Module Phenology - Session 1

Geoprocessing III - 2025

Spatio-temporal applications in ecology using R

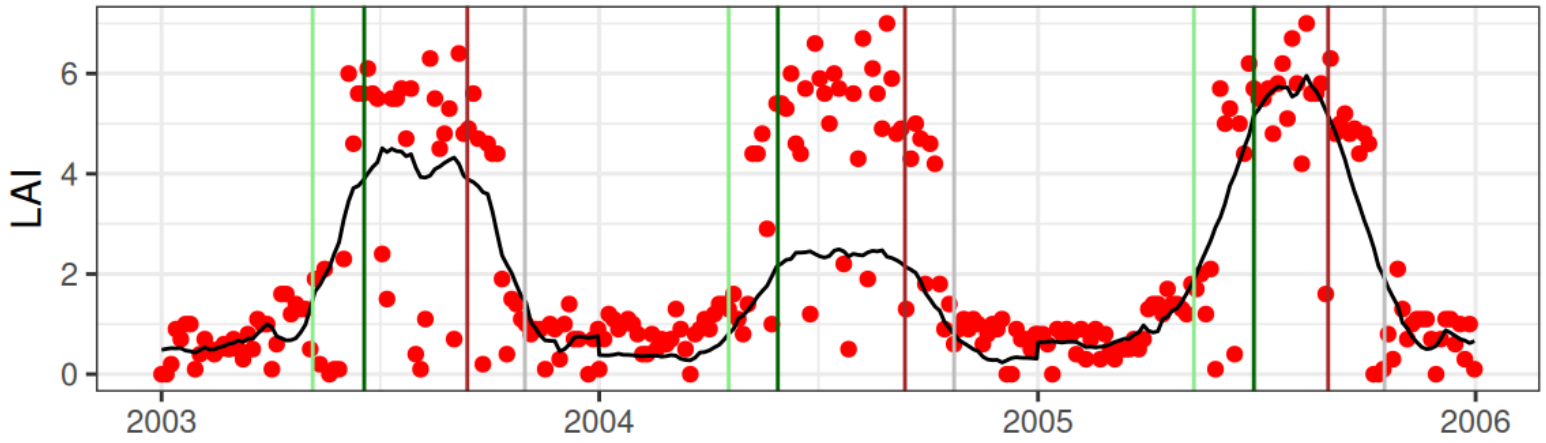
Fabian Bernhard & Koen Hufkens

2025-05-15

Phenology

Split image of canopy change between winter and spring/summer at Harvard Forest, Petersham, MA, USA

Phenology algorithms



Sesonal evolution of leaf area index (LAI) and detected phenology

Module 6 - Phenology trends and algorithms

- course material at: https://fabern.github.io/handfull_of_pixels/
  - additional content & exercises: https://geco-bern.github.io/handfull_of_pixels/
- exercises: chapter 7.1 Phenology trends and algorithms
- exercise: work alone or in teams of two
- hand-in: **by June 5th, 2025** (in ILIAS)
- hand-in: as a zip file containing a reproducible copy of your R project folder
  - Rmd files with your analysis and responses to the exercise questions
  - as well as rendered ("knitted") html-files of your Rmd-files
  - including other data and source code (if needed)

Module 6 - These sessions (15/22.05.2025)

- R and R markdown based
- all accessible / when reading the chapters in class + brief intro
- here to teach, not to grade

Online material

```
https://fabern.github.io/handfull_of_pixels/
```

- 1 Crash course R
- 2 Accessing data
- 3 Geospatial data in R
- 4 Phenology trends
- 5 Phenology algorithms
- (6 Phenology modelling)
- (7 Landcover classification)
- 8 Exercises

Spatial processing in R

- World is data rich
- Finding the right data is key
- Objectives
    - Finding and using data
    - Getting data
    - Spatial processing in R

Finding & using data

- Identify how to **get the data**
- Understand the format (extension)
- **Confirm the format**
  - meta-data
  - headers

Getting data

- Direct downloads
    - manually
    - not always structured
- Application Programming Interfaces (APIs)
    - automatic

```
library("MODISTools") # load the library for use in this script

products <- MODISTools::mt_products() # list all available products

data <- MODISTools::mt_subset(product = "MOD11A2", ...) # download one data product
```

Spatial processing in R

- Focus on two main tools
- raster data `terra`
- vector data `sf`

Only raster data processing will be discussed in depth.

