

SA-TIED Geospatial Analysis Workshop

Overview



Dr Justin van Dijk



j.t.vandijk@ucl.ac.uk



Objectives

We will explore the following topics:

- Fundamentals of using R for data analysis .
- Creating thematic maps using R.
- Quantifying the degree of spatial dependence in a dataset.
- Incorporating space into statistical models.

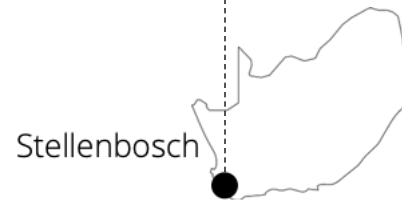
Schedule

Day 1 – Morning	R for Data Analysis
Day 1 – Afternoon	R for Spatial Analysis
Day 2 – Morning	Spatial Autocorrelation
Day 2 – Afternoon	Spatial Models

Background



B.Sc. Human Geography and Planning
M.Sc.. Human Geography and Planning



Ph.D. Transport Economics



Lecturer in Social and Geographic
Data Science

SA-TIED Geospatial Analysis Workshop

S01 – R for Data Analysis



This session

- What is a programming language?
- A gentle introduction to working with R.
- Why use R for data analysis?

Programming languages

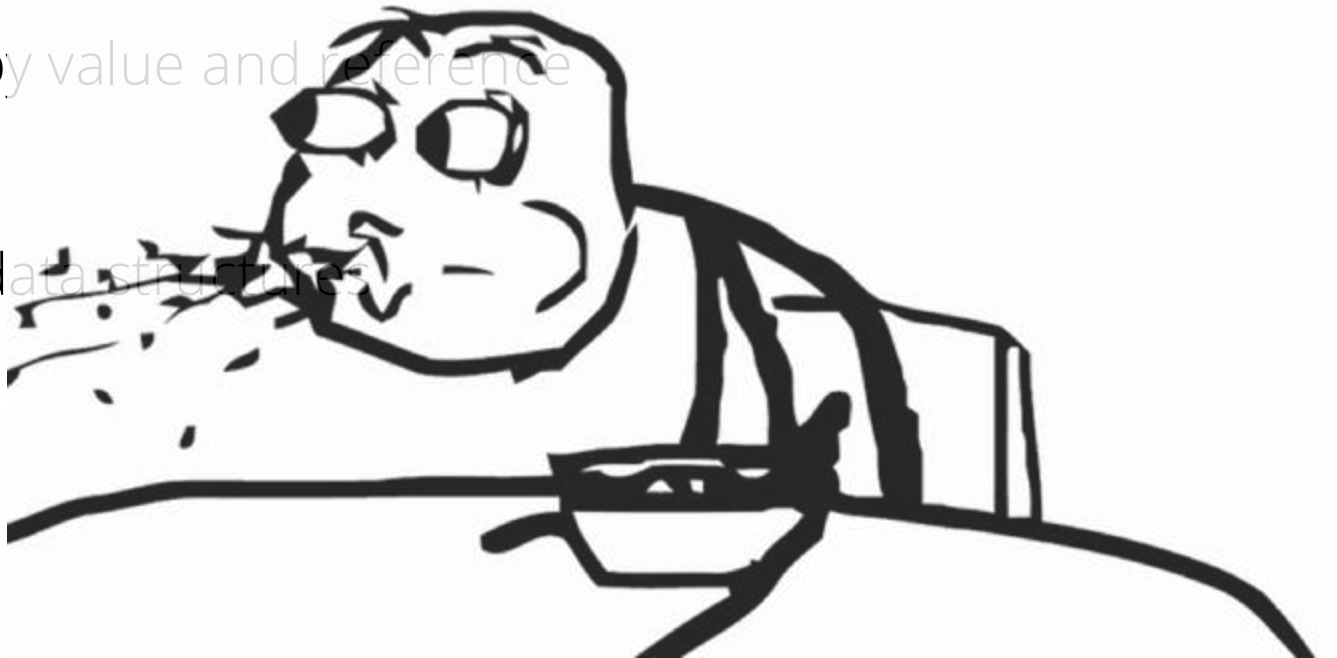
- Consist of a formal set of instructions that you can use to write software or perform computational tasks.
- Require users to write code, which involves typing commands in a text-based environment.
- Are highly flexible and powerful, allowing for custom solutions, automation, and complex operations.

Programming languages

- Identifiers and primitive data types
- Assignment, arithmetic, logical and relational operators
- Expression and statements, debugging
- Flow of control: selection and repetition
- Functions, parameters passing, call by value and reference
- Object-oriented programming
- 1/2 dimensional arrays, strings and data structures

Programming languages

- Identifiers and primitive data types
- Assignment, arithmetic, logical and relational operators
- Expression and statements, debugging
- Flow of control: selection and repetition
- Functions, parameters passing, call by value and reference
- Object-oriented programming
- 1/2 dimensional arrays, strings and data structures



Programming languages

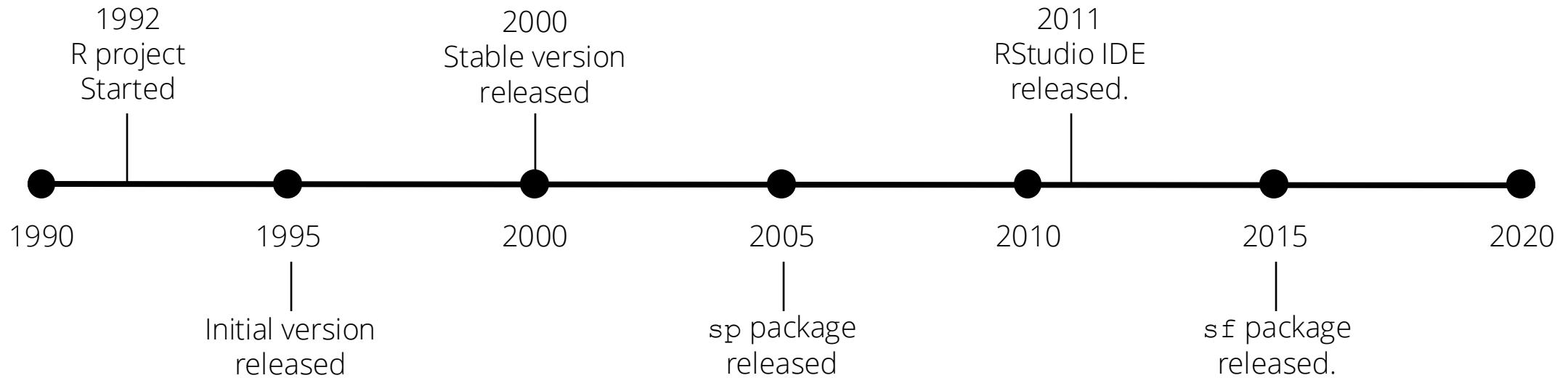
- Ιδεντιφιερες ανδ πριμιτιφε data τυπες
- Ασσιγνμεντ, αριτημετιξ, λογιξαλ ανδ ρελατιοναλ οπερατορς
- Εχπρεσσιον ανδ στατεμεντς, δεβυγγινγ
- Φλω οφ ξοντρολ· σελεξιτιον ανδ ρεπετιτιον
- Φυνξιτιονς, παραμετερς πασσινγ, ξαλλ βψ αφλυε ανδ ρεφερενξε
- Οβσεξτ-οριεντεδ προγραμμινγ
- 1/2 διμενσιοναλ αρραψς, στρινγς ανδ data στρυξτυρες

The absolute basics

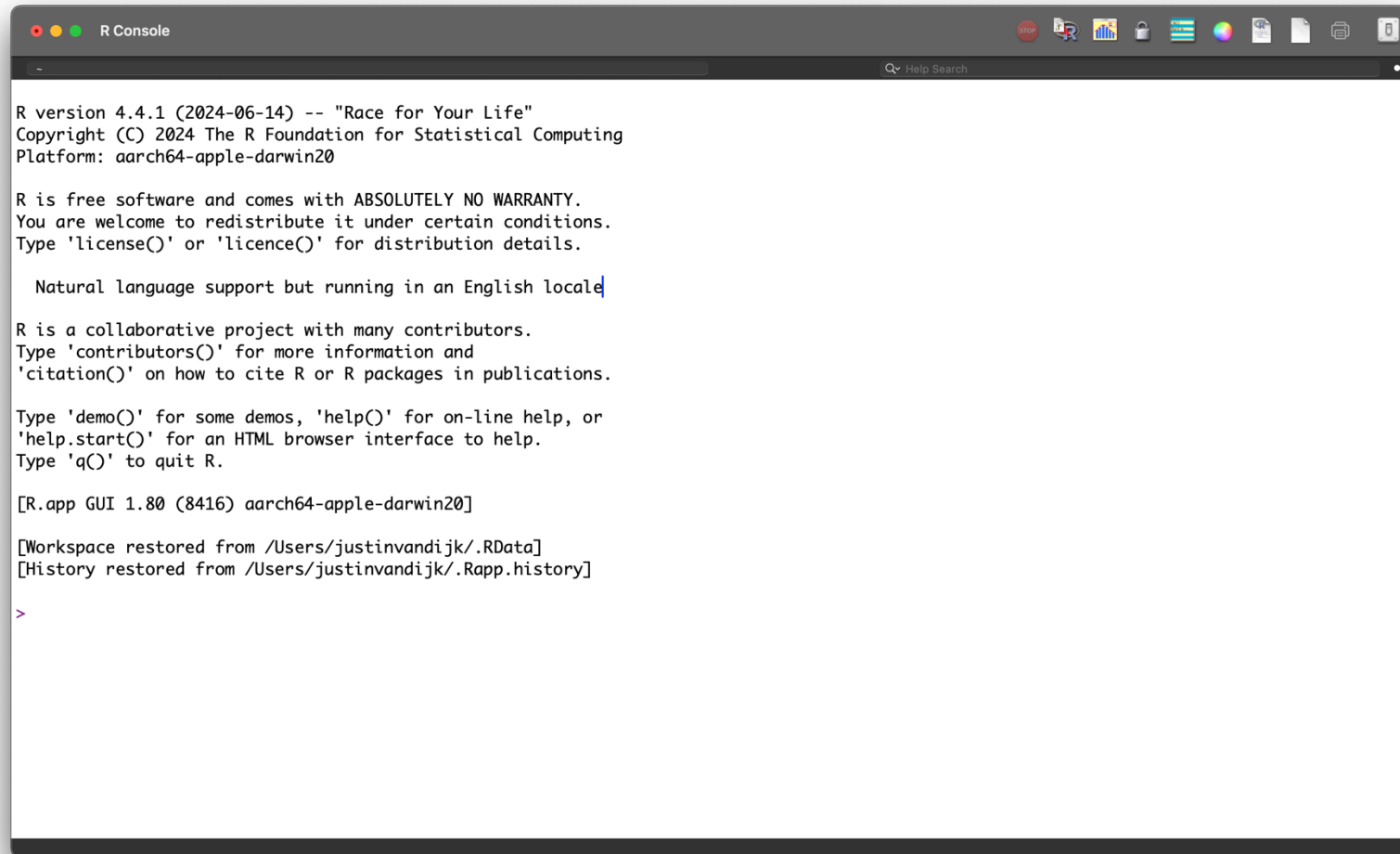
- R is primarily used through interactive command-line interfaces and scripts.
- R efficiently handles various data types, particularly vectors and tables.
- R's functionality can be extended through a vast ecosystem of packages.
- R is the programming language, but we can interact with it using other software.
- R is free and open source.

