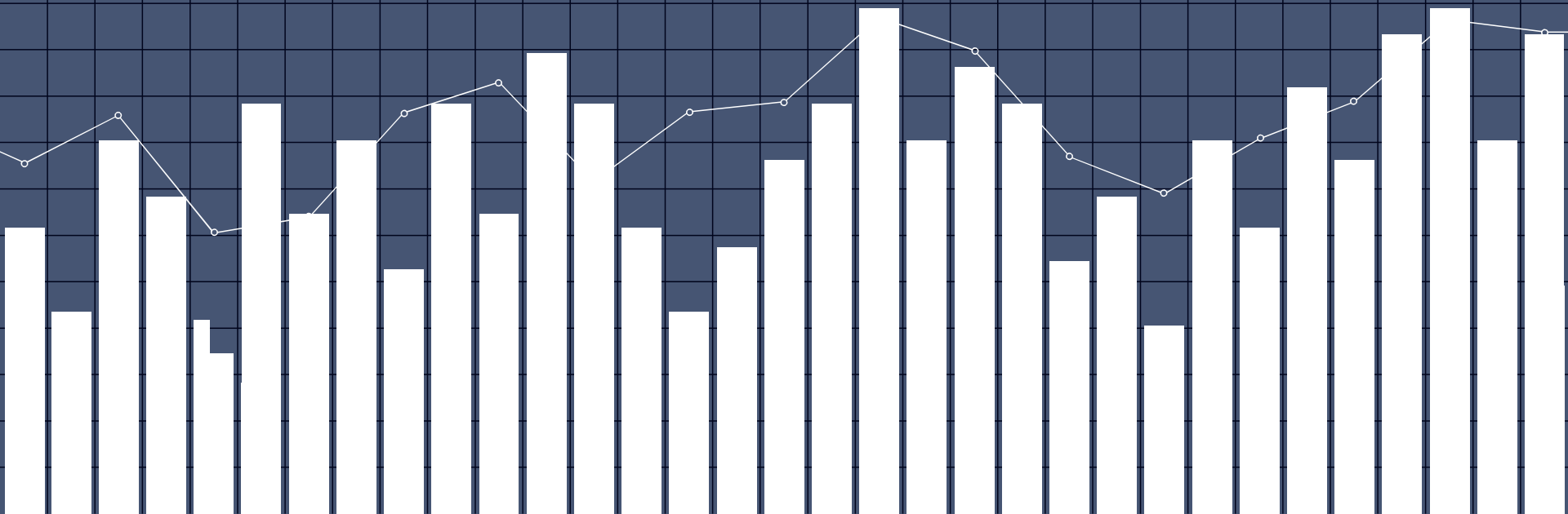
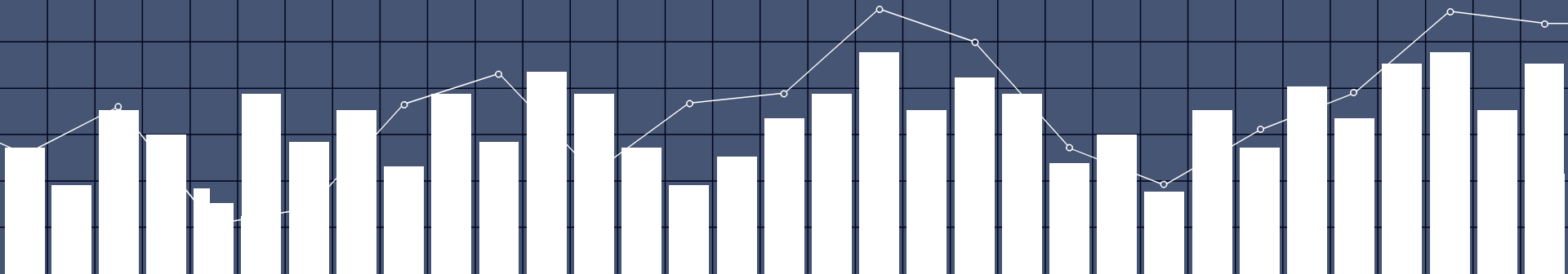