Spatial processing in R: package setup

```
# Installing the required packages
# (additional software modules)
install.packages("terra")
install.packages("sf")
install.packages("dplyr")
install.packages("ggplot2")
install.packages("patchwork")
...
install.packages("MODISTools")
```

- already done for you
- on personal laptop: see https://fabern.github.io/handfull_of_pixels/appendix_setup.html
    - for Geoprocessing-III you don't need last block of packages (`phenocamr` to `xgboos`)

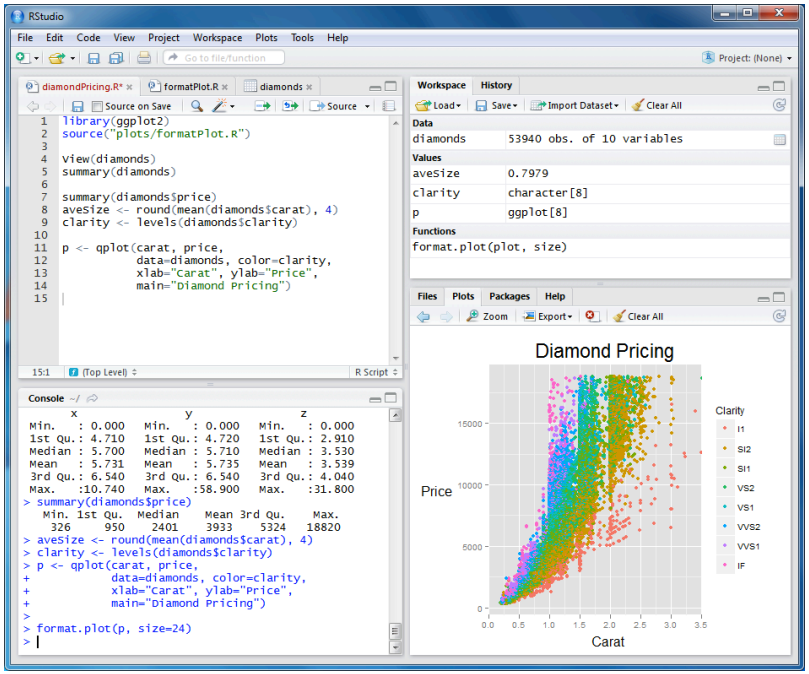
Spatial processing in R: the basics

```r
# load the library
library("terra")
filepath <- system.file("ex/elev.tif", package="terra")
elev <- terra::rast(filepath)

# show the resulting object
elev
```