A little history

- R is programming language originally developed for statistical purposes.
- 1975: Bell Labs develops a language for Statistical Analysis ("S").
- 1992: Ross Ihaka and Robert Gentleman develop opensource version of "S".



Working with R



```
R Console

R version 4.4.1 (2024-06-14) -- "Race for Your Life"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

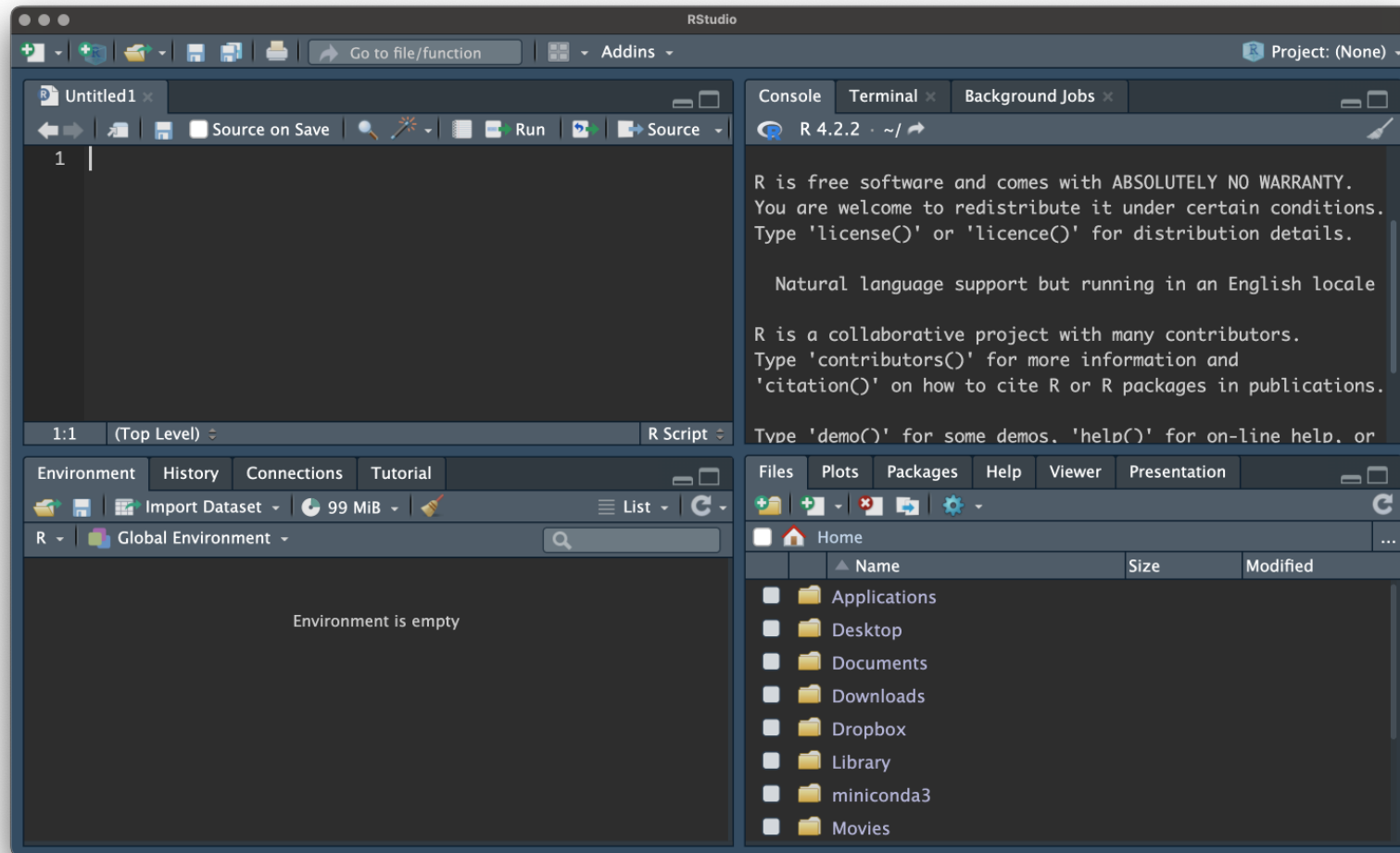
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.80 (8416) aarch64-apple-darwin20]

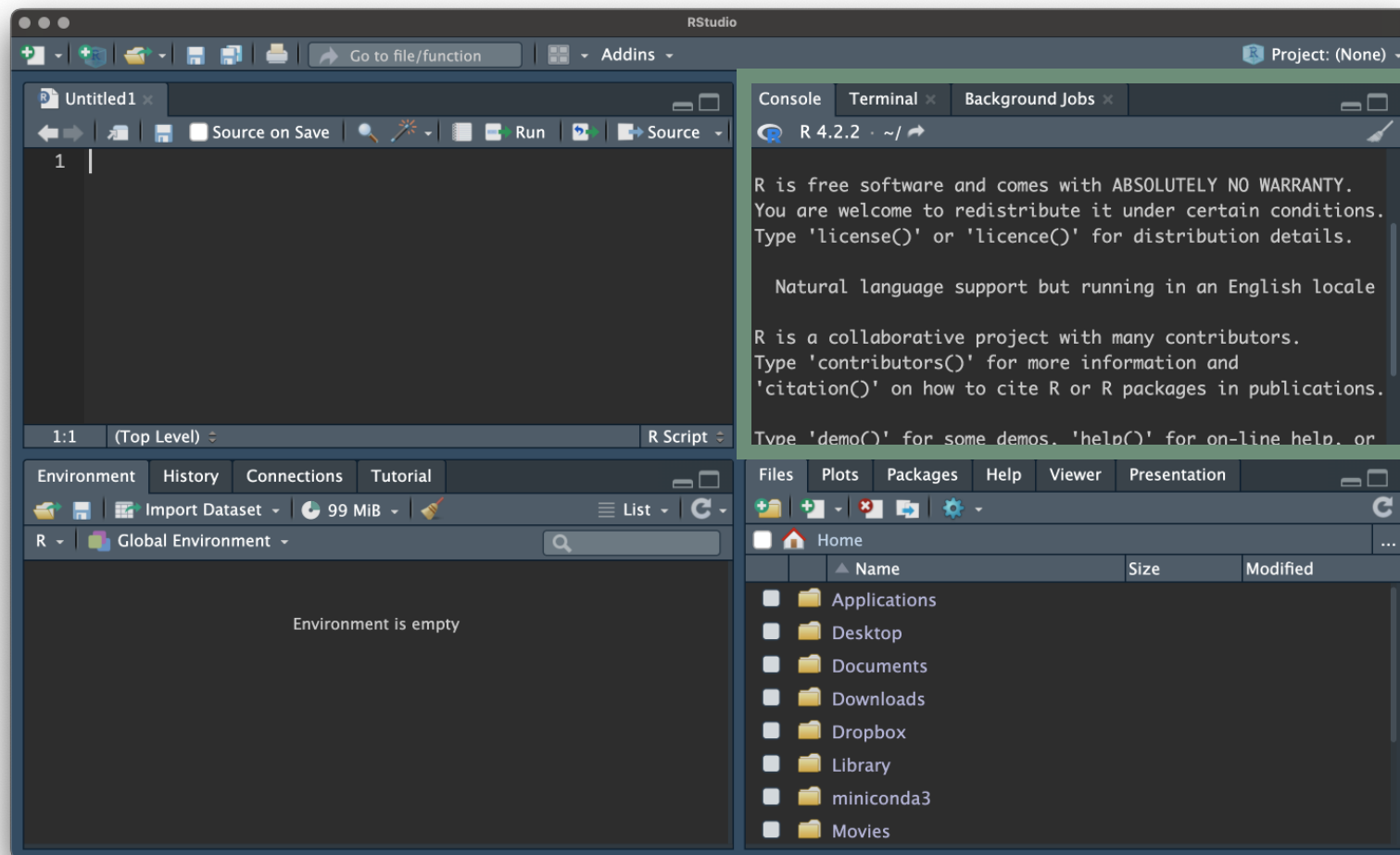
[Workspace restored from /Users/justinvandijk/.RData]
[History restored from /Users/justinvandijk/.Rapp.history]

>
```

