Obtaining and Using R

This book relies on Anaconda for demonstrating the data science code. You must begin by installing Anaconda, as described in Chapter 4 of the book. After you have an Anaconda installation in place, you can use the instructions in the following sections to create an R environment in which you can execute the book's R code.

Obtaining and installing Anaconda for R

To install R, you must open the Anaconda Prompt. For Windows users, this means choosing Start@@-->All Programs@@-->Anaconda3@@-->Anaconda Prompt. You see the Anaconda Prompt, shown in Figure 1. Notice the word *(base)* at the beginning of the prompt. This is the current environment—the base environment.

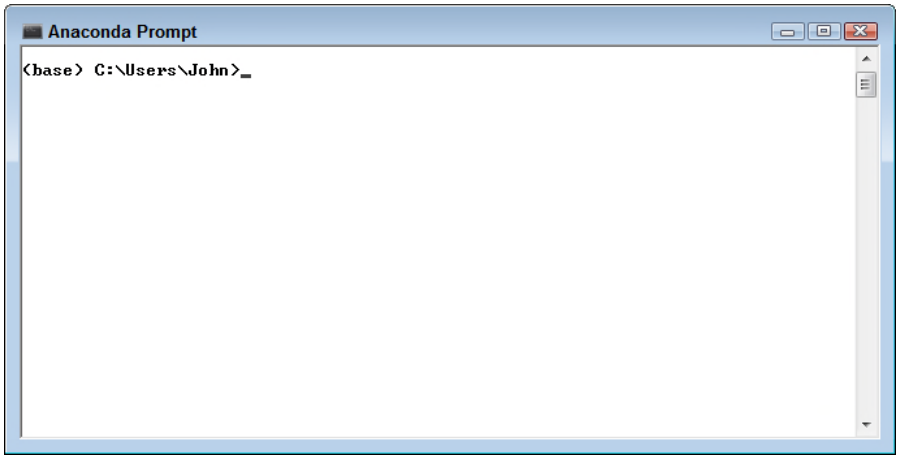

Figure 1: Open an Anaconda Prompt to install R.

At the Anaconda Prompt, type **conda create -n R_env r-essentials r-base** and press Enter. This command creates a new Anaconda environment called R_env. Whenever you want to work with R code, you use the R_env environment. Within this environment, `conda`, the Anaconda command-line utility, installs R essentials and base packages. This set of packages is enough to get you started. However, you can install other packages later as needed for specific kinds of analysis. The installation process displays a series of messages that ends with a listing of packages that `conda` will install, like those shown in Figure 2.

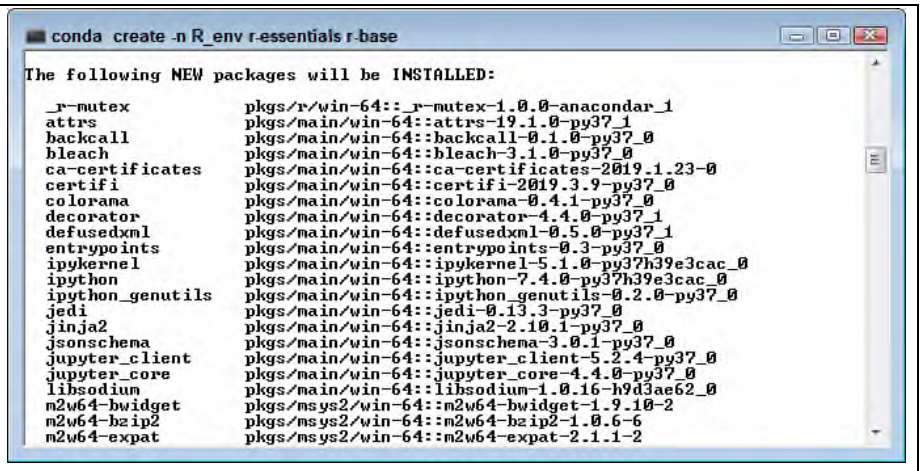

Figure 2: The conda utility tells you which packages it will install.

You can begin the installation process by typing **y** and pressing Enter. At this point, conda begins to download and extract packages. When the downloads complete, conda performs various transactional processes that can include compiling some of the packages it downloaded into executables (or other forms). This process can take a while, so you might want to have a good book ready or some coffee to drink.

<Remember>

After the installation process completes, you're still in the (base) environment. To work with the (R_env), you must type **conda activate R_env** at the Anaconda Prompt. You can how use whatever directory command your platform supports to see a list of the files that conda installed. To leave the (R_env) environment and go back to the (base) environment, you type **conda deactivate** and press Enter.

Starting the R environment

Fortunately, you won't work at the Anaconda Prompt when you want to work with R code. However, you won't start Notebook directly, either, because doing so starts the (base) environment and you want the (R_env) environment. Instead, you start Anaconda Navigator, which appears in Figure 3. Notice that this utility provides you with access to all the GUI tools that Anaconda supports (and if you don't see what you need, you can always install more).