06. Explorace dat



Harmonogram

- ▣ 01. Rekapitulace
- ▣ 02. Kategorická data
- ▣ 03. Numerická data



Rekapitulace

readr



readr part of the tidyverse

[Home](#)[Overview](#)[Locales](#)[Reference](#)[News ▾](#)

Introduction to readr

The key problem that readr solves is **parsing** a flat file into a tibble. Parsing is the process of taking a text file and turning it into a rectangular tibble where each column is the appropriate part. Parsing takes place in three basic stages:

1. The flat file is parsed into a rectangular matrix of strings.
2. The type of each column is determined.
3. Each column of strings is parsed into a vector of a more specific type.

It's easiest to learn how this works in the opposite order. Below, you'll learn how the:

1. **Vector parsers** turn a character vector into a more specific type.
2. **Column specification** describes the type of each column and the strategy readr uses to guess types so you don't need to supply them all.
3. **Rectangular parsers** turn a flat file into a matrix of rows and columns.

Each `parse_*()` is coupled with a `col_*()` function, which will be used in the process of parsing a complete tibble.

Contents

- [Vector parsers](#)
- [Column specification](#)
- [Rectangular parsers](#)

Rekapitulace

data.table



[Home](#) [About](#) [RSS](#) [add your blog!](#) [Learn R](#) [R jobs](#) [Contact us](#)

WELCOME!

[Follow](#) 63.3K followers

Here you will find daily news and tutorials about R, contributed by over 750 bloggers. There are many ways to follow us -

By e-mail:

Your e-mail here

Subscribe

49212 readers
BY FEEDBURNER

On Facebook:



R blog...
74K likes

[Liked](#)

You and 17 other friends like this



If you are an R blogger yourself you are invited to add your own R content

Intro to The data.table Package

June 14, 2016

By Stevie P

[Like 2](#) [Share](#) [in Share](#)

(This article was first published on Rolling Your Rs, and kindly contributed to R-bloggers)

[f Share](#)

[Tweet](#)

Data Frames

R provides a helpful data structure called the “data frame” that gives the user an intuitive way to organize, view, and access data. Many of the functions that you would use to read in external files (e.g. `read.csv`) or connect to databases (RMySQL), will return a data frame structure by default. While there are other important data structures, such as the **vector**, **list** and **matrix**, the data frame winds up being at the heart of many operations not the least of which is aggregation. Before we get into that let me offer a very brief review of data frame concepts:

- A data frame is a set of rows and columns.
- Each row is of the same length and data type
- Every column is of the same length but can be of differing data types
- A data frame has characteristics of both a matrix and a list
- Bracket notation is the customary method of indexing into a data frame

SEARCH R-BLOGGERS

Google Custom Search



RECENT POPULAR POSTS

future.apply - Parallelize Any Base R Apply Function
Let R/Python send messages when the algorithms are done training
A primer in using Java from R - part 1
Forecasting my weight with R
Why R 2018 Winners

MOST VISITED ARTICLES OF THE WEEK

1. How to write the first for loop in R
2. Installing R packages
3. Using apply, sapply, lapply in R
4. R – Sorting a data frame by the contents of a column
5. How to perform a Logistic Regression in R
6. How to Make a Histogram with Basic R
7. Tutorials for learning R
8. How to Make a Histogram with ggplot2
9. Creating Slopegraphs with R

Rekapitulace styler

Tidyverse

Packages

Articles

Learn

Help

Contribute

styler 1.0.0



Photo by Heng Films

We're pleased to announce the release of **styler 1.0.0**. **styler** is a source code formatter - a package to format R code according to a style guide. It defaults to our implementation of the tidyverse style guide, but there is plenty of flexibility for a user to specify their own style. A coherent style is important for consistency and legibility. Just as it is important to put spaces between words. You can install **styler** from CRAN:

```
install.packages("styler")
```

styler can style text, single files, packages and entire R source trees with the following functions:

- `style_text()` styles a character vector.
- `style_file()` styles R and Rmd files.
- `style_dir()` styles all R and/or Rmd files in a directory.
- `style_pkg()` styles the source files of an R package.
- An RStudio Addin that styles the active R or Rmd file, the current package or the highlighted code.