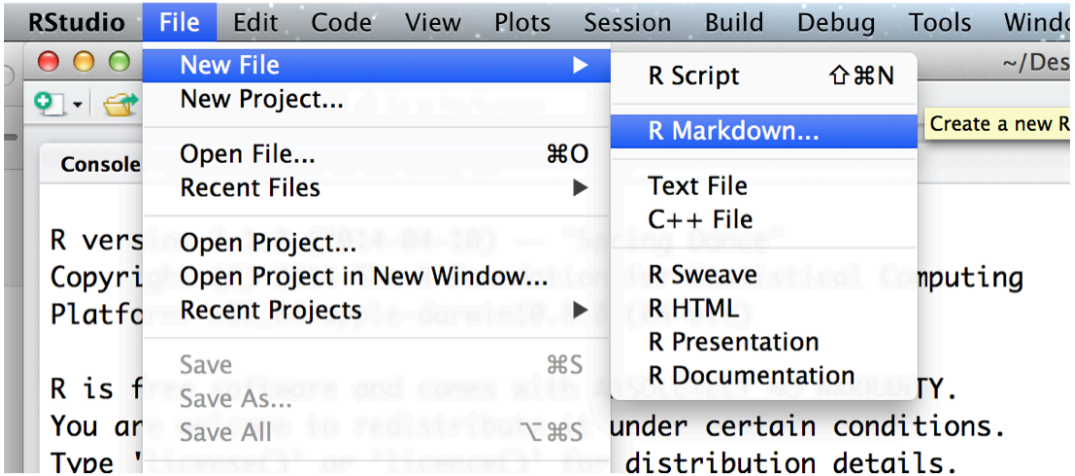
```
## class       : SpatRaster
## dimensions  : 90, 95, 1  (nrow, ncol, nlyr)
## resolution  : 0.008333333, 0.008333333  (x, y)
## extent      : 5.741667, 6.533333, 49.44167, 50.19167  (xmin, xmax, ymin, ymax)
## coord. ref. : lon/lat WGS 84 (EPSG:4326)
## source      : elev.tif
## name        : elevation
## min value   :       141
## max value   :       547
```

Crash course R/RStudio

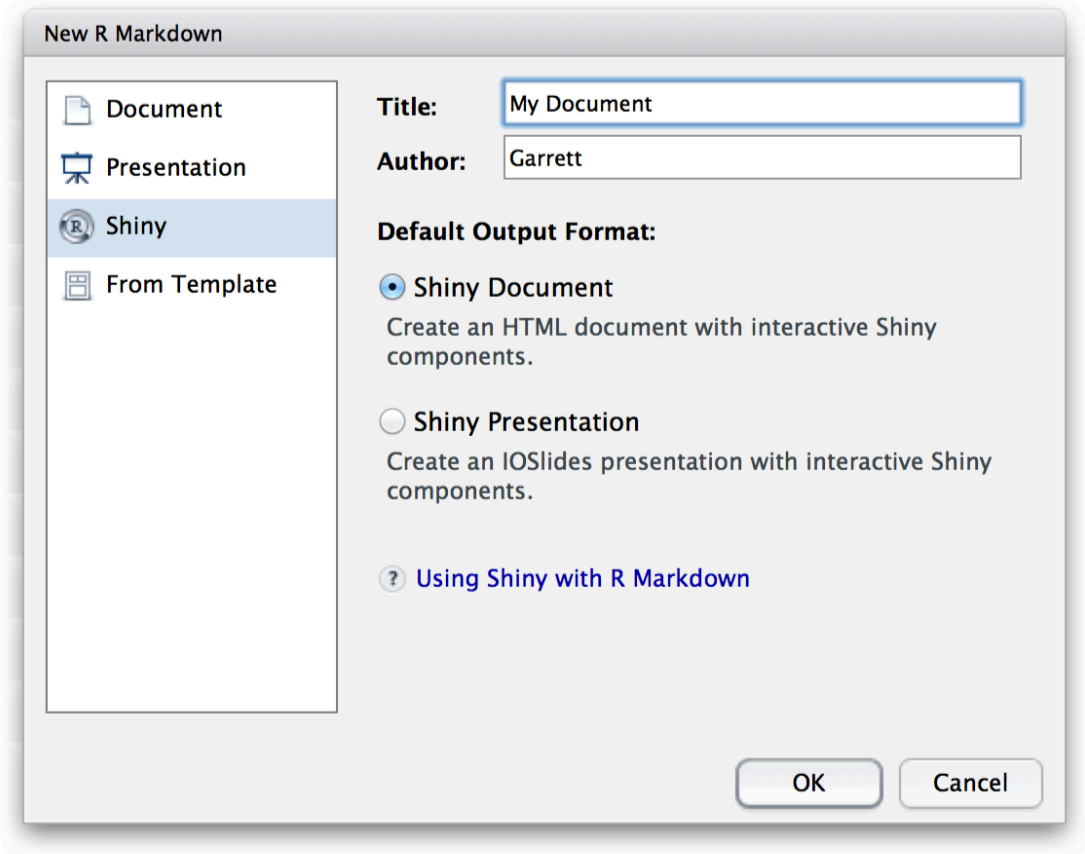


- RStudio based (Integrated Development Environment)
- code heavy (focus on automation / computation)
- parallels with python

