Pre-Workshop Handout

Preparing to use R

original by François Michonneau & Auriel Fournier (Data Carpentry)
modified by Ethan Deyle (BU Marine Semester)

Fall 2023

To minimize troubleshooting during the workshop, learners should follow the instruction below to download and install everything beforehand. If they are using their own computers this should be no problem, but if you are using a computer that is managed by the IT department, we might need help from an IT administrator.

Install R and RStudio

R and RStudio are two separate but related pieces of software:

- **R** is a programming language that is especially powerful for data exploration, visualization, and statistical analysis
- **RStudio** is an integrated development environment (IDE) that makes using R easier. In the workshop, we will use RStudio to interact with R.

If you don't already have R and RStudio installed, follow the instructions for your operating system below. You have to install R before you install RStudio.

Windows

- Download R from the CRAN website.
- Run the `.exe` file that was just downloaded
- Go to the RStudio download page
- Under *All Installers*, download the RStudio Installer for Windows.
- Double click the file to install it
- Once it's installed, open RStudio to make sure it works and you don't get any error messages.

MacOS

- Download R from the CRAN website.
- Select the `.pkg` file for the latest R version
- Double click on the downloaded file to install R
- It is also a good idea to install XQuartz (needed by some packages)
- Go to the RStudio download page
- Under *All Installers*, download the RStudio Installer for MacOS.
- Double click the file to install RStudio
- Once it's installed, open RStudio to make sure it works and you don't get any error messages.

Linux

- Follow the instructions for your distribution from CRAN, they provide information to get the most recent version of R for common distributions. For most distributions, you could use your package manager (e.g., for Debian/Ubuntu run `sudo apt-get install r-base`, and for Fedora `sudo yum install R`),

but we don't recommend this approach as the versions provided by this are usually out of date. In any case, try to make sure you have at least R 4.0.0.
- Go to the RStudio download page
- Under *All Installers*, select the version that matches your distribution and install it with your preferred method (e.g., with Debian/Ubuntu `sudo dpkg -i rstudio-YYYY.MM.X-ZZZ-amd64.deb` at the terminal).
- Once it's installed, open RStudio to make sure it works and you don't get any error messages.

Update R and RStudio

If you already have R and RStudio installed, first check if your R version is up to date:

- When you open RStudio your R version will be printed in the console on the bottom left. Alternatively, you can type `sessionInfo()` into the console. If your R version is 4.0.0 or later, you don't need to update R for this lesson. If your version of R is older than that, download and install the latest version of R from the R project website for Windows, for MacOS, or for Linux
- It is not necessary to remove old versions of R from your system, but if you wish to do so you can check How do I uninstall R?
- Note: The changes introduced by new R versions are usually backwards-compatible. That is, your old code should still work after updating your R version. However, if breaking changes happen, it is useful to know that you can have multiple versions of R installed in parallel and that you can switch between them in RStudio by going to `Tools > Global Options > General > Basic`.
- After installing a new version of R, you will have to reinstall all your packages with the new version. For Windows, there is a package called `installr` that can help you with upgrading your R version and migrate your package library.

To update RStudio to the latest version, open RStudio and click on `Help > Check for Updates`. If a new version is available follow the instruction on screen. By default, RStudio will also automatically notify you of new versions every once in a while.

Install required R packages

During Marine Semester courses you will likely need to use an R package or two. Packages contain useful R code written by other people. In the Quantitative Fisheries Analysis course, for example, students will use the packages `dplyr`, `rEDM`, `ggplot2` and possibly others.

Like most tasks on the computer, there's more than one way to go about this. Your Rstudio, for example, should have a **Packages** tab in the low right hand quadrant. It's good to know how to do things from the **Console**, however.

To install these packages, open RStudio and copy and paste the following command into the **Console** window (look for a blinking cursor on the bottom left), then press the Enter (Windows and Linux) or Return (MacOS) to execute the command.

```
install.packages("ggplot2")
```

Alternatively, you can install the packages using RStudio's graphical user interface by going to `Tools > Install Packages` and typing the names of the packages separated by a comma.

R tries to download and install the packages on your machine. When the installation has finished, you can try to load the packages by pasting the following code into the console:

```
library(ggplot2)
```

If you do not see an error like `there is no package called '...'` you are good to go, and **we'll see you at the workshop!**

Updating R packages

Generally, it is recommended to keep your R version and all packages up to date, because new versions bring improvements and important bugfixes. To update the packages that you have installed, click `Update` in the `Packages` tab in the bottom right panel of RStudio, or go to `Tools > Check for Package Updates...`.

Sometimes, package updates introduce changes that break your old code, which can be very frustrating. To avoid this problem, you can use a package called `renv`. It locks the package versions you have used for a given project and makes it straightforward to reinstall those exact package version in a new environment, for example after updating your R version or on another computer. However, the details are outside of the scope of this lesson.

Contributors

** This material was adapted from **

> François Michonneau, Tracy Teal, Auriel Fournier, Brian Seok, Adam Obeng, Aleksandra Natalia Pawlik, ... Ye Li. (2019, July 1). datacarpentry/R-ecology-lesson: Data Carpentry: Data Analysis and Visualization in R for Ecologists, June 2019 (Version v2019.06.1). Zenodo. http://doi.org/10.5281/zenodo.3264888

The list of contributors to the original Data Carpentry lesson is available here.