

Biodiversity Summer School: Pre-course Installation and downloading of data

Here are the instructions for downloading software and data for the Biodiversity Summer School.

General aims of practicals

These practicals aim to teach students who already have some experience of using R how to manage, model, and analyse biodiversity using the tidyverse group of packages and selected other packages in the Rstudio Integrated Development Environment (IDE).

For students who are new to R

For students who are new to R and especially new to programming in general, we recommend you make use of the resources at the end of this document to familiarise yourself with some of the basics in R. Start with [R Introduction to Selected Topics](#).

Getting set up

You must be set up before the course begins. Please work through these instructions to download data and software and install the software.

Help available

In case you encounter difficulties, volunteers will be available to assist you set up on the 08 May from 10 am – noon and 2 pm – 4 pm and 09 May, 2021 from 2 pm – 4 pm. All times are in CET = GMT+2.

Installing QGIS and registering on GEE

To set up for the section of the course on spatial biodiversity data, lectured by Dr Sandra MacFadyen, follow instructions given [here](#).

Be careful to install [QGIS version 3.18.2](#), register on [Google Earth Engine](#) (GEE) and [install the required plugins and look at the listed resources](#)

Installing R and Rstudio

If you have never installed R or Rstudio before

1. Follow [this link](#) to download the latest version of R for your operating system.
2. Install R by opening the installer and following the steps
3. Launch R to check it has installed correctly
4. After you have downloaded and installed R, follow [this link](#) to download the latest version of Rstudio for your operating system.
5. Install Rstudio by opening the installer and following the steps
6. Launch Rstudio to check it has installed correctly.

If you already have R and Rstudio installed updated them by following instructions on the webpage below. Also, update any packages you have installed ([instructions also on this webpage](#)).

Do this in the following order.

1. Update R
2. Update R-studio
3. Update your packages (I recommend you do this from Rstudio)

Learning to work in Rstudio

If you are not used to working in Rstudio, please read the following helpful [introduction](#)

Projects, scripts, markdowns, notebooks

To work in an organised way, you need to keep all your files together, including data files, scripts, outputs, and even notebooks (more about notebooks shortly).

In Rstudio, click **File -> New Project**

If you have been working in R already, you will be asked to save your work up to this point before R creates the new project.

You can then create a new folder or choose an existing folder to contain all these files. This folder will become the working directory for this project.

Ensure that when you download your data, it goes into the correct project folder. It is fine to have subfolders within this project folder.

Now you are ready to start some work. You can execute code directly from the console, but this is not a good place to do work on writing code, so you will want to make a file in which to write your analysis. The simplest way to do this is in a script file.

In Rstudio, click **File -> New File -> R Script**

You can write your code directly in the script file and annotate it with comments. “#” is used to denote comments. Anything that comes after # is treated by R as a comment and it does not try to run it

For example, you copy the following into a script and run it. To run code line by line, place the cursor on the first line and repeatedly click run from the bar at the top of the script tab (or with Ctrl+Enter enter from your keyboard). You can also highlight a whole section of code and run it.

```
#This is an example comment
x <- c(3, 5, 9) #this is an example of assignment
y <- c(1,2,3)
x #this makes R print the value assigned to x
plot(y~x) #this is to show you where the plots go
```

You can see that that output appears with the code in the console, except that figures go to the figures tab in the bottom right pane.

Alternatively, instead of writing in a script, you can use a notebook. An R notebook can be seen as an interactive document containing sections of code (called “chunks”), where output, including figures, appears below each for each chunk

To create a new R notebook

In Rstudio, click **File -> New File -> R Notebook**

Read the instructions given in your new notebook. These instructions show how notebooks work, as well as telling you how they work.

In your R notebook, the grey sections are code chunks in which you write the code you want to run. The white sections can be used for headings and other text. Output will appear underneath each chunk. Create a new chunk by clicking

Insert -> R

From the bar at the top of the notebook. For more advanced users, it is possible to directly insert chunks of code in other languages too, such as Python.

Paste the same code given above into your new code chunk, and then run it by clicking on the run button (the green triangle) in the top right-hand corner of the chunk.

Now you can see output below the chunk, including textual output

```
[1] 3 5 9
```

and the plot.

You can hopefully see how convenient it is to use R notebooks to write your code and check the results. When you work in an R notebook, the code and results are automatically used to create a preview file with the extension *.nb.html*. You can use these to share results with collaborators who don't use R. Be careful though, this is not "rendered". That means that R does not run the code again to make the preview nb.html file. If you have edited code chunks and not run them again, and made mistakes, you may not realise it. R notebooks are great for work in progress, but R markdowns are often better for sharing results

R markdown

To create a new R Markdown

File -> New File -> R Markdown

R Markdowns are very similar to notebooks (the notebook can be seen as a special kind of markdown). The R markdown is there to create documents for sharing your code and results. Once again, paste the example code from above into a code chunk.

To run all the code in the markdown, you can click

Run -> Run All

Now to create a document to share (i.e. a report), from the bar at the top of the markdown file, click

Knit -> Knit to html

This tells Rstudio to run all the code in the markdown and create a formatted html document. If any line throws an error, this process will fail. You will see that there is formatting in the report (there was also formatting in the notebook preview). Use the [R markdown cheat sheet](#) to learn how to format.

To summarise... You should be writing your code either in R scripts or in R notebooks. You should share your results or progress using R notebooks or R Markdowns (or if you use latex, you will probably want to use Sweave).

Installing packages

Packages provide specialised functions not included in the base version of R.

You can install packages from the menu in Rstudio

Tools -> Install packages

And then entering individual package names.

For this summer school, you should install the following packages:

tidyverse, vegan, zetadiv, popdemo, popbio, bipartite, dismo, maptools, rgdal, raster, sp, NetworkExtinction, meta

tidyverse is not a single package but a set of packages. By asking R to install “tidyverse” you will automatically get all of these packages installed.

Download the data

These instructions for downloading the data will ensure that the code I supply to import your data into Rstudio works on your computer.

Download data and unzip (extract) the data as follows if you are a Windows User.

Data to download

Create a project and download these three datasets into it.

In Rstudio, click **File -> New Project**

[Tree survey data](#) for the Barro Colorado 10ha tree plot

[Tree survey data](#) for multiple tree plots in Panama

[Trait data](#) for the Barro Colorado 10ha tree plot

Create a second project and download these two datasets into it.

[Compadre](#) plant demography dataset

[Comadre](#) animal demography dataset

Remember to unzip data after downloading it!

Loading data in R

Create a new notebook or script in your project using the instructions above. Load the dplyr package and the data using this code.

```
library(dplyr)

load("../doi_10.15146_mdpr-
pm59__v1/marena_tree_17Jul2019/marena1cns.tree1.rdata")
```

If it works, the data will appear in the environment pane as an object called "marena1cns.tree1". If it doesn't work, you will get an error message.

If it does not work, try to replace the text inside the quotations with the full path to the file.

R Resources

[R Introduction to Selected Topics](#), an overview of base R, the tidyverse, and data management

[The pirates guide to R](#), a humorous guide to base R

[R for data science](#), a guide by Hadley Wickham, the main author of the tidyverse.

[R studio 101](#), and introduction to Rstudio.

[R markdown cheat sheet](#)

[A page of links to more cheat sheets](#)

[Keyboard shortcuts](#)