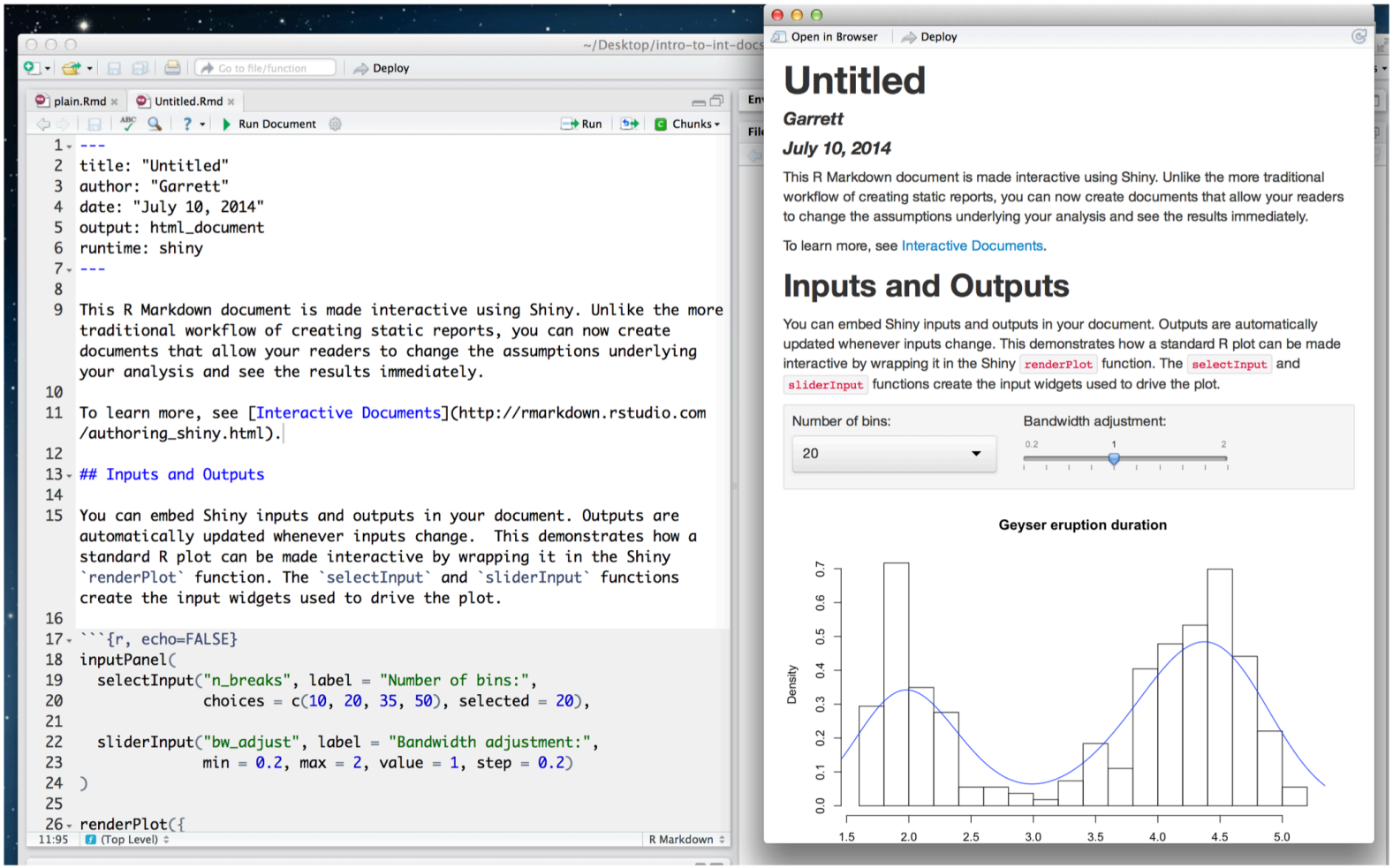
R markdown


RStudio
File
Edit
Code
View
Plots
Session
Build
Debug
Tools
New File
New Project...
Open File... ⌘O
Recent Files
Open Project...
Open Project in New Window...
Recent Projects
Save ⌘S
Save As...
Save All ⌥⌘S
R Script ⇧⌘N
R Markdown...
Text File
C++ File
R Sweave
R HTML
R Presentation
R Documentation
Create a new R
Console

R markdown


New R Markdown
Document
Presentation
Shiny
From Template
Title:
My Document
Author:
Garrett
Default Output Format:
Shiny Document
Create an HTML document with interactive Shiny components.
Shiny Presentation
Create an IOSlides presentation with interactive Shiny components.
Using Shiny with R Markdown
OK
Cancel

