Search earth data science site

Earth Data Analytics Online Certificate

Enroll now! Learn more.

Lesson 1. How to Reuse Functions That You Create In Scripts - Source a Function in R

Leah Wasser

▤ R Source Functions - Efficient Programming - Earth analytics course module

Welcome to the first lesson in the **R Source Functions - Efficient Programming** module. Learn how to source a function in R by saving the function in another R script.

🎓 Learning Objectives

After completing this tutorial, you will be able to:

- Calculate NDVI using NAIP multispectral imagery in R .

- Describe what a vegetation index is and how it is used with spectral remote sensing data.

☑ What You Need

You will need a computer with internet access and `R` / `R Studio` loaded to complete this lesson.

Where to Store Your Functions

When you create a function to use in your analysis, you often create it and store it at the top of your script or `.Rmd` file as a first step. However, lots of functions at the top of your code can make your code dense and harder to read.

It is good practice to create separate `R` scripts that you can use to store sets of related functions. You can then call those functions using the `source()` function, at the top of your script in the same way that you call an `R` package. `R` will then load those functions into memory and you can use them!

Sourcing functions is good practice because it is:

1. **Reusable:** It allows you to reuse functions over and over using the same code (i.e. you don't have to copy and paste the function into each new analysis script).

2. **Easy to Maintain:** It allows you to quickly fix a function that doesn't work properly - only once.

3. **Sharable:** In the same way that a library can be used by anyone, you can share your `R` script containing your functions with anyone, too. This is the first step towards creating an `R` package!

How to Source Functions in R

To source a set of functions in `R` :

1. Create a new `R Script` (.R file) in the same working directory as your `.Rmd` file or `R` script. Give the file a descriptive name that captures the types of functions in the file.

2. Open that `R Script` file and add one or more functions to the file.

3. Save your file.

Next,

- Open your `.Rmd` file or `R` script.
- At the top of your file, add the `source(path/tofile/here.R)` function.

```
source("remote-sensing-functions.R")
```

If the `.R` script is in your main working directory, then it won't have a path element before it like `week_06/functionfile.R` vs `functionfile.R` .

If it's in a different directory, adjust the path accordingly. Once you run the code containing the `source()` function, all of the functions in your `.R` file will load into your global environment. You can now use them in your script!

Additional Resources

- NicerCode.com Guide to Functions

🏷 Tags   🏷 **Reproducible science and programming:** literate

🏠 **earth-analytics Home**

expressive programming , functions

📅 **Updated:** March 30, 2020

LEAVE A COMMENT