# A Beginners Guide to R's Galaxy

*Michal J. Czyz*

*2018-03-16*

# Contents

# Chapter 1

# In the beginning there was only darknes...

**R** (R Core Team, 2017) is one of the most common used languages in Data Science. It is so called fourth-generation programming language (4GL), meaning it is *user-friendly*, while still quite powerful. **R** is powered by huge open-source oriented community. Thanks to their work, during many years of development, enormous number of *packages* (also incorrectly called *libraries*) were established, making using **R** for common works related to Data Science easy even for Beginners.

The purpose of this document is to familiarize with **R** people who have at least some basics in statistics or modelling and no knowledge on programming. Thus, examples you will find in this book are driven by making life easier for all of those who struggle with data in their work.

To give you an example and how awesome and powerful **R** is, I wrote whole this book in **R** using package **bookdown** (Xie, 2016, 2018). Hoping this short description encouraged you to dive into *World of R*, we can start learning opportunities of this programming language.

# Chapter 2

# Introduction

Placeholder

## 2.1 RStudio

## 2.2 Few tips to make life easier

## 2.3 Installing packages

## 2.4 Conventions

# Chapter 3

# Basics

Placeholder

## 3.1 Getting started

## 3.2 Help

## 3.3 Internet is a great source of information

## 3.4 More on internet sources

## 3.5 Syntax

## 3.6 Common operators

## 3.7 Variables

### 3.7.1 Naming Variables

## 3.8 Math operations

## 3.9 Logics

## 3.10 Functions

# Chapter 4

# Somwehere between basic and useful

Placeholder

## 4.1 Addressing

## 4.2 Vectors

## 4.3 Data frames (and matrices)

## 4.4 Lists

## 4.5 Operation on Vectors

## 4.6 Randomization and distribution

## 4.7 `tidyverse` idea and `dplyr` library

## 4.8 Wide vs. long tables

## 4.9 World of `dplyr`

## 4.10 `select` columns and `filter` rows

## 4.11 `mutate` and `transmute`

## 4.12 `group_by` and `summarise`

## 4.13 Is there anything more in `dplyr` library?

# Chapter 5

# Lets do some math!

Placeholder

## 5.1 Simple statistical model

## 5.2 Other models

## 5.3 Packages

## 5.4 Simple mechanistic model and noise

## 5.5 Polynomial models in R

## 5.6 Solving differential equations

# Chapter 6

# Functions

Placeholder

## 6.1 Simple math functions.

## 6.2 Building your own calculator

## 6.3 It is not over yet... *Calculator shouldn't divide be 0*!

# Chapter 7

# Graphics

What would be our work value without visualization? Not much. **R** provides use with some tools to make plots, charts and other visual stuff. However base version has very limited graphic design by default and making it pretty needs a lot of time and code. Nowadays, however, in `tidyverse` there is a very powerful library with dozens of extensions - `ggplot2`. *GG* stands for *Gramar of Graphics* and in practice it means that we build our visualization layer after layer. The vast spectrum of `ggplot` functions and its extensions is out of the scope of this book. Here we will focus on the basic and most useful things as well as how to find not so common functionalities.

## 7.1 Base plots

Before we get to `ggplot2`, we should begin with base graphic functions. It is true that the do not look as pretty as plots from dedicated libraries, but they are extremely useful in quick checking of our workflow. Not to mention, that many packages are still operating with old fashioned base graphics.

