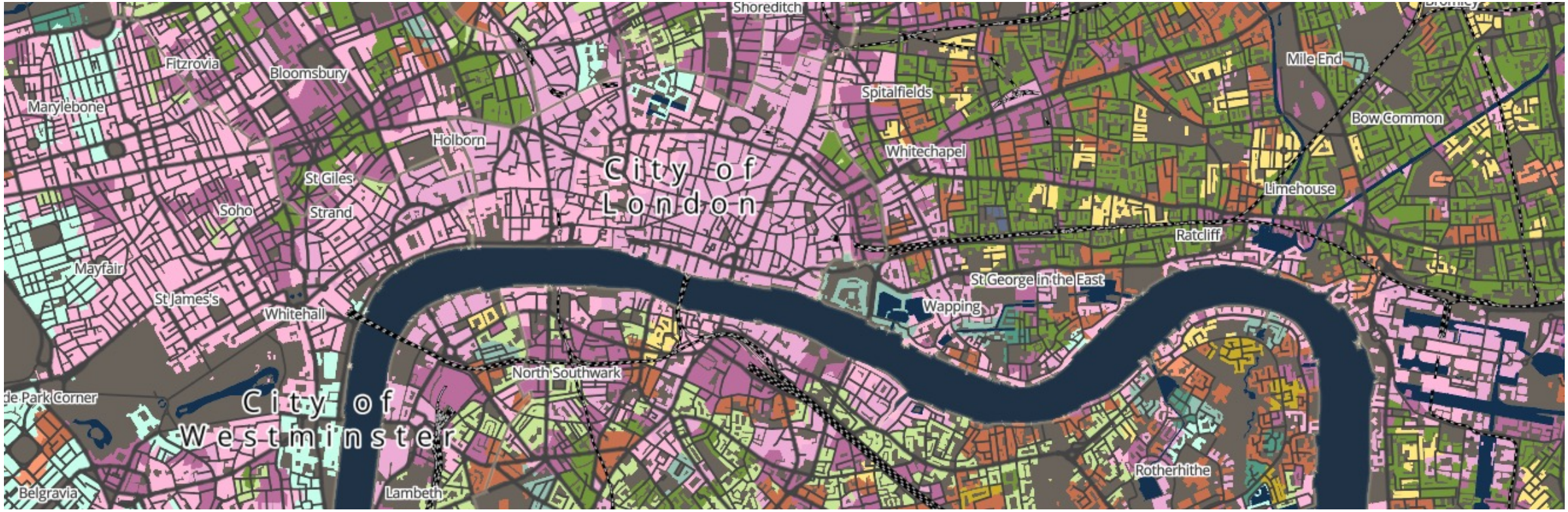
Geocomputation

Programming for Data Analysis



UCL

Where are we at?

Part I: Foundational Concepts

W1	Geocomputation: An Introduction	
W2	GIScience and GIS software	QGIS
W3	Cartography and Visualisation	QGIS
W4	**Programming for Data Analysis**	R
W5	Programming for Spatial Analysis	R

This week

- Short recap
- What is a programming language?
- Principles of R

Before we start

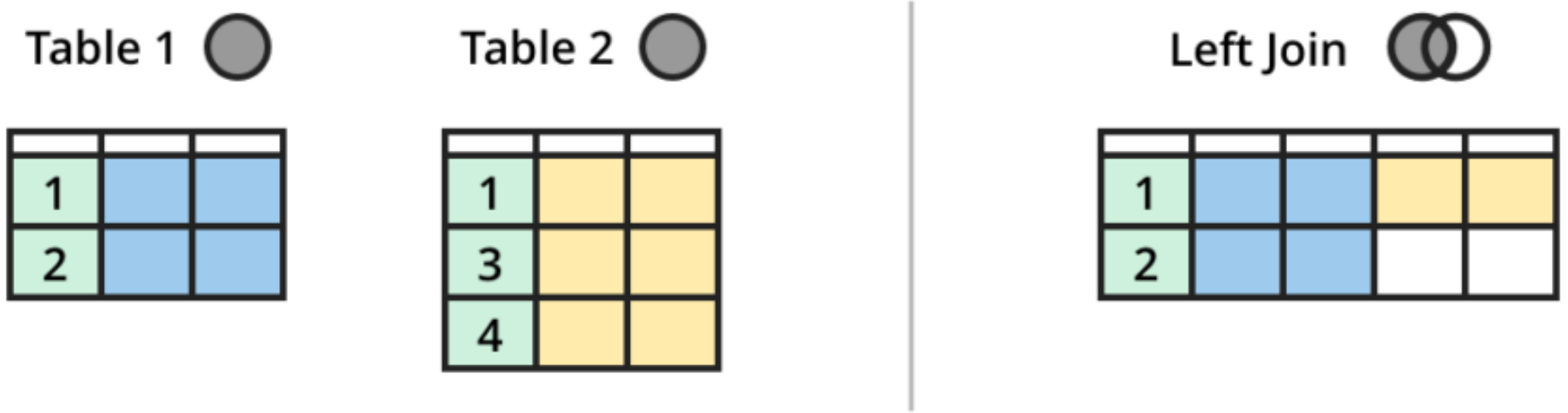
- Go to www.menti.com
- Use code: XXXX XXXX

Recap

- Vector versus raster
- Joining attributes to spatial (vector layers)
- Data classification
- Geographic Coordinate Systems and Projected Coordinate Systems

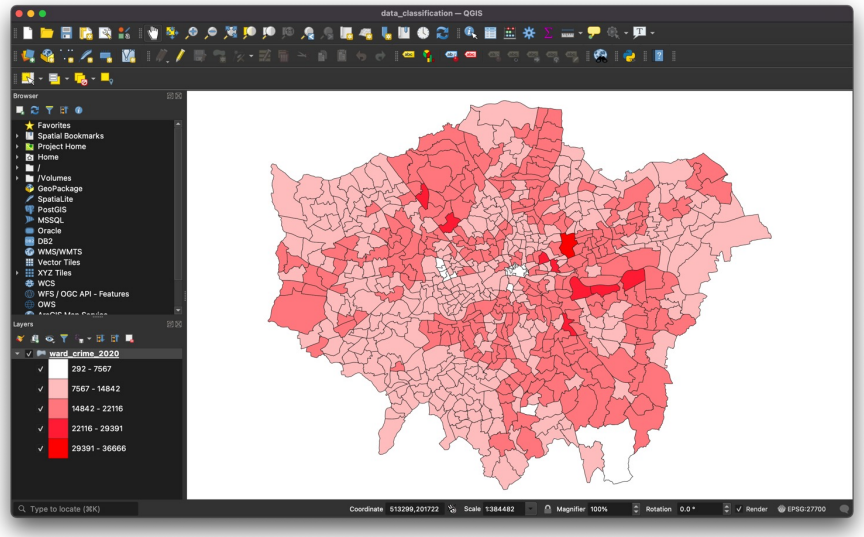
8	9	9	10	0	10	10	10	0	0	0	7	5	3	0	0	0	0	1
8	9	9	10	10	0	10	9	9	0	0	5	3	0	0	0	0	0	0
8	8	9	9	10	0	0	9	8	7	5	0	0	0	1	0	0	0	0
5	8	8	9	10	10	0	9	7	5	0	0	5	5	5	0	0	0	1
3	5	8	9	9	10	0	0	3	0	0	0	5	0	0	1	0	0	2
2	5	8	8	9	9	10	0	0	0	1	5	0	0	0	0	0	0	1
2	4	6	8	8	9	0	0	0	1	5	0	0	5	5	5	0	0	1
0	3	6	8	8	0	0	0	0	5	0	5	5	5	5	5	0	0	0
2	2	5	8	0	0	0	0	0	0	5	5	5	5	5	5	3	0	0
0	2	5	0	0	1	2	3	4	4	4	4	4	4	4	5	0	0	0
0	0	0	0	1	1	1	1	4	4	4	4	4	4	4	5	0	0	0
0	0	1	1	2	2	2	2	3	3	3	3	3	3	3	4	0	3	0
1	1	1	1	2	2	3	3	3	3	1	1	1	1	1	2	3	4	3

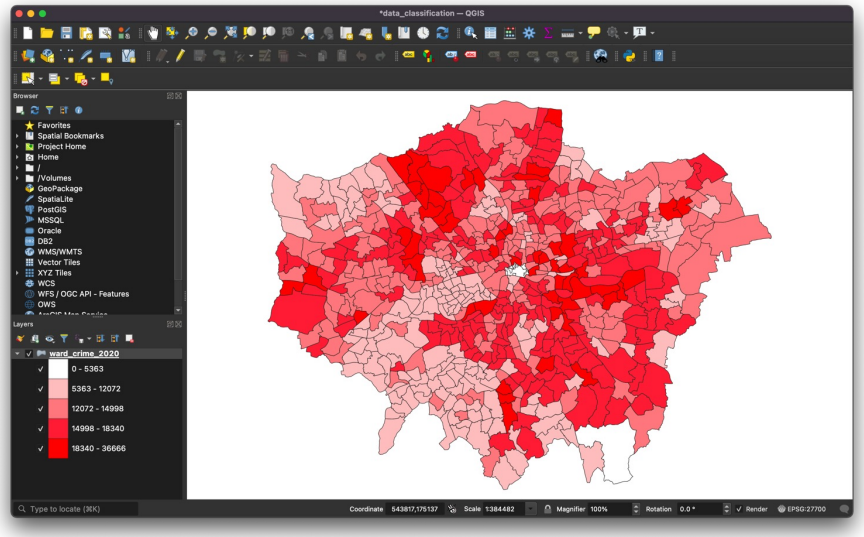
Attribute join

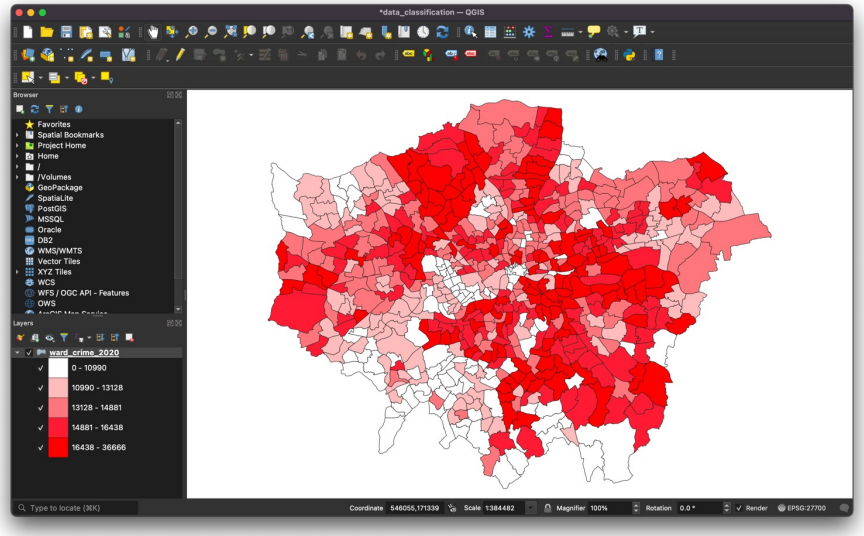


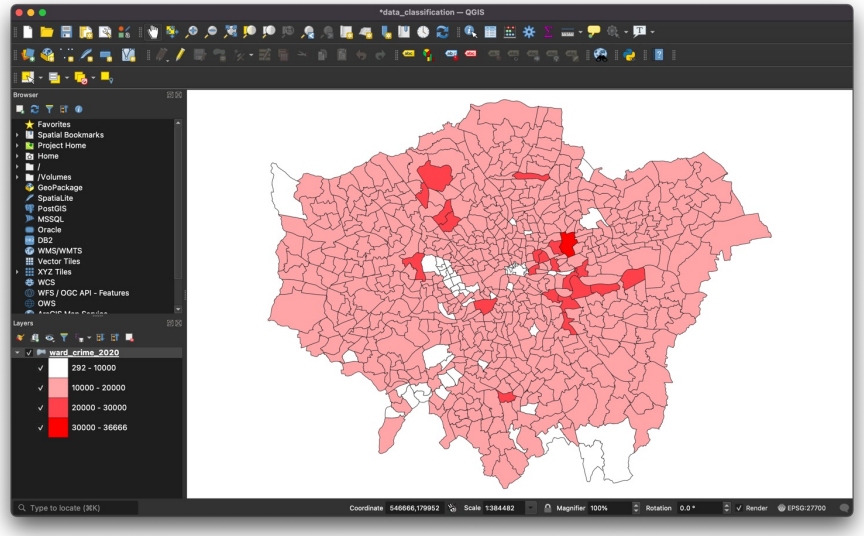
The Data School. 2019. *SQL Join Types Explained Visually*.

