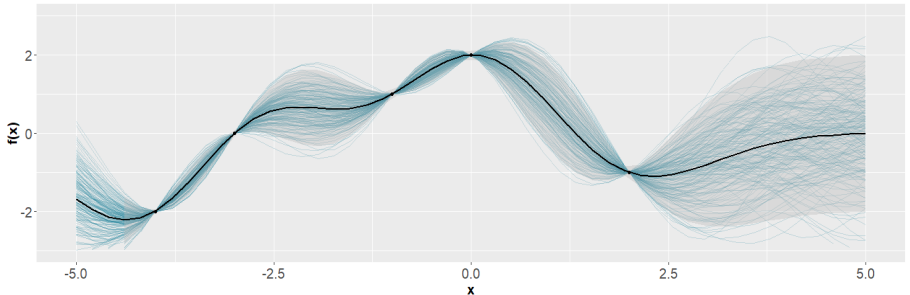


Geovisualization

Lab1: Introduction to R

Joaquin Cavieres

Geoinformatics, Bayreuth University



Outline

- 1 Introduction
- 2 What is R?
- 3 Installing R and RStudio
- 4 What can we do with R?
- 5 Simple commands
- 6 Creating objects
- 7 Types of data

Introduction

Introductory books:

- An Introduction to R (R Development Core Team) [▶ Link](#)
- Yet another R Introduction (Andreas Handel) [▶ Link](#)

What is R?

R is an open source software to allows us:

- Data summarization
- Data cleaning
- Data visualization
- Statistical analysis
- Make graphs
- Statistical models
- Machine learning models
- etc,...

It is in constant improvement due to the contribution of many authors of packages (libraries).

R also allows us to use/create.

- C++ via (Rcpp)
- Personal pages (Blogdown)
- APIs (e.g. plumber)
- Dashboard (Shiny)
- Documents (Rmarkdown)
- Javascript
- etc...

And now we have Quarto! [▶ Link](#)

Let's to work!



Important links for R

- <http://www.r-project.org> ▶ Link
- Download R: <http://cran.r-project.org> ▶ Link
- Download Rstudio: <https://posit.co/download/rstudio-desktop/> ▶ Link

Installing R and RStudio

Installing R

- Go to R project
- Select the link to download R under the Getting Started section
- Select a CRAN mirror in a country closest to you (they are all copies of the same CRAN server)
- Select the R download for your operating system (Windows/Mac/Linux)
- Download the most recent version of base R

Installing RStudio

- Go to RStudio
- In the menu, go to Products > RStudio
- Select download RStudio Desktop
- Select Download for RStudio desktop (free) and select the download for you operating system.

RStudio is an interface that provides features to make using and managing R much easier.

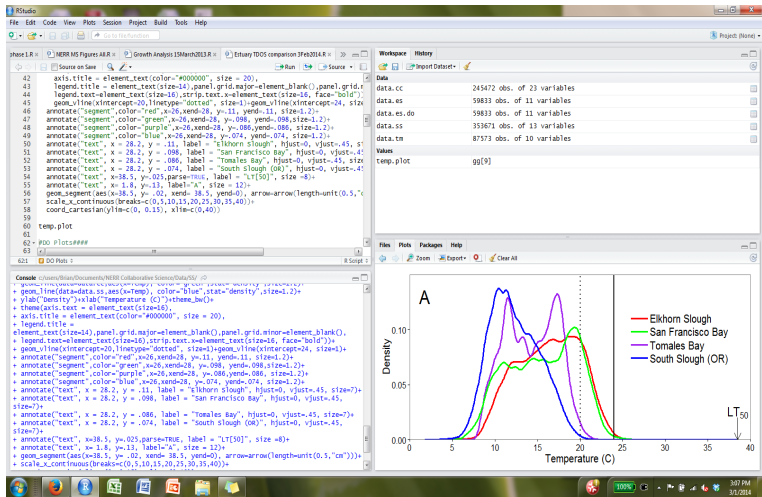


Figure: Rstudio software (picture taken from Internet)

In the interface of Rstudio you can find

- Workspace/environment tab which tells you what objects are saved and what exists in memory
- History tab which shows previous commands you have run.

What can we do with R?

High quality plots

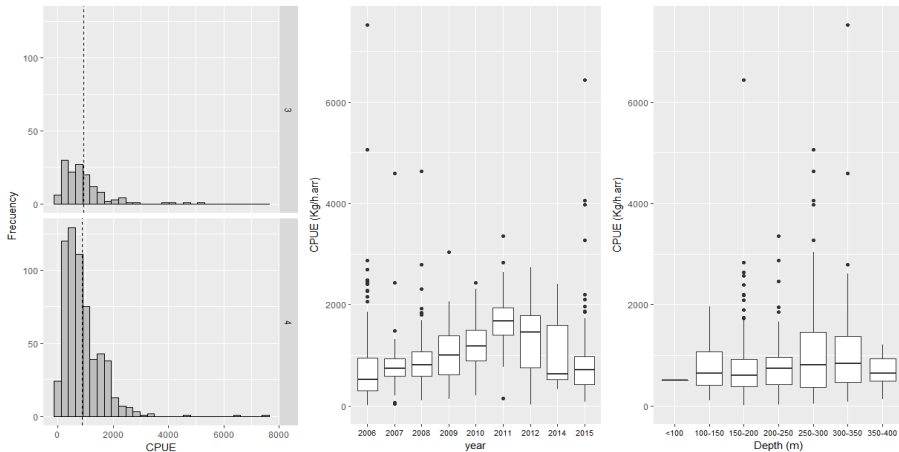


Figure: Histogram and box-plot.