## 7.2 How to make plot?

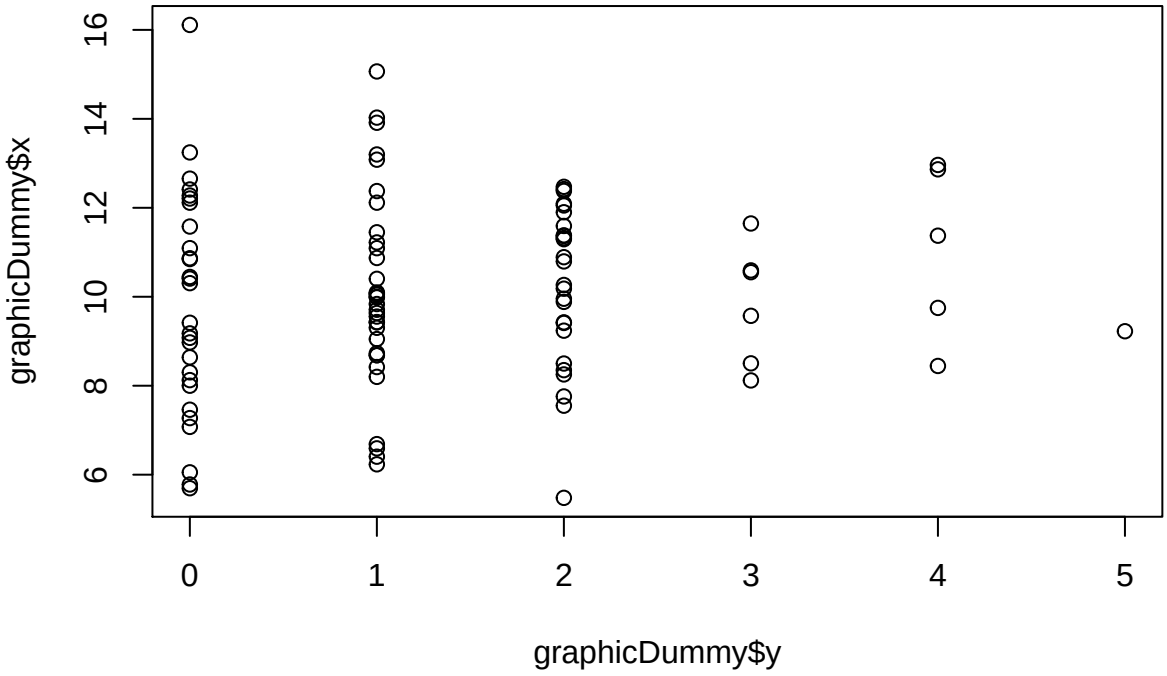
In previous chapters we made some plots, but we did not cover the details. Most important thing is to acknowledge that we can make plots with `hist()`, `plot()`, `barplot()` and `boxplot()` functions (to mention only the most common). There are of course other types of plots you might need, however those three are the *classic* ones. If you are looking for more examples of base plots, you should visit help page of `graphics` package. To change the look of your plot, just provide proper arguments to this function, like `col` for color of line/points or, `xlab` and `ylab` to change names of axis labels. Unfortunately a list of arguments that you can control and adjust is long. The good news is that all of them have their default value, so you need to change only the ones you want, without thinking about rest. Full list of parameter you can change you will find after executing: `?par` command. As this help page contains many different kinds of parameters you should scroll to find *Graphical Parameters* chapter.

