

RStudio:
untuk Ekonomi dan Bisnis

Tedy Herlambang

—

—

—

—

2025

For The Rising Star

Ucapan Terima Kasih

Terima kasih yang sebesar-besarnya saya sampaikan kepada: Prof. Abdul Haris, Dr. Titik Musriati dan Dr. Ngatimun (Universitas Panca Marga), Muhammad Agus Nugroho dan Rifaldi Kadir (UIN Gorontalo), Rudi Masniadi (Universitas Teknologi Sumbawa), Erlyn Yuniashri dan Ajeng Kartika Galuh (Universitas Brawijaya Malang), Oktaviani Ika Wijayanti, Frederic Winston Nalle (Universitas Timor), Yeni Puspita (Universitas Negeri Jember) dan Muhammad Rizal (Universitas Islam Malang).

Tedy Herlambang
UPM
4 Januari 2024

Sinopsis

Buku ini merupakan *pengantar* cara melakukan analisis ekonomi dan bisnis secara kuantitatif dengan menggunakan R dan RStudio. Persoalan yang dibahas umumnya tingkat S1 atau S2.

Saya berharap pembaca dapat mengambil satu gagasan dari buku ini dalam melakukan investigasi ekonomi dan bisnis secara kuantitatif: data dan alat-alat analisis tidak sempurna. Namun, ketika kita memahami kekuatan dan kelemahan alat-alat ini, kita dapat menggunakannya untuk menemukan hal-hal menarik di dalam ekonomi dan bisnis.

Jangan ragu untuk menghubungi saya dengan pemikiran Anda tentang buku ini, ide perbaikan/materi tambahan atau mungkin kesalahan yang Anda temukan di buku ini.

Daftar Isi

Daftar Gambar	vi
Daftar Tabel	vii
Daftar Istilah	viii
1 Pengenalan R dan Rstudio	1
1.1 Tata Letak RStudio	3
1.2 R dalam Gaya “Tanya Jawab”	4
1.3 Saatnya menggunakan skrip	8
1.4 Manajemen Data dan Projek di R	10
1.5 Paket-Paket R	11
1.6 Mengimpor Data ke dalam R	12
Appendices	
A Beberapa Rujukan untuk Belajar R	14
Referensi	15