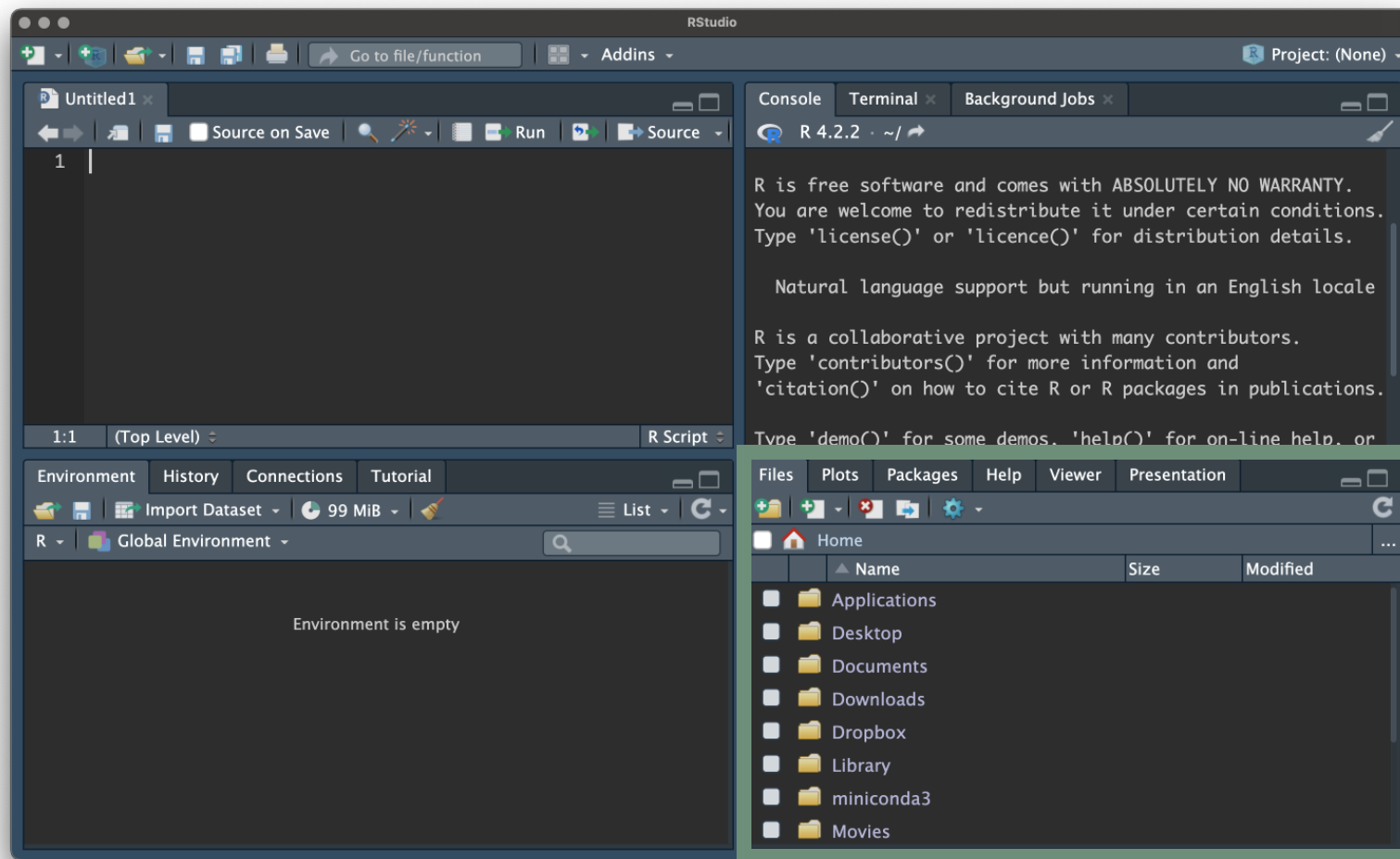

Working with R



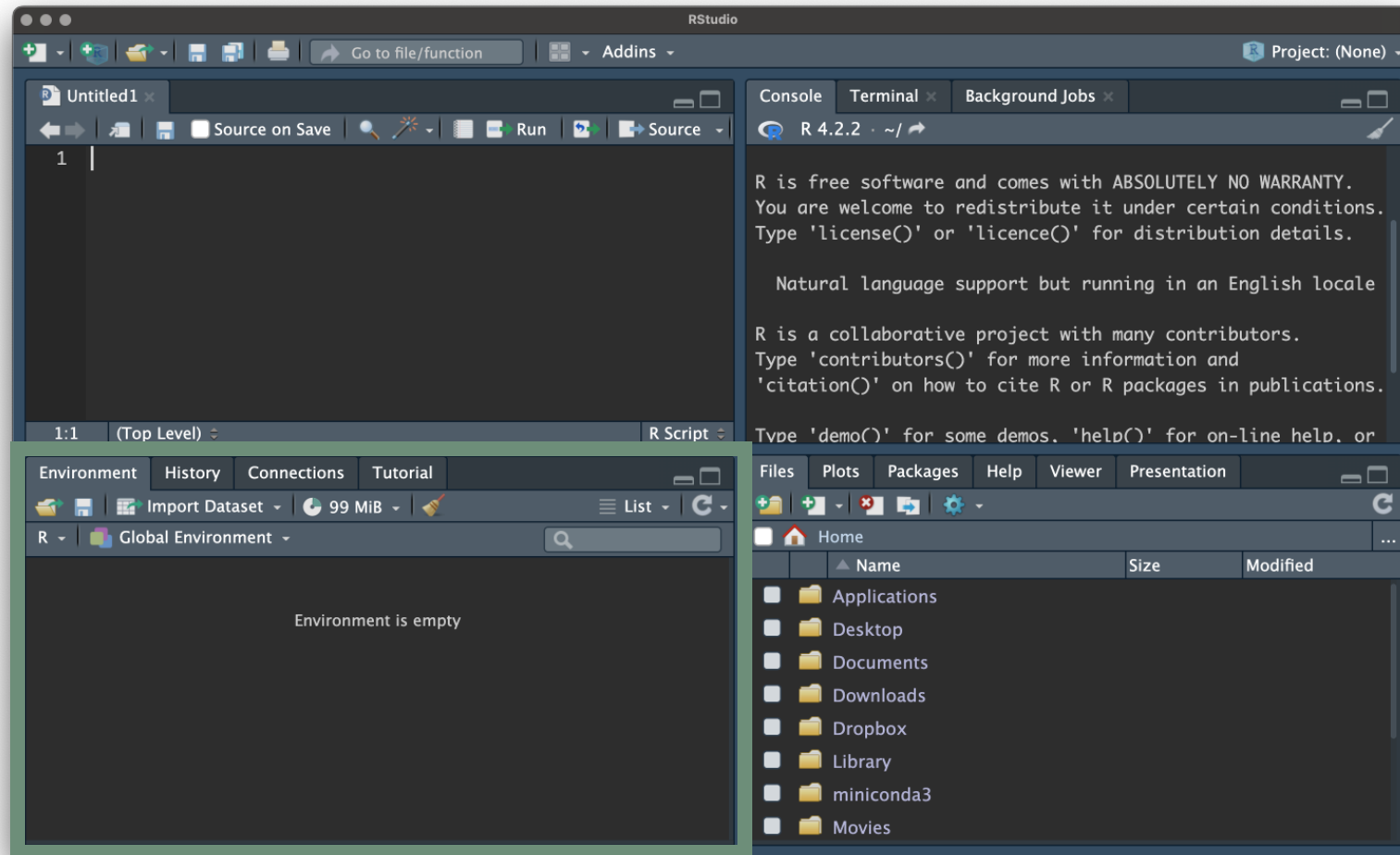
Working with R



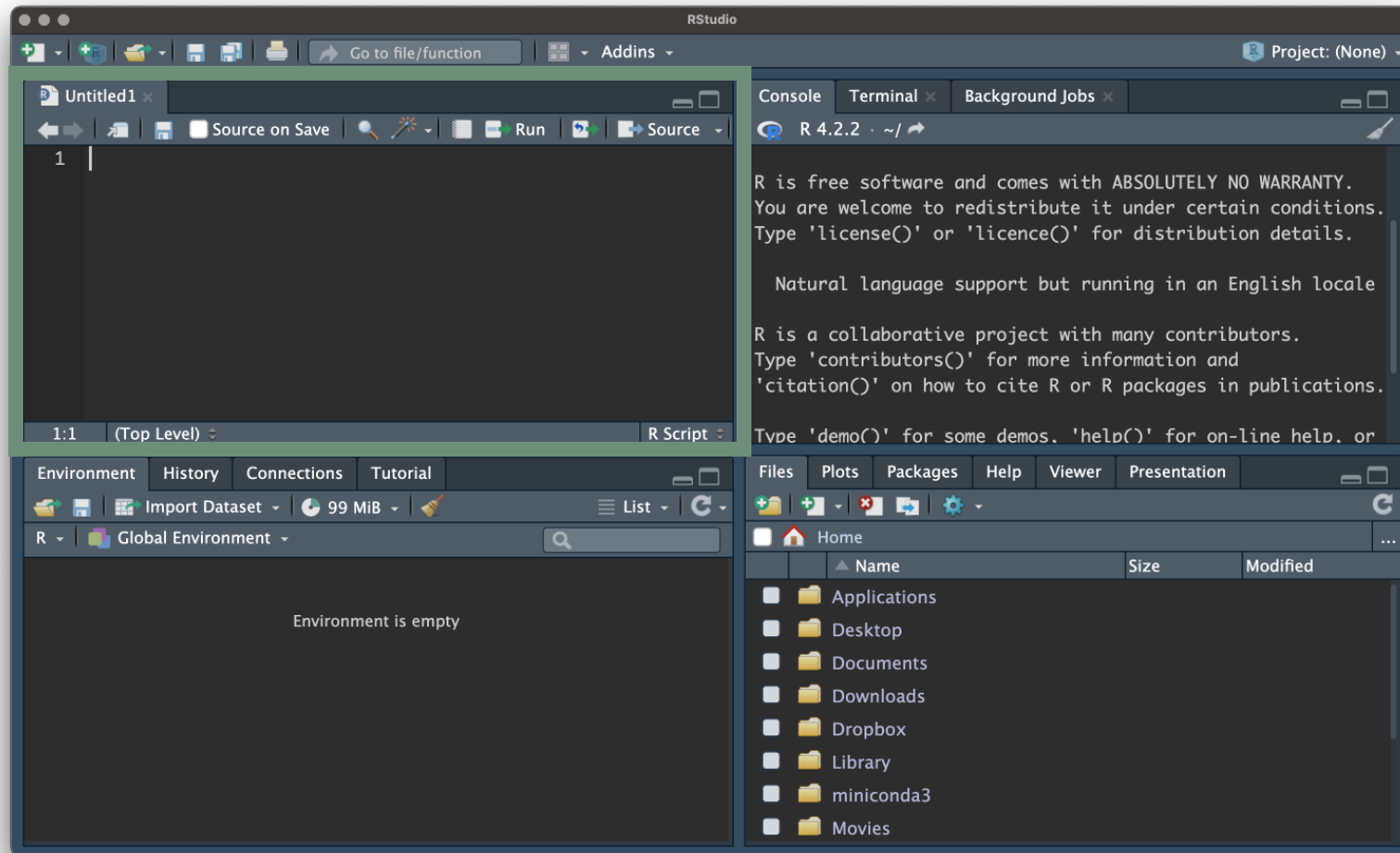
Working with R



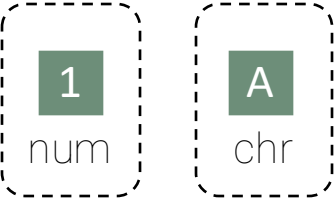
Working with R



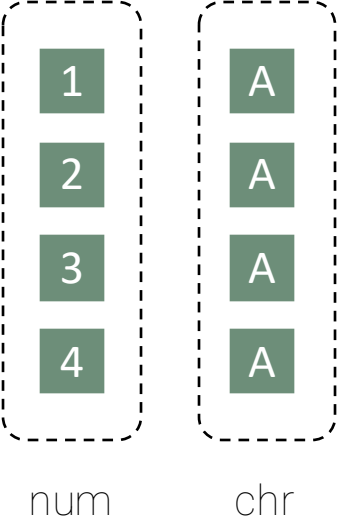
Working with R



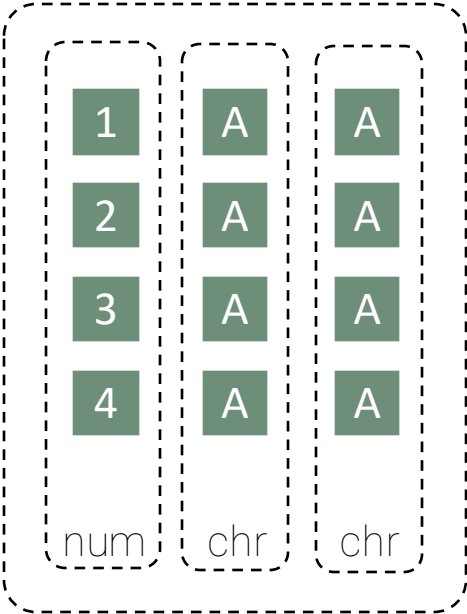
