

# YaRrr! The Pirate's Guide to R

*Nathaniel D. Phillips*

*2017-02-22*



# Contents

<b>1</b>	<b>Preface</b>	<b>5</b>
1.1	Where did this book come from? . . . . .	7
1.2	Who is this book for? . . . . .	7
1.3	Why is R so great? . . . . .	7
1.4	R is like a relationship... . . . . .	9
1.5	Who am I? . . . . .	11
<b>2</b>	<b>Getting Started</b>	<b>13</b>
<b>3</b>	<b>Jump In!</b>	<b>15</b>
<b>4</b>	<b>The Basics</b>	<b>17</b>
<b>5</b>	<b>Scalars and vectors</b>	<b>19</b>
<b>6</b>	<b>Vector functions</b>	<b>21</b>
<b>7</b>	<b>Indexing Vectors with [ ]</b>	<b>23</b>
<b>8</b>	<b>Matrices and Dataframes</b>	<b>25</b>
<b>9</b>	<b>Test your R might! Solutions</b>	<b>27</b>
<b>10</b>	<b>Placeholder</b>	<b>29</b>





## Chapter 1

## Preface



The purpose of this book is to help you learn R from the ground-up.

## 1.1 Where did this book come from?

Let me make something very, very clear...

*I did not write this book.*

This whole story started in the Summer of 2015. I was taking a late night swim on the Bodensee in Konstanz and saw a rusty object sticking out of the water. Upon digging it out, I realized it was an ancient usb-stick with the word YaRrr inscribed on the side. Intrigued, I brought it home and plugged it into my laptop. Inside the stick, I found a single pdf file written entirely in pirate-speak. After watching several pirate movies, I learned enough pirate-speak to begin translating the text to English. Sure enough, the book turned out to be an introduction to R called The Pirate's Guide to R.

This book clearly has both massive historical and pedagogical significance. Most importantly, it turns out that pirates were programming in R well before the earliest known advent of computers. Of slightly less significance is that the book has turned out to be a surprisingly up-to-date and approachable introductory text to R. For both of these reasons, I felt it was my duty to share the book with the world.

If you or spot any typos or errors, or have any recommendations for future versions of the book, please write me at [YaRrr.Book@gmail.com](mailto:YaRrr.Book@gmail.com) or tweet me @YaRrrBook.

## 1.2 Who is this book for?

While this book was originally written for pirates, I think that anyone who wants to learn R can benefit from this book. If you haven't had an introductory course in statistics, some of the later statistical concepts may be difficult, but I'll try my best to add brief descriptions of new topics when necessary. Likewise, if R is your first programming language, you'll likely find the first few chapters quite challenging as you learn the basics of programming. However, if R is your first programming language, that's totally fine as what you learn here will help you in learning other languages as well (if you choose to). Finally, while the techniques in this book apply to most data analysis problems, because my background is in experimental psychology I will cater the course to solving analysis problems commonly faced in psychological research.

### What this book is

This book is meant to introduce you to the basic analytical tools in R, from basic coding and analyses, to data wrangling, plotting, and statistical inference.

### What this book is not

This book does not cover any one topic in extensive detail. If you are interested in conducting analyses or creating plots not covered in the book, I'm sure you'll find the answer with a quick Google search!

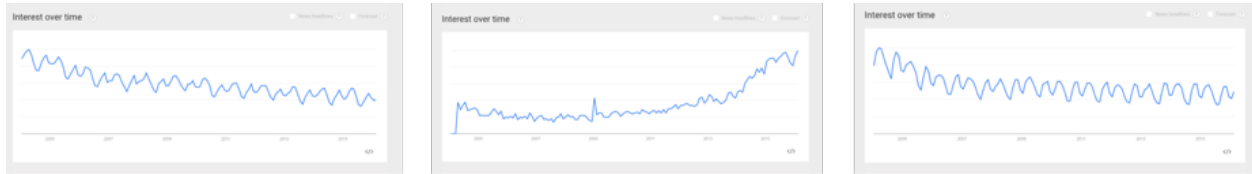
## 1.3 Why is R so great?

