

Supported by:



Federal Ministry
for the Environment, Nature Conservation
and Nuclear Safety



Federal Agency for
Nature Conservation



EOCap4Africa

5 Introduction to R and QGIS

c) Introduction to RStudio



Learning Objectives



Understand the RStudio interface and its key components

Learn basic computer science concepts (data types, functions, libraries)

Execute simple commands in R

Install and load essential libraries for geospatial analysis

What is R?



R is a programming language for data analysis and statistics

Why is R great for geospatial data?

- Open-source and widely used in research
- Supports large-scale data processing
- Integrates with GIS tools like QGIS and Google Earth Engine
- Provides specialized spatial libraries (e.g., sf, terra, ggplot2)



Starting RStudio for the first time



Simply double click on your RStudio application
to begin your programming journey!



The image shows a screenshot of the RStudio interface with several annotations. At the top, a dark blue box labeled "Menu bars" points to the menu bar (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help) and the toolbar. The main editor window contains R code for creating a world map with ggplot2. A dark blue box labeled "Code" points to the code editor. Below the code editor is the Console window, which shows the execution of the code. A dark blue box labeled "Console" points to the console output, stating "code is executed here or error messages are displayed". On the right side of the interface, there are several panels: Environment, History, Connections, Build, and Tools. A dark blue box labeled "Environment" points to the Environment panel, stating "Environment shows all variables you have". Below the Environment panel is the Plots panel, which shows a world map. A dark blue box labeled "Plot view" points to the plot, stating "Plot view see your data visualised here".

Menu bars

```
24 # Check the structure of the loaded data
25 print(countries) # Displays the first few rows and geometry information
26 summary(countries) # Summary of attributes and geometry types
27
28 # 3. Visualize the data: Create a basic map
29 # Use ggplot2 to create a simple map of the countries
30 ggplot(data = countries) +
31   geom_sf(fill = "lightblue", size = 0.2) +
32   labs(title = "World Map", subtitle = "Visualization of vector data") +
33   theme_minimal()
34
35 # 4. Explore the data
36 # Access attribute data
37 head(countries) # Inspect the first few rows of attribute data
38
39 # Select and plot specific attributes
40 # Example: Highlight a single country (e.g., Brazil)
41 brazil <- countries[countries$NAME == "Brazil", ]
42 ggplot(data = countries) +
43   geom_sf(fill = "gray90", color = "black", size = 0.2) +
44
```

Code

Environment shows all variables you have

Plot view see your data visualised here

Console
code is executed here or error messages are displayed

Data Types in R



Type	Example	Description
Character (Text)	"Hello, world!"	Stores words and letters
Numeric (Numbers)	3.14	Stores decimal or whole numbers
Logical (Boolean)	TRUE, FALSE	Represents true/false conditions
Vector	c(1, 2, 3, 4)	A list of numbers or text

Variables in R



Variables: Store values for later use

```

my_variable <- 10
my_other_variable <- 11

my_variable + my_other_variable
|
  
```

Navigate to a line of code and press ctrl + ENTER to execute your code

Results are displayed in the console at the bottom

```

> my_variable <- 10
> my_other_variable <- 11
> my_variable + my_other_variable
[1] 21
> |
  
```



What are libraries and why do they matter?

What is a library (or Package) in R?

- A library (also called a package) is a collection of pre-written functions and datasets that extend R's capabilities
- Libraries allow users to perform specialized tasks without writing all the code from scratch

Why do we need libraries?

- R's base functions are limited – Libraries add new tools for specific tasks
- Saves time – Instead of writing complex code, we can use existing functions
- Community-driven – Thousands of open-source libraries are available, maintained by experts
- Essential for geospatial analysis – R does not natively support GIS, so libraries like sf and raster are required



What are libraries and why do they matter?

Install the package first

Then you can load the library in to use its functions

```

# install.packages is only needed when you use a library for the first time
install.packages("ggplot2")

# Loads the installed library
library(ggplot2)
  
```



What are libraries and why do they matter?

Want to install multiple packages at once?

Create a vector using "c"!

```

# 1. Install and load necessary packages
# First, we need to install some packages that help us load and display visual data
install.packages(c("sf", "ggplot2"))

# Load the libraries
library(sf)           # For handling spatial data
library(ggplot2)     # For visualization, from maps to plots
  
```



Data structures in R

Data Structure	Definition	Example use case
Vector	Ordered list of same data type	A list of temperatures: c(25, 27, 30, 29)
List	Ordered collection of different data types	Storing mixed values: list("tree", 45, TRUE)
Matrix	2D table with same data type	Storing image pixel values
Data Frame	2D table with mixed data types	Storing survey data (name, age, city)

For example:

```

# Creating the Dataframe
df <- data.frame(
  Name = c("Alice", "Bob", "Charlie"),
  Age = c(25, 30, 35),
  City = c("New York", "London", "Paris")
)

# Checking the dataframe
print(df)
  
```



Tasks

1. Start the RStudio Script of this Lecture and follow the Instructions
 1. Assign your own variables and use them for more complicated calculations
 1. Create your own dataframe with any data you like



Common errors in RStudio

“Function not found” error? → Make sure the package is installed and loaded (install.packages("package_name"))

“Object not found” error? → Check variable names and assignments

RStudio crashing? → Restart the session (Session → Restart R)

Accidentally deleted code? → Use the History tab to recover past commands

Summary & key takeaways



RStudio is the primary interface for writing and executing R scripts

Functions are essential for performing specific tasks and improving efficiency

Data structures like vectors, lists, matrices, and data frames organize information

Libraries extend R's capabilities, especially for spatial data analysis

Supported by:



Federal Ministry
for the Environment, Nature Conservation
and Nuclear Safety



Federal Agency for
Nature Conservation



Thank you for your attention!

Dr. Insa Otte, Hanna Schulten
(on behalf of the EOCap4Africa Team)
and colleagues

insa.otte@uni-wuerzburg.de

