

# Doing Meta-Analysis in R

A Hands-On Guide

*Mathias Harrer, B.Sc. & Dr. habil. David Ebert*

*Friedrich-Alexander-University Erlangen-Nuremberg*



# Contents

<b>1</b>	<b>About this Guide</b>	<b>5</b>
<b>2</b>	<b>RStudio &amp; Basics</b>	<b>7</b>
2.1	Getting RStudio to run on your computer . . . . .	7
2.2	Basics . . . . .	8



# Chapter 1

## About this Guide

This guide shows you how to conduct Meta-Analyses in R from scratch. The focus of this guide is primarily on clinical outcome research in psychology. The guide is designed for staff and collaborators of the **PROTECT Lab**, which is headed by **Dr. David D. Ebert**.



### The guide will teach you how to:

- Get **R** and **RStudio** set for your Meta-Analysis
- Get your data into R
- Prepare your data for the meta-analysis
- Perform fixed- and random-effects meta-analysis using the `meta` and `metafor` packages
- Analyse the heterogeneity of your results
- Tackle heterogeneity using subgroup analyses and meta-regression
- Check if selective outcome reporting is a present in your data
- Control for selective outcome reporting
- Analyse the risk of bias in your data
- Do advanced types of meta-analyses, such as
  - network analyses or
  - meta-analyses with more than one outcome

### What this guide will not cover

Although this guide will provide some information on the statistics behind meta-analysis, it will not give you an in-depth introduction into how meta-analyses are calculated mathematically.

It is also beyond the scope of this guide to advise in detail which meta-analytical strategy is suited best in certain contexts, and on how the search, study inclusion and reporting of meta-analyses should be conducted. The *Cochrane Handbook for Systematic Reviews of Interventions*, however, should be a great source to find more information on these topics.

**Generally, there are two other books to recommend when conducting Meta-Analyses:**

- If you're looking for an easily digestible, ready-to-use book on how Meta-Analyses can be conducted, we can recommend **Pim Cuijpers' online courses on Meta-Analysis**. The courses are freely available on YouTube. To have a look, [click here](#).
- If you're interested in more details on how to conduct Meta-Analyses in R, you can either have a look at Wolfgang Viechtbauer's page for the `metafor` package ([Link](#)). Or you can consult a book on the `meta` package which was recently published (Schwarzer et al., 2015).

**How to get the R code for this guide** All code behind this book is available online on **GitHub**. We have created a website containing a **download link** for all codes, and a **quick guide** on how to get the code running on your computer. The site can be found [here](#).

**To get started, proceed to the next chapter!**

## Chapter 2

# RStudio & Basics

Before we start with our meta-analysis, we have to download and prepare a **computer program** which allows us to use *R* for programming.

Probably the best option for this at the moment is **RStudio**. This program gives us a user interface which makes it easier to overlook our data, packages and output. The best part is that RStudio is completely free and can be downloaded anytime.

In this Chapter, we'll focus on how you can install RStudio on your computer. We'll also provide some general information on *R*, and how you can get help if you get error messages.

If you already have RStudio installed on your computer, and if you're an experienced *R* user already, all of this might be nothing new for you. You may **skip** this chapter then.

Especially if you have **never used *R* before**, we would like to consider this Chapter essential, as it gives you some input on how *R* works, and how we can use it for our data.

### 2.1 Getting RStudio to run on your computer



As a prerequisite for this guide, you need to have **RStudio** and a few essential **R packages** installed.

**You have to follow these steps to get your RStudio set.**

1. Download RStudio on the **RStudio** Website (Link). It's free!
2. If you do not have **R** installed yet, you will have to install the latest *R* Version before you can use RStudio. You can get the latest *R* version [here](#).
3. Once RStudio is running, open the **Console** on the bottom left corner of your screen.
4. We will now install a few packages using *R* Code. Here's an overview of the packages, and why we need them:

Package	Description
tidyverse	This is a large package containing various functions to tidy up your data
meta	This package is a so-called wrapper for many meta-analytic functions and makes it easy to code different meta-analyses
metafor	This package is used by the meta package for many applications, so we need to have it installed

5. To install these packages, we use the `install.packages()` function in R. One package after another, our code should look like this:

```
install.packages("tidyverse")
install.packages("meta")
install.packages("metafor")
```

Don't forget to put the packages in `"`. Otherwise, you will get an error message.

**You are now set and ready to proceed. Below, you can find some basic information on RStudio and troubleshooting**

## 2.2 Basics

Order to get the most out of this guide, it's helpful if you have some programming experience already. If you've never programmed before, you might find *Hands-On Programming with R* (Grolemund, 2014) to be a useful primer.

### 2.2.1 Running R Code

There are three things you need to run the code: **R**, **RStudio**, and collection of **R packages**. Packages are the fundamental units of reproducible R code. They include reusable functions, the documentation that describes how to use them, and sample data.

Gladly, once you've reached this point successfully, these three things are set already. Nevertheless, we will have to install and load a few new packages at some place in this guide, for which you can use the `install.packages()` the same way as you did before.

Throughout the guide, a consistent set of conventions is used to refer to code:

- Functions are in a code font and followed by parentheses, like `sum()` or `mean()`.
- Other R objects (like data or function arguments) are in a code font, without parentheses, like `seTE` or `method.tau`.
- Sometimes, we'll use the package name followed by two colons, like `meta:metagen()`. This is also valid R code. This is used so as to not confuse the functions of different packages with the same name.

### 2.2.2 Getting Help

As you start to apply the techniques described in this guide to your data you will soon find questions that the guide does not answer. This section describes a few tips on how to get help.

- If you get stuck, start with Google. Typically, adding "R" to a search is enough to restrict it to relevant results: if the search isn't useful, it often means that there aren't any R-specific results available. Google is particularly useful for error messages. If you get an error message and you have no idea what it means, try googling it. Chances are that someone else has been confused by it in the past, and there will be help somewhere on the web. (If the error message isn't in English, run `Sys.setenv(LANGUAGE = "en")` and re-run the code; you're more likely to find help for English error messages.)
- If Google doesn't help, try [stackoverflow](#). Start by spending a little time searching for an existing answer; including [R] restricts your search to questions and answers that use R.
- Lastly, if you stumble upon an error (or typos!) in this guide's syntax, feel free to contact **Mathias Harrer** at [mathias.harrer@fau.de](mailto:mathias.harrer@fau.de).



# Bibliography

Grolemund, G. (2014). *Hands-On Programming with R: Write Your Own Functions and Simulations*. O'Reilly.

Schwarzer, G., Carpenter, J. R., and Rücker, G. (2015). *Meta-analysis with R*. Springer.