## 7.3 Can we actually do something?

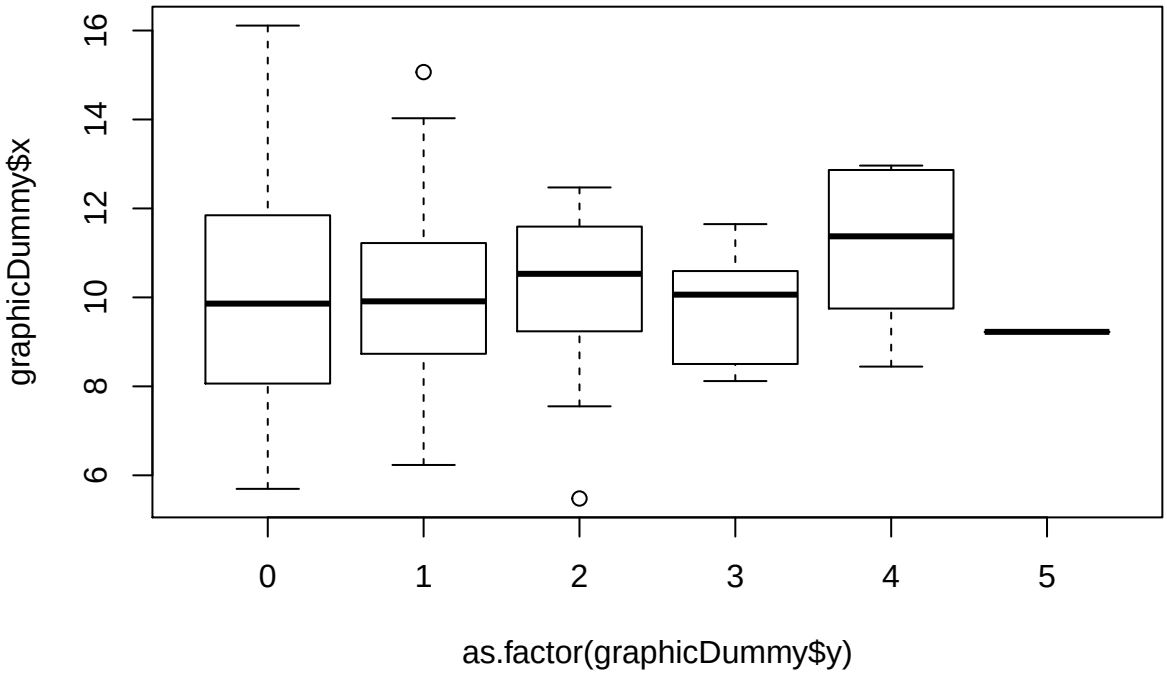
Yes we can. But as I said earlier scope of this book is not graphics. Thus I will cover here only the basic things you should know, or at least know where to find them. Under I will present only code and its output. Try to figure out yourself what is going on. You can also copy - paste the code in your script or console,

edit it and learn how it works by doing. Even the best book is just a book, and nothing will substitute the knowledge you gain by experiencing.

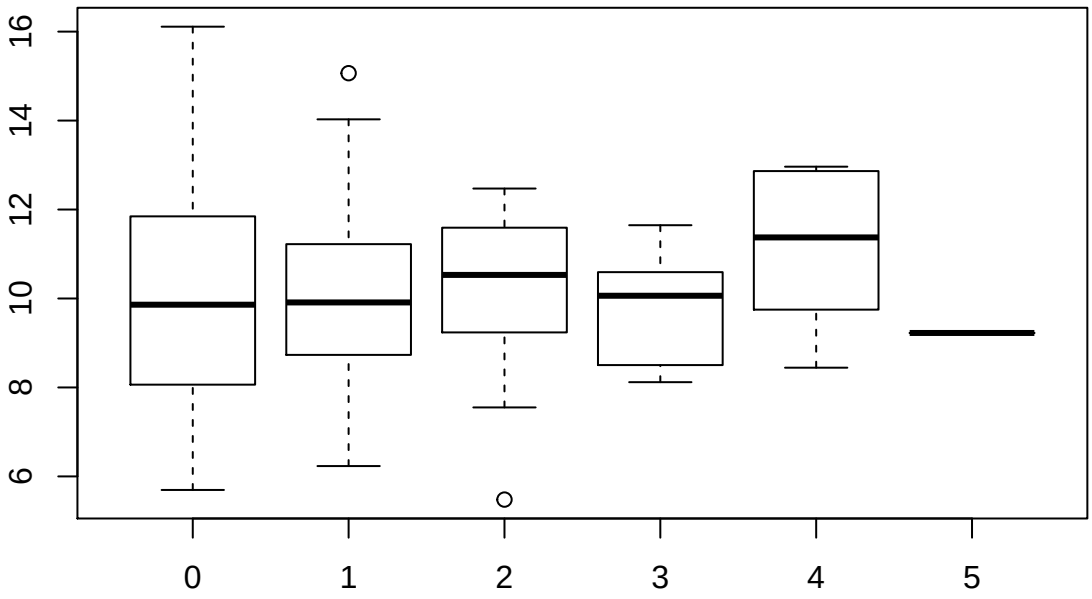
```
graphicDummy <- data.frame(x = rnorm(100, 10, 2), y = rpois(100, 1.2))
plot(graphicDummy$x~graphicDummy$y)
```