Maps for spatial data

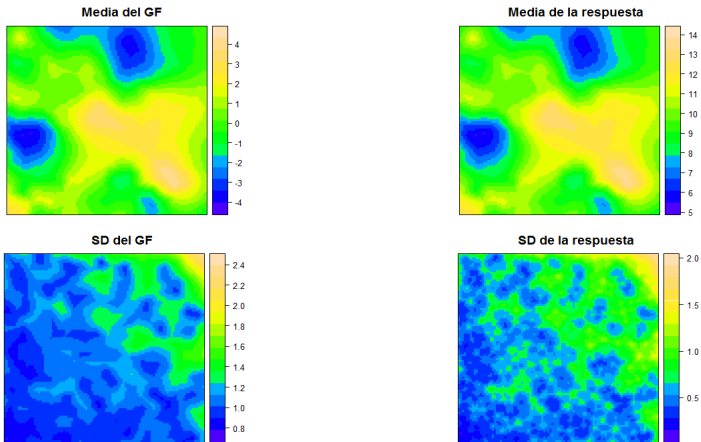


Figure: Mean and sd for a spatial random field.

Important shortcuts in RStudio

- Ctrl+Enter (or CMD+Enter on Mac) will run the current line of code in an R script (the same as copying and pasting the code from your script to the R console).
- Ctrl+1 take you to the script page
- Ctrl+2 takes you to the console

You can check other shortcuts of Rstudio in the next page: [▶ Link](#)

Help?

In R we have a special command to ask about some type of instruction:

- `help(log)` [▶ Link](#)
- `help(glm)` [▶ Link](#)

Simple commands

- The instruction “#” allow us “comment” lines and in this way it does not read.
- If we want to write an algorithm, then:
 - Use correct names
 - Don't use repeated lines in the code
 - Don't assume that the names should be remembered in the next lines of code

$> 2 + 2$

[1] 4

$> 2^2$

[1] 4

$> 2 * (1 + 1)$

[1] 4

Examples

Write in R and calculates:

```
> 1 + 2(3 + 4)
```

Examples

Write in R and calculates:

$$> 1 + 2(3 + 4)$$

$$> \log(4^3 + 3^{2+1})$$

Examples

Write in R and calculates:

$$> 1 + 2(3 + 4)$$

$$> \log(4^3 + 3^{2+1})$$

$$> \sqrt{(4 + 3)(2 + 1)}$$

Examples

Write in R and calculates:

$$> 1 + 2(3 + 4)$$

$$> \log(4^3 + 3^{2+1})$$

$$> \sqrt{(4 + 3)(2 + 1)}$$

$$> \left(\frac{1+2}{2+4}\right)^2$$

Creating objects

- Each result of any mathematical operation can be saved in a "object".
 - Numbers
 - Characters
 - Tables
 - Vectors/Matrices
 - Graphs
 - Statistical model
 - Etc...

We can **assign** a specific value for a variable, for example:

```
variable <- x
```

The “variable” have now the value of x.

but also we can assign with the sign “=”, for example:

```
variable = x
```

Also we can assign characters to a variable:

```
name <- "John"
```

To "name" we assigned the character "John".

If we want to assign a name with a space, we have to do:

```
name <- "John Cavieres"
```


Seeing the created objects

There are different ways to visualise the created objects:

```
print(name)
```

```
[1] John Cavieres.
```

```
name
```

```
[1] John Cavieres.
```

If we want to manipulate an object we can do: `x <- 2`

```
x*2
```

```
[1] 4.
```

Removing created objects

We can write `list()` to see the created objects

```
ls()
```

Removing created objects

We can write `list()` to see the created objects

```
ls()
```

```
[1] "name" "variable"
```

Removing created objects

We can write `list()` to see the created objects

```
ls()
```

```
[1] "name" "variable"
```

To remove a particular object we write `rm()`, for example, if we want to remove the object “name”, we write:

```
rm(name)
```

Removing created objects

We can write `list()` to see the created objects

```
ls()
```

```
[1] "name" "variable"
```

To remove a particular object we write `rm()`, for example, if we want to remove the object “name”, we write:

```
rm(name)
```

```
ls()
```

Removing created objects

We can write `list()` to see the created objects

```
ls()
```

```
[1] "name" "variable"
```

To remove a particular object we write `rm()`, for example, if we want to remove the object “name”, we write:

```
rm(name)
```

```
ls()
```

```
[1] "variable"
```

If we want to remove all the created objects `rm(list=ls())`

If we want to remove all the created objects `rm(list=ls())`

Review if there are more objects with `list()`...

Types of data

There are different types of data to be used in R and they differ in the way that are saved in the computer.

- Numeric (integer, floating point, etc)
- Logic (boolean, true/false)
- Characters(text)

The type of data is not obvious, furthermore when they are read from external sources, for the same is necessary to know what type the data are.

```
>variable2 <- 2
```

```
>variable2
```

```
[1] 2.
```

```
>mode(variable2)
```

```
[1] "numeric".
```

```
>is.numeric(variable2)
```

```
[1] "TRUE".
```

Similar functions can be applied to character objects:

```
>is.character(variable2)
```

```
[1] "FALSE".
```

```
>is.character(name)
```

```
[1] "TRUE".
```

The class “numeric” and “character” are the most commonly found in the declaration of objects.

See you next class!...



Lamigueiro, O. P. (2014). Displaying time series, spatial, and space-time data with R. CRC Press.