Classifying data

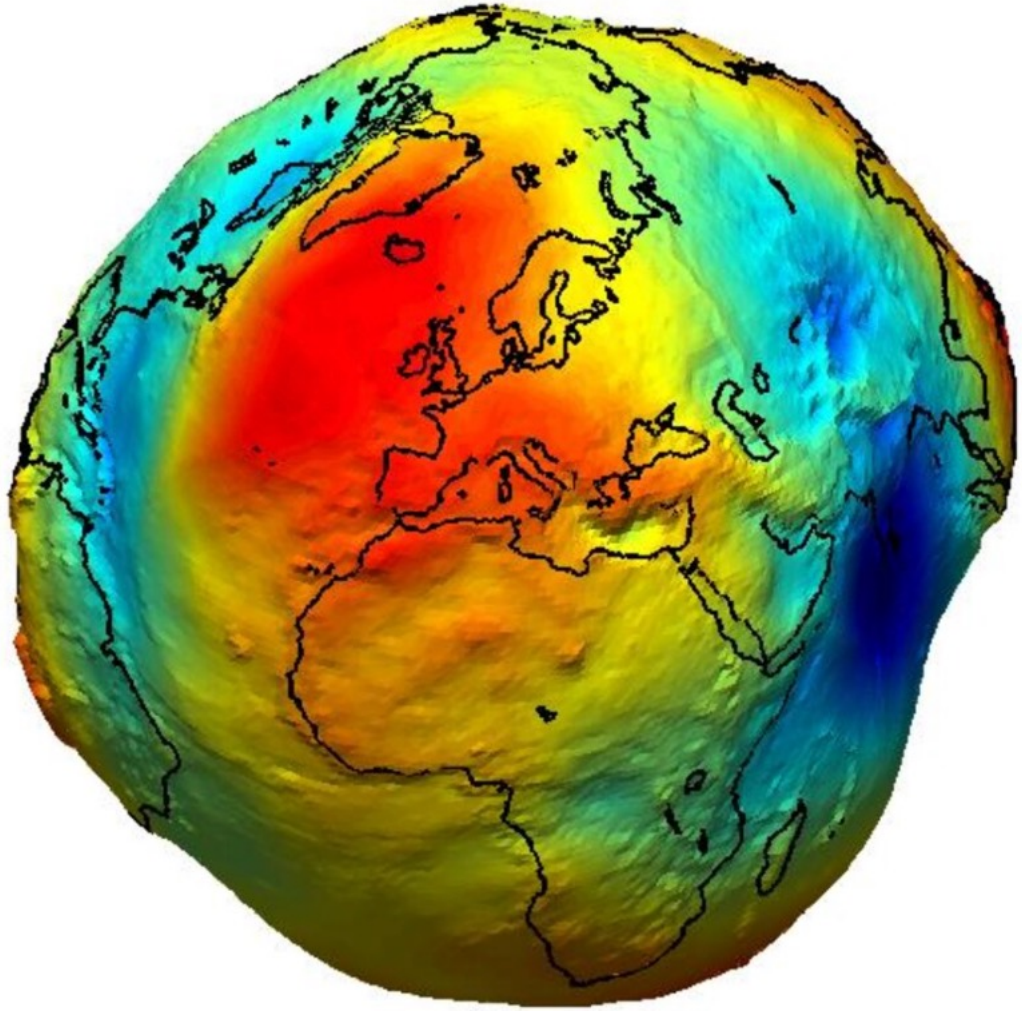




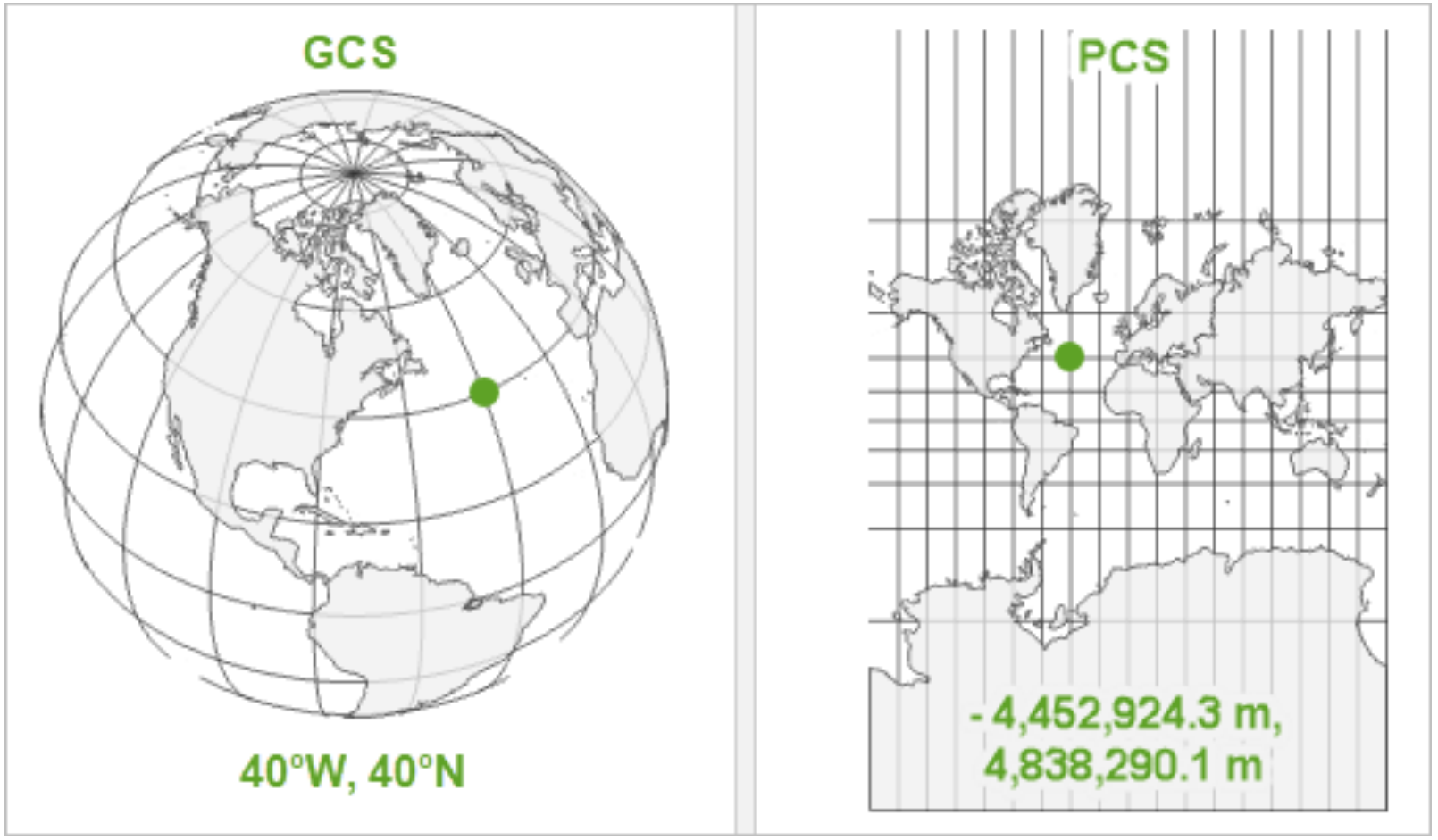




Representing the globe



Moving from an ellipsoid to a plane



QGIS

- Powerful open-source GIS, widely used both within and outside of academia.
- Extensive documentation and user community.
- Indispensable when learning about core spatial analysis / GIScience concepts: layering spatial data, attribute joins

- However (1): QGIS can be a bit fiddly at times. *Aligning multiple maps on a canvas?*
- However (2): Graphical User Interface involves lots of manual actions, time-involved to repeat.

Programming languages

“Everyone does need to learn to code. It is no longer sufficient for a GI Scientists to just work with a standard GIS interface: menus, buttons and black boxes.”

Brunsdon and Comber 2020

Programming languages

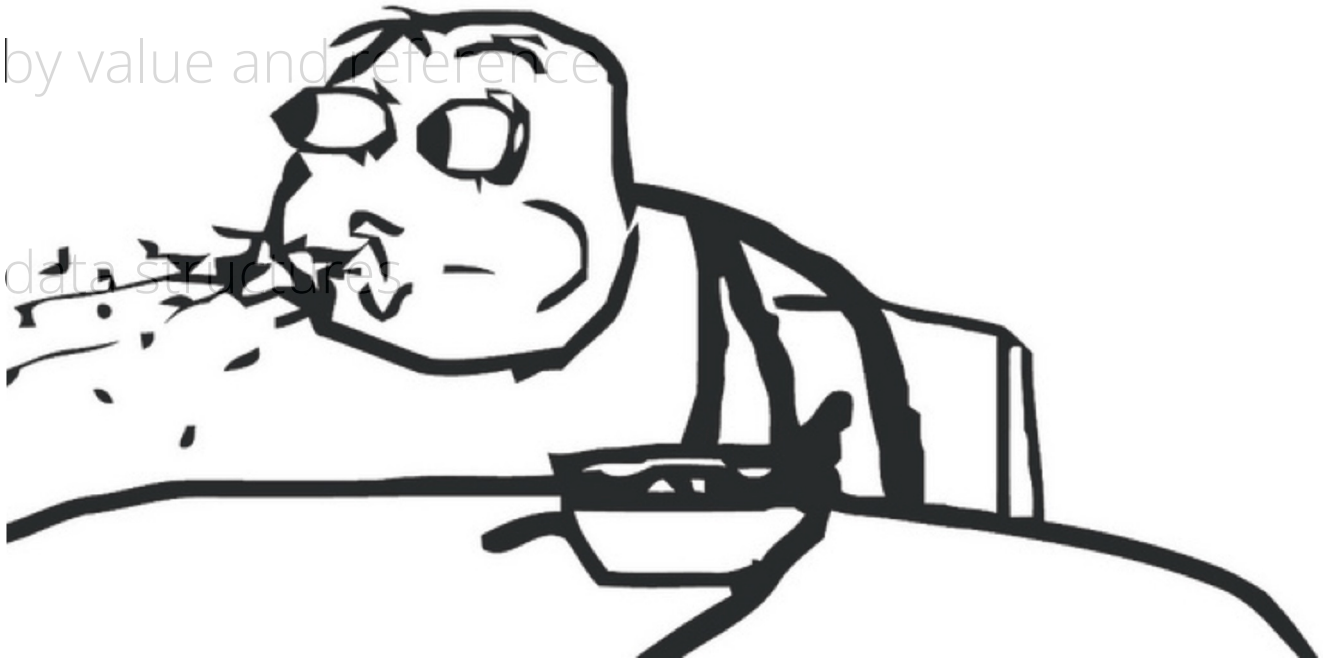
- From ”point-and-click” to “writing commands”.
- We will be using a programming language called R.
- Reproducible research