As you've already gotten this book, you probably already have some idea why R is so great. However, in order to help prevent you from giving up the first time you run into a programming wall, let me give you a few more reasons:

1. R is 100% free and as a result, has a huge support community. Unlike SPSS, Matlab, Excel and JMP, R is, and always will be completely free. This doesn't just help your wallet - it means that a huge community of R programmers will constantly develop and distribute new R functionality and packages at a speed that leaves all those other packages in the dust! Unlike Fight Club, the first rule of R is "Do talk about R!" The size of the R programming community is staggering. If you ever have a question

about how to implement something in R, a quick Poog<sup>1</sup> search will lead you to your answer virtually every single time.

2. R is the presnt, and future of statistical programming. To illustrate this, look at the following three figures. These are Google trend searches for three terms: R Programming, Matlab, and SPSS. Try and guess which one is which.



3. R is incredibly versatile. You can use R to do everything from calculating simple summary statistics, to performing complex simulations to creating gorgeous plots like the chord diagram on the right. If you can imagine an analytical task, you can almost certainly implement it in R.
4. Using RStudio, a program to help you write R code, You can easily and seamlessly combine R code, analyses, plots, and written text into elegant documents all in one place using Sweave (R and Latex) or RMarkdown. In fact, I translated this entire book (the text, formatting, plots, code...yes, everything) in RStudio using Sweave. With RStudio and Sweave, instead of trying to manage two or three programs, say Excel, Word and (sigh) SPSS, where you find yourself spending half your time copying, pasting and formatting data, images and test, you can do everything in one place so nothing gets misread, mistyped, or forgotten.

```
circlize::chordDiagram(matrix(sample(10),
                               nrow = 2, ncol = 5))
```