Daftar Gambar

1.1	GUI R	2
1.2	Antar Muka RStudio	4
1.3	Antar Muka RStudio dengan Skrip	9

Daftar Tabel

Daftar Istilah

1-D, 2-D	_____
Otter	_____
Hedgehog	_____

1

Pengenalan R dan Rstudio

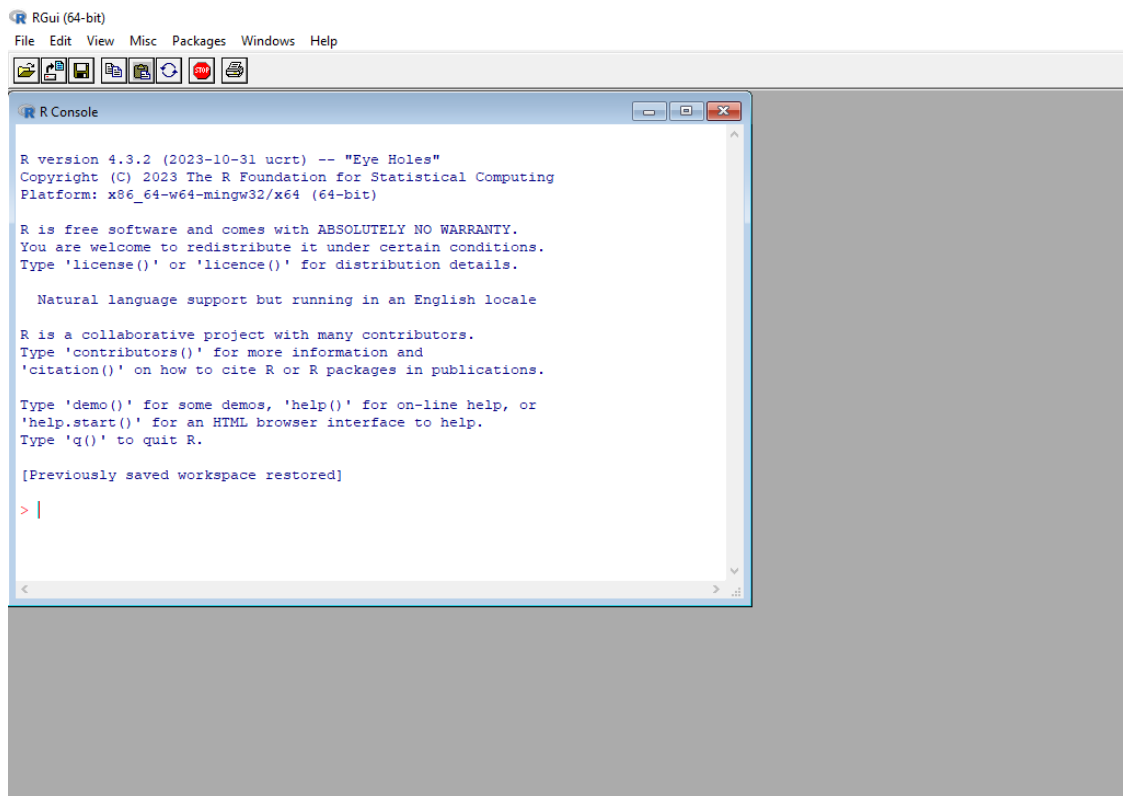
Contents

1.1	Tata Letak RStudio	3
1.2	R dalam Gaya “Tanya Jawab”	4
1.3	Saatnya menggunakan skrip	8
1.4	Manajemen Data dan Projek di R	10
1.5	Paket-Paket R	11
1.6	Mengimpor Data ke dalam R	12

Ada banyak situs rujukan bagus untuk belajar R. Tempat pertama menurut saya adalah *R Project for Statistical Computing* di <http://www.r-project.org/>. Situs ini berisi tautan ke buku, manual, *R Journal* dan lain-lain. Silakan unduh R di situs ini. Gratis! Sedangkan untuk menginstal RStudio silakan kunjungi [posit](#). Silakan unduh *RStudio Desktop* dari situs ini. Juga gratis! Untuk pengenalan ini, saya berasumsi bahwa Anda telah mengunduh dan menginstal R dan RStudio di komputer Anda. Kurva pembelajaran untuk R tidak terlalu curam. Dalam waktu singkat, Anda akan dapat menggunakan R secara aktif.

R tersedia secara gratis untuk berbagai platform komputasi seperti Windows, Macintosh dan Unix. Ini membuat R lebih fleksibel. Tetapi di sisi lain, ini berarti tidak ada pengembangan antarmuka pengguna grafis (GUI) yang sesuai dengan

keinginan pengguna seperti halnya program berbayar SPSS atau Stata. GUI bawaan R terlihat lebih mirip dengan konsol DOS lama. Lihat Gambar 1.1.



saja dari kemampuan R dan RStudio yang sangat banyak. Ada banyak rujukan bagus dan gratis untuk mempelajari dasar-dasarnya secara *online*. Untuk pengantar R dapat ditemukan di Crawley (2012) dan Wickham & Grolemund (2017), dan untuk pengenalan RStudio dapat dibaca Verzani (2011).