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

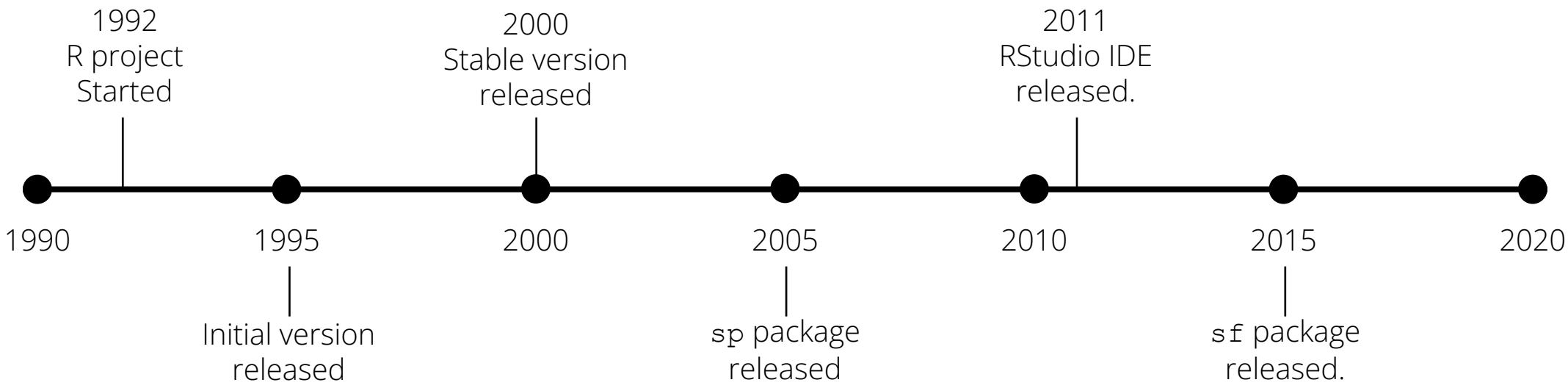
- Ιδεντιφιερς ανδ πριμιτιfε δατα τψπες
- Ασσιγνμεντ, αριτημετιξ, λογιξαλ ανδ ρελατιοναλ οπερατορς
- Εχπρεσσιον ανδ στατεμεντς, δεβυγγινγ
- Φλοω οφ ξοντρολ· σελεξτιον ανδ ρεπετιτιον
- Φυνξτιονς, παραμετερς πασσινγ, ξαλλ βψ αfλυε ανδ ρεφερενξε
- Οβσεξτ-οριεντεδ προγραμμινγ
- 1/2 διμενσιοναλ αρραψς, στρινγς ανδ δατα στρυξτυρες

What is R?

- R is used primarily through interactive command-line.
- R can create and use different types of data but works predominantly very well with a linear collection of things (vector) and tables.
- R is extremely extendable through packages.
- R is brilliant when it comes to static graphics and dealing with spatial data.

A little history

- R is programming language which specialises in statistics.
- 1975: Bell Labs develops a language for Statistical Analysis
- 1992: Ross Ihaka and Robert Gentleman develop opensource version of S



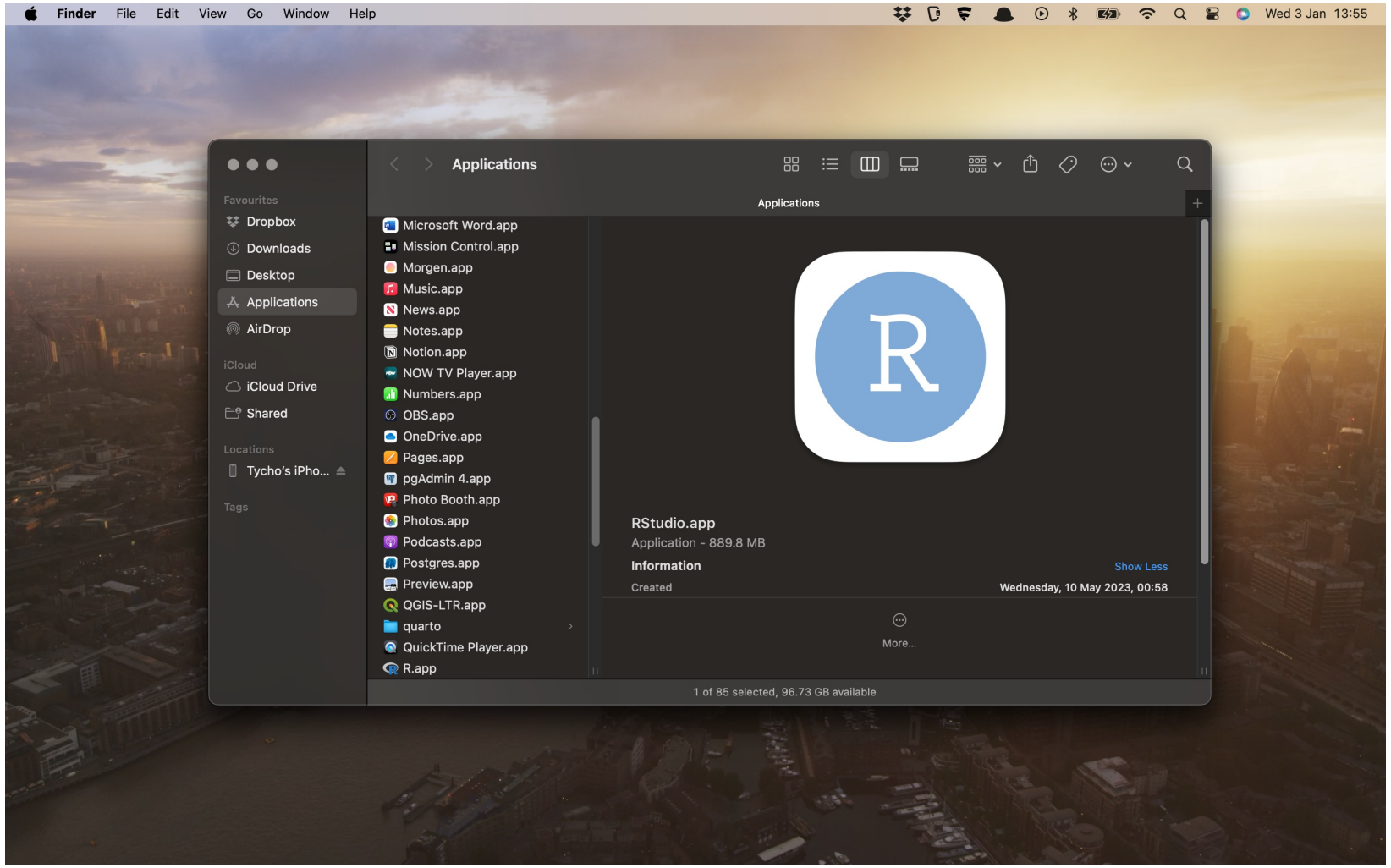
Principles of R

- R is used primarily through interactive command-line.
- R can create and use different types of data but works predominantly very well with a linear collection of things (vector) and tables.
- R is extremely extendable through packages.
- R is brilliant when it comes to static graphics and dealing with spatial data.

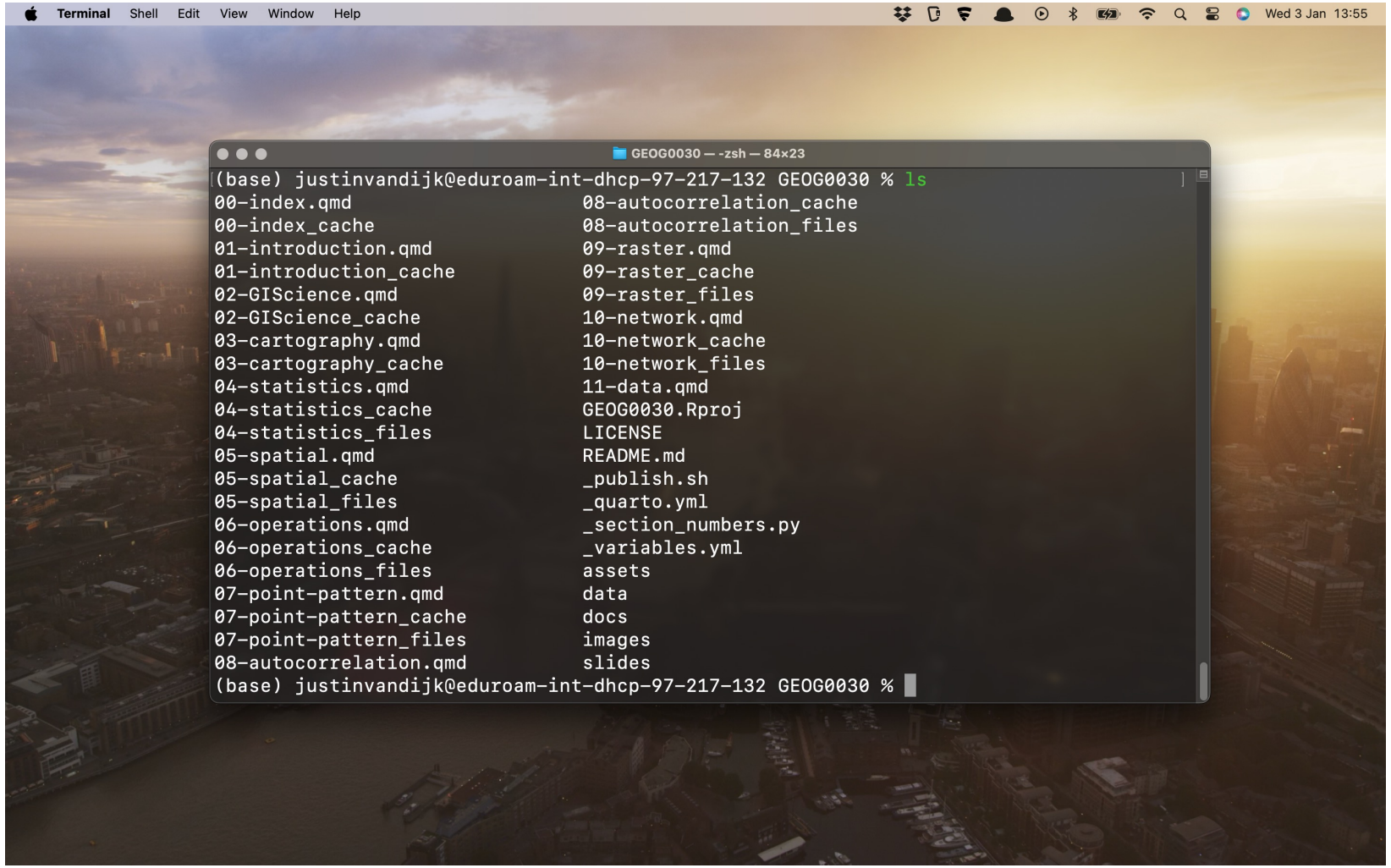
Command-Line Interface

- Command-Line Interface versus (CLI) Graphical User Interface (GUI)
- The GUI allows the user to interact with the system using graphical elements such as windows, icons, menus while the CLI allows the user to interact with the system using commands.

