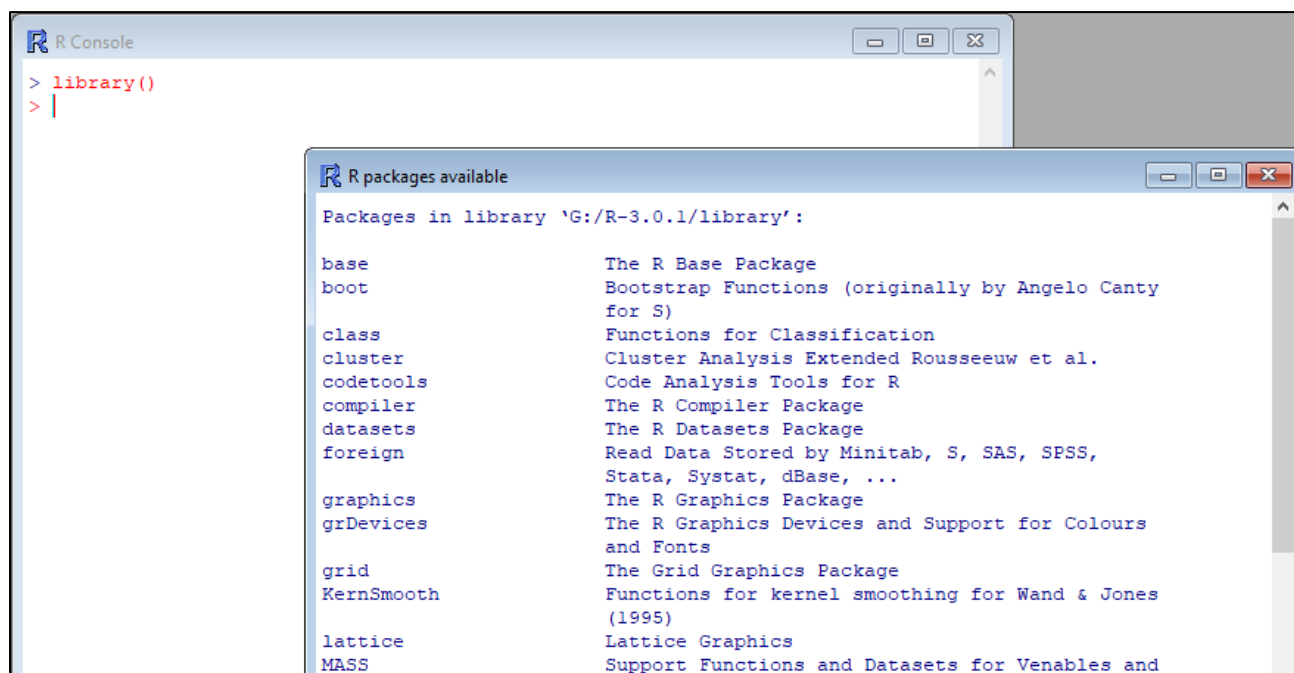# Installing R Packages Using R

Many R functions that you will use are "built in" to the base R library, meaning that we can use those functions as long as we just have our basic R installed. However, some of the functions are built into packages (sets of functions and commands) that are not in the base R library.

In such cases, the necessary package must be downloaded and installed before we are able to use any functions in that package. Luckily, most of this process is pretty automatic; R should know where to automatically store and install the package(s). This set of R instructions will discuss how to go about locating, installing, and using non-base packages in R.

For example, in STAT 213, there are a few functions that will be used that are not in the base R library. In order to use these functions, we will need to install the **mosaic** package (and some related packages). So let's use that package as our example for this set of instructions.

Checking if a Package is Installed

Depending on the computer you're using (your own, one of the lab computers through the Virtual Desktop, etc.), a package of interest may or may not already be installed in your R library. To check a list of available packages, open R and type **library()** in the R console. This will bring up a new window of available packages in your R library.
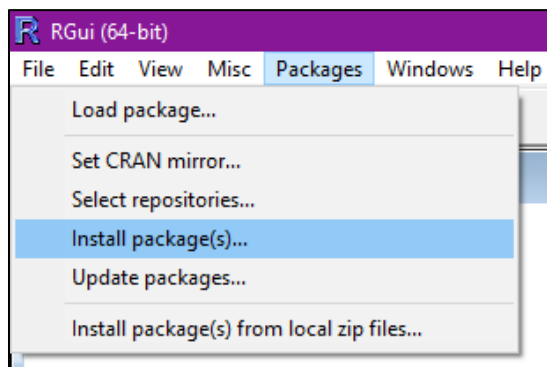