1.1 Tata Letak RStudio

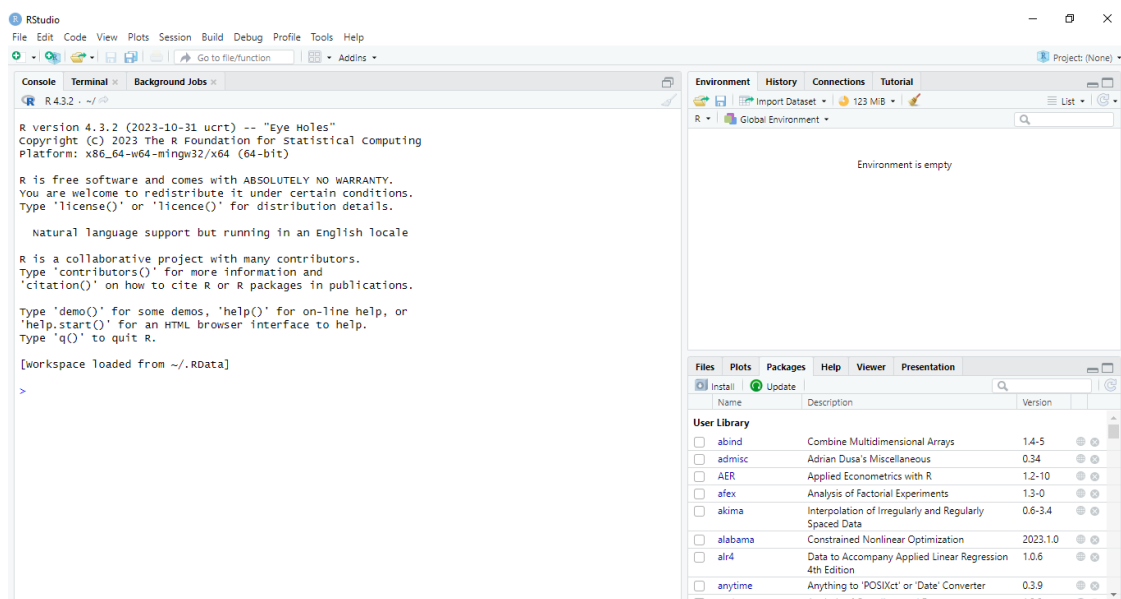
Sebagai permulaan, kita akan mempelajari tata letak RStudio dan elemen inti yang akan kita gunakan saat bekerja (nanti Anda dapat mengubah tata letak dan tampilan RStudio sesuai selera). Antarmuka RStudio dibagi menjadi beberapa kuadran (lihat Gambar 1.2).

Saat Anda memulai RStudio untuk pertama kalinya, Anda akan melihat tiga panel. Panel kiri menunjukkan konsol R. Kuadran ini juga dapat digunakan sebagai sumber masukan langsung untuk perintah. Kita belum membuka skrip, jadi kita tidak melihat panel skrip. Anda mempunyai dua pilihan disini: mengetik langsung di konsol (panel di sebelah kiri) atau membuat skrip.

Jika kita mengetikkan perintah langsung ke konsol, kita menggunakan R secara interaktif atau R dalam gaya “tanya jawab” (Dalgaard (2008)). Kelemahannya cara ini adalah kita lebih sulit untuk mengulangi atau menyimpan perintah-perintah kita. Semua fungsi akan dijalankan segera setelah kita menekan tombol **Enter**. Jika kita ingin menjalankan perintah cepat yang tidak perlu didokumentasikan dalam skrip, kita cukup menggunakan cara ini yaitu memasukkan perintah tersebut di konsol dan tekan Enter (Windows).

Di sebelah kanan, panel atas berisi lima tab: *Environment*, *History*, *Connections*, *Build* dan *Tutorial*. Di dalam Environment semua kumpulan data, variabel, model, objek, dan plot yang berbeda akan disimpan. Kuadran ini juga menunjukkan daftar berbagai objek yang ada dalam sesi R yang sedang berjalan. Jika kita menginput suatu dataset atau membuat variabel baru, semuanya akan muncul di kuadran ini.

Sedangkan panel kanan bawah menampilkan enam tab: *File*, *Plots*, *Packages*, *Help*, *Viewer* dan *Presentation*. Anda dapat mengklik setiap tab untuk menelusuri berbagai fitur di dalamnya. Tab *Plots* adalah tempat mencetak atau mengekspor plot dan grafik. Tab *Packages* menunjukkan berbagai paket yang terpasang di komputer Anda. Tab yang penting disini adalah *Help*: kita bisa bantuan jika kita lupa dengan fungsi-fungsi tertentu.



Gambar 1.2: Antar Muka RStudio

1.2 R dalam Gaya “Tanya Jawab”

Disini kita akan mempraktikkan R dalam gaya tanya jawab yaitu mengetik perintah langsung ke konsol. Kita menggunakan gaya ini ketika hanya penghitungan rumus tunggal yang diperlukan, seperti kalkulator. Ketika komputasi melibatkan beberapa langkah, kita akan beralih menggunakan skrip (dibagian selanjutnya).

Dalam buku ini, kita jarang mengetik langsung ke konsol, namun jika Anda merasa perlu mengulangi atau mengunjungi kembali perintah yang telah Anda gunakan di konsol, di panel kanan atas ada tab *History* berisi catatan lengkap dari setiap perintah yang telah Anda gunakan sebelumnya.