Graphical User Interface



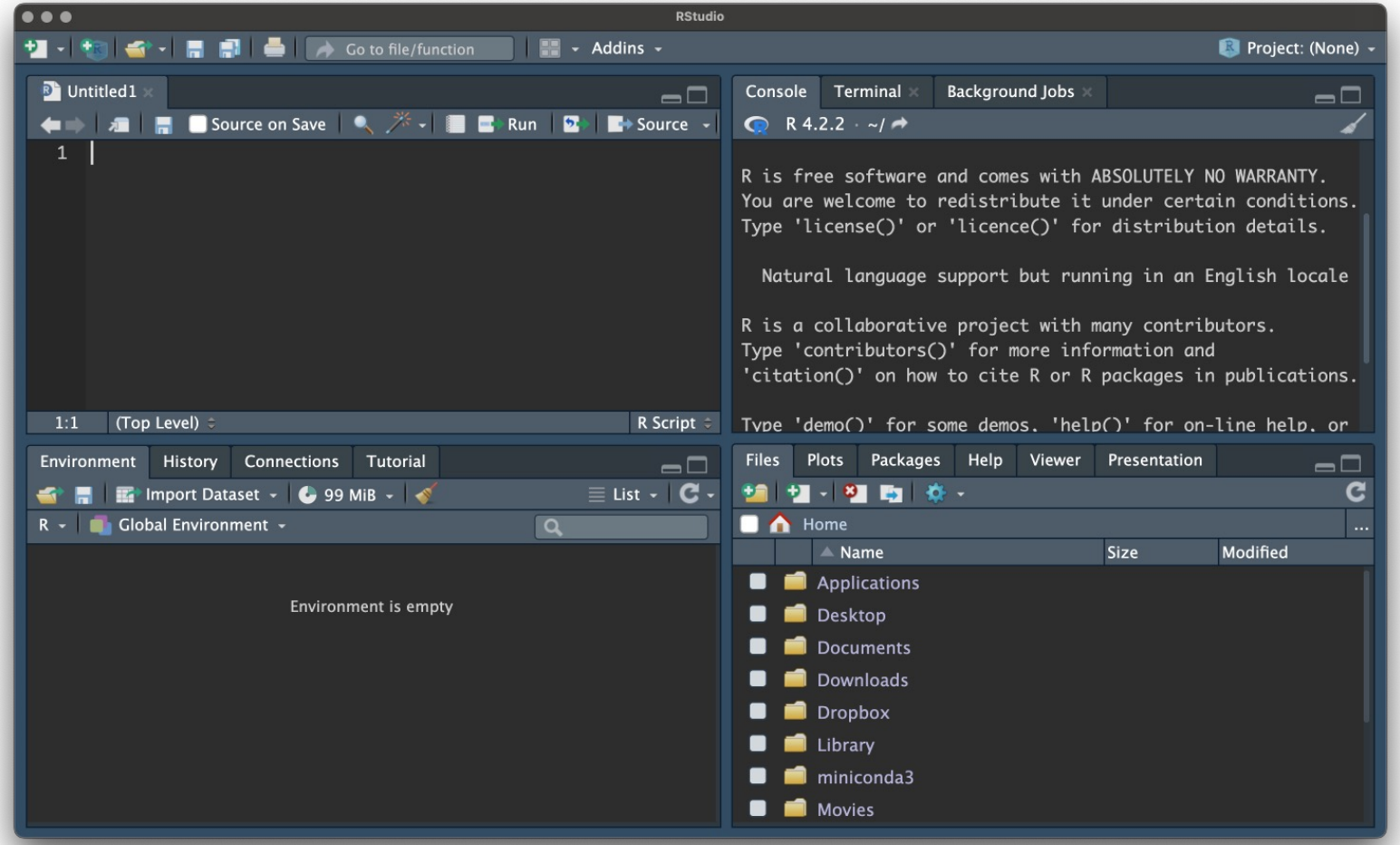
Command-Line Interface



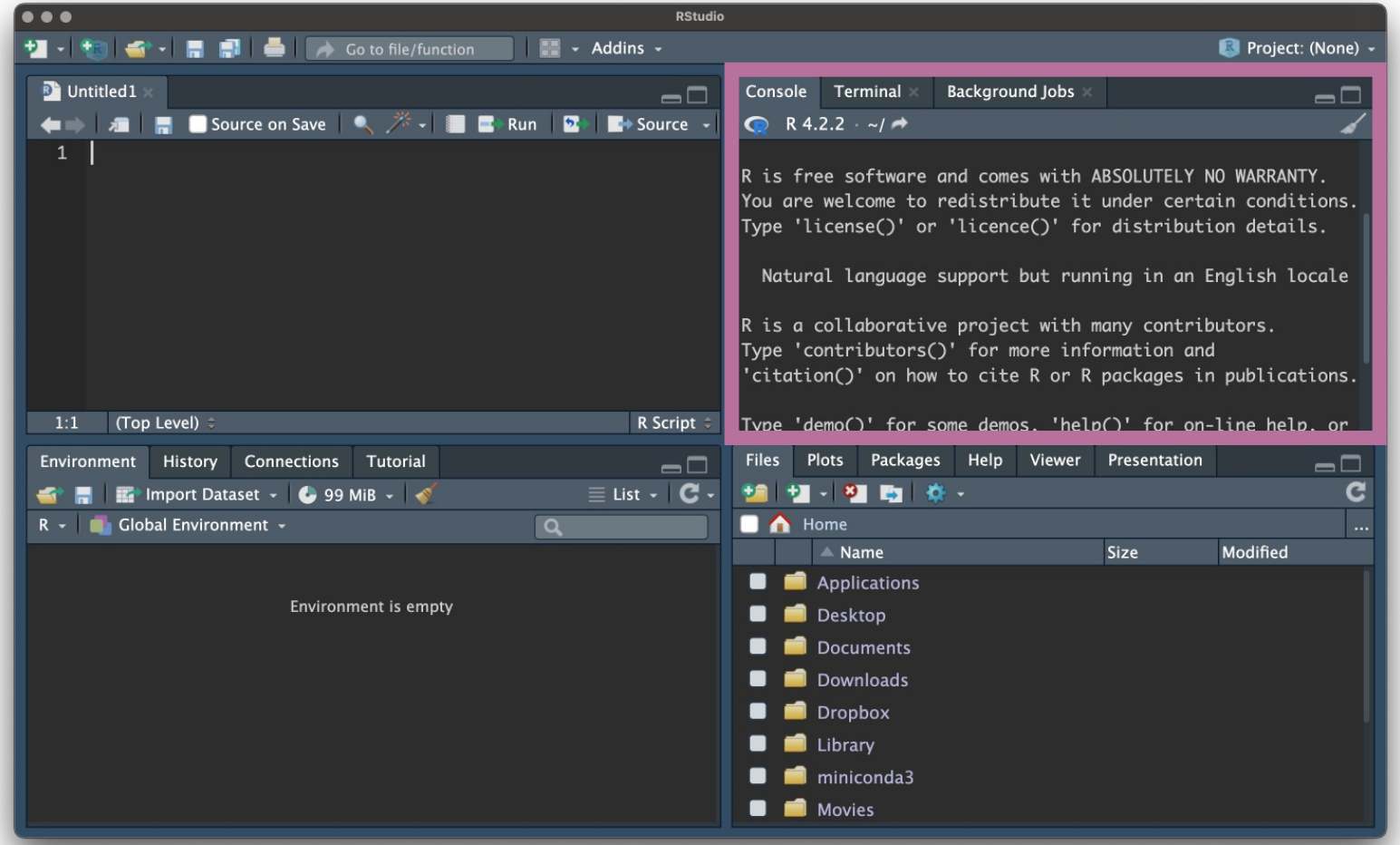
Interaction through language

- Capturing complex instruction with language is much easier than with skeuomorphism.
- Repeating stuff is easy.
- Much steeper learning curve but greater rewards as well.