Contents

Upcoming events

rstudio::conf 2019

Austin, TX
Jan 15-18

rstudio::conf 2019 covers all things RStudio, including workshops to teach you the tidyverse, and talks to show you the latest and greatest features.

tidyverse developer day

Austin, TX
Jan 19

Help the tidyverse team improve our code and documentation. First-time contributors are welcome.

Dataset

Videogames

 Search kaggle  Competitions Datasets Kernels Discussion Learn ...  



Silver
Videogames

last run a year ago · IPython Notebook HTML · 303 views
using data from [Video Game Sales with Ratings](#) ·  Public

0
voters

Notebook Code **Data (1)** Comments (0) Log Versions (4) Forks (3) [Fork Notebook](#)

Data

Data Sources

 Video Game Sales w...
 Video_Games_Sales_... 16.7k x 16



Video Game Sales with Ratings
Video game sales from Vgchartz and corresponding ratings from Metacritic
Last Updated: 2 years ago (Version 2)

About this Dataset

Context
Motivated by Gregory Smith's web scrape of VGChartz [Video Games Sales](#), this data set simply extends the number of variables with another web scrape from [Metacritic](#). Unfortunately, there are missing observations as Metacritic only covers a subset of the platforms. Also, a game may not have all the observations of the additional variables discussed below. Complete cases are ~ 6,900

Kategorická data

Kontingenční tabulky

Data

```
Videogames <- read_csv("Video_Games_Sales_as_at_22_Dec_2016.csv")
```

```
Videogames2 <- read.csv("Video_Games_Sales_as_at_22_Dec_2016.csv")
```

Skríning dat

```
view(dfSummary(Videogames))
```

Kontingenční tabulka – absolutní četnosti

```
table(Videogames$Genre, Videogames$Rating)
```

Kontingenční tabulka – hry pro všechny

```
Videogames_Everyone <- Videogames %>%  
  filter(Rating == "E") %>%  
  droplevels()
```

Kategorická data

Sloupcový graf

Data

```
Videogames_Everyone_Teen <- Videogames %>%  
  filter(Rating %in% c("E", "T")) %>%  
  droplevels()
```

Balíček

```
library(ggplot2)
```

Sloupcový graf

```
ggplot(Videogames_Everyone_Teen, aes(x = Genre, fill = Rating)) +  
  geom_bar(position = "dodge")
```

Sloupcový graf s úpravou popisků

```
ggplot(Videogames_Everyone_Teen, aes(x = Genre, fill = Rating)) +  
  geom_bar(position = "dodge") +  
  theme(axis.text.x = element_text(angle = 90))
```


Kategorická data

Kontingenční tabulky – relativní četnosti

Tabulka s žánry a hodnocením

```
GenreRating = table(Videogames_Everyone_Teen$Genre, Videogames_Everyone_Teen$Rating)
```