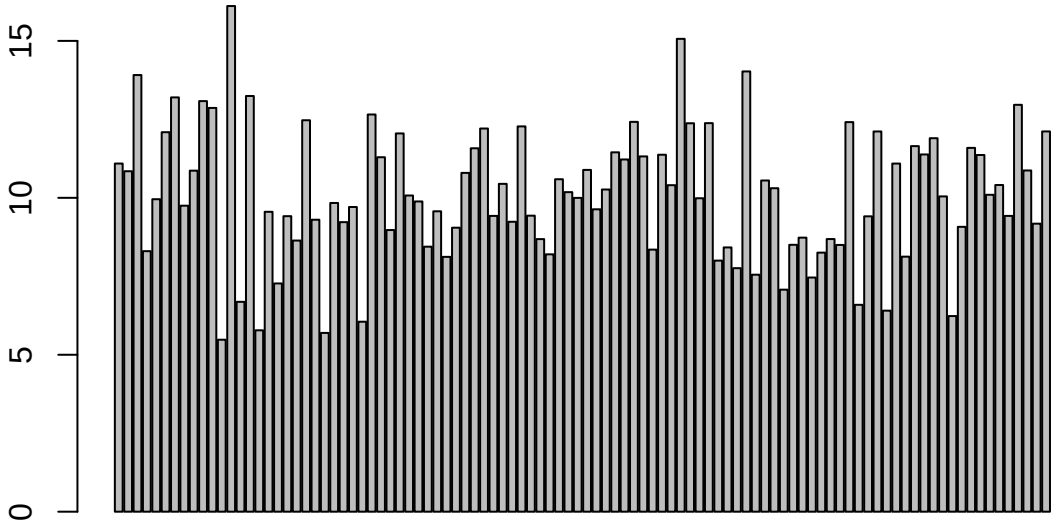

```
plot(graphicDummy$x~as.factor(graphicDummy$y))
```



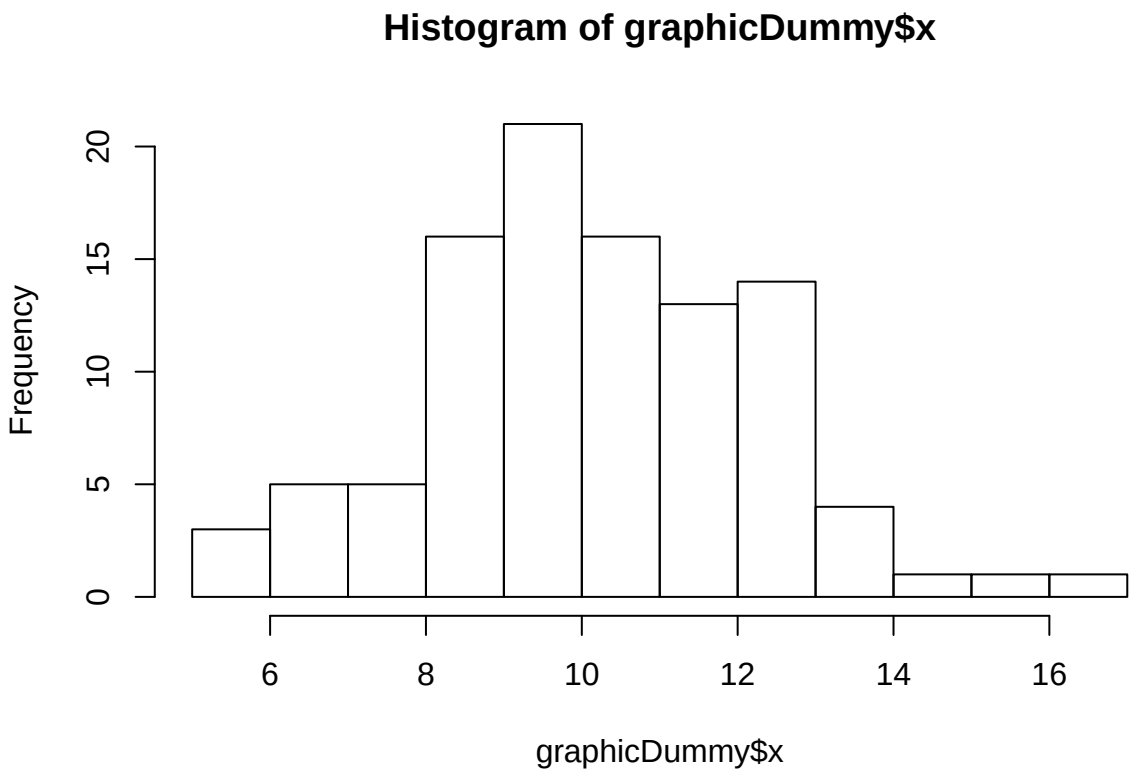

```
boxplot(graphicDummy$x~graphicDummy$y)
```