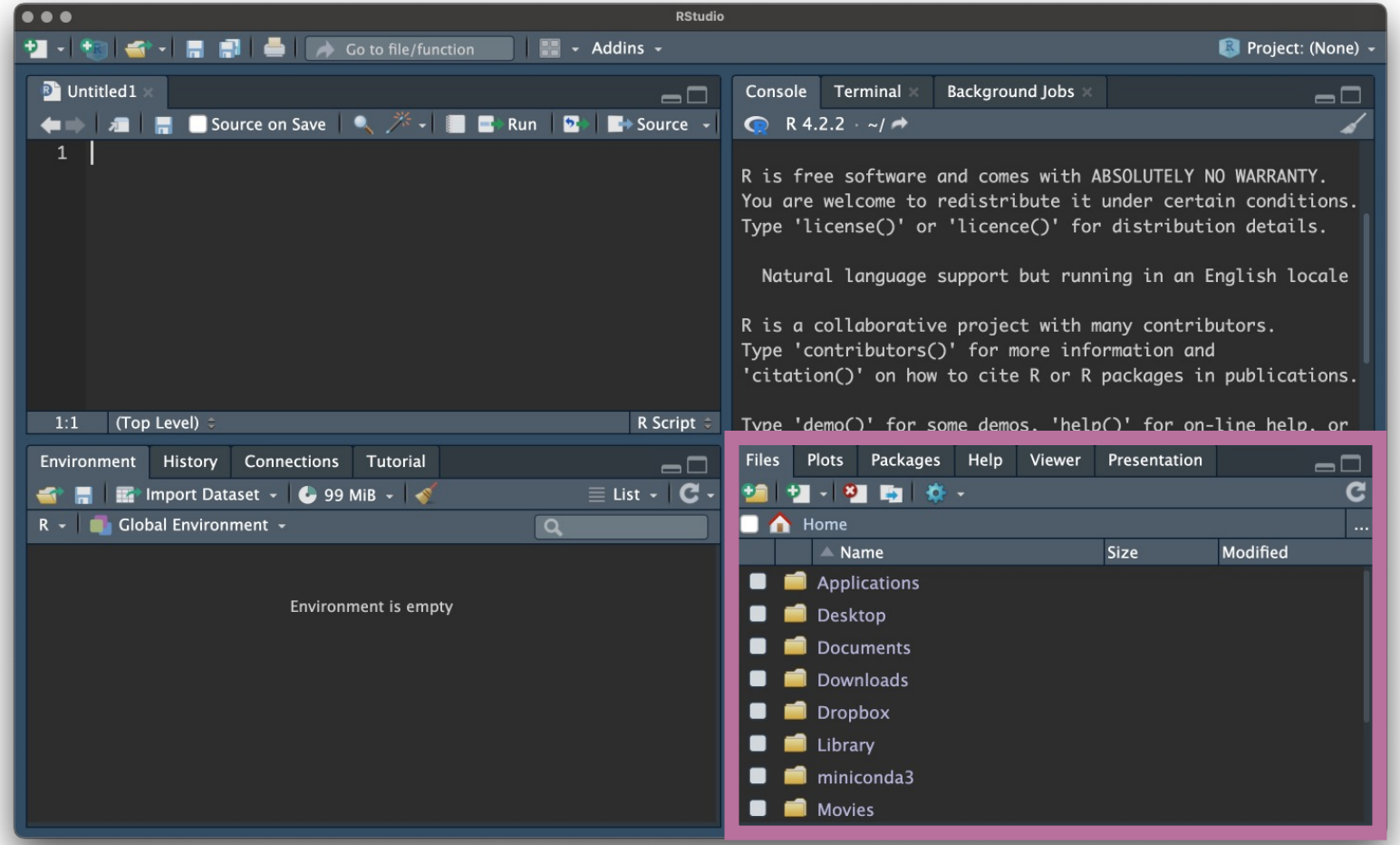
How does this work in R



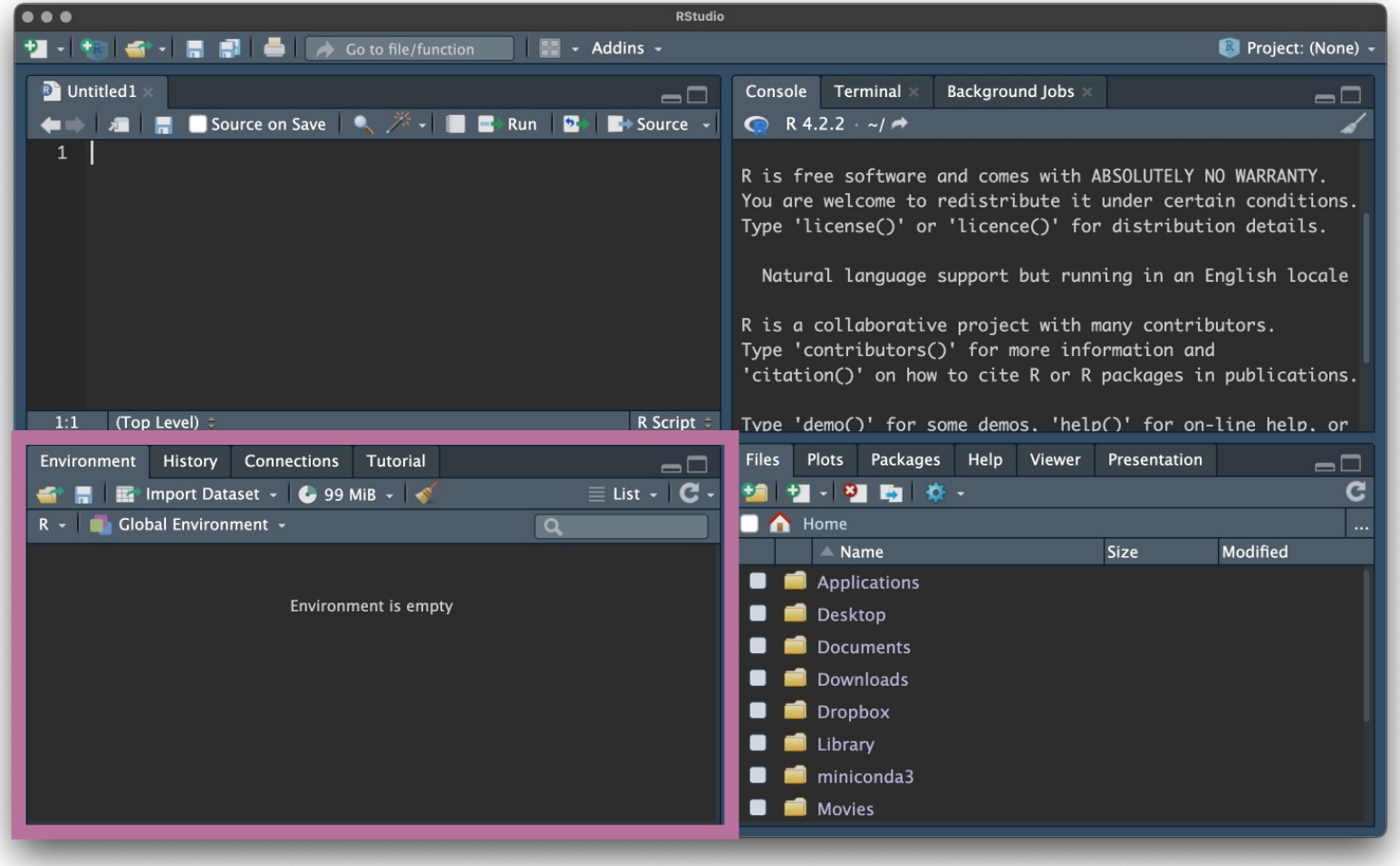
How does this work in R



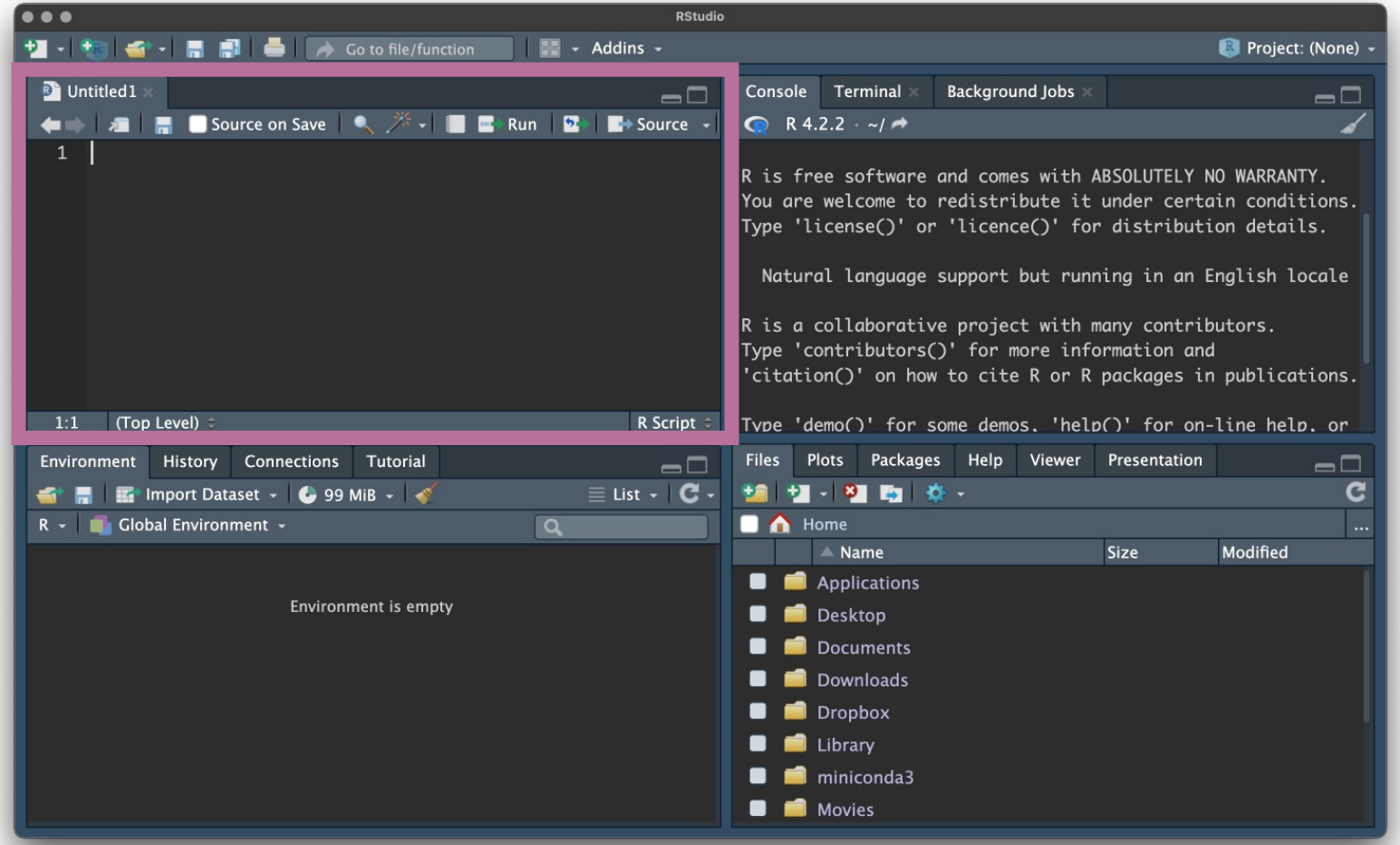
How does this work in R



How does this work in R



How does this work in R

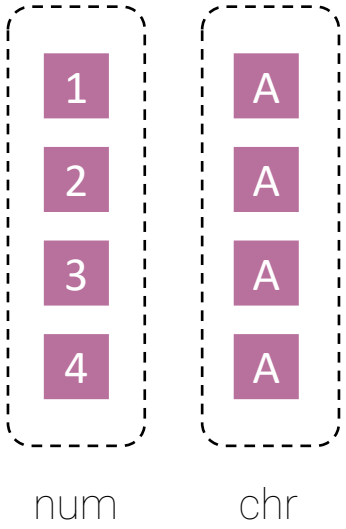


Principles of R

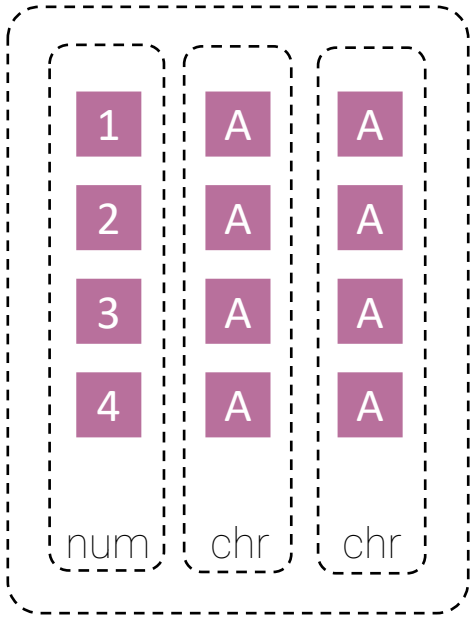
- R is used primarily through interactive command-line.
- R can create and use different types of data but works predominantly very well with a linear collection of things (vector) and tables.
- R is extremely extendable through packages.
- R is brilliant when it comes to static graphics and dealing with spatial data.

Data types

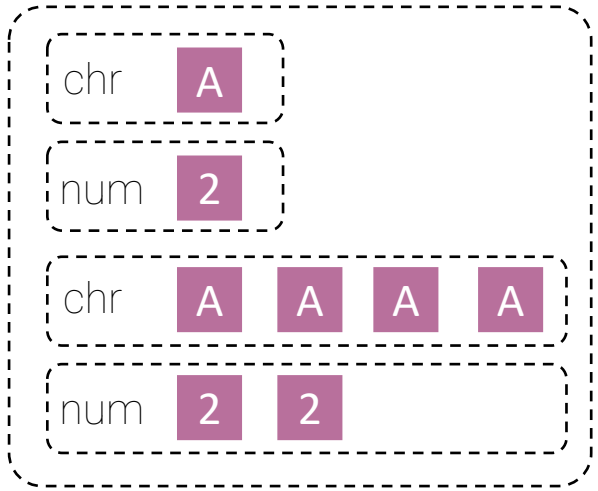
Scalar



Vector



Dataframe



List

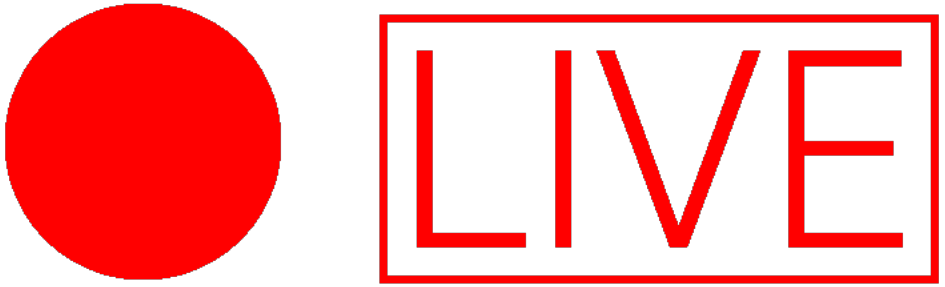
Variables

- 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.

Functions

- Variables can be used as an input for functions.
- Functions are pieces of code that accomplish a specific task.
- Functions usually "take in" data, process it, and "return" a result.
- Once a function is written, it can be used over and over and over again.