Celkové relativní četnosti

```
prop.table(GenreRating)
```

Relativní četnosti pro řádky

```
prop.table(GenreRating, 1)
```

Relativní četnosti pro sloupce

```
prop.table(GenreRating, 2)
```

Kategorická data

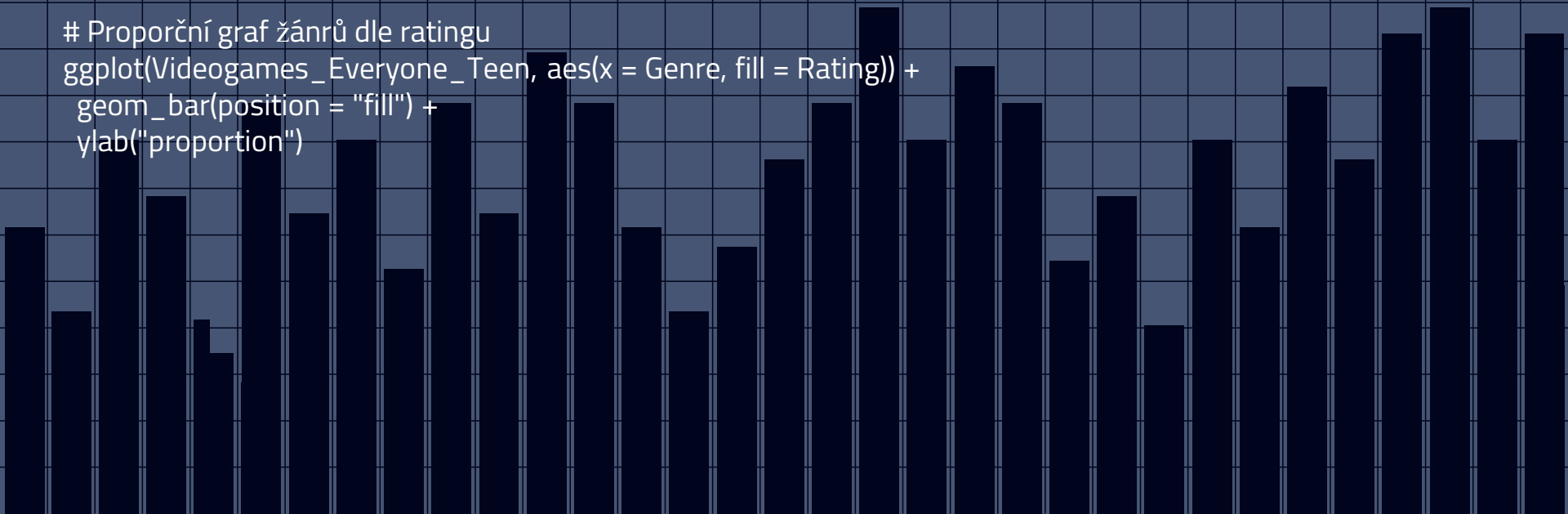
Sloupcový graf pro relativní četnosti

Sloupcový graf žánrů dle ratingu

```
ggplot(Videogames_Everyone_Teen, aes(x = Genre, fill = Rating)) +  
  geom_bar()
```

Proporční graf žánrů dle ratingu

```
ggplot(Videogames_Everyone_Teen, aes(x = Genre, fill = Rating)) +  
  geom_bar(position = "fill") +  
  ylab("proportion")
```



Kategorická data

Sloupcový graf pro hodnoty právě jedné proměnné a sloupcový graf s tříděním

```
# Faktorizace proměnné Rating
```

```
Videogames_Everyone_Teen$Rating <- factor(Videogames_Everyone_Teen$Rating,  
      levels = c("E", "T"), labels = c("Everyone", "Teen"))
```

```
# Sloupcový graf pro proměnnou Rating
```

```
ggplot(Videogames_Everyone_Teen, aes(x = Rating)) +  
  geom_bar()
```

```
# Sloupcový graf pro proměnnou žánr dle proměnné Rating
```

```
ggplot(Videogames_Everyone_Teen, aes(x = Genre)) +  
  geom_bar() +  
  facet_wrap(~ Rating) +  
  theme(axis.text.x = element_text(angle = 90))
```

Numerická data

Grafické zobrazení

Histogram s facetami (vrstvy)

```
ggplot(Videogames_Everyone_Teen, aes(x = Critic_Score)) +  
  geom_histogram() +  
  facet_wrap(~ Rating)
```

Filtrace her dle žánrů: střílečky, strategie a RPG

```
Shooter_Strategy_RPG <- filter(Videogames, Genre %in% c("Shooter", "Strategy", "Role-Playing"))
```