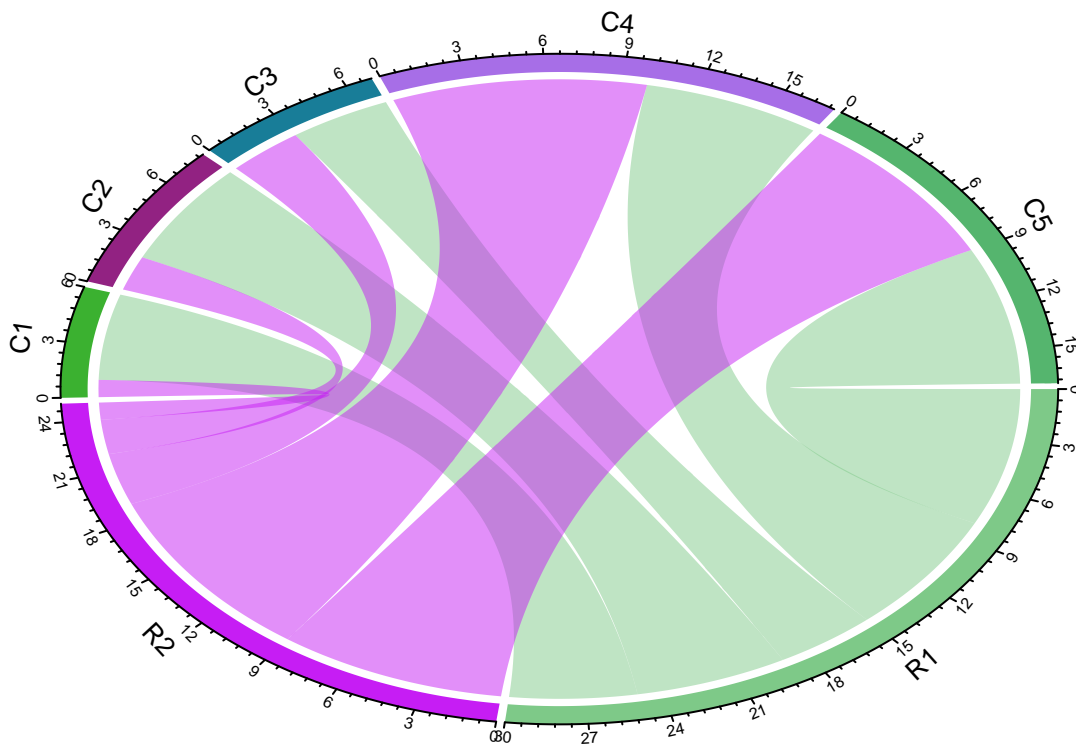


Figure 1.1: A super cool chord diagram from the circlize package

<sup>1</sup>I am in the process of creating Poog<sup>1</sup> - Google for Pirates. Kickstarter page coming soon...





Figure 1.2: Yep, R will become both your best friend and your worst nightmare. The bad times will make the good times oh so much sweeter.

5. Analyses conducted in R are transparent, easily shareable, and reproducible. If you ask an SPSS user how they conducted a specific analyses, they will either A) Not remember, B) Try (nervously) to construct an analysis procedure on the spot that makes sense - which may or may not correspond to what they actually did months or years ago, or C) Ask you what you are doing in their house. I used to primarily use SPSS, so I speak from experience on this. If you ask an R user (who uses good programming techniques!) how they conducted an analysis, they should always be able to show you the exact code they used. Of course, this doesn't mean that they used the appropriate analysis or interpreted it correctly, but with all the original code, any problems should be completely transparent!
6. And most importantly of all, R is the programming language of choice for pirates.

## 1.4 R is like a relationship...

Yes, R is very much like a relationship. Like relationships, there are two major truths to R programming:

1. There is nothing more *frustrating* than when your code does *not* work
2. There is nothing more *satisfying* than when your code *does* work!

Anything worth doing, from losing weight to getting a degree, takes time. Learning R is no different. Especially if this is your first experience programming, you are going to experience a *lot* of headaches when you get started. You will run into error after error and pound your fists against the table screaming: "WHY ISN'T MY CODE WORKING?!?!? There must be something wrong with this stupid software!!!" You will spend hours trying to find a bug in your code, only to find that - frustratingly enough, you had had an extra space or missed a comma somewhere. You'll then wonder why you ever decided to learn R when (::sigh::) SPSS was so "nice and easy."

**Fun Fact!** SPSS stands for "Shitty Piece of Shitty Shit". True story.

This is perfectly normal! Don't get discouraged and DON'T GO BACK TO SPSS! That would be quitting on exercise altogether because you had a tough workout.