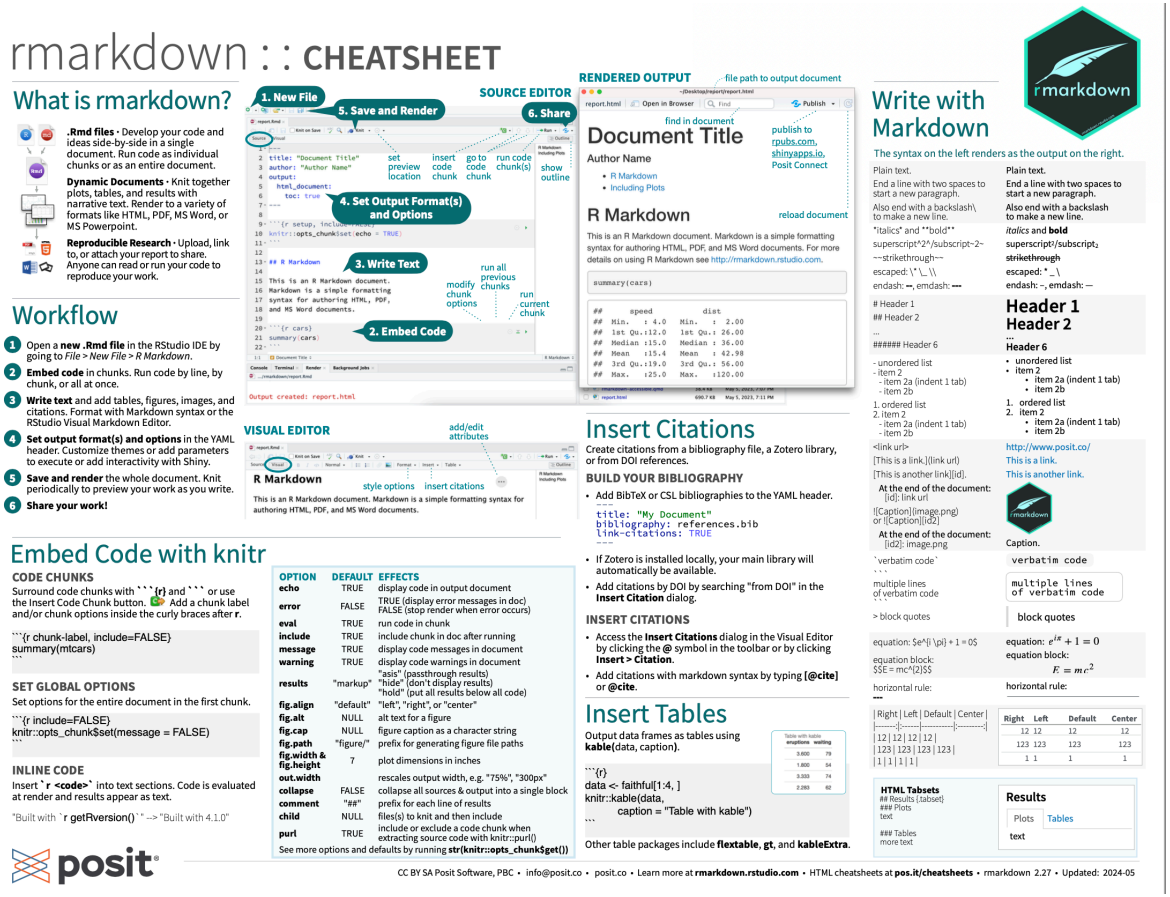
R markdown


~/Desktop/intro-to-int-docs
Deploy
plain.Rmd
Untitled.Rmd
Run Document
Run
Chunks
---
title: "Untitled"
author: "Garrett"
date: "July 10, 2014"
output: html_document
runtime: shiny

This R Markdown document is made interactive using Shiny. Unlike the more traditional workflow of creating static reports, you can now create documents that allow your readers to change the assumptions underlying your analysis and see the results immediately.
To learn more, see [Interactive Documents](http://rmarkdown.rstudio.com/authoring_shiny.html).
Inputs and Outputs
You can embed Shiny inputs and outputs in your document. Outputs are automatically updated whenever inputs change. This demonstrates how a standard R plot can be made interactive by wrapping it in the Shiny `renderPlot` function. The `selectInput` and `sliderInput` functions create the input widgets used to drive the plot.
```{r, echo=FALSE}
inputPanel(
selectInput("n_breaks", label = "Number of bins:",
choices = c(10, 20, 35, 50), selected = 20),
sliderInput("bw_adjust", label = "Bandwidth adjustment:",
min = 0.2, max = 2, value = 1, step = 0.2)
)
renderPlot({
11:95
(Top Level)
R Markdown
Open in Browser
Deploy
Untitled
Garrett
July 10, 2014
This R Markdown document is made interactive using Shiny. Unlike the more traditional workflow of creating static reports, you can now create documents that allow your readers to change the assumptions underlying your analysis and see the results immediately.
To learn more, see Interactive Documents.
Inputs and Outputs
You can embed Shiny inputs and outputs in your document. Outputs are automatically updated whenever inputs change. This demonstrates how a standard R plot can be made interactive by wrapping it in the Shiny renderPlot function. The selectInput and sliderInput functions create the input widgets used to drive the plot.
Number of bins:
20
Bandwidth adjustment:
0.2
1
2
Geyser eruption duration
Density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