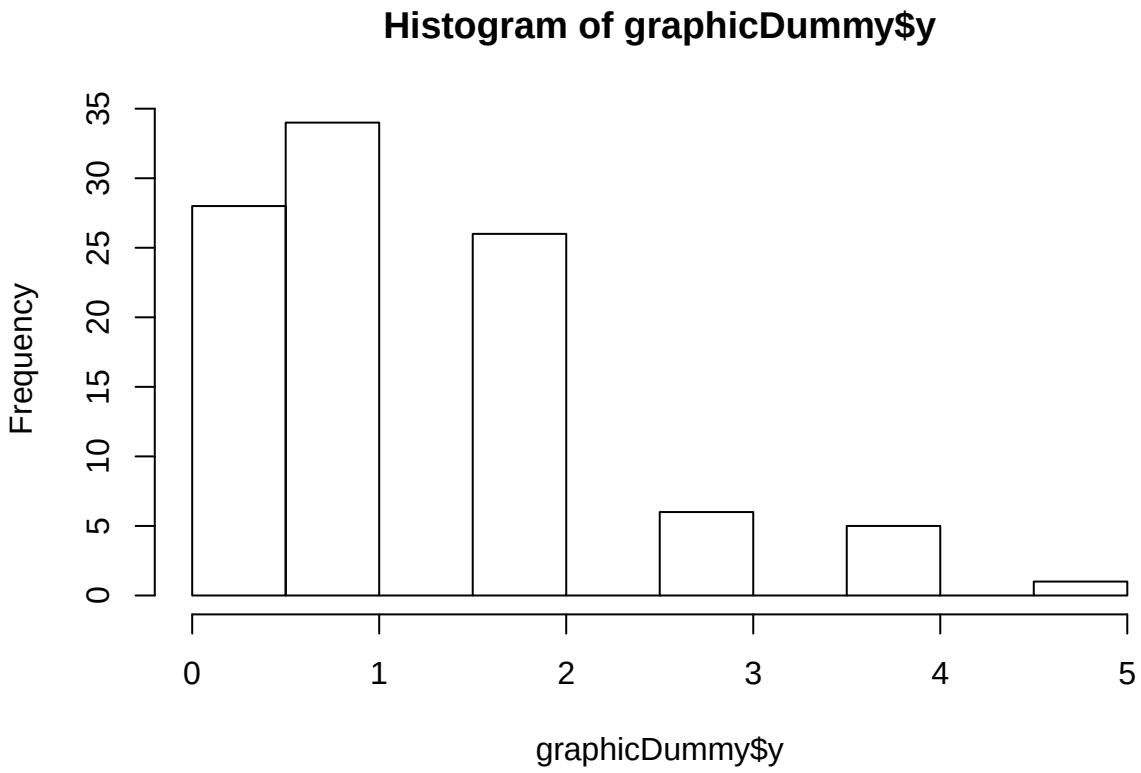

```
barplot(graphicDummy$x)
```



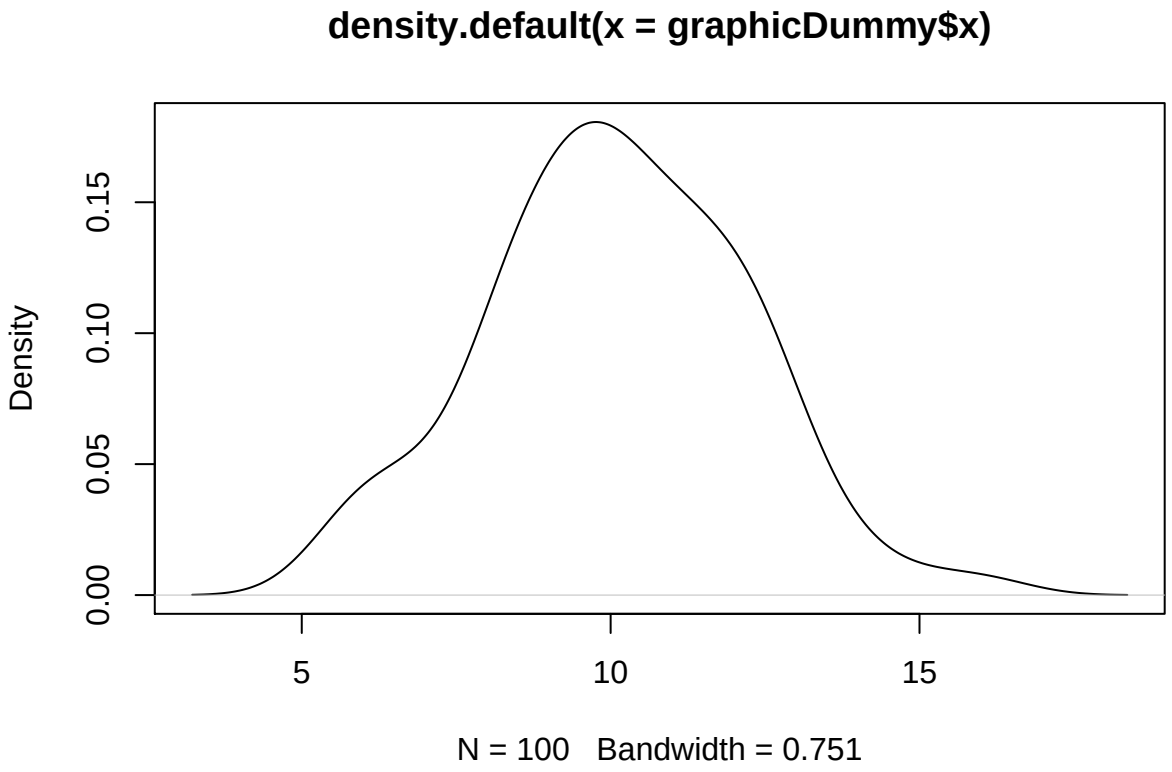

```
hist(graphicDummy$x)
```



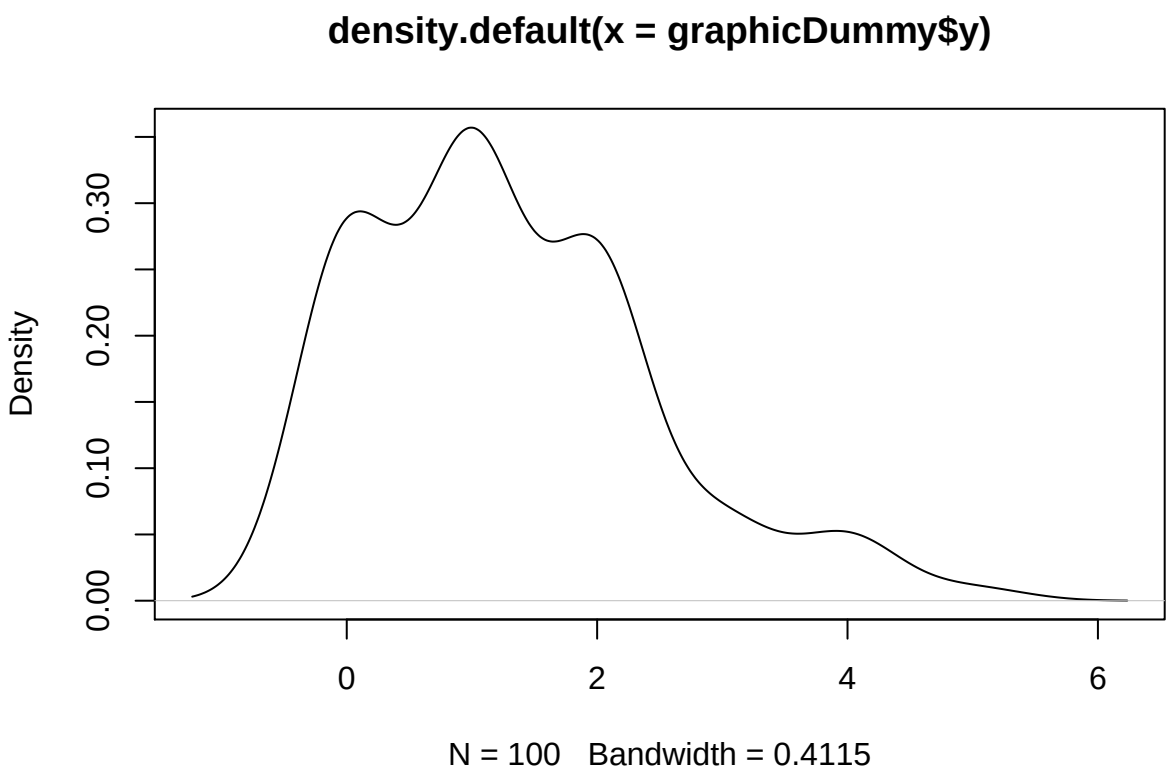

```
hist(graphicDummy$y)
```




```
plot(density(graphicDummy$x))
```




```
plot(density(graphicDummy$y))
```



# Chapter 8

# Final indications

## 8.1 Use Projects

To organize your work, not only in **RStudio**, but also on hard drive you should use projects. It is extremely easy in this IDE. Just click button – *Create new project* – choose *New Directory* and type of project you need. **RStudio** will take care of everything for you. Later you can easily add files of sub-folders with specifying content into your project and use relational links.

## 8.2 Use RMarkodow

To make your work more reproducible, and also to make nice looking output (in html or pdf) it is a good idea to work in RMarkdown files instead of RScripts. The syntax of **RMarkdown** is nearly identical