Getting started with R

FW5121 – Jan. 25, 2019

Dee

Tutorial modified from Chris Brown’s R spatial course: https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxzZWFzY2FwZW1vZGVsbGluZ3xneDoxZGNiMWMwZjZkZTI0Nzhk

Obtaining R

R can be downloaded from <http://cran.r‐project.org/>

Using a script editor, such as “RStudio,” can also be helpful. RStudio can be downloaded from

<http://www.rstudio.com/>

Starting RStudio

Click the RStudio icon to open R Studio. The interface is divided into several panels (clockwise

From top left):

1. The source code (supporting tabs)

2. The currently active objects/history

3. A File browser/plot window/package install window/R help window (tabbed)

4. The R console The source code editor (top left) is where you type, edit and save your R code.

The source code editor (top left) is where you type, edit and save your R code. The editor

supports text highlighting, autocompletes common functions and parentheses, and allows the

user to select R code and run it in the R console (bottom right) with a keyboard shortcut (Ctrl R

on windows, command-enter on macs). Code will appear in this font: plot(x,y)

Installing packages

R offers a variety of functions for importing, editing, manipulating, analysing and exporting

spatial data. Most of these functions rely on add-on packages that can be loaded to an R session

using the library(packagename) command. If you close R and restart you will have to load the

packages again.

If your required package is not already installed on your computer, it can be simply installed by

implementing the following command (you must be connected to the internet):

install.packages("maps")

Multiple packages can be loaded at the same time by listing the required spatial packages in a

vector. Here we will install all the packages you will require for today's in class exercise:

install.packages(c("maps”, “rgdal”))

Also a couple of optional packages:

install.packages(c("RColorBrewer", “ggplot2”))

Select the local CRAN mirror (e.g. a location in the US). If you are using a mac, rgdal cannot be installed in this way. See Dr. Chris Brown’s webpage for more help (<https://sites.google.com/site/seascapemodelling/installing-rgdal-on-osx)>

Starting a new script

Let's open a new script and save it to your harddrive.

When writing R scripts, use numerous # comments throughout your R scripts, you will thank

yourself when you go back to the analysis later! Similarly at the start of your code, put some

meta-information, such as:

# who wrote the code?

# what does the code do?

# when did you write the code? etc.

The more projects you work on and analyses you do, the more important it is to have this meta-

information at the beginning of your code.

Load packages

We usually start a script by loading the necessary packages with the library() function:

library(rgdal)

library(maps)

library(ggplot2)

These add-on packages contain everything we need for today. rgdal is R's interpretation of the Geospatial Data Abstraction Library. It contains definitions and translations for projections for rasters and vectors.

Setting your working directory folder so you can read in data to map ranges

Moving on. You will need to set your working directory to the folder where the datafiles for this

course are: setwd('mypath/Datasets') You can get the path by using your folder browser and right clicking and selecting properties, then just copy the path. Note on windows paths will be copied with \ rather than / so make sure you turn them around in your script.

In the script I provided, the example links to a folder where I saved the data for each species that is read in to plot the range maps.

setwd("~/Dropbox/RARE spp MS/Revision1\_rarespp\_TREE/Figures/RareSp\_WorldMap")

You’ll need to change this to the folder where you saved the data that you will want to plot.