Sebagai ilustrasi, misalnya Anda ingin menghitung 2×2 , kita menulis setelah *command prompt* `> 2*2` dan tekan **Enter**. Selanjutnya kita dapat memberi nama dan menyimpan hasilnya (sebagai nama variabel), misalnya *opat*. Maka kita tuliskan `opat <- 2*2`. Artinya komputer menghitung 2×2 dan menyimpan hasilnya dalam variabel bernama *opat*. Perhatikan operator penugasan `<-` sebagai ciri khas R, walaupun kita bisa juga menggunakan tanda `=`. Contoh perintah/kode ditampilkan dalam kotak abu-abu, sedangkan luarannya diawali dengan tanda pagar ganda `##`. Dengan cara yang sama kita buat variabel *onom* `onom <- 2*3`.

```
2*2
```

```
## [1] 4
```

```
opat <- 2*2  
opat # memanggil variabel opat
```

```
## [1] 4
```

```
onom = 3*2  
onom
```

```
## [1] 6
```

Hal ini juga berlaku untuk operasi aritmatika (+, -, /, ^,...), operator perbandingan (==, <=,...), operator logika (&, |, !,...) dan fungsi matematika dasar seperti sin, cos, exp, logaritma (log), akar kuadrat (sqrt).

Variabel paling dasar dalam R adalah vektor. Vektor R adalah barisan nilai yang bertipe sama. Misalnya, untuk membuat vektor lima dimensi beranggotakan angka 7, kita menuliskan `tujuh5 <- rep(7,5)`. Maka *tujuh5* adalah vektor dengan masing-masing komponen 7; untuk melihatnya kita ketik *tujuh5* dan tekan Enter. Fungsi *rep* merupakan fungsi khusus dari R yang merupakan kependekan dari pengulangan (repeat). Argumennya ditulis sebagai *rep(apa yang diulang, berapa kali)*.

```
tujuh5 <- rep(7,5)
tujuh5
```

```
## [1] 7 7 7 7 7
```

Jika a dan b bilangan bulat, perintah a:b membuat vektor bilangan bulat dari a ke b.

```
vektor17 <- 1:7
vektor17
```

```
## [1] 1 2 3 4 5 6 7
```

Untuk lebih bisa mengontrol vektor yang dibuat, Anda gunakan perintah sekuen *seq*.

```
sek305 <- seq(1,30,5) # membuat vektor bilangan dari 1-30 dengan
↳ beda 5
sek305
```

```
## [1] 1 6 11 16 21 26
```

```
sek301 <- seq(30,1,-5) # membuat vektor bilangan dari 30-1 dengan
↳ beda -5
sek301
```

```
## [1] 30 25 20 15 10 5
```

Elemen vektor dapat diindeks dengan tanda kurung []. Argumen braket dapat berupa bilangan bulat tunggal atau vektor.

```
sek305[4]
```

```
## [1] 16
```

```
sek305[1:4]
```

```
## [1] 1 6 11 16
```

```
sek305[c(1,4,5,10)]
```

```
## [1] 1 16 21 NA
```

Untuk perintah terakhir saya meminta elemen vektor `sek305` yang pertama, keempat, kelima, dan kesepuluh. Karena tidak ada elemen kesepuluh di dalam vektor `sek305`, maka hasilnya `NA` (*not available*). Seringkali kita juga ingin mencari elemen suatu vektor yang memenuhi suatu kondisi tertentu, misalnya elemen `sek305` yang kurang dari 10. Hal ini dilakukan dengan meminta indeks yang memenuhi

```
sek305

## [1] 1 6 11 16 21 26

which(sek305 < 10) # elemen beberapa yang kurang dari 10

## [1] 1 2

index <- which(sek305 < 10) # bilangan berapa saja yang kurang dari
↪ 10 dari vektor sek305
sek305[index]

## [1] 1 6
```

Vektor dapat terdiri dari angka, karakter dan bahkan rangkaian karakter.

```
omahku <- c("Malang", "memang", "kota", "yang", "sueejuk") #
↪ perhatikan penggunaan ""
omahku

## [1] "Malang" "memang" "kota" "yang" "sueejuk"

omahku[c(3,4,5)]

## [1] "kota" "yang" "sueejuk"
```

