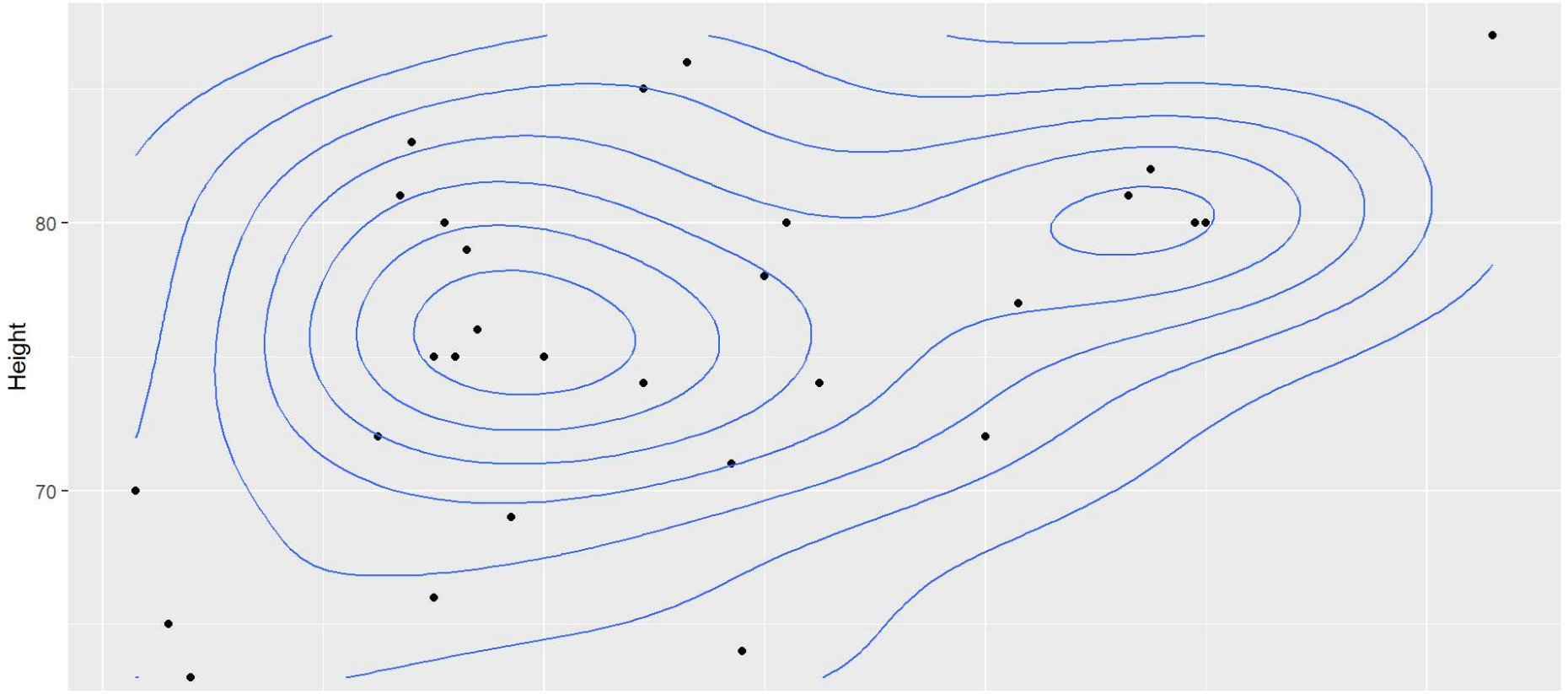
Taktéž je možné znázornit si graficky **regresní linii** (pokud jsou data lineárně závislá).

```
ggplot(data = trees) +
  aes(x = Girth, y = Height) +
  geom_point() +
  geom_smooth(method = "loess", #mame malo hodnot, proto minimum
              level = 0.95) #interval spolehlivosti
```



# Je možné scatterplot taky vyjádřit pomocí hustoty.

```
graf <- ggplot(data = trees) +
  aes(x = Girth, y = Height) +
  geom_point() +
  geom_density_2d(); graf
```



# Uložení grafu

Graf si můžete ulžit buď skrz okno Plots, a nebo pomocí kódu.

```
ggsave(plot = graf,
       filename = "ukazka_grafu.jpg",
       width = 10,
       height = 5,
       units = "cm")
```

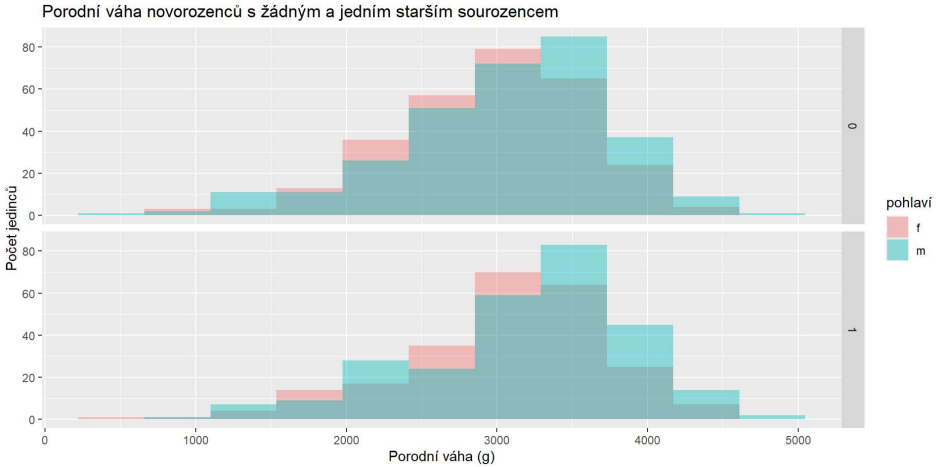
# CVIČENÍ

Vytvořte vizualizaci váhy novorozenců s maximálně jedním starším sourozencem s ohledem na jejich pohlaví. Přejmenujte legendu, popisky os a pojmenujte graf.