```

https://rmarkdown.rstudio.com/

Cheat sheet: https://github.com/rstudio/cheatsheets/raw/main/rmarkdown.pdf



rmarkdown : : CHEATSHEET

What is rmarkdown?

.Rmd files · Develop your code and ideas side-by-side in a single document. Run code as individual chunks or as an entire document.

Dynamic Documents · Knit together plots, tables, and results with narrative text. Render to a variety of formats like HTML, PDF, MS Word, or MS Powerpoint.

Reproducible Research · Upload, link to, or attach your report to share. Anyone can read or run your code to reproduce your work.

Workflow

1. Open a **new .Rmd file** in the RStudio IDE by going to *File > New File > R Markdown*.
2. **Embed code** in chunks. Run code by line, by chunk, or all at once.
3. **Write text** and add tables, figures, images, and citations. Format with Markdown syntax or the RStudio Visual Markdown Editor.
4. **Set output format(s) and options** in the YAML header. Customize themes or add parameters to execute or add interactivity with Shiny.
5. **Save and render** the whole document. Knit periodically to preview your work as you write.
6. **Share your work!**

SOURCE EDITOR

1. New File
5. Save and Render
6. Share
set preview location
insert code chunk
go to code chunk
run code chunk(s)
show outline
4. Set Output Format(s) and Options
3. Write Text
modify chunk options
run all previous chunks
run current chunk
2. Embed Code

RENDERED OUTPUT

file path to output document
find in document
publish to rpubs.com, shinyapps.io, Posit Connect
reload document

Document Title

Author Name

- R Markdown
- Including Plots

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

```
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

VISUAL EDITOR

add/edit attributes
style options
insert citations

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.

Insert Citations

Create citations from a bibliography file, a Zotero library, or from DOI references.

BUILD YOUR BIBLIOGRAPHY

- Add BibTeX or CSL bibliographies to the YAML header.