Principles of R: Data types



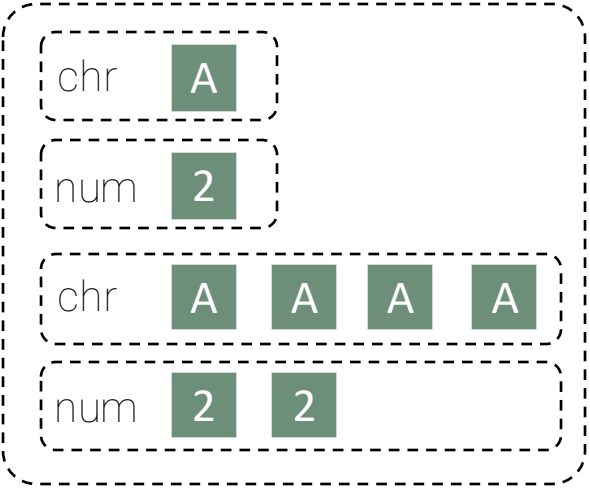
Scalar



Vector



Dataframe



List

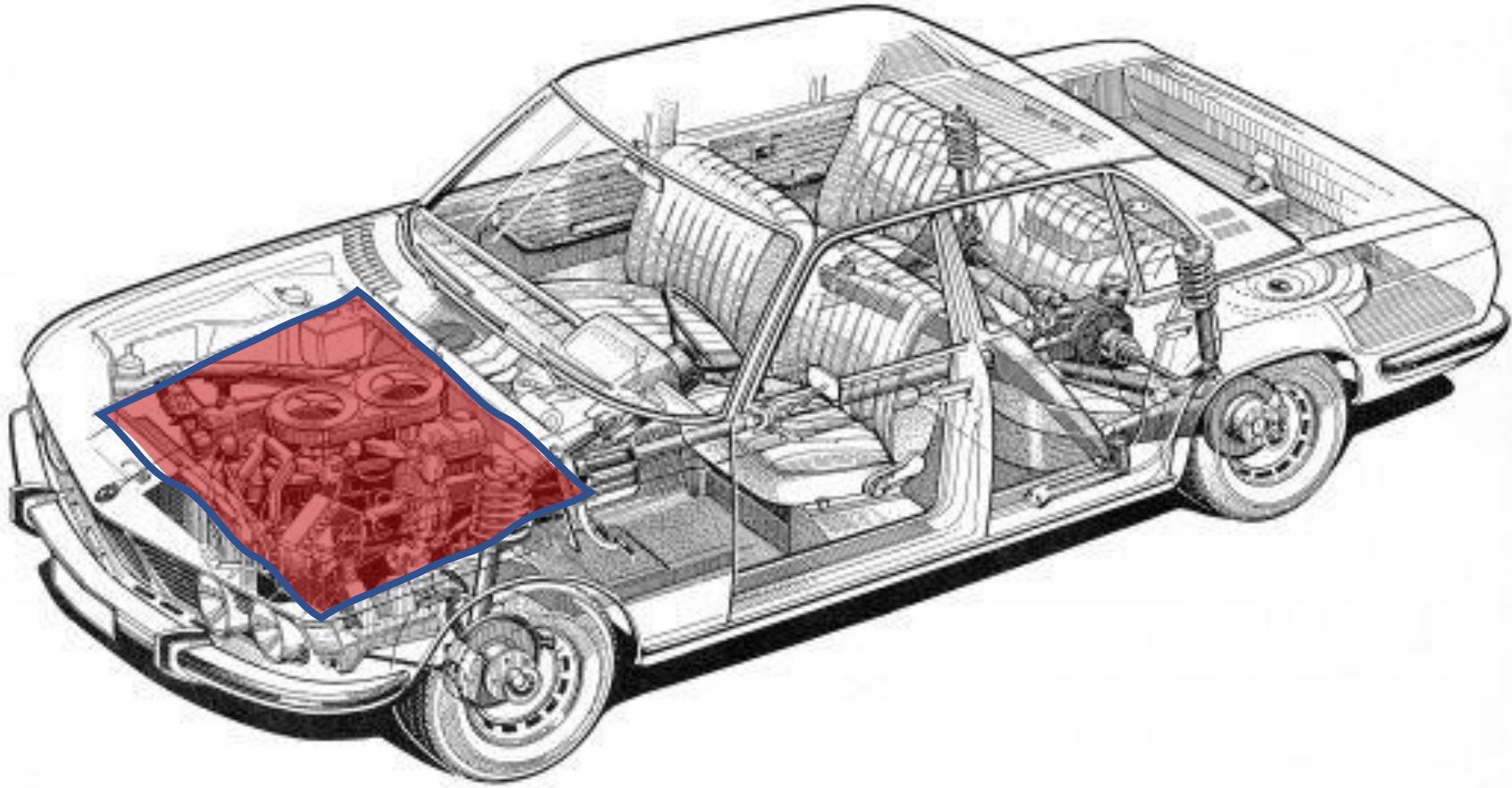
Principles of R: Variables

- All scalars, vectors, tables, and lists can be assigned to a variable.
- Variables are used to store information to be referenced and manipulated in a computer programme.