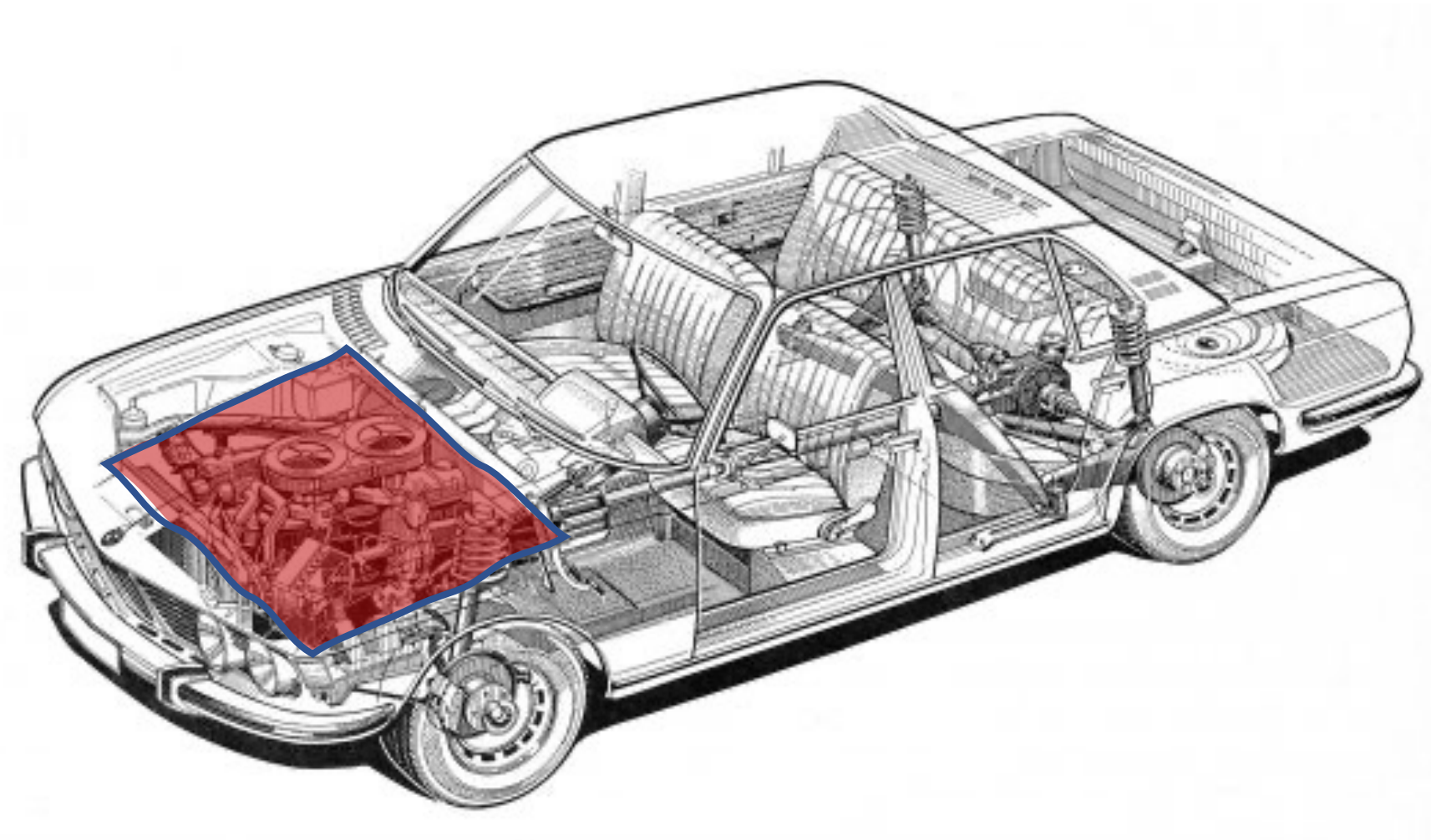
RStudio



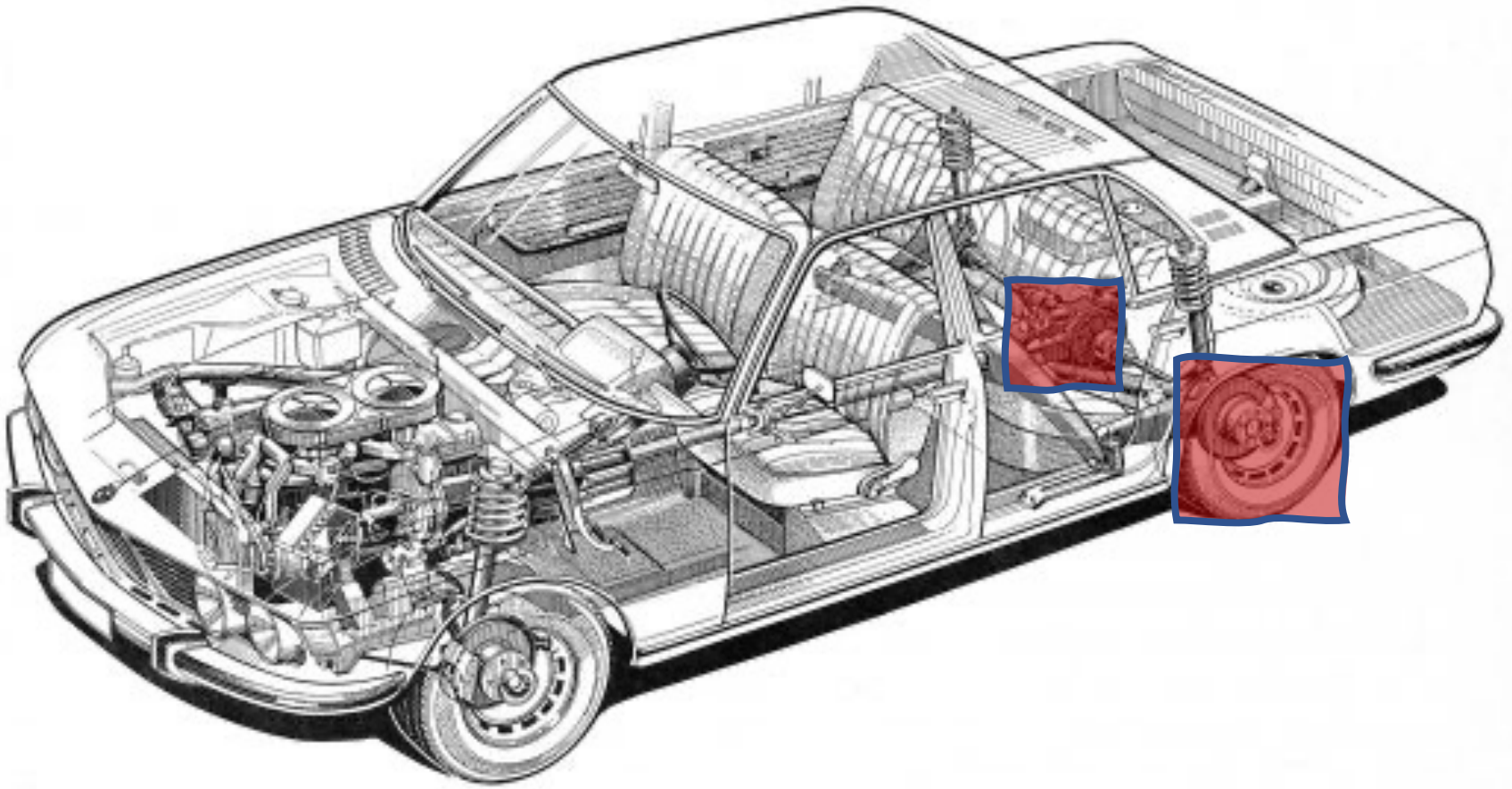
Principles of R

- R is used primarily through interactive command-line.
- R can create and use different types of data but works predominantly very well with a linear collection of things (vector) and tables.
- R is extremely extendable through packages.
- R is brilliant when it comes to static graphics and dealing with spatial data.