Untuk membuat matriks juga dapat dilakukan dengan mudah. Misalnya, `mat.pi.10.5 <- matrix(pi, nrow=10, ncol=5)` akan membuat matriks 10×5 dengan nama `mat.pi.10.5` dengan semua entri π . Sekali lagi, untuk melihat hasilnya, ketik `mat.pi.10.5` dan tekan Enter. R peka penulisan huruf besar-kecil. Jadi `mat.pi.10.5` dan `mat.Pi.10.5` memiliki arti yang berbeda. Semua perintah menggunakan huruf kecil, kecuali perintah `View()`.

```
mat.pi.10.5 <-matrix(pi, nrow=10,ncol=5)
mat.pi.10.5
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [2,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [3,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [4,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [5,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [6,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [7,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [8,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [9,] 3.141593 3.141593 3.141593 3.141593 3.141593
## [10,] 3.141593 3.141593 3.141593 3.141593 3.141593
```

1.3 Saatnya menggunakan skrip

Biasanya penghitungan statistik melibatkan banyak baris kode. Jika demikian kita beralih menggunakan skrip. Untuk memulai skrip baru, Anda dapat mengklik File, New File, lalu R Script atau **Ctrl+Shift+N**. Maka dilayar RStudio akan muncul layar baru di kiri atas sehingga Anda sekarang punya empat kuadran. Inilah salah satu keunggulan R dibandingkan perangkat lunak analisis *point and click* yaitu Anda dapat menyimpan pekerjaan Anda sebagai skrip. Lihat Gambar

Ini juga merupakan kebiasaan baik untuk mendeskripsikan proyek dan menulis komentar singkat di badan fungsi yang kita buat. Sekali lagi ini berguna bagi penulis skrip dan bagi orang ketiga yang akan membaca kode tersebut.

1.4 Manajemen Data dan Proyek di R

R adalah bahasa pemrograman berorientasi objek. Saat Anda membukanya, Anda memiliki lingkungan (*Environment*) kosong yang dapat diisi dengan objek sebanyak kemampuan memori komputer Anda. Segala sesuatu yang ingin Anda simpan atau manipulasi di lain waktu perlu didefinisikan sebagai objek di lingkungan ini. Termasuk disini file data, objek atau hasil model, grafik, dan sebagainya. Artinya, tidak seperti perangkat lunak statistik standar, yang biasanya hanya mengizinkan analis untuk membuka satu kumpulan data, R memungkinkan analis untuk bekerja dengan beberapa file data secara bersamaan.

Kita bisa mendapatkan direktori kerja saat ini menggunakan fungsi `getwd()`. Direktori kerja adalah cara kita memberi tahu R di mana mendapatkan file, dan juga di mana mengeksport file. Dengan kata lain, di sinilah R akan beroperasi di dalam komputer kita. Kita dapat mengatur direktori kerja kita dengan menentukan path file ke lokasi yang kita inginkan dengan perintah `setwd`, atau Anda dapat mengatur jalur menggunakan menu drop-down (Session -> Set Working Directory-> Choose Directory).

```
## Working directory di atur ke desktop
setwd("/Users/Username/Desktop")
## untuk memastikan working directory yang dipakai
getwd()
[1] "/Users/Username/Desktop"
```

Sering kali kita memiliki banyak file data sekaligus yang dimuat di lingkungan kita. Jika demikian manajemen data menjadi penting dalam konteks ini. Saya sangat menyarankan Anda membuat *project* setiap kali Anda memulai pekerjaan yang berbeda, yang tidak terkait dengan pekerjaan sebelumnya. Proyek

disini adalah tempat untuk menyimpan pekerjaan Anda yang berkaitan dengan proyek spesifik.

Untuk membuat proyek klik simbol R di pojok kanan atas, klik *New Directory > New Project* lalu tulis nama direktori (misalnya *rstudio4eb*) dan klik *Create project*. Ketika Anda sudah berada didalam proyek folder, sebaiknya dibuat subfolder tambahan misalnya khusus untuk data, grafik, referensi dan lain-lain. Setelah folder ditentukan, aturlah R ke direktori proyek Anda, sehingga folder itu menjadi lokasi default tempat R mencari file.

Ekstensi file R untuk file data adalah *.RData*. Anda dapat menyimpan satu atau lebih objek dalam file tersebut menggunakan fungsi simpan (*Ctrl + s*). Di dalam *RData* tersimpan hasil model, grafik, atau objek apa pun. Hal ini sangat berguna ketika model membutuhkan waktu lama untuk dalam penghitungannya. Hasil tersebut dapat disimpan dan digunakan kembali nanti.