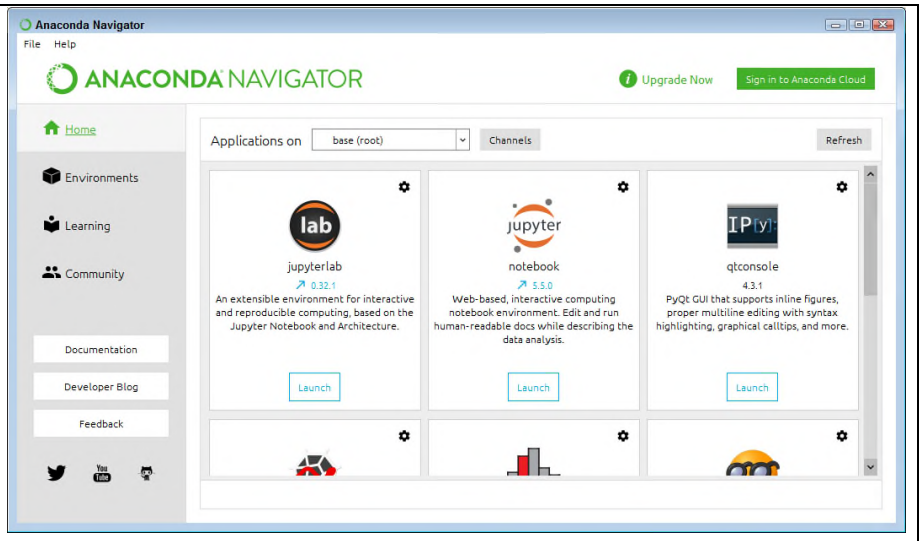

Figure 3: Anaconda Navigator provides access to a number of useful tools.

To start R, you must first select R_env in the Applications On drop-down list box, as shown in Figure 4. The tools you see will change to reflect what you can do with this environment. The important tool for this book is Notebook. Note the gear icon in the upper-right corner of the square. If you need to change the version of Notebook, click the icon and choose Install Specific Version from the list.

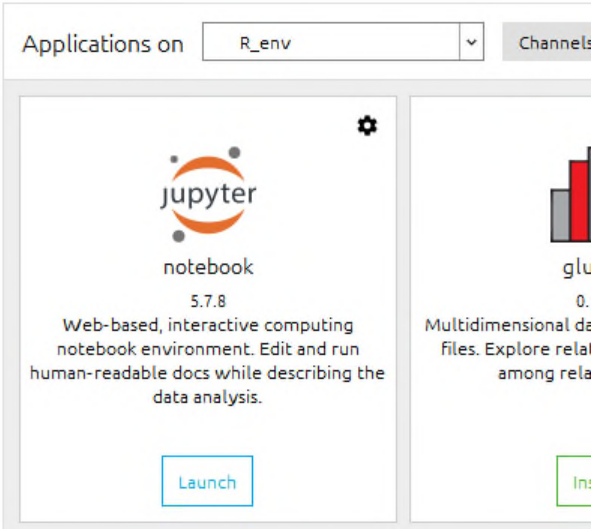

Figure 4: Changing your environment will often change the available tool list.

Click Launch in the Notebook square to start the application. You see Notebook start in a browser, just as you do when working with Python, but now you're working with R instead.

Defining an R Code Repository

Because you're using Notebook, everything with R works precisely the same as it does for Python, as described in the "Defining the code repository" section of Chapter 4. The name of the R repository for this book is `ML4D2ER`. Use the instructions in the "Defining a code repository" section to create an R repository and experiment with an R file. However, instead of choosing Python 3 from the New drop-down list, you choose R instead. Otherwise, everything works as it would for Python. The test file for this section is `ML4D2E_R; 04; Sample.ipynb`.

<Tip>

Something important to note when using R is that you can download your code as an `.r` file, as shown in Figure 5. In fact, you have not only the Notebook formats to choose from, but a number of R-specific formats as well. To keep things simple, the downloadable source for this book relies on .ipynb files for all source code to ensure that you get the required comments with the file.

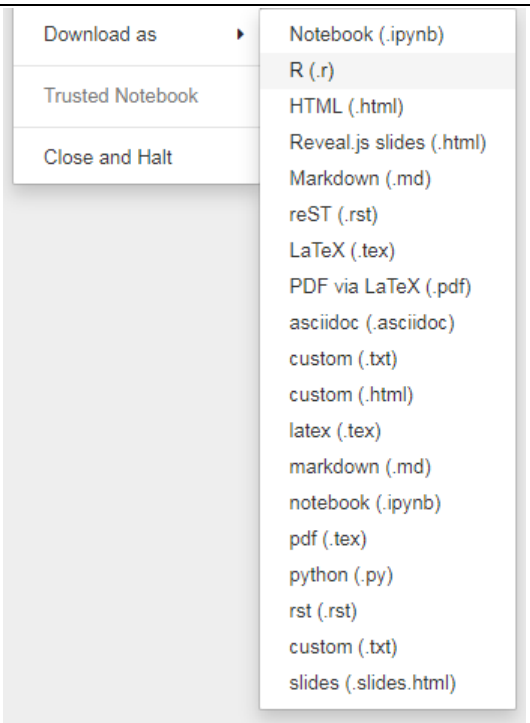

Figure 5: You can save R code in .r files, but the .r files lack Notebook comments.

You can perform a quick test of your R environment using the following code:

```
print(R.version.string)

result <- system2('conda',
                  args='list -n base anaconda$',
                  stdout=TRUE, wait=TRUE)
print('Anaconda Version: ')
print(result)
```

Using `system2()` to run `conda` works best and you must set the `stdout` and `wait` arguments as shown. Notice that `conda` is the command and what you want to do with `conda` must appear as part of the `args` argument. After you click Run, you see the following output:

```
[1] "R version 4.0.3 (2020-10-10)"
[1] "Anaconda Version: "
[1] "# packages in environment at C:\\Users\\John\\ANACON~1:"
[2] "#"
[3] "# Name                    Version                   Build  Channel"
[4] "anaconda                 2020.07                  py38_0  "
```

This output shows the versions of products use for this book, which include version 4.0.3 for R and 2020.07 for Anaconda. Make sure your version information matches this version information for best results.