If you scroll through this list and do not see **mosaic** (or whatever package you need), then we need to install it!
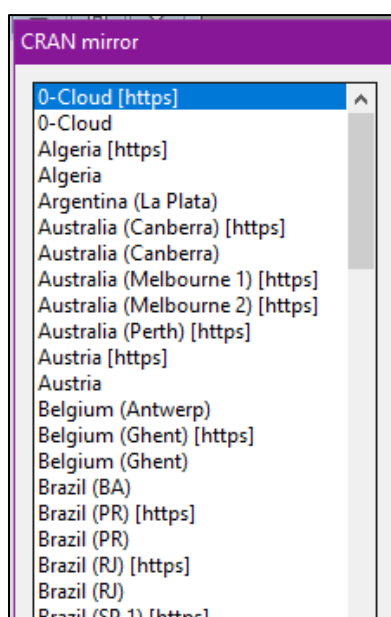
Installing a Package Using R

To install a package of interest using R, follow these instructions:

1. Open R. In the top menu, click **Packages → Install package(s)…**

2. Select a CRAN mirror. It doesn't really matter which mirror you select; I usually pick the one closest to me geographically. Double click the mirror name or click "OK" to select it.

3. After you select the CRAN mirror, this menu will close. If you go back into **Packages → Install package(s)…**, a new window should now open that contains a list of all R packages. It may take a moment for this window to appear, as there are a lot of R packages! In this window, scroll down until you reach the name of the package you want to install. For this example, we want to install **mosaic**.
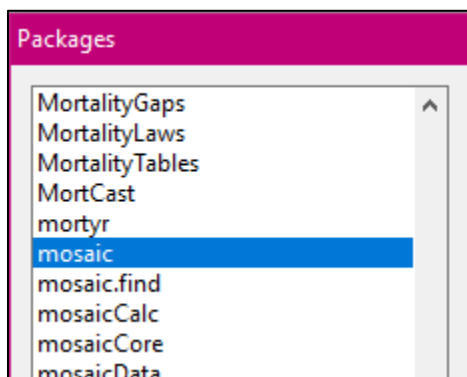
The mosaic package is big and also requires several other packages for it to run; it *should* automatically install any related packages that it needs. Be patient while it installs; it may take a few minutes! **You will notice a lot of text show up in your R console window. This is normal!** It means the packages are installing.

The **mosaic** package will have been successfully installed when you get a message similar to this and your cursor (the little red caret **>**) appears after it.
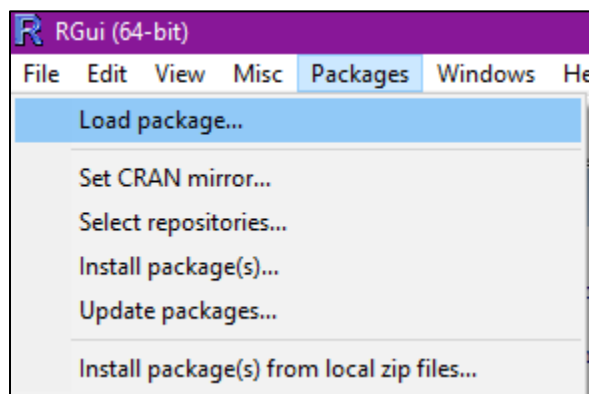

This general process should be the same for other packages (though smaller packages may not take as much time to install as mosaic).

Loading a Package in R

To use an R function from a non-base package, you need to load the package first. You can do this by either typing `library(packagename)` or by going to **Packages → Load Package** and selecting the package from the list that pops up.

```
R  RGui (64-bit)
File  Edit  View  Misc  Packages  Windows  He
      Load package...

      Set CRAN mirror...
      Select repositories...
      Install package(s)...
      Update packages...

      Install package(s) from local zip files...
```

Either way, you should see something like this in your R console:

```
> library(mosaic)
Registered S3 method overwritten by 'mosaic':
  method                          from
  fortify.SpatialPolygonsDataFrame ggplot2

The 'mosaic' package masks several functions from core packages in order to add
additional features.  The original behavior of these functions should not be affected by this.

Attaching package: 'mosaic'

The following objects are masked from 'package:dplyr':

    count, do, tally

The following object is masked from 'package:Matrix':

    mean

The following object is masked from 'package:ggplot2':

    stat

The following objects are masked from 'package:stats':

    binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
    quantile, sd, t.test, var

The following objects are masked from 'package:base':

    max, mean, min, prod, range, sample, sum
```

There's a lot of text there, but if yours looks similar to this, it means that **mosaic** loaded and you can use the functions in that package and in the related packages needed by **mosaic**. Again, this general process should be the same for other packages. Depending on the package, you may or may not see any blue text after you type typing `library(packagename)`.