```
title: "My Document"
bibliography: references.bib
link-citations: TRUE
```

- If Zotero is installed locally, your main library will automatically be available.
- Add citations by DOI by searching "from DOI" in the **Insert Citation** dialog.

INSERT CITATIONS

- Access the **Insert Citations** dialog in the Visual Editor by clicking the **@** symbol in the toolbar or by clicking **Insert > Citation**.
- Add citations with markdown syntax by typing **[@cite]** or **@cite**.

Insert Tables

Output data frames as tables using **kable**(data, caption).

```
{r}
data <- faithful[1:4, ]
knitr::kable(data,
      caption = "Table with kable")
```

Other table packages include **flextable**, **gt**, and **kableExtra**.

Write with Markdown

The syntax on the left renders as the output on the right.

Plain text. End a line with two spaces to start a new paragraph. Also end with a backslash\ to make a new line.	Plain text. End a line with two spaces to start a new paragraph. Also end with a backslash to make a new line.
italics and **bold** superscript^2^/subscript~2~ ~~strikethrough~~ escaped: * _ \\ endash: --, emdash: ---	*italics* and **bold** superscript^2/subscript$_2$ ~~strikethrough~~ escaped: * _ \ endash: –, emdash: —
# Header 1 ## Header 2 ... ###### Header 6	**Header 1** **Header 2** ... **Header 6**
- unordered list - item 2 - item 2a (indent 1 tab) - item 2b 1. ordered list 2. item 2 - item 2a (indent 1 tab) - item 2b	• unordered list • item 2 • item 2a (indent 1 tab) • item 2b 1. ordered list 2. item 2 • item 2a (indent 1 tab) • item 2b
<link url> [This is a link.](link url) [This is another link][id]. At the end of the document: [id]: link url	http://www.posit.co/ This is a link. This is another link.
![Caption](image.png) or ![Caption][id2] At the end of the document: [id2]: image.png	Caption.
`verbatim code` ``` multiple lines of verbatim code ```	`verbatim code` `multiple lines of verbatim code`
> block quotes	block quotes
equation: $e^{i \pi} + 1 = 0$	equation: $e^{i\pi} + 1 = 0$
equation block: $$E = mc^{2}$$	equation block: $$E = mc^2$$
horizontal rule: ---	horizontal rule:

Right	Left	Default	Center
12	12	12	12
123	123	123	123
1	1	1	1

HTML Tabsets
Results {.tabset}
Plots
text
Tables
more text

Results

Plots | Tables

text

Embed Code with knitr

CODE CHUNKS

Surround code chunks with ```` ```{r} ```` and ```` ``` ```` or use the Insert Code Chunk button. Add a chunk label and/or chunk options inside the curly braces after **r**.

```
```{r chunk-label, include=FALSE}
summary(mtcars)
```
```

SET GLOBAL OPTIONS

Set options for the entire document in the first chunk.

