# R and RStudio - The Basics

*Fu-Sheng Chou, MD, PhD*

**Installing R and Rstudio:**

R

As I mentioned in the previous post, R is open-source software maintained by a group of talents. R is freely available for download from multiple mirror sites (servers around the globe that host exactly the same files and data for the sake of convenience to the end-users). The most recent version is 4.0.3 ("Bunny-Wunnies Freak Out").

R is available for download here. (1)

Installation in Windows and Mac environments are straightforward. Double-click on the downloaded file and the installation wizard will take you through the process. If you are using Linux, I assume you know the basics of command-line tools and can figure them out on your own.

> *"As I mentioned in the previous post, R is open-source software maintained by a group of talents. R is freely available for download from multiple mirror sites (servers around the globe that host exactly the same files and data for the sake of convenience to the end-users)."*

**Rstudio:**

The next step is to install RStudio. RStudio comes in different flavors. There is a desktop version (RStudio Desktop) and a server version (RStudio Server). The latter version can be installed on a computer that runs the Linux operating system and allows users to log in remotely to use RStudio via a web browser. As a Delaware Public Benefit Corporation, RStudio always comes with an open-source version free for download. The paid Pro version provides advanced functionality and priority technical support. RStudio Desktop open-source, free version is quite sufficient to build web apps for personal or small-group use. The installation file can be downloaded here. (2) Make sure you choose the one on the left side for RStudio Desktop Open-Source License. The website will detect your operating system directly and point you to the right version. The current stable version is 1.3. Version 1.4 is available for public preview.

After the file is downloaded, installation is also pretty straightforward on Windows or Mac. Just follow the installation wizard. Again, installing RStudio open-source on a machine running Linux can be tricky, but if you are using Linux, I am sure you can figure it out!

**RStudio Cloud:**

If you do not like the hassle of installing software on your personal computer, and you know you will always have a high-speed internet connection when you work on an R project, you may want to try RStudio Cloud offered by the RStudio company. Again, a free version provides 1 CPU and 1GB RAM per project for up to 15 projects in total. The free version should be sufficient for simple webApp development.

> *"Again, a free version provides 1 CPU and 1GB RAM per project for up to 15 projects in total. The free version should be sufficient for simple webApp development."*

**Navigating through the RStudio interface:**

R is the programming language. RStudio is the integrated development environment (IDE) that accesses R to perform tasks. When you launch RStudio, R opens automatically in the background. In other words, all you need is to open RStudio, and you are ready to carry out all the coding tasks.

When you open RStudio, you see a window that is similar to Figure 1. There are three panels, one on the left and two on the right. The panel on the left defaults to a tab called "Console." This location is the place where all the codes are executed. The right upper panel defaults to the tab "Environment." This location is where all the "objects" that are created are listed. The right lower panel has multiple tabs, with "Files" being the default view. If you are using RStudio locally, the folders in the Files tab are the folders on your computer. If you use RStudio Cloud, the default view is the Project folder. Next to "Files," you have "Plots," where plots are displayed. Remember we said that R has a superb graphic engine for plotting! The next tab is "Packages," where all the installed packages are listed. The first one on top (in my R version) is "askpass," followed by "assertthat," and so on. The checkboxes left to the names of the packages can be clicked to "load" the library. Alternatively, you can also type **library(askpass)** in the Console to load the library you need. Next to the tab "Packages" is "Help," where all the resources are listed. The last tab is "Viewer," the tab that shows all the HTML rendering. For building webApps, you probably only need to access the "Files" tab.

There are numerous menu items and buttons in the RStudio IDE to improve coding efficiency. We will get to them when we need to. Interested readers may refer to the RStudio IDE cheat sheet for details. (3)

**Install the *shiny* package:**

Before we can dive in to make the first webApp, we need to install the package first. Type **install.packages("shiny")**. As you are typing **install**, you should see hints displaying next to the cursor to guide you to the right "function." You can select the one you want and press ENTER.

> *"There are numerous menu items and buttons in the RStudio IDE to improve coding efficiency. We will get to them when we need to. Interested readers may refer to the RStudio IDE cheat sheet for details. (3)"*

Let us pause for a second to dissect this "function" first. A "function" is a packaged task that you tell R to perform. All functions ideally should have their own unique names. In the above syntax, the name of the function is **install.packages**. Followed by the name is a set of parentheses needed for R to know that **install.packages** is a function to be executed. Inside the parentheses are "arguments" that the coders need to provide for the function to be executed meaningfully. In this case, if we do not provide the name of the package to be installed, R does not know what to do. Here we provided a character string *shiny*. A character string is a text; it is always labeled with quotation marks. Single or double quotations both work, in most cases.

One thing to point out is, if after pressing ENTER, you see a "+" sign showing up in the next line, and it looks like R is waiting for something to happen, usually, it is because there is a syntax error in the first line, or the syntax is not complete. For example, if you only provide a left-sided quotation mark, R thinks that the syntax is not complete and patiently waits for you to complete the syntax. In other words, R allows syntax entered on multiple lines.

The other thing to point out is, R is case-sensitive, so **install.packages("shiny")** is NOT equal to **install.packages("Shiny")**. R tells you that package 'Shiny' is not available for this version of R if you type in the latter.

It takes a few minutes to install the package, depending on your computer's speed and the internet speed.

Now, if you type **?install.packages()** in the Console, you will see that a document that describes the function **install.packages()** pulled up to the Help tab of the right lower panel. In the document, the top left corner has the name of the function, and the package that the function resides in is in curvy brackets. The function **install.packages()** resides in the package called utils, which stands for *utilities*. Moving down, you can see the Description of the function, as well as Usage, which gives you all the arguments you can provide to R when executing the function. The first argument is pkgs, which stands for *packages*. R goes by the order inside the parentheses of a function if an argument's name is not specified. Therefore, **install.packages("shiny")** means the same as **install.packages(pkgs= "shiny")**. In Usage, you can see that the default answer for the argument *repos* is getOption("repos"), whatever that means. For installing packages, all you need is to provide the name of the package with quotation marks. That is all you need.

**Summary:**

In this post, we installed R and RStudio, went through a short introduction of the RStudio IDE interface, and installed the *shiny* package. We will start making our first webApp next month. Thanks for reading!

> *"In this post, we installed R and RStudio, went through a short introduction of the RStudio IDE interface, and installed the shiny package. We will start making our first webApp next month."*

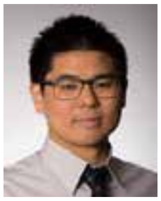
***References:***

1. https://ftp.osuosl.org/pub/cran/
2. https://rstudio.com/products/rstudio/download/
3. https://www.rstudio.com/wp-content/uploads/2016/01/rstudio-IDE-cheatsheet.pdf

*Disclosure: The author identifies no conflict of interest*

NT

*Corresponding Author*



*Fu-Sheng Chou, MD, PhD -*
*Senior Associate Editor,*
*Director, Digital Enterprise*
*Neonatology Today*
*Assistant Professor of Pediatrics*
*Division of Neonatology, Department of Pediatrics*
*Loma Linda University Children's Hospital*
*FChou@llu.edu*