Troubleshooting

The easiest way to make sure a package is working is to just try one of the functions that is built into that package. As an example, the function `favstats()` is built into the **mosaic** package, meaning that in order to use `favstats()`, the **mosaic** package must be installed, loading, and working properly.

If you are running into an issue with a function that should be working, here is a set of troubleshooting steps you may want to try.

Step 1

First, **make sure the package that you need is installed**. Type `library()` in the R console. This will bring up a list of installed packages. If you do not see the package that you need in the list, install it (see the "Installing a Package Using R" section above).

Step 2

If the package that you need is installed, but your function is still not working, **make sure the package is loaded into your R session**. To load a package, type `library(packagename)` or go to **Packages → Load Package** and select the package from the list that pops up. Once the package is loaded, you should be able to use any functions from it.

Step 3

If the package is installed and loaded but your function is still not working, **check to see if you have the latest version of R installed.** To check your version of R, open R and look at the top row of the text that automatically appears in the R console. This tells you what version of R you're running. In the following screenshot, I am running version 4.0.4.
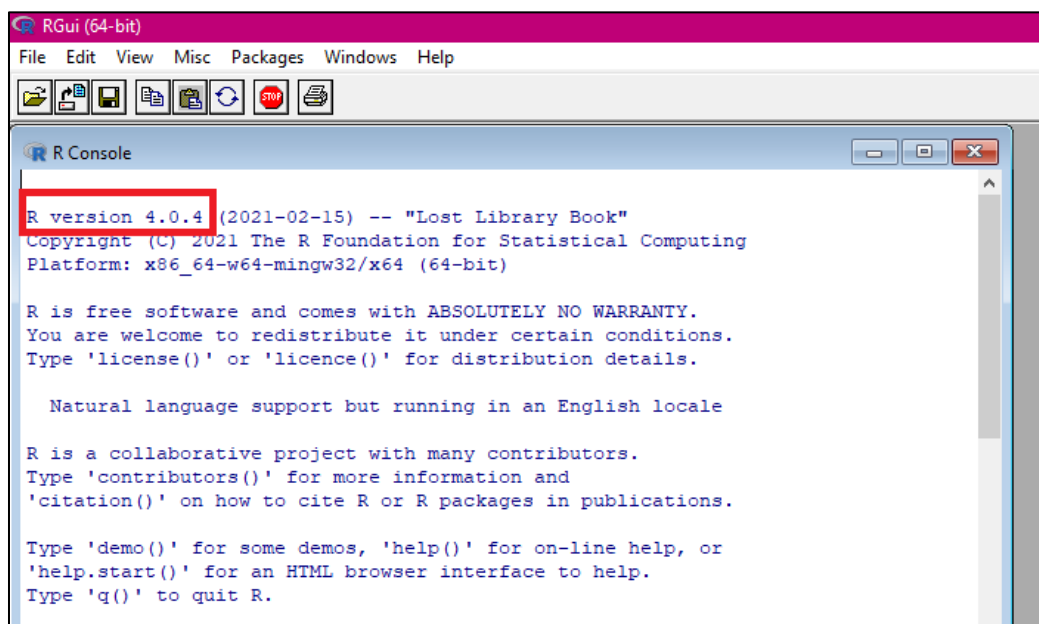
To check if this is the latest version of R, visit the R homepage at https://www.r-project.org/. The latest release version will be listed on the main page. In the following screenshot, it shows that the latest version as of the writing of this guide is version 4.1.0.

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want o

- Download R for Linux (Debian, Fedora/Redhat, Ubuntu)
- Download R for macOS
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. If you do not know what this means, you probably do not want to do it!

- The latest release (2021-05-18, Camp Pontanezen) R-4.1.0.tar.gz, ead what's new in the latest version.
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).

Some packages may not work with older versions of R. So if you do not have the latest version, you can get it by just re-installing R from https://www.r-project.org/ (if you need to refer to the "Installing R" guide). You do not have to uninstall the older version you have; it will just be overwritten by the newest version.

Once you have the latest version of R, try re-installing the package you need, load it into your R session, and try running a function from that package. This typically solves the problem!

Every once and a while, though, R can just be a little buggy. So if you're still having trouble, just closing and re-opening R can sometimes fix the problem!