to original **Markdown**, but allows you to execute and store **RCode**. You can find RMarkdown introduction here. There is also nice cheat sheet which contains all commands you will need. As there is strong pressure to make research and science more open and reproducible, it is strongly advised that you work with proper tools to do it like **RMarkdown** or **Project Jupyter**. If you want to learn more, there is very nice guide by British Ecological Society you should read.

## 8.3 Use .rds files

Usually we work with plain text files like csv or tsv in **R**. However, there is also a highly valuable format of files called rds, that makes work even more pleasant. It not only preserves all classes of variables, but also takes less space than plain text file, due to compression algorithms. Also when using `saveRDS()` you don't need to define hundreds of parameters, just name of object you want to save and its file name. The downside is that it is format to use directly with R.

## 8.4 Nice resources you SHOULD read

- Basic Transformations and Explorations in the Tidyverse
- RECONlearn free resources on epidemiology
- R Graphic CookBook free on-line version
- Collection of R cheat sheets
- Collection of best news from data science world

- Great blog discussing issues with statistics and data science

## 8.5 Whole Game

We manage to go through whole introduction to *R's Galaxy*. But that's not the end. For a final word I would like to go with you with once again through through all the stuff you will need more or less in your everyday job. We will build few functions to analyse data, and present them with nice graphics and tables.

# Chapter 9

# References

# Bibliography

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Xie, Y. (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. Chapman and Hall/CRC, Boca Raton, Florida. ISBN 978-1138700109.

Xie, Y. (2018). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.7.