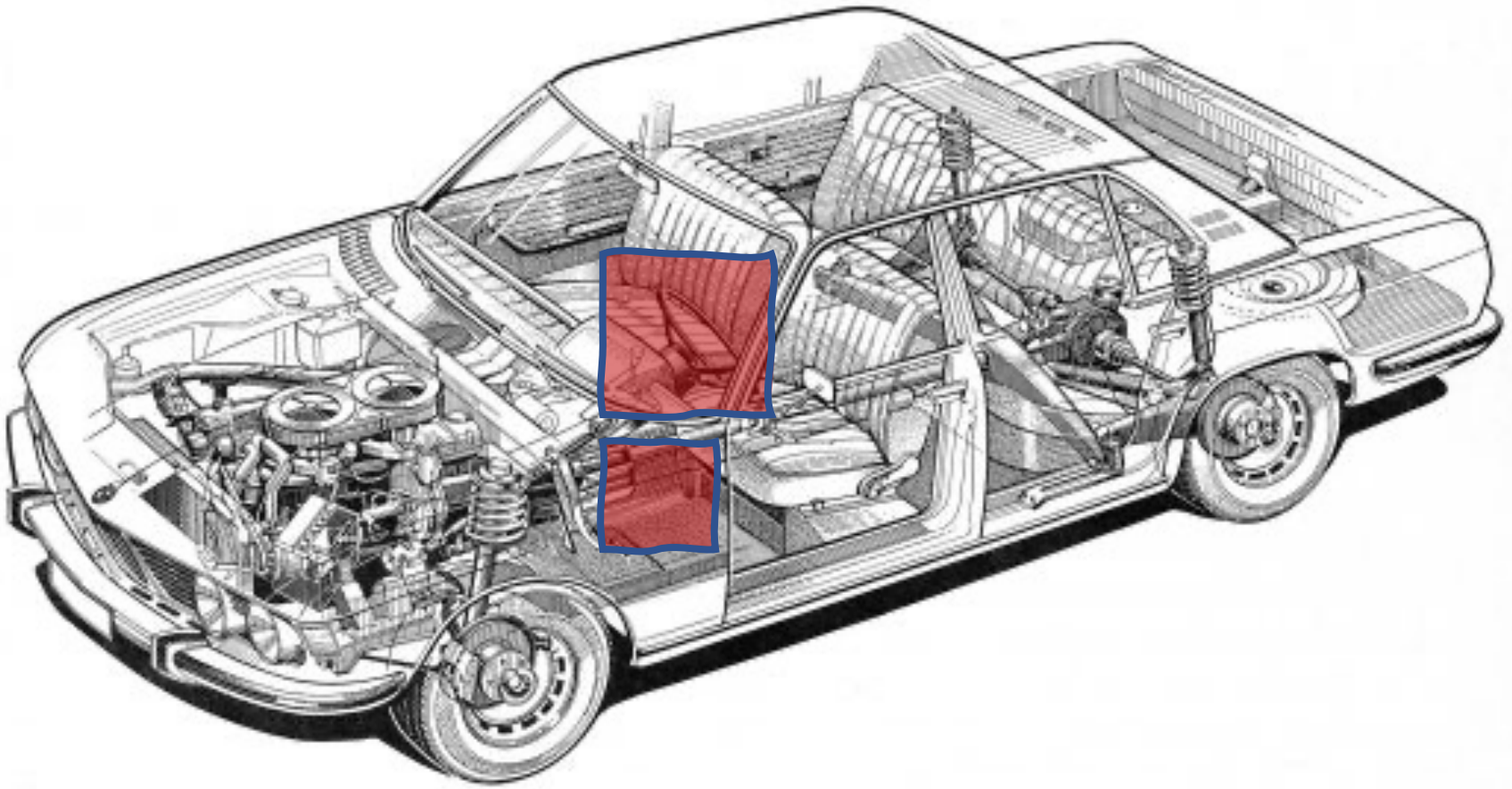
Packages



Packages



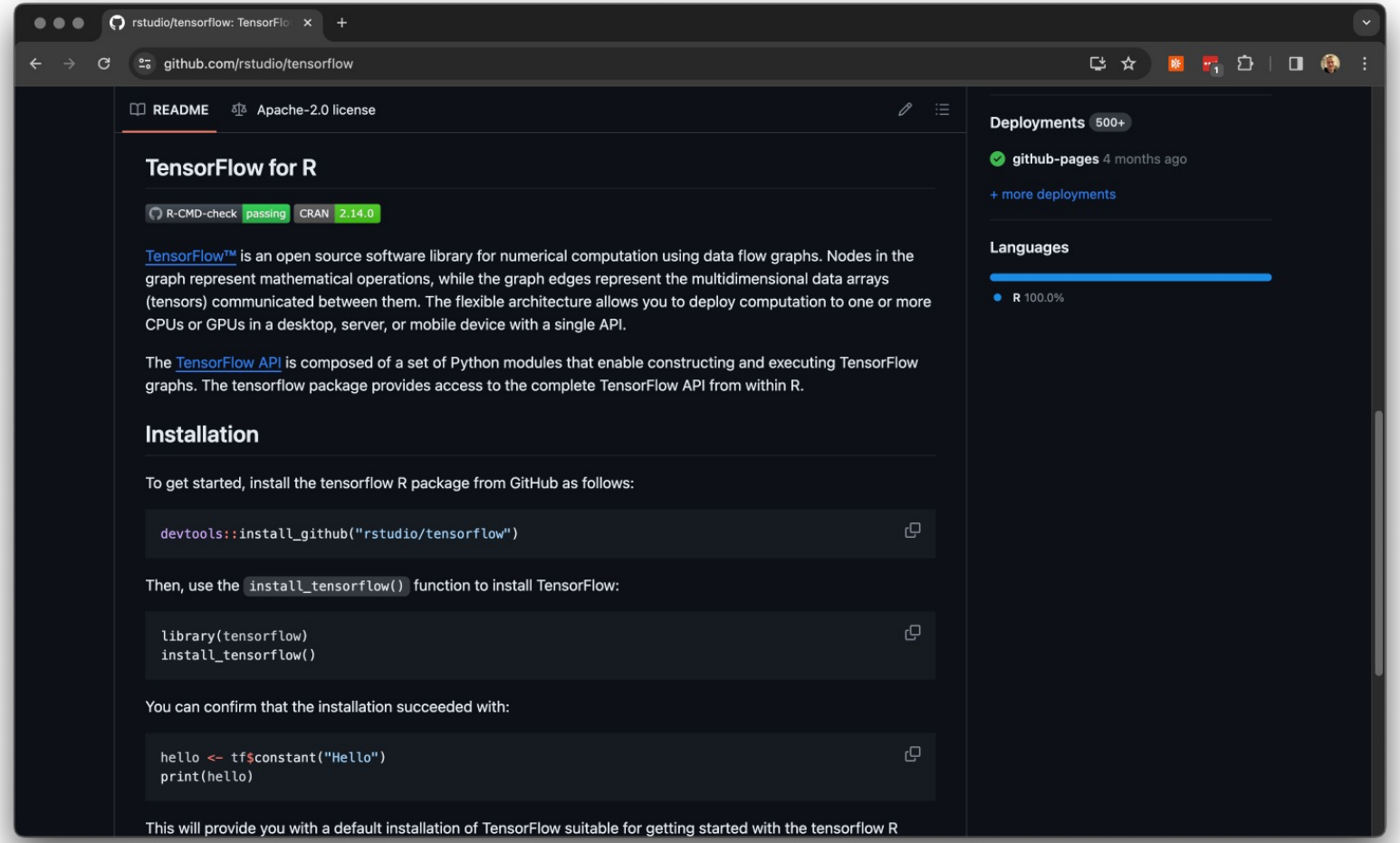
Packages



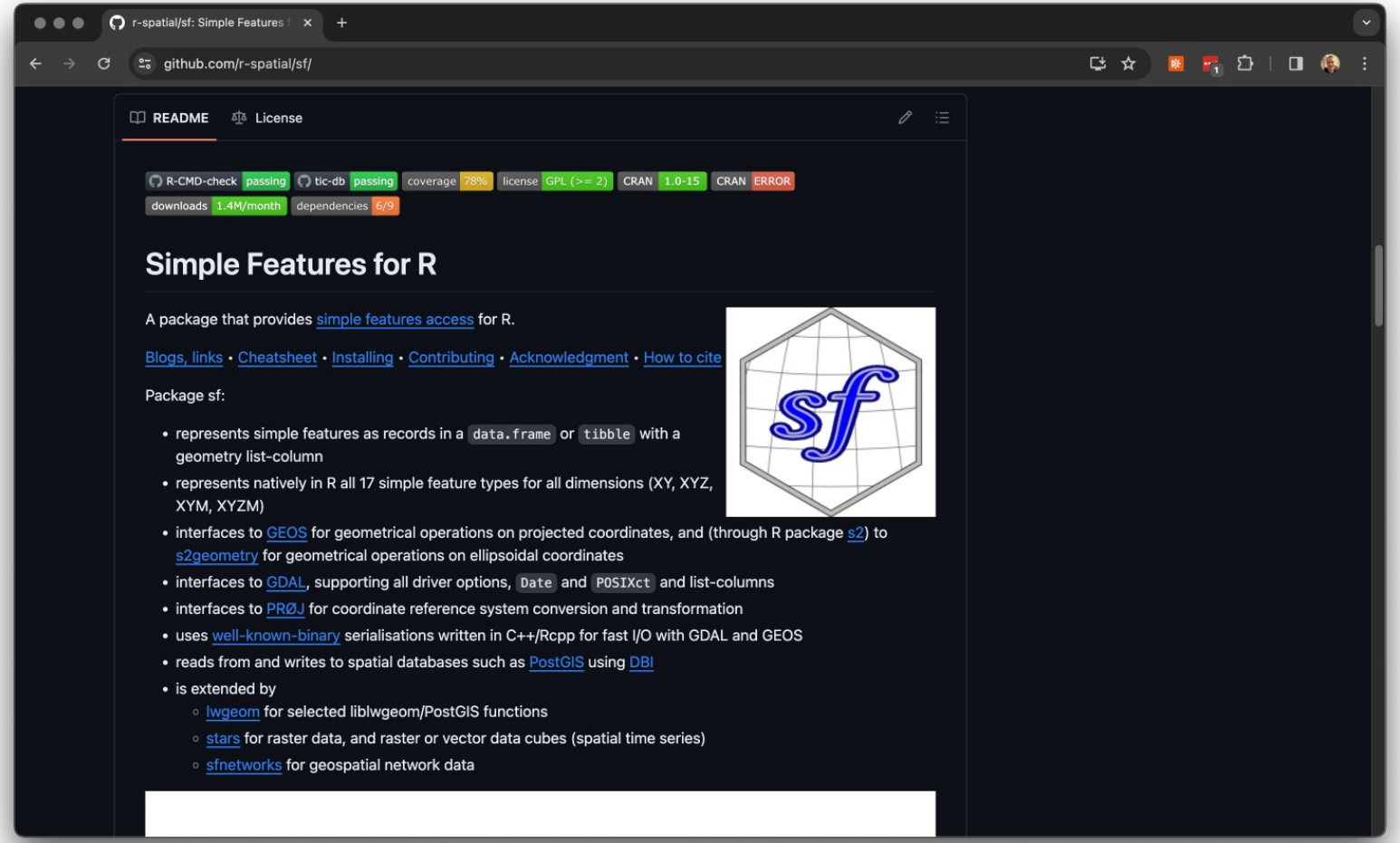
Packages

- Available on The Comprehensive R Archive Network (CRAN).
- CRAN package repository features over 20,249 available packages (01/01/2024).
- Packages provide extensions to R.

tensorflow


rstudio/tensorflow: TensorFlo
github.com/rstudio/tensorflow
README
Apache-2.0 license
TensorFlow for R
R-CMD-check passing
CRAN 2.14.0
TensorFlow™ is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API.
The TensorFlow API is composed of a set of Python modules that enable constructing and executing TensorFlow graphs. The tensorflow package provides access to the complete TensorFlow API from within R.
Installation
To get started, install the tensorflow R package from GitHub as follows:
devtools::install_github("rstudio/tensorflow")
Then, use the install_tensorflow() function to install TensorFlow:
library(tensorflow)
install_tensorflow()
You can confirm that the installation succeeded with:
hello <- tf$constant("Hello")
print(hello)
This will provide you with a default installation of TensorFlow suitable for getting started with the tensorflow R
Deployments 500+
github-pages 4 months ago
+ more deployments
Languages
R 100.0%

sf

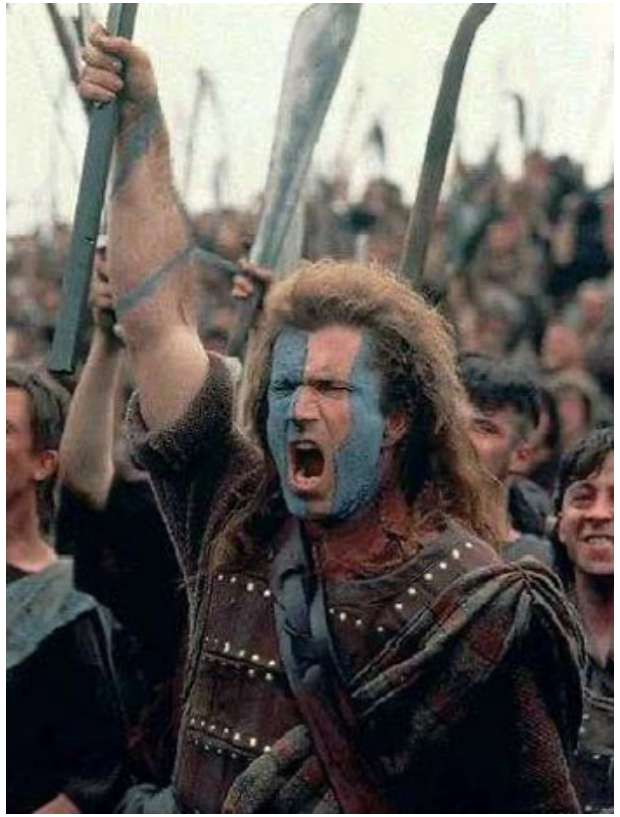

r-spatial/sf: Simple Features
github.com/r-spatial/sf/
README
License
R-CMD-check passing
tic-db passing
coverage 78%
license GPL (>= 2)
CRAN 1.0-15
CRAN ERROR
downloads 1.4M/month
dependencies 6/9
Simple Features for R
A package that provides simple features access for R.
Blogs, links • Cheatsheet • 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 PRØJ 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)
◦ sfnetworks for geospatial network data

Principles of R

- R is used primarily through interactive command-line.
- R can create and use different types of data but works predominantly very well with a linear collection of things (vector) and tables.
- R is extremely extendable through packages.
- R is brilliant when it comes to static graphics and dealing with spatial data.