Selain file *RData*, menyimpan file dalam format *.csv* juga efektif saat bekerja di R. Ada perintah dasar R untuk mengimpor dan mengekspor file *csv* (masing-masing *read.csv* dan *write.csv*). R dapat langsung mengimpor data dari paket perangkat lunak statistik lain (*SPSS*, *Stata*), namun hal ini memerlukan pengunduhan paket (*package*) yang tidak disertakan dalam instalasi bawaan R. Kita membahas paket di bagian selanjutnya.

1.5 Paket-Paket R

R bekerja melalui paket termasuk paket dasar dan ribuan paket atau ekstensi. Dengan paket dasar dan ekstensinya, banyak analisis data statistik dapat dilakukan dan grafik statistik kualitas tinggi dihasilkan.

Karena sifatnya yang *open source*, pengguna dapat mengembangkan dan memperluas fitur-fitur dasar R dengan penambahan fungsionalitas. Ketika seseorang ingin menambahkan fungsi tambahan ke R, mereka membuat sebuah paket: fitur tambahan yang dapat diinstal pada mesin pengguna dan kemudian

dimuat ke R bila diperlukan. Paket juga dapat menyimpan fungsi dan kumpulan data R.

Anda perlu mengunduh paket tertentu untuk menyediakan perintah yang diperlukan dan dapat menjalankan model atau analisis tertentu jika fungsi itu tidak tersedia secara *default*. Paket-paket itu memerlukan tiga langkah untuk menggunakannya. Pertama, kita harus menginstal paket itu sendiri, memuatnya dari *library*, dan terakhir memanggil salah satu fungsi paket tersebut.

Paket dapat diunduh di CRAN (*Comprehensive R Archive Network*) dan dapat diinstal melalui kode `install.packages("nama paket")` atau melalui opsi drop-down (*Tools -> Install Package*). Paket di CRAN telah dievaluasi untuk memastikan paket tersebut berfungsi di seluruh platform. Paket bisa juga tersedia bagi pengguna tetapi belum diarsipkan di CRAN tetapi masih berada di Github. Paket tersebut dapat diinstal menggunakan fungsi `devtools::install_github` (Wickham et al. (2022)). Untuk setiap sesi R yang baru, paket yang diperlukan perlu dimuat (bukan diinstal ulang, cukup dimuat ulang dengan fungsi `library()` atau `require()`).

1.6 Mengimpor Data ke dalam R

Dalam pelaksanaan analisis data, data sering kali disimpan dalam berbagai format. R dapat membaca berbagai jenis file data seperti free format text files, comma separated value files (csv), file Excel, file SPSS, file SAS, dan file Stata. Mengimpornya ke dalam R juga mudah. Kita dapat mengimpor .csv file menggunakan basis R. Tetapi jika data dalam format lain, kita perlu menginstal paket untuk mengimpor data misalnya paket *foreign* (Team et al. (2023)) yang berfungsi untuk membaca dan menulis file data dari SAS, SPSS dan Stata. Saat memasukkan nama file, jangan lupa untuk menempatkan nama file diantara tanda kutip ("nama-file.csv").

Appendices



Beberapa Rujukan untuk Belajar R

Referensi

- Crawley, M. J. (2012). *The R Book* (2nd edition). Wiley.
- Dalgaard, P. (2008). *Introductory Statistics with R*. Springer. <https://doi.org/10.1007/978-0-387-79054-1>
- Ismay, C., & Kim, A. Y. (2019). *Statistical Inference via Data Science: A Modern Dive into R and the Tidyverse*. CRC Press.
- Team, R. C., Bivand, R., Carey, V. J., DebRoy, S., Eglen, S., Guha, R., Herbrandt, S., Lewin-Koh, N., Myatt, M., Nelson, M., Pfaff, B., Quistorff, B., Warmerdam, F., Weigand, S., Foundation, F. S., & Inc. (2023). *Foreign: Read Data Stored by 'Minitab', 'S', 'SAS', 'SPSS', 'Stata', 'Systat', 'Weka', 'dBase', ...*
- Verzani, J. (2011). *Getting Started with RStudio*. O'Reilly.
- Wickham, H., & Grolemund, G. (2017). *R for Data Science*. O'Reilly Media.
- Wickham, H., Hester, J., Chang, W., & Byan, J. (2022). *Devtools: Tools to Make Developing R Packages Easier*. CRAN.