```
```{r include=FALSE}
knitr::opts_chunk$set(message = FALSE)
```
```

INLINE CODE

Insert **`r <code>`** into text sections. Code is evaluated at render and results appear as text.

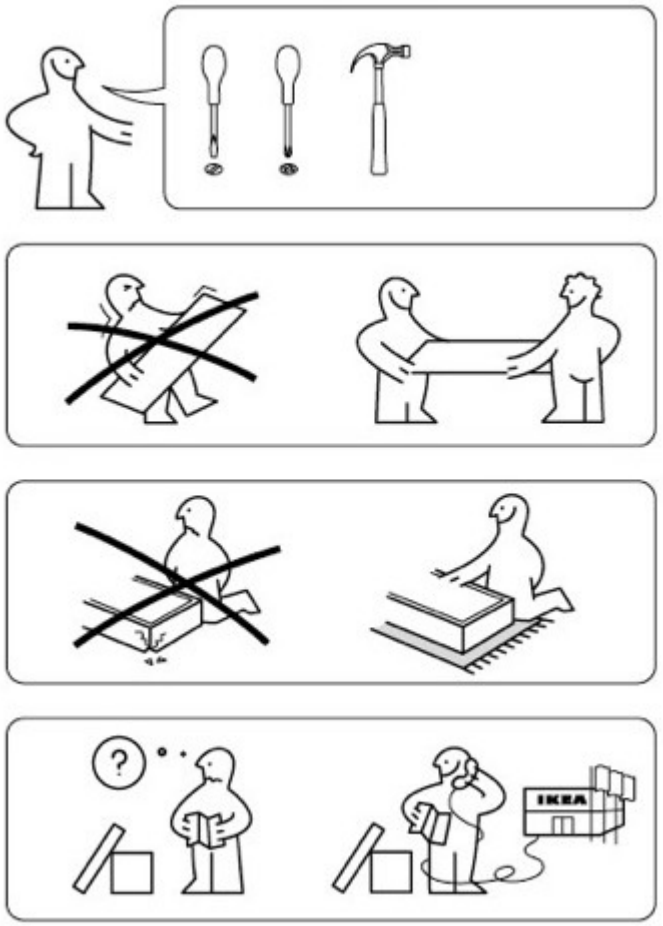
"Built with `r getRversion()`" --> "Built with 4.1.0"

OPTION	DEFAULT	EFFECTS
echo	TRUE	display code in output document
error	FALSE	TRUE (display error messages in doc) FALSE (stop render when error occurs)
eval	TRUE	run code in chunk
include	TRUE	include chunk in doc after running
message	TRUE	display code messages in document
warning	TRUE	display code warnings in document
results	"markup"	"asis" (passthrough results) "hide" (don't display results) "hold" (put all results below all code)
fig.align	"default"	"left", "right", or "center"
fig.alt	NULL	alt text for a figure
fig.cap	NULL	figure caption as a character string
fig.path	"figure/"	prefix for generating figure file paths
fig.width & fig.height	7	plot dimensions in inches
out.width		rescales output width, e.g. "75%", "300px"
collapse	FALSE	collapse all sources & output into a single block
comment	"##"	prefix for each line of results
child	NULL	files(s) to knit and then include
purl	TRUE	include or exclude a code chunk when extracting source code with knitr::purl()

See more options and defaults by running **str(knitr::opts_chunk$get())**

posit

Reporting



- Read the documentation/course
- Motivate the use of particular data and or methods
- Make notes on potential limitations
- Report in Rmarkdown (rendered html)
- Ask for help during the sessions

Spatial processing in R: project setup

```
# Download the project zip file
https://github.com/geco-bern/R_proj_template/archive/refs/heads/main.zip
```

- Download and unzip the file
- Rename and open the Rproj file (opens RStudio)
- Place / save data in `data/`
- Place R markdown scripts for report in `vignettes/`
- Place e.g. copy-pasted R code from tutorial `analysis/trying_out.R`
- Let's get started: https://fabern.github.io/handfull_of_pixels/

Spatial processing in R: package setup (fixing missing admin rights)

```r
# Installing the required packages
install.packages("terra")
# ....
install.packages("here")
# This does not work on lab computers, since you do not have the rights.
# In that case, you must install packages into a personal folder. (see below)
```

```r
# Setup your personal package library (needs to be done in each session again)
.libPaths()
# [1] "C:/Program Files/R/R-4.5.0/library"

dir.create("C:\\Users\\fb24k097\\Documents\\R2")      # create a folder
.libPaths(new = "C:\\Users\\fb24k097\\Documents\\R")  # set the personal folder
.libPaths()
# [1] "C:/Users/fb24k097/Documents/R"
# [2] "C:/Program Files/R/R-4.5.0/library"
```