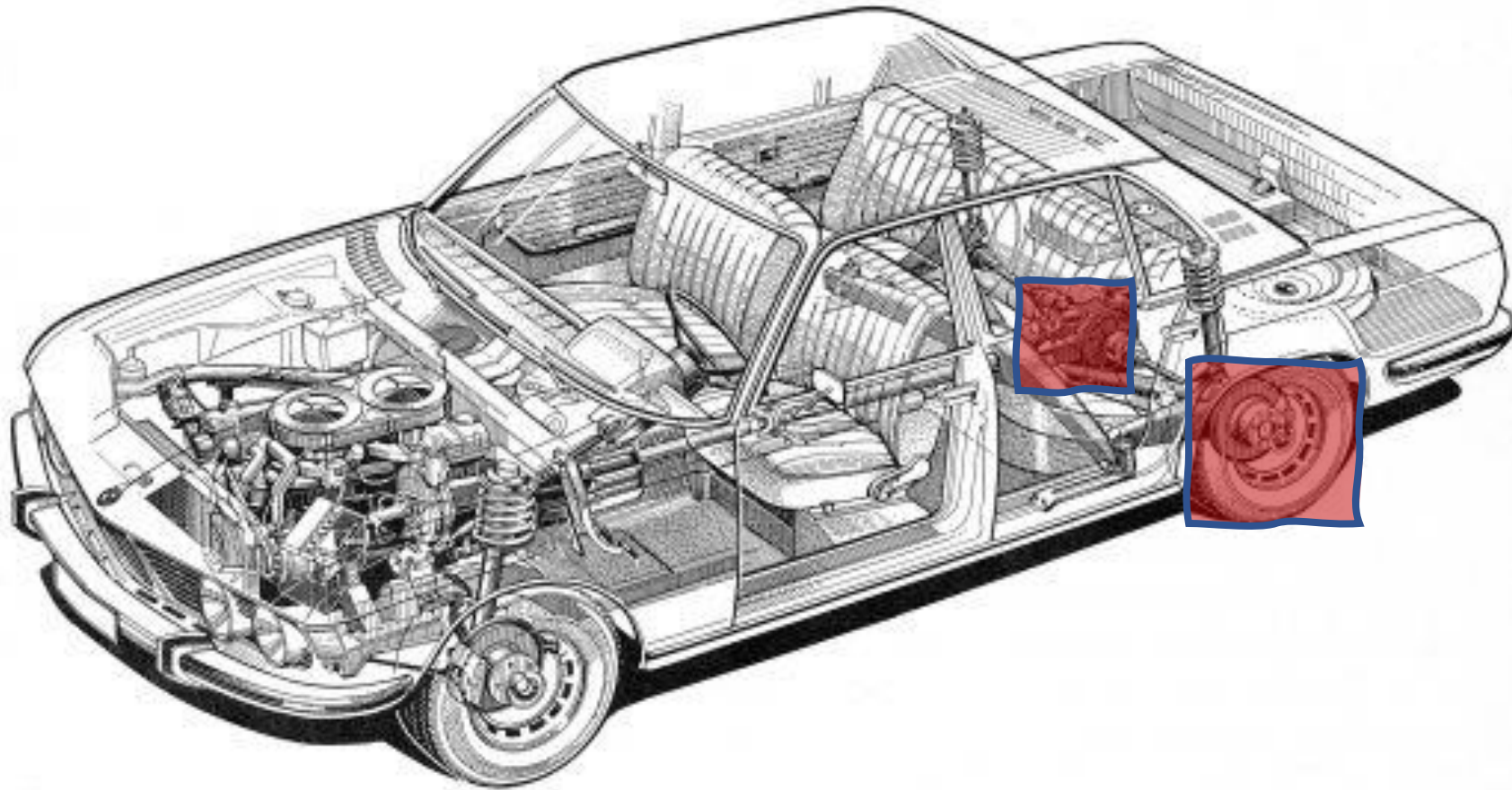
Principles of R: Functions

- Variables can be used as an input for functions.
- Functions are pieces of code designed to accomplish specific tasks.
- Once a function is written, it can be reused.

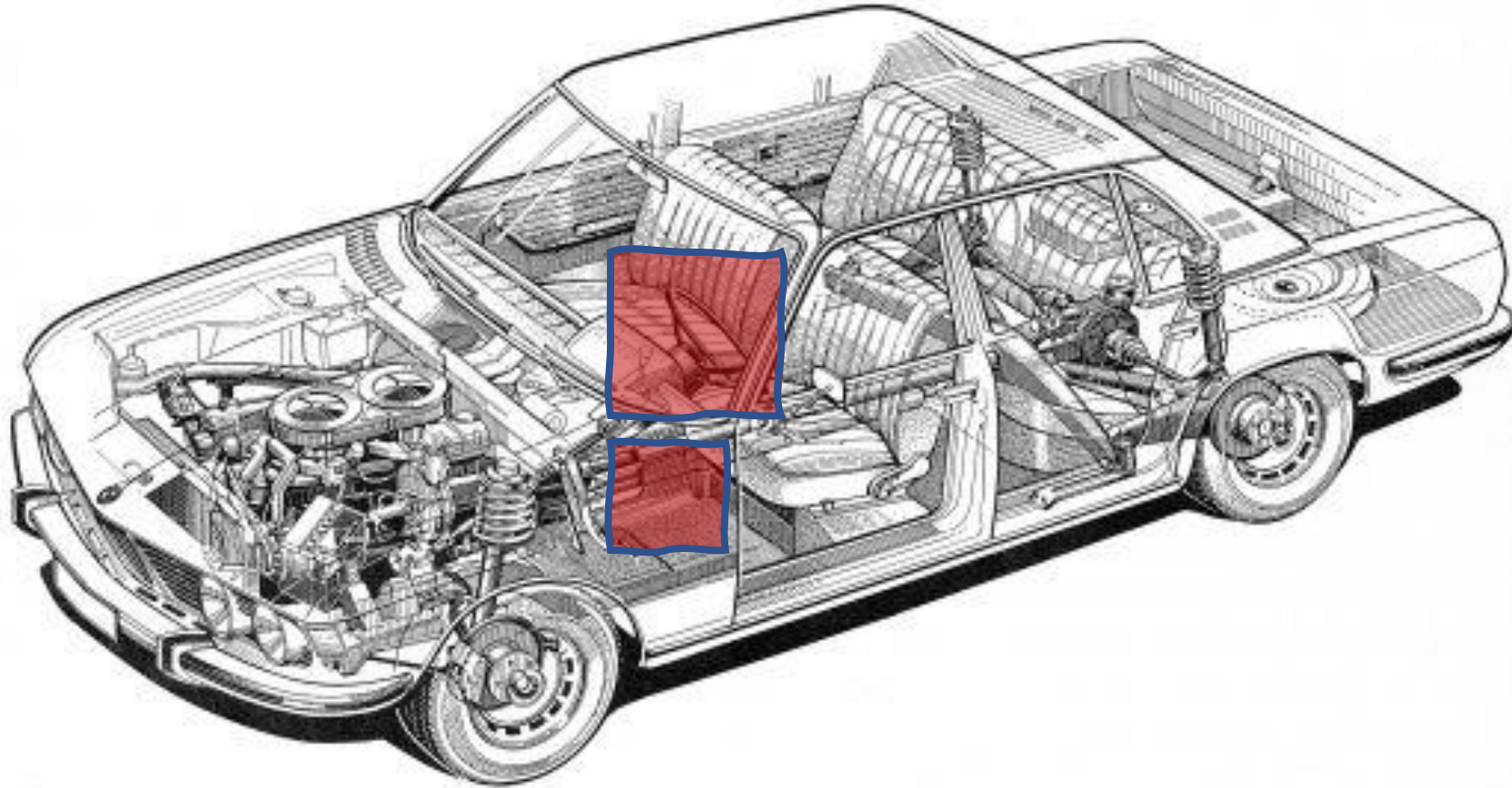
Principles of R: Extensions



Principles of R: Extensions



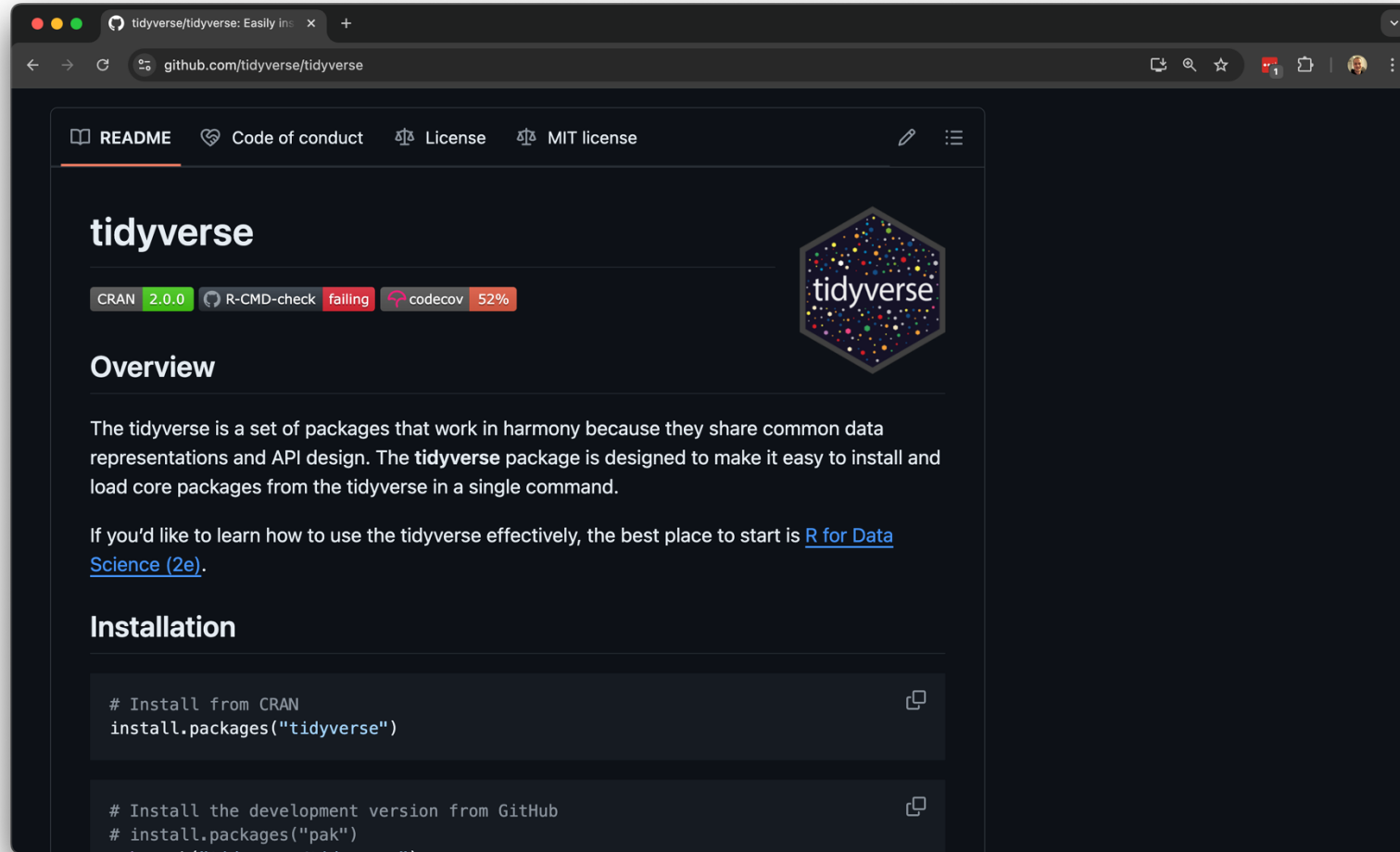
Principles of R: Extensions



Principles of R: Extensions

- Available on The Comprehensive R Archive Network: CRAN
- CRAN is a repository for R packages and software.
- Currently CRAN features over 21,140 available packages (22/08/2024).
- Allows developers to submit and maintain their own R packages.