# ŘEŠENÍ - první možnost

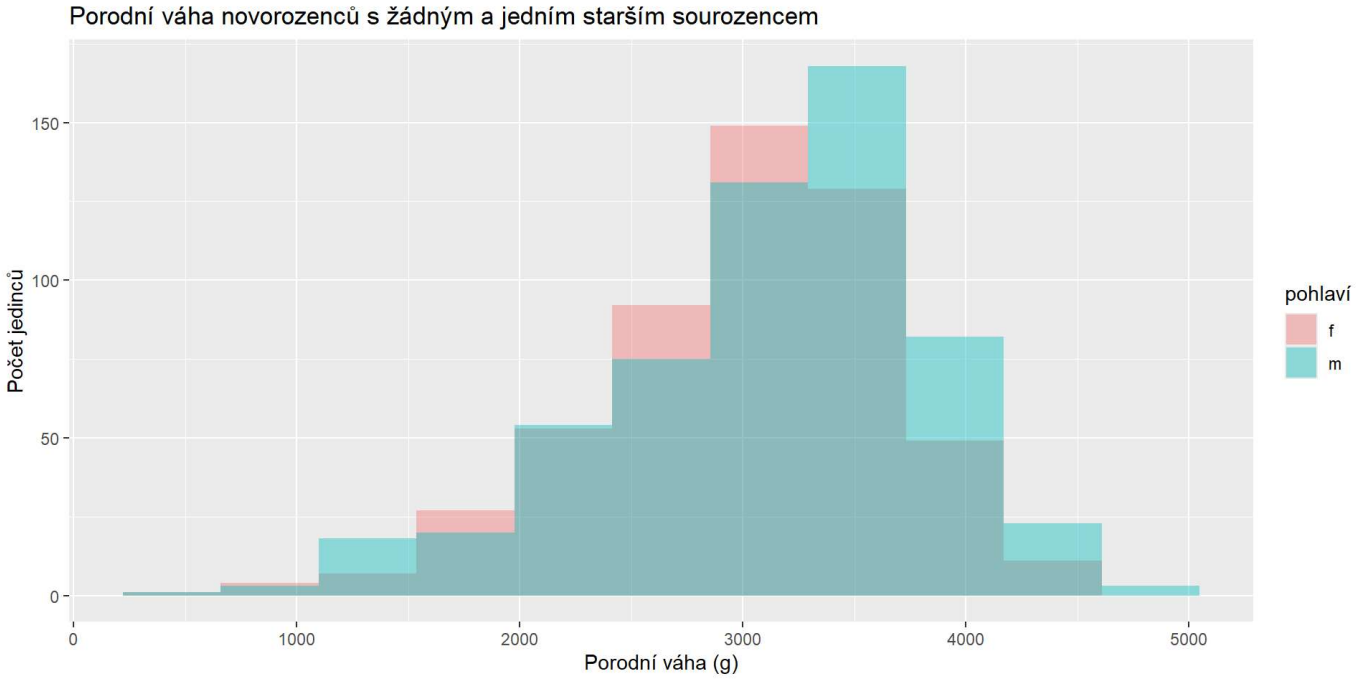
```r
sourozenci <- select(data, prch.N, weight.C, sex.C) %>%
  filter(prch.N <=1)

ggplot(sourozenci,
       aes(x = weight.C, fill = sex.C)) +
  geom_histogram(bins = 11,
                 alpha = 0.4,
                 position = position_identity()) +
    labs(title = "Porodní váha novorozenců s žádným a jedním starší
         x = "Porodní váha (g)", y = "Počet jedinců", fill = "pohla
    facet_grid(prch.N ~ .)
```



# ŘEŠENÍ - druhá možnost

```
1  ggplot(sourozenci,
2         aes(x = weight.C, fill = sex.C)) +
3    geom_histogram(bins = 11,
4                   alpha = 0.4,
5                   position = position_identity()) +
6      labs(title = "Porodní váha novorozenců s žádným a jedním starší
7           x = "Porodní váha (g)", y = "Počet jedinců", fill = "pohla
```



# Co dál?

## Rozcestník ale úplně na všechno

The Big Book of R

## Knihy (online)

R for Data Science

R Graphics Cookbook

Hands-on Programming with R

## Balíčky

tidyverse

ggplot2

## Cheatsheets

base R, base R no.2

dplyr

National Czech Programme | eosc

# Děkuji za pozornost.

lucie.hoskova@ruk.cuni.cz

Spolufinancováno Evropskou unií

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

MUNI ICS

cesnet

VŠB TECHNICKÁ UNIVERZITA OSTRAVA | IT4INNOVATIONS NÁRODNÍ SUPERPOČÍTAČOVÉ CENTRUM

**Registrační číslo IPs EOSC-CZ**
CZ.02.01.01/00/22_004/0007682