Box plot

```
ggplot(Shooter_Strategy_RPG, aes(x = as.factor(Genre), y = Critic_Score)) +  
  geom_boxplot()
```

Density plot s překryvem kategorií

```
ggplot(Shooter_Strategy_RPG, aes(x = Critic_Score,  
  fill = as.factor(Genre))) +  
  geom_density(alpha = .3)
```

Numerická data

Grafické zobrazení

Histogram počtu prodaných kusů her (v milionech) v EU

```
Videogames %>%
```

```
  ggplot(aes(x = EU_Sales)) +
```

```
  geom_histogram() +
```

```
  ggtitle("Počet prodaných kusů her (v milionech) v EU")
```

Histogram počtu prodaných kusů her se sportovní tematikou (v milionech) v EU

```
Videogames %>%
```

```
  filter(Genre == "Sports") %>%
```

```
  ggplot(aes(x = EU_Sales)) +
```

```
  geom_histogram() +
```

```
  xlim(c(0, 3)) +
```

```
  ggtitle("Počet prodaných kusů sportovních her (v milionech) v EU")
```

Numerická data

Grafické zobrazení

Hodnocení her uživateli – šířka sloupce 30

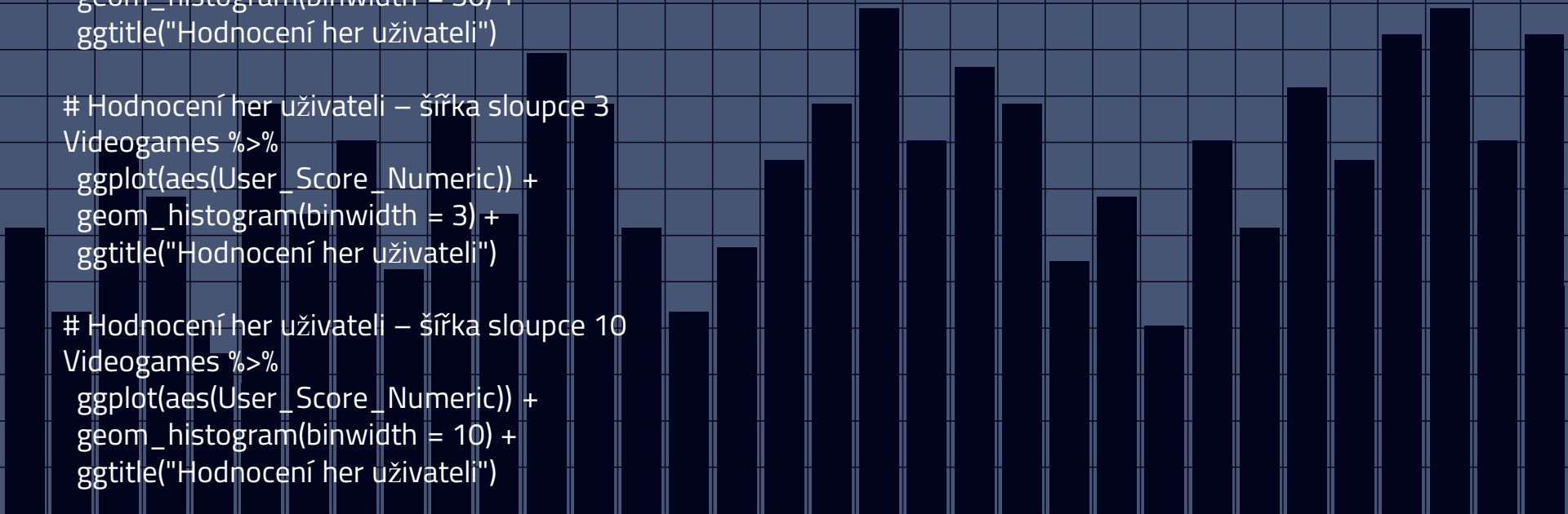
```
Videogames %>%  
  ggplot(aes(User_Score_Numeric)) +  
  geom_histogram(binwidth = 30) +  
  ggtitle("Hodnocení her uživateli")
```