Tidyverse



The screenshot shows the GitHub repository page for tidyverse/tidyverse. The browser address bar displays 'github.com/tidyverse/tidyverse'. The repository page includes a navigation bar with links to README, Code of conduct, License, and MIT license. The main content area features the 'tidyverse' logo, a status bar showing 'CRAN 2.0.0', 'R-CMD-check failing', and 'codecov 52%'. Below this is an 'Overview' section with a paragraph describing the tidyverse as a set of packages sharing common data representations and API design. It also includes a link to 'R for Data Science (2e)'. The 'Installation' section follows, providing two methods: installing from CRAN and installing the development version from GitHub. Each method is accompanied by a code block with R commands and a copy icon.

tidyverse/tidyverse: Easily install and load the tidyverse

github.com/tidyverse/tidyverse

README Code of conduct License MIT license

tidyverse

CRAN 2.0.0 R-CMD-check failing codecov 52%

Overview

The tidyverse is a set of packages that work in harmony because they share common data representations and API design. The **tidyverse** package is designed to make it easy to install and load core packages from the tidyverse in a single command.

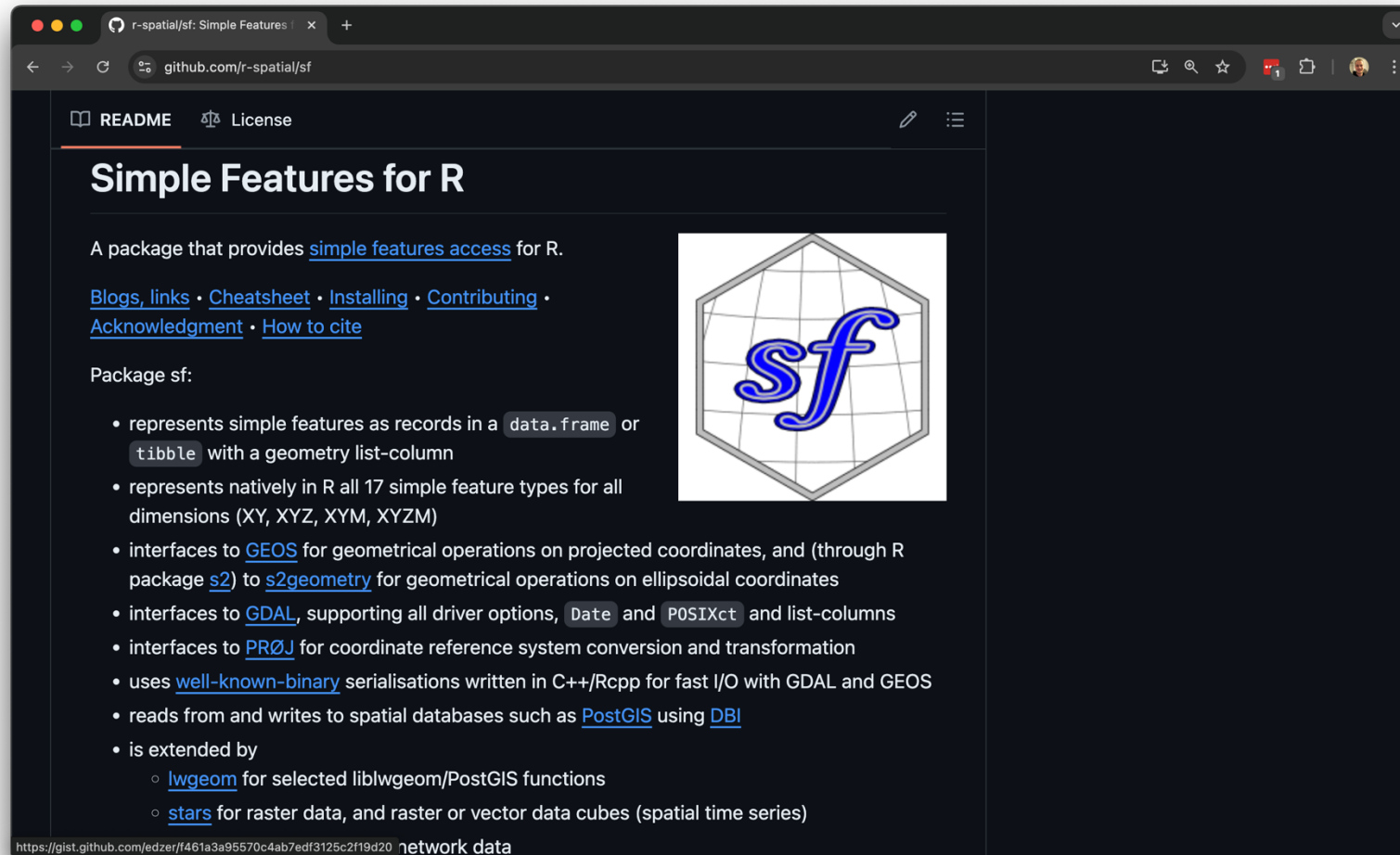
If you'd like to learn how to use the tidyverse effectively, the best place to start is [R for Data Science \(2e\)](#).

Installation

```
# Install from CRAN
install.packages("tidyverse")
```

```
# Install the development version from GitHub
# install.packages("pak")
# pak::install_tidyverse()
```

Simple features



The screenshot shows a web browser displaying the GitHub repository for 'r-spatial/sf: Simple Features for R'. The page has a dark theme. At the top, there are tabs for 'README' and 'License'. The main heading is 'Simple Features for R'. Below it, a paragraph states: 'A package that provides [simple features access](#) for R.' To the right of this text is a logo consisting of a blue 'sf' inside a wireframe cube. Below the paragraph is a list of links: '[Blogs, links](#) • [Cheatsheet](#) • [Installing](#) • [Contributing](#) • [Acknowledgment](#) • [How to cite](#)'. Further down, the text 'Package sf:' is followed by a bulleted list of features. At the bottom of the page, a URL is visible: 'https://gist.github.com/edzer/f461a3a95570c4ab7edf3125c2f19d20' followed by the text 'network data'.

README License

Simple Features for R

A package that provides [simple features access](#) for R.

[Blogs, links](#) • [Cheatsheet](#) • [Installing](#) • [Contributing](#) • [Acknowledgment](#) • [How to cite](#)

Package sf:

- represents simple features as records in a `data.frame` or `tibble` with a geometry list-column
- represents natively in R all 17 simple feature types for all dimensions (XY, XYZ, XYM, XYZM)
- interfaces to [GEOS](#) for geometrical operations on projected coordinates, and (through R package [s2](#)) to [s2geometry](#) for geometrical operations on ellipsoidal coordinates
- interfaces to [GDAL](#), supporting all driver options, `Date` and `POSIXct` and list-columns
- interfaces to [PROJ](#) for coordinate reference system conversion and transformation
- uses [well-known-binary](#) serialisations written in C++/Rcpp for fast I/O with GDAL and GEOS
- reads from and writes to spatial databases such as [PostGIS](#) using [DBI](#)
- is extended by
 - [lwgeom](#) for selected liblwgeom/PostGIS functions
 - [stars](#) for raster data, and raster or vector data cubes (spatial time series)

<https://gist.github.com/edzer/f461a3a95570c4ab7edf3125c2f19d20> network data

Tmap

README

GPL-3.0 license

tmap: thematic maps in R

R-CMD-check

passing

codecov

12%

CRAN

3.3-4

CRAN checks

Downloads

License

GPL v3

r-universe

3.99.9002

`tmap` is an actively maintained open-source R-library for drawing thematic maps. The API is based on [A Layered Grammar of Graphics](#) and resembles the syntax of [ggplot2](#), a popular R-library for drawing charts.

Reproducibility with R Markdown

- Markdown is a lightweight `markup language` designed for creating formatted text using a plain-text editor. It allows you to write in a simple, readable format that can be easily converted to HTML, PDF, or other formats.
- R Markdown extends Markdown by integrating R code with the text. This allows you to: embed code, execute code, create dynamic reports, format and typeset your content.

Reproducibility with R Markdown

HTML (on the right). You could also choose to render it into other formats like PDF, MS Word, etc.

On this page

Get Started
Tutorial: Hello, Quarto
Tutorial: Computations
Tutorial: Authoring

hello.qmd


```
---  
title: "Hello, Quarto"  
format: html  
editor: visual  
---  
  
{r}  
#| label: load-packages  
#| include: false  
  
library(tidyverse)  
library(palmerpenguins)
```

Meet Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

Meet the penguins

The `penguins` data from the [palmerpenguins](#) package contains size measurements for 344 penguins from three species observed on three islands in the Palmer Archipelago, Antarctica.

