

Introducing R and RStudio

James H. Steiger
Vanderbilt University

Introduction

In this module, we get started in R and RStudio by illustrating a few key points and demonstrating some basic capabilities of both programs.

R is an amazingly powerful and *extensible* system for statistical analysis. There is a huge range of statistical packages that implement many advanced analyses in R. In fact, R has quickly become the vehicle through which new methods are introduced. And, with a little practice, you can learn to write your own functions in R.

Before the period is over, we'll see how R can do things in a couple of lines that used to take weeks of careful programming.

If you use R properly, it can serve as an important aid to *studying* statistics, as we'll see in the next module.

Introduction

RStudio is a **development environment** for R.

RStudio has a vast range of capabilities. Here are a few:

- RStudio can organize many R routines into complex projects.
- While you are running R code, you can **debug** the code in RStudio by single-stepping through the code and examining the results of the code while it is executing.
- RStudio gives you 1-button access to compiling complex documents that combine LaTeX and R. You can embed R code in a document, and RStudio will automatically run the code and embed both the code and the output in a beautifully typeset document.
- You can embed all your data and calculations in the document. Imagine writing a 50 page paper with 10 figures and 6 tables. Then you discover that 5 data points were inadvertently omitted.
- With the R system, you can add the 5 data points, click on a single button, and the entire document can reconstitute itself in a couple of seconds. If you need to add a figure in the middle of the document, all the figure numbers and labels will adjust themselves.

Introduction

Let's get started.

Getting Started in R

Choosing a Working Directory

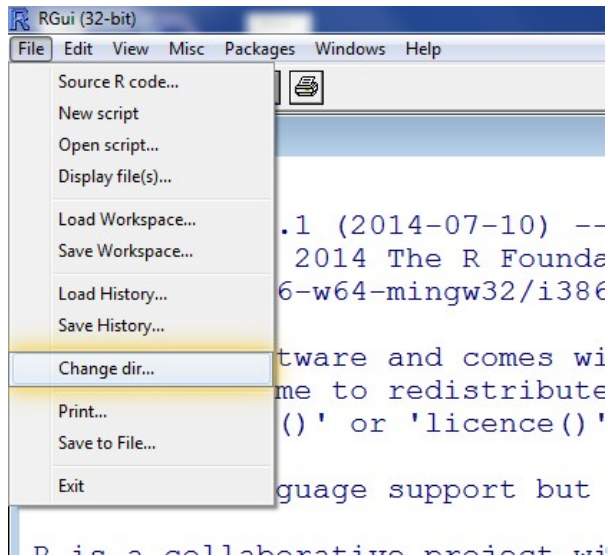
After starting an R session, you should make sure that you have selected the correct **working directory**, the place where R will search for files and save files by default.

It is a good idea to have a different working directory for every homework and lab assignment.

In the Windows version, just select `File --> Change Dir` from the menu.

Getting Started in R

Choosing a Working Directory



Getting Started in R

Choosing a Working Directory

Of course, you'll want to remember the name of the directory!!

Getting Started in R

Some Simple Calculations

Let's do a few basic operations in R.

To perform a simple calculation, type in a line and press the <Enter> key.

Here we add 4 and 2.

```
> 4 + 2
```

```
[1] 6
```


Getting Started in R

Some Simple Calculations

But what happens if you type just `4 +` and press the `<Enter>` key? Try it.

You see a `'+'` character. This is the **continuation character**.

When you see this character at the beginning of the line, it is an invitation to “keep going.” In this case, just enter the `2` to complete the statement and press `<Enter>` again.

Now you will see the result

```
> 4 + 2  
[1] 6
```

Getting Started in R

Some Simple Calculations

Let's divide 2 numbers

```
> 27/3  
[1] 9
```

That was easy! How about subtraction?

```
> 76 - 12  
[1] 64
```

Multiplication is slightly trickier, in that it requires use of the '*' character.

For example, to multiply 8×7 , we enter

```
> 8 * 7  
[1] 56
```

Getting Started in R

Opening a Script Window

When you are experimenting with R, it is a very good idea to open a **Script Window**. This is a place where you can type code, edit it, and then run it directly from the window.

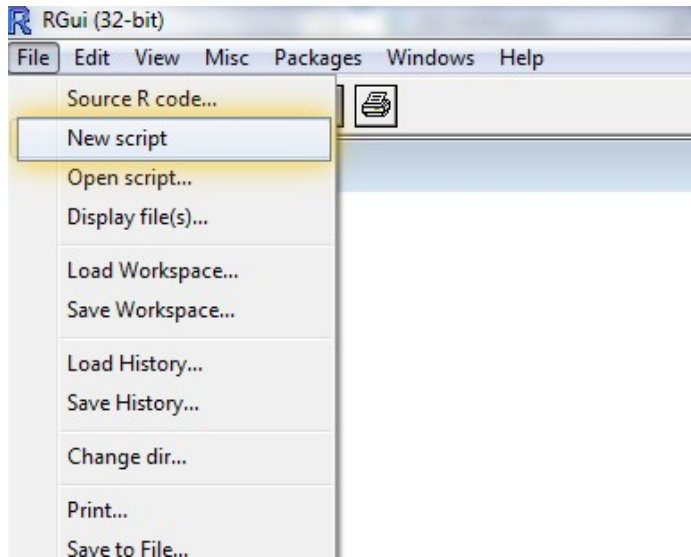
If your code is several lines long, and you discover an error, you can correct the error quickly and efficiently in the window.

You'll also discover that the up-arrow key is a really good friend when you are entering commands directly into R. This key allows you to recall previous commands. If you enter a command and realize it had a minor typo, you can recall the line and change it in an instant.

Let's open a Script Window and experiment with the two types of error correction.

Getting Started in R

Opening a Script Window



Getting Started in R

Opening a Script Window

You want to multiply the sum of 3 and 4 by 2. Suppose that, forgetting that multiplication has to be indicated explicitly in R, you type the following, and see the error message.

```
> 2(3 + 4)
```

```
Error: attempt to apply non-function
```

The error message is cryptic. Parentheses are used in R to demarcate a list of arguments to a function. R misreads your code, and things you are trying to execute the function “2” on the input “3+4”.

Since there is no function “2”, you get the error message.

You could retype the whole line. But it is faster to press the up-arrow key, recall the erroneous line, and quickly insert the multiplication sign, as follows:

```
> 2 * (3 + 4)
```

```
[1] 14
```

Getting Started in R

Opening a Script Window

Another way of developing code is to type it into a Script Window and execute it from there.

Type `2(3+4)` into the Script Window and then press `<Ctrl> + R`

The line is passed to the console and executed, resulting in the same error message we saw before.

You can simply edit the line in the Script Window, re-execute it, and get the answer.

The advantage of this approach is that, once you are done, you have your correct code still sitting there in the Script Window. You can save it, edit it, add comments, etc.

Getting Started in RStudio

The RStudio Interface

In the beginning, “Raw R” will work well for us with its simple console and Script Window interface.

However, RStudio multiplies the power of this approach by making available a number of additional windows with a wide range of capabilities.

RStudio includes a special window for handling graphical output, a full debugging facility, and a window for exploring currently loaded packages and their related help files.

It won't be long before you'll be running the vast majority of your R sessions from within RStudio.

Let's start up RStudio and take a quick look.