Hodnocení her uživateli – šířka sloupce 3

```
Videogames %>%  
  ggplot(aes(User_Score_Numeric)) +  
  geom_histogram(binwidth = 3) +  
  ggtitle("Hodnocení her uživateli")
```

Hodnocení her uživateli – šířka sloupce 10

```
Videogames %>%  
  ggplot(aes(User_Score_Numeric)) +  
  geom_histogram(binwidth = 10) +  
  ggtitle("Hodnocení her uživateli")
```



Numerická data

Grafické zobrazení

Boxplot počtu hodnotících uživatelů

```
Videogames %>%  
  ggplot(aes(x = 1, y = User_Count)) +  
  geom_boxplot()
```

Vyřazení odlehlých hodnot

```
Videogames_no_out <- Videogames %>%  
  filter(User_Count < 1000)
```

Boxplot počtu hodnotících uživatelů bez odlehlých hodnot

```
Videogames_no_out %>%  
  ggplot(aes(x = 1, y = User_Count)) +  
  geom_boxplot()
```



37

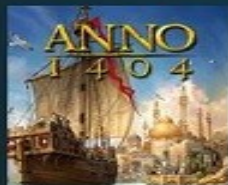
790

4600 XP to next level

My Games

Free Games

Hidden



Anno 1404 - GoL...



Anno 2070



Anno 2205



Assassin's Cree...



Assassin's Cree...



Assassin's Cree...



Assassin's Cree...



Assassin's Cree...



Assassin's Cree...



Child of Light



Far Cry 2



Far Cry® 4



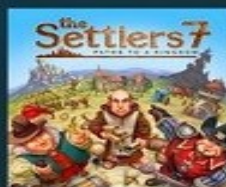
Might & Magic ...



Silent Hunter®: ...



The Crew



The Settlers 7: ...



Tom Clancy's R...



Trials Fusion

Numerická data

Sumarizace

```
# Data Ubisoftu
```

```
Ubisoft = filter(Videogames,  
  ggplot(aes(x = 1, y = User_Count)) +  
  geom_boxplot())
```

```
# Průměr a medián prodaných her (v milionech kusů) v Severní Americe Ubisoftu jako vydavatele dle žánru
```

```
Ubisoft %>%  
  group_by(Genre) %>%  
  summarize(mean(NA_Sales),  
    median(NA_Sales))
```

```
# Boxplot prodaných her (v milionech kusů) Ubisoftu v Severní Americe jako vydavatele dle žánru
```

```
Ubisoft %>%  
  ggplot(aes(x = Genre, y = NA_Sales)) +  
  geom_boxplot()
```

Numerická data

Sumarizace

Density plot pro nezměněné hodnoty

```
Ubisoft %>%  
  ggplot(aes(x = NA_Sales)) +  
  geom_density()
```

Logaritmická transformace proměnné

```
Ubisoft <- Ubisoft %>%  
  mutate(log_NA_Sales = log(NA_Sales))
```

Density plot pro transformované hodnoty

```
Ubisoft %>%  
  ggplot(aes(x = log_NA_Sales)) +  
  geom_density()
```

Numerická data

Identifikace odlehlých hodnot

Filtr pro akční hry od Ubisoftu, přidání proměnné indikující, zda jde o odlehlou hodnotu

```
Ubisoft_Action <- Ubisoft %>%  
  filter(Genre == "Action") %>%  
  mutate(is_outlier = (NA_Sales < 1))
```

Odstranění odlehlých hodnot z analýzy, vypracování boxplotu

```
Ubisoft_Action %>%  
  filter(is_outlier == FALSE) %>%  
  ggplot(aes(x = 1, y = NA_Sales)) +  
  geom_boxplot()
```

Zdroje

Data Wrangling with dplyr and tidyr Cheat Sheet – <https://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf>