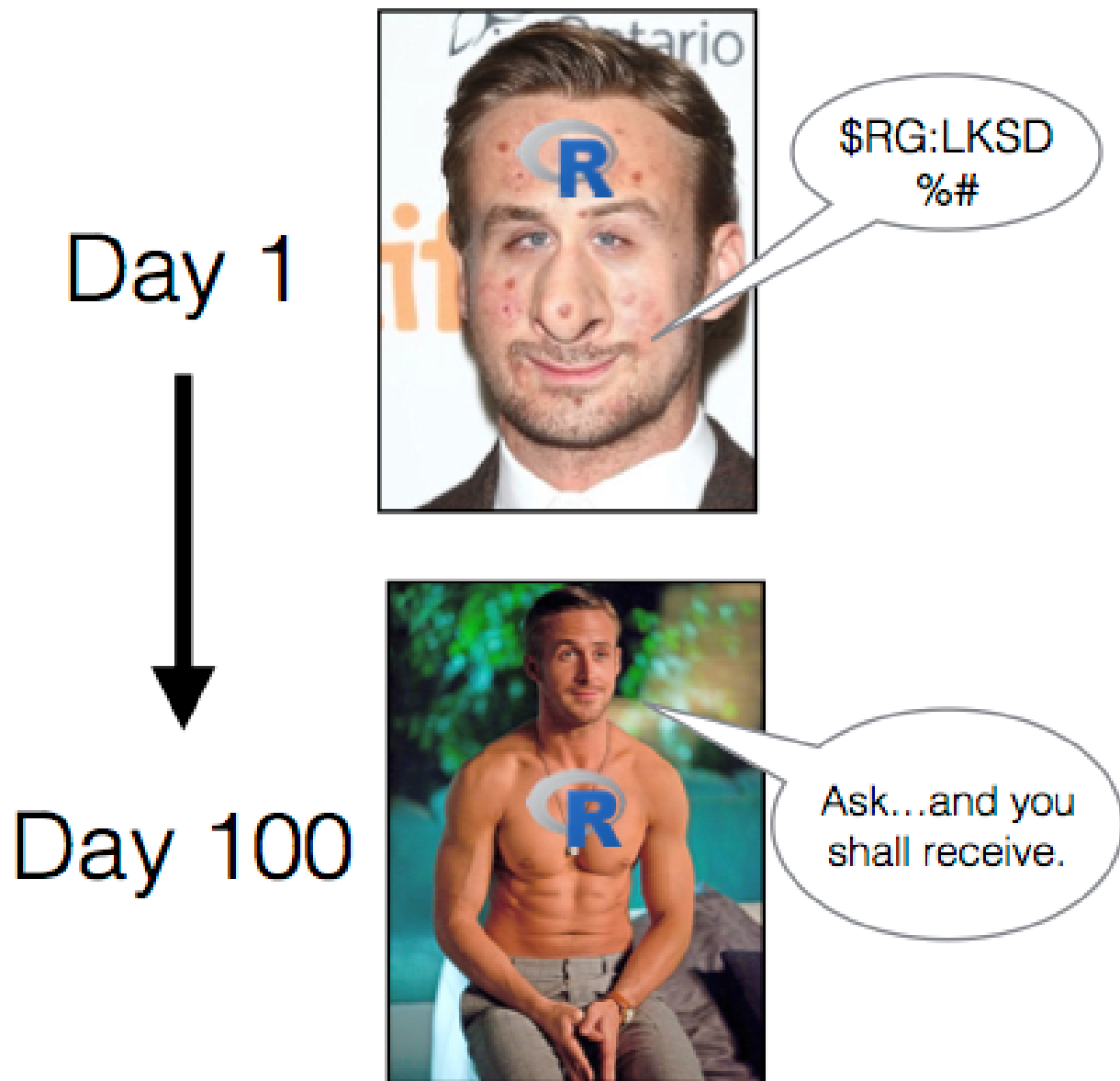


Figure 1.3: When you first meet R, it will look so fugly that you'll wonder if this is all some kind of sick joke. But trust me, once you learn how to talk to it, and clean it up a bit, all your friends will be crazy jealous.

Trust me, as you gain more programming experience, you'll experience fewer and fewer bugs (though they'll never go away completely). Once you get over the initial barriers, you'll find yourself conducting analyses much, much faster than you ever did before.

## 1.5 Who am I?

My name is Nathaniel – not Nathan...not Nate...and *definitely* not Nat. I am a psychologist with a background in statistics and judgment and decision making. You can find my R (and non-R) related musings at <http://ndphillips.github.io>



Figure 1.4: Like a pirate, I work best with a mug of beer within arms' reach.

## Chapter 2

# Getting Started



## Chapter 3

# Jump In!





## Chapter 4

# The Basics



## Chapter 5

# Scalars and vectors



## Chapter 6

# Vector functions



## Chapter 7

# Indexing Vectors with [ ]





## Chapter 8

# Matrices and Dataframes



## Chapter 9

# Test your R might! Solutions



## Chapter 10

# Placeholder



# Bibliography