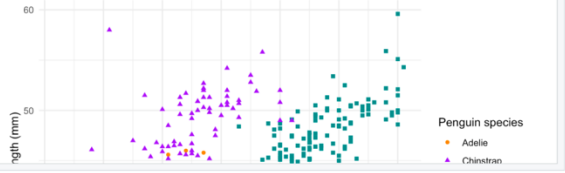


The plot below shows the relationship between flipper and bill lengths of these penguins.

```
{r}  
#| label: plot-penguins
```

Flipper and bill length

Dimensions for penguins at Palmer Station LTER

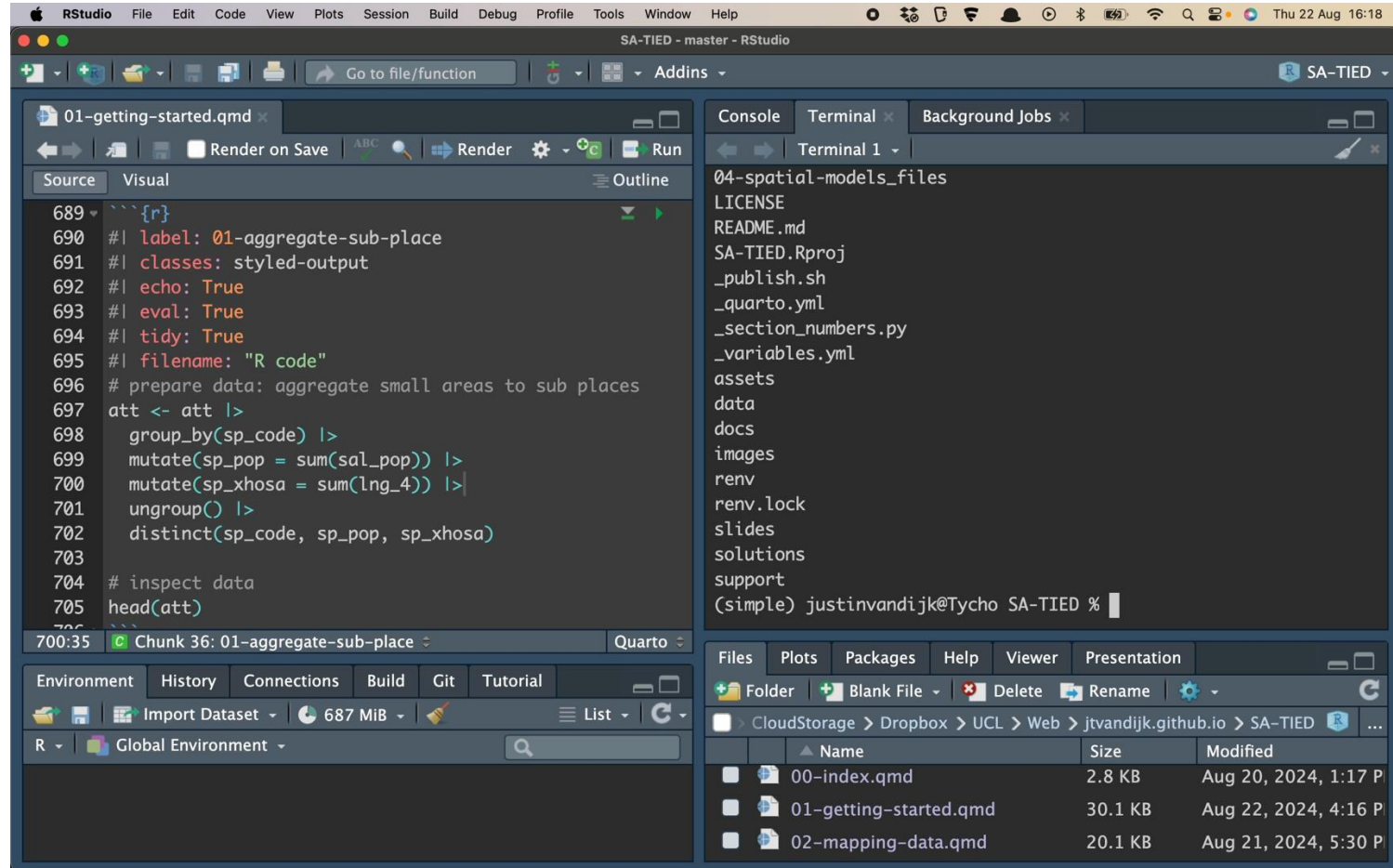


Penguin species

- Adelle
- Chinstrap

This is the basic model for Quarto publishing—take a source document and render it to a variety of output

Reproducibility with R Markdown



The screenshot displays the RStudio environment with the following components:

- Source Editor:** Shows R code for a Quarto document chunk. The code includes metadata (label, classes, echo, eval, tidy, filename) and data processing steps using `dplyr` functions.
- Console:** Displays the output of the R code, showing a directory listing of files and folders.
- Environment:** Shows the current environment as 'Global Environment' with 687 MIB of memory used.
- File Explorer:** Shows the project structure, including files like `00-index.qmd`, `01-getting-started.qmd`, and `02-mapping-data.qmd`.

Source Editor Code:

```
689 ```{r}
690 #| label: 01-aggregate-sub-place
691 #| classes: styled-output
692 #| echo: True
693 #| eval: True
694 #| tidy: True
695 #| filename: "R code"
696 # prepare data: aggregate small areas to sub places
697 att <- att |>
698   group_by(sp_code) |>
699   mutate(sp_pop = sum(sal_pop)) |>
700   mutate(sp_xhosa = sum(lng_4)) |>
701   ungroup() |>
702   distinct(sp_code, sp_pop, sp_xhosa)
703
704 # inspect data
705 head(att)
706 ```
```

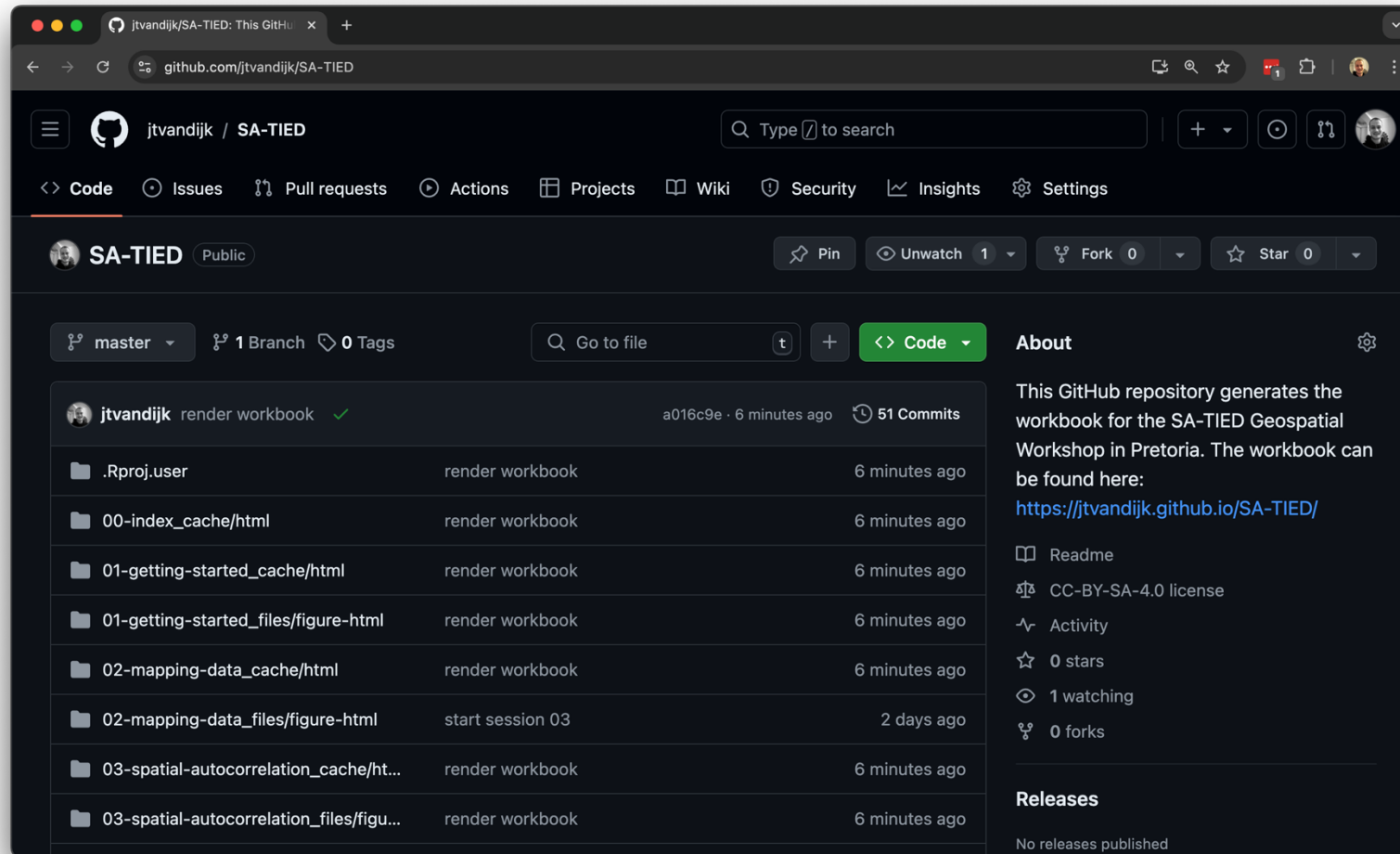
Console Output:

```
04-spatial-models_files
LICENSE
README.md
SA-TIED.Rproj
_publish.sh
_quarto.yml
_section_numbers.py
_variables.yml
assets
data
docs
images
renv
renv.lock
slides
solutions
support
(simple) justinvandijk@Tycho SA-TIED %
```

File Explorer:

Name	Size	Modified
00-index.qmd	2.8 KB	Aug 20, 2024, 1:17 P
01-getting-started.qmd	30.1 KB	Aug 22, 2024, 4:16 P
02-mapping-data.qmd	20.1 KB	Aug 21, 2024, 5:30 P