5 more reasons on why you should use R

R is free as in

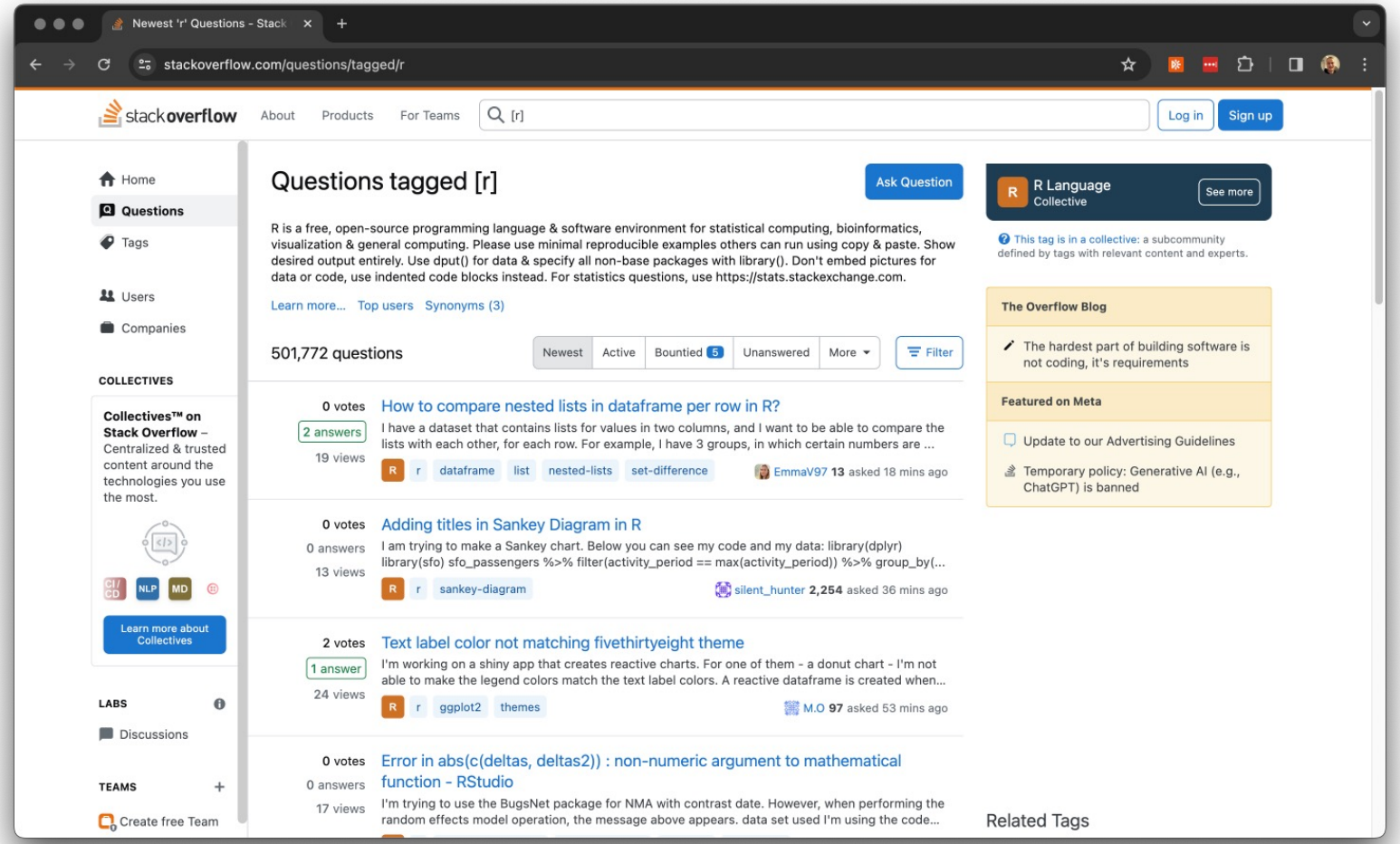


Freedom

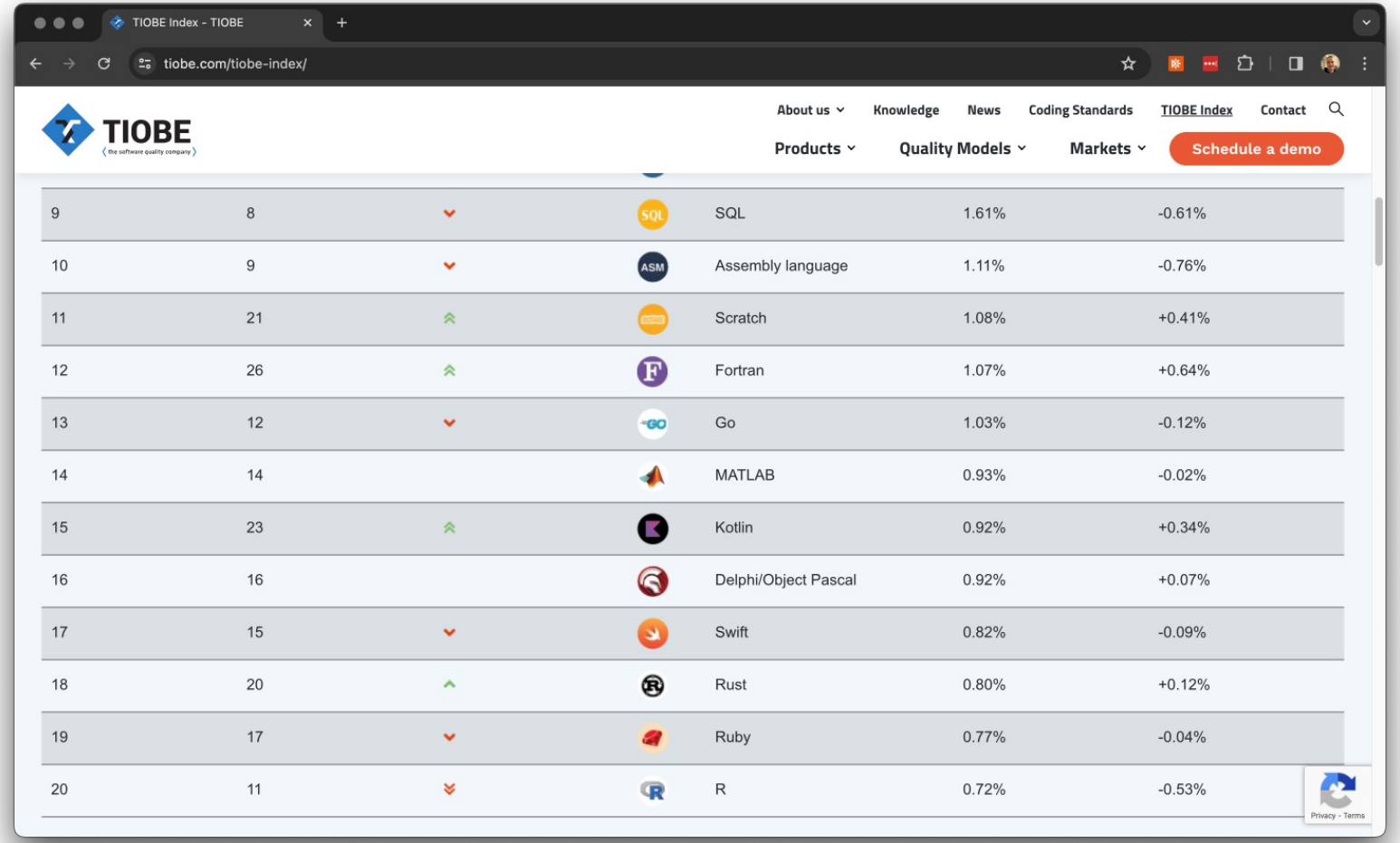
Free beer

R allows you to produce your outputs programmatically

R is supported by a large vibrant community



R is popular and in-demand



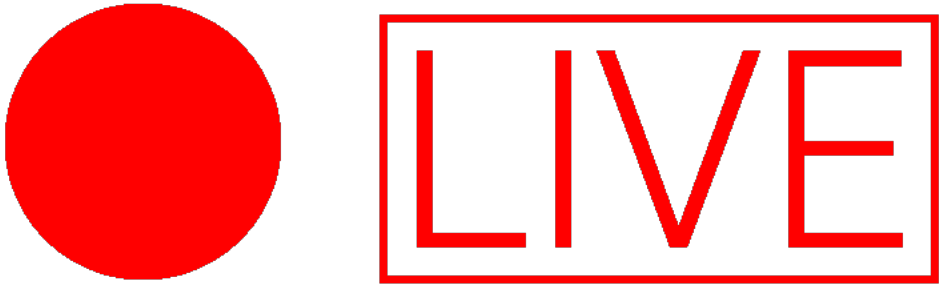
Python?


YES, I WOULD LOVE TO SEE ANOTHER
R VS PYTHON FOR DATA SCIENCE COMPARISON
makeameme.org

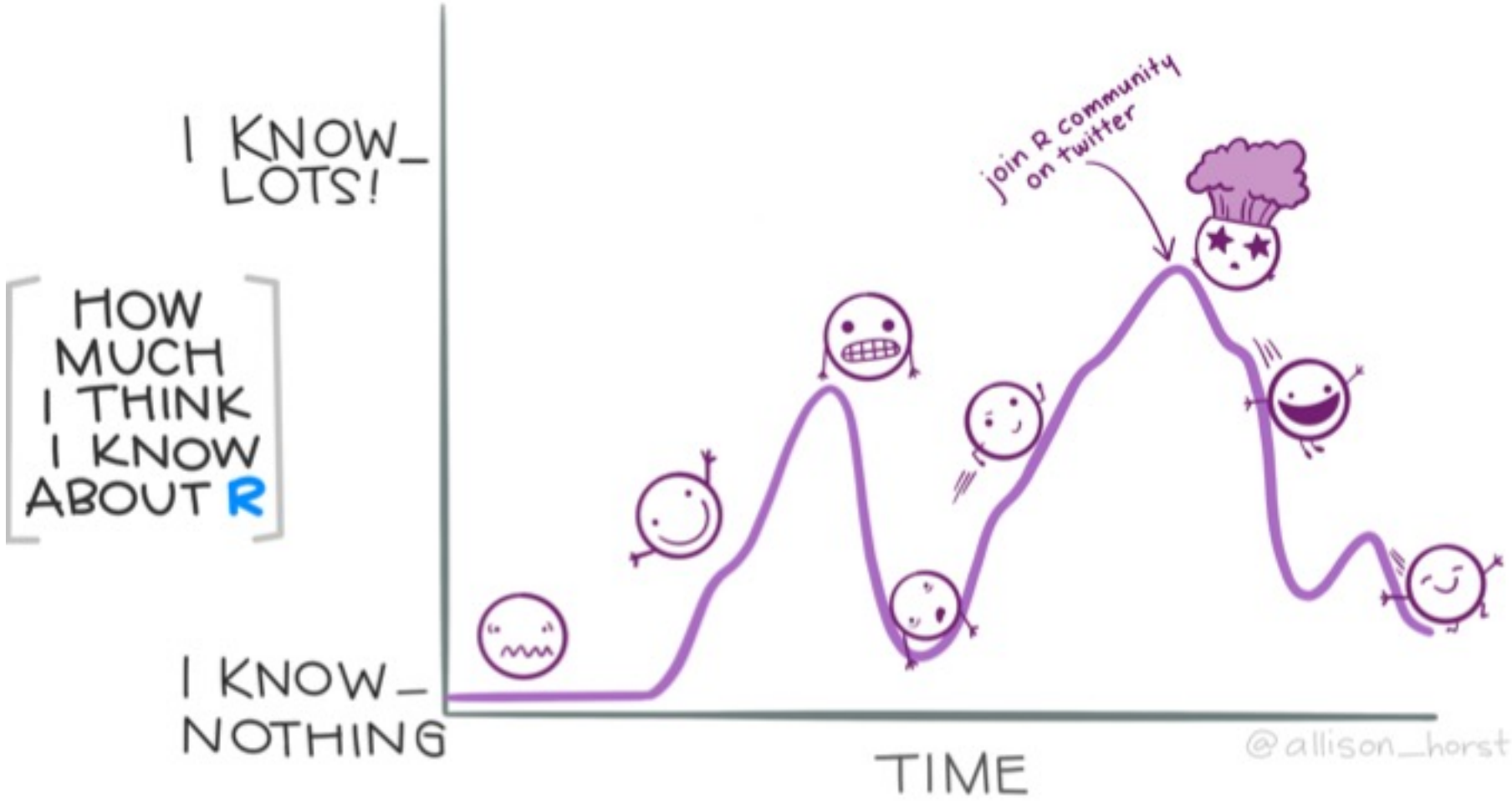
RMarkdown

- Markdown is a lightweight markup language for creating formatted text using a plain-text editor.
- RMarkdown is an extension of the markdown syntax that enables R code to be embedded in them in a way which can later be executed.
- Why do we want this? Typesetting.

RStudio



Learning curve



Conclusion

- R comes from the rock-stars of the computer science industry.
- R is primarily command line based.
- R is extremely powerful, versatile and popular.
- R is free and Open Source.
- Plenty of tools and community around R.
- It is arguably one of the best transferable skills you can learn.
- Supports reproducible academic research.

Questions

Justin van Dijk

j.t.vandijk@ucl.ac.uk

Conclusion

Conclusion