Reproducibility with R Markdown

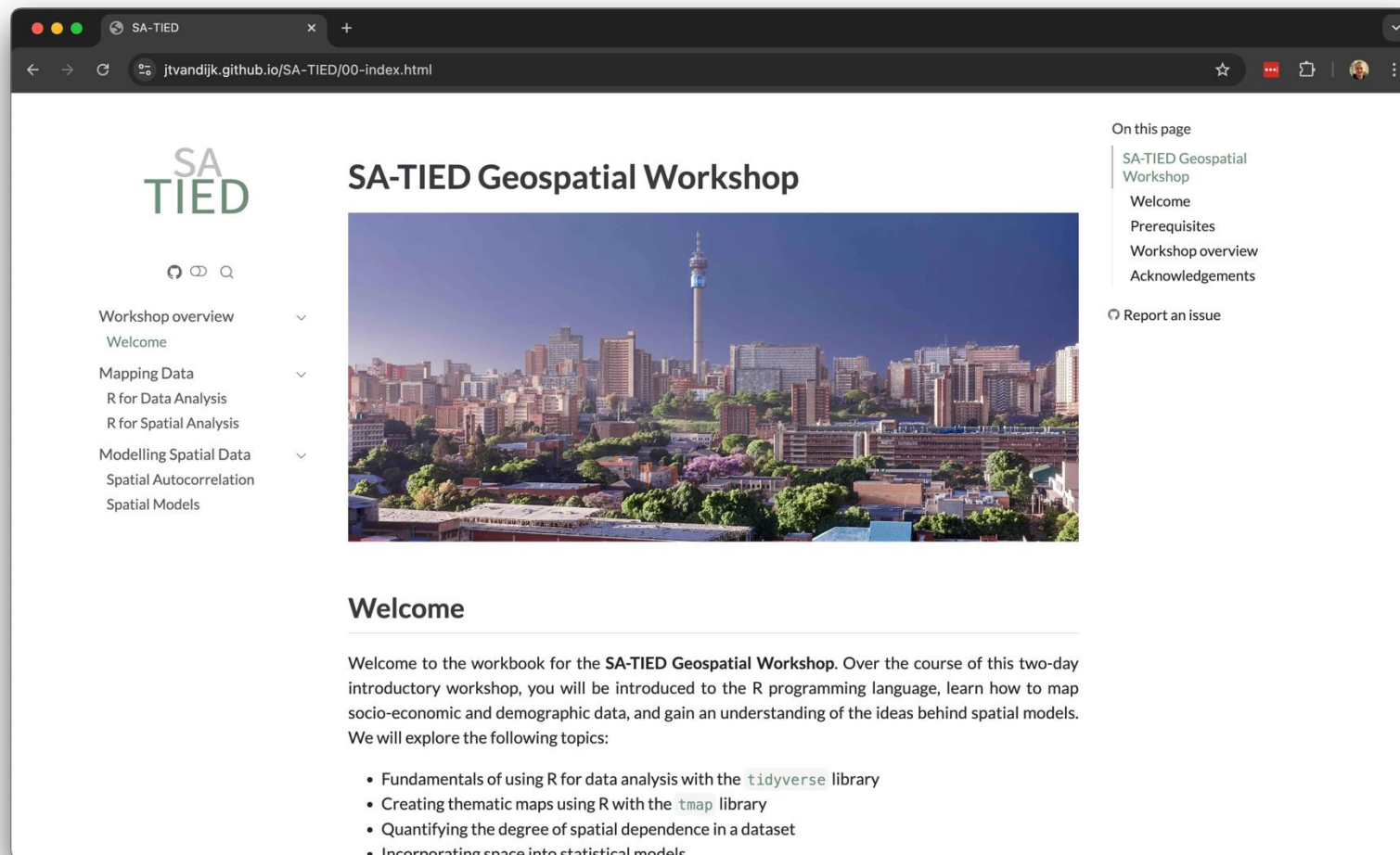


The screenshot shows a web browser displaying the GitHub repository page for 'jtvandijk / SA-TIED'. The repository is public and has 1 branch and 0 tags. The main content area shows a list of files and folders, all rendered from a workbook. The files include:

- .Rproj.user
- 00-index_cache/html
- 01-getting-started_cache/html
- 01-getting-started_files/figure-html
- 02-mapping-data_cache/html
- 02-mapping-data_files/figure-html
- 03-spatial-autocorrelation_cache/ht...
- 03-spatial-autocorrelation_files/figu...

The right sidebar contains the 'About' section, which states: 'This GitHub repository generates the workbook for the SA-TIED Geospatial Workshop in Pretoria. The workbook can be found here: <https://jtvandijk.github.io/SA-TIED/>'. It also lists the repository's metadata: Readme, CC-BY-SA-4.0 license, Activity, 0 stars, 1 watching, and 0 forks. The 'Releases' section indicates that no releases have been published.

Reproducibility with R Markdown



The screenshot shows a web browser window with the address bar displaying `jtvandijk.github.io/SA-TIED/00-index.html`. The page features the SA-TIED logo on the left, a main heading "SA-TIED Geospatial Workshop" with a cityscape image, and a "Welcome" section. A sidebar on the right lists page contents, and a left sidebar lists navigation links.

SA-TIED

SA-TIED Geospatial Workshop

Welcome

Welcome to the workbook for the **SA-TIED Geospatial Workshop**. Over the course of this two-day introductory workshop, you will be introduced to the R programming language, learn how to map socio-economic and demographic data, and gain an understanding of the ideas behind spatial models. We will explore the following topics:

- Fundamentals of using R for data analysis with the `tidyverse` library
- Creating thematic maps using R with the `tmap` library
- Quantifying the degree of spatial dependence in a dataset
- Incorporating space into statistical models

On this page

- SA-TIED Geospatial Workshop
- Welcome
- Prerequisites
- Workshop overview
- Acknowledgements

[Report an issue](#)

Workshop overview

- Welcome

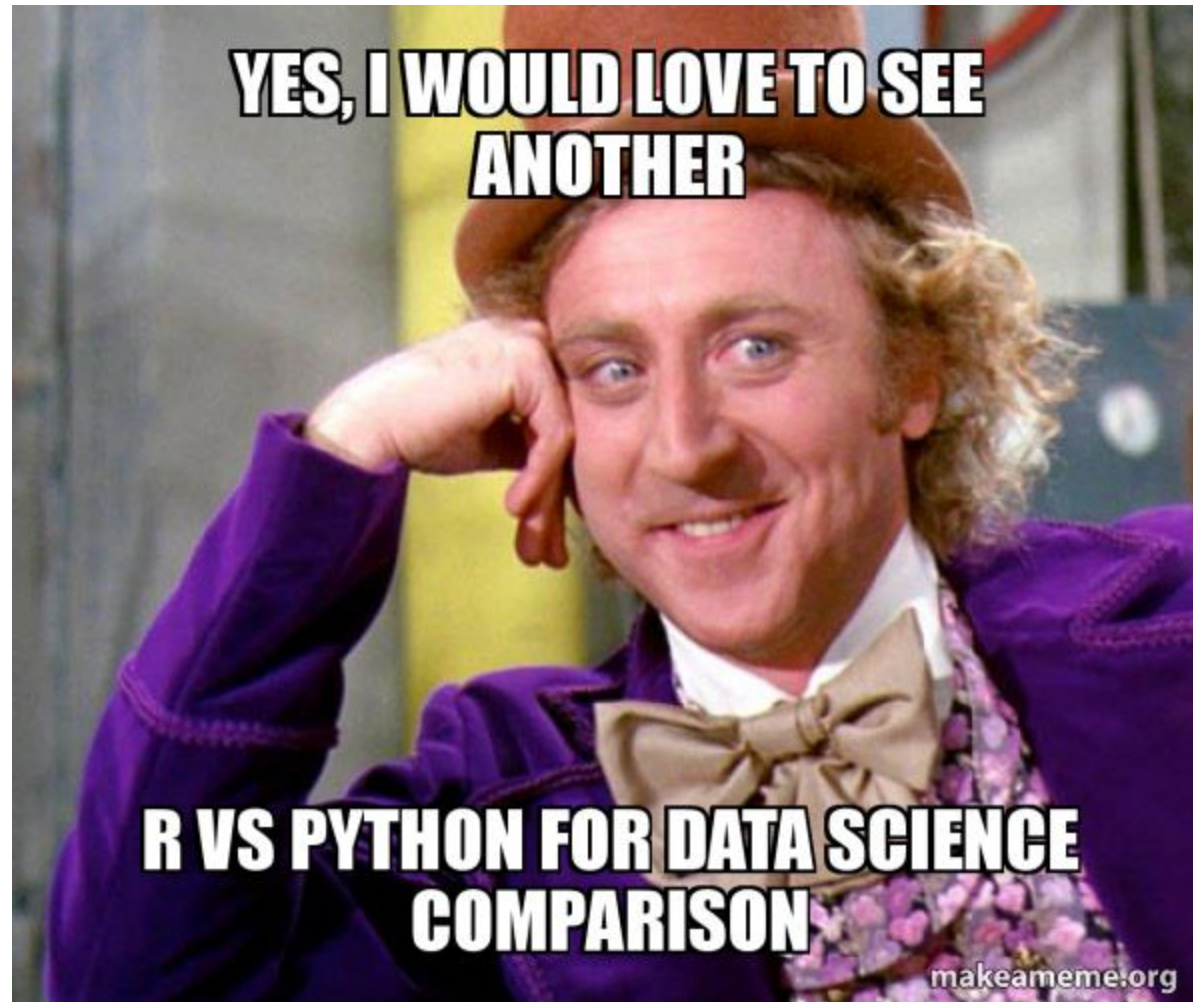
Mapping Data

- R for Data Analysis
- R for Spatial Analysis

Modelling Spatial Data

- Spatial Autocorrelation
- Spatial Models

Python?



Questions

Justin van Dijk
j.t.vandijk@ucl.ac.uk

