

00_Introduction

Aside from this document (00_Introduction), this series of pdfs and html documents correspond to R scripts of the same name. 01_basicHist.pdf describes in detail what you'll see in 01_basicHist.R and so on. This allows us keep the annotations in the R script fairly light and allows you to have this pdf open or printed out while you work with the script.

The idea is that while you're working with the R scripts, you can have a printout of the PDF to refer to or have the html file open in a browser window. The html files can be printed too. Use whatever works best for you.

If you need some assistance with these, please let me know.

Important note

Especially if you're new to coding, you're not going to understand everything in these files.

That's OK. That's normal. You don't have to understand everything to use it.

Think about this like you're opening a door. All you need to know is how to turn the door knob, you don't have to know how to design and machine all the parts and install a door knob in a home.

At some point you might want to or need to, but you don't have to right now. You're not reading this because you want to become a computer scientist, you're reading this to become a better journalist.

If at any time you want to learn more about something, copy the code snippet and paste it into a google search. Chances are lots of people have written about it and some of them may have even written a tutorial about it.

BTW, that's a valuable tip: google what you're curious about and add the word "tutorial." It's how most of the links listed below were found.

But generally, don't feel bad because you don't 'get' something right off. No one does.

If things go wrong

They will. You may have an extra comma, or forgotten to close parenthesis. It happens to everyone.

R is pretty good about telling you where the problem is. Or, try comparing your code to code that works and see if you can spot the difference.

Sometimes problems can get complicated. If so, you can try copying the warning message into a google search. A lot of times you'll come across the answer because you're not the only person to have this problem. You can also contact me.

And sometimes, it's really complicated. All these files were created using R version 3.3.3 and R Studio 1.0.143. That shouldn't be an issue until it is (welcome to coding).

How these guides are structured

These guides assume you've installed R and R Studio, and have done some of the first Swirl tutorials.

- From 01 to 07, you'll be working with R's basic plots. For exploring data yourself, these are good tools to know. However, their design isn't great for print or online. And, it takes some work to get a decent graphic out of them. But these are also good exercises for understanding R, so try hard to work through them.

- At 08, we start working with ggplot2, the R graphics library that takes a sophisticated approach to graphics. 08_qplot is part of ggplot - q stands for quick, and it's meant to create quick visualizations that help you understand the data. Qplot is an excellent way of quickly looking at your data, and it's a lot easier to use than the basic plots. Along with ggplot2, qplot should replace all those basic plots.
- 09 on looks at the various graphic types available through ggplot. You'll also learn to export your work for publication.
- Finally, M01 and on looks at R's ability to do maps - specifically choropleth maps. Mapping is different from anything you've done before, so work with R a bit before you tackle it.

If you want to skip the rest of this, go ahead

Below this are some useful links and a discussion of open source software like R. I recommend you look through this stuff when you get the chance.

But if you want to jump into the scripts and guides, go ahead.

Resources

Below are links where you can get the latest versions of R and R studio, as well as how to install the Swirl tutorials.

Along with those, there are links to tutorials, books and other places where you can learn more about R.

I highly recommend you explore these - they are the source material for the files you're going to be using. They were created by talented professionals who have freely shared their knowledge, including investigative reporters at some of the top news organizations in the country. The work they're doing is the kind of thing we should aspire to.

Installing R

To get R, go to <https://www.r-project.org/>

Once that's installed, get the free version of R-studio here <https://www.rstudio.com/> You can run R without R-studio. But R-studio is a friendlier interface for R, and most people recommend it.

Be sure to install the versions compatible with your operating system, especially version number.

Learning R

Swirl is a set of interactive - free - tutorials that will take you through the basics of learning R and R-studio. <http://swirlstats.com/> To install it, type: `install.packages("swirl")` in the R console. It'll download and install it for you.

To run swirl, type: `library(swirl)` and that will activate the tutorials.

I'm also going through this Udemy class: <https://www.udemy.com/r-programming/> I find it to be a nice compliment to the swirl tutorials.

If you've never done a Udemy class, the first time you go to their website and look at a specific class, it'll offer that class to you for a drastically reduced price. This r-programming class was only \$10.

Other resources - graphics

Find annotated scripts in folder R Graphics that start with basic plotting in R and progress through to plots using ggplot2. The point with these is to produce various chart types suitable for print or online - and for online, specifically viewable on mobile devices.

I found these tutorials invaluable for developing our R scripts for graphics <http://t-redactyl.io/tag/r-graphing-tutorials.html>

This is an excellent introduction to R for reporters. <https://paldhous.github.io/NICAR/2017/r-analysis.html>
This was taught at NICAR 2017, but the step-by-step worksheet is fairly easy to follow.

Want to see how 538 does their graphics? They use R, and they're putting all their code online for free. Here's a description of what they're doing: <https://cran.r-project.org/web/packages/fivethirtyeight/vignettes/fivethirtyeight.html>

And here's the R scripts and datasets they examined <https://github.com/rudeboybert/fivethirtyeight/tree/master/data-raw>

Here's the book on data science and R, with one of the authors being the person who created the visualization library ggplot2 <http://r4ds.had.co.nz/>

Here's a free book on advanced R: <http://adv-r.had.co.nz/>

David Montgomery of the Pioneer Press shared this analysis of police data, from working the data to visualizing it: <https://github.com/pioneerpress/code/tree/master/st-anthony-police>

He's constantly doing little things to develop his skills - just like an athlete will work out every day. Here's where he has that, some of it fun: <https://github.com/dhmontgomery/personal-work>

Other resources - mapping

There's a way bigger challenge doing maps in R, especially if you don't have experience in GIS at all. My annotated scripts on this are in folder mapping. The focus on here is to create exploratory choropleth maps with your choice of color palattes. To also have these able to be exported for print or online would be gravy.

Here's an extensive online tutorial to get started with mapping and R: <https://github.com/Robinlovelace/Creating-maps-in-R/blob/master/README.md> It had some rough parts, but I worked through it and my annotations may help: <https://github.com/dailyherald/R-work/blob/master/mapping/mapping.R>

This was a good compliment to the above: <http://www.computerworld.com/article/3038270/data-analytics/create-maps-in-r-in-10-fairly-easy-steps.html>

Another step by step here: https://reubenfb.github.io/mapping_in_r/presentation.html#/

This is just plain beautiful to look at, but it also has good information on working your data so you can do choropleth maps with custom breaks: <https://timogrossenbacher.ch/2016/12/beautiful-thematic-maps-with-ggplot2-only/#discrete-classes-with-quantile-scale>

This may be a good compliment to the above: <http://www.kevjohnson.org/making-maps-in-r/>

More info on the scales package: <https://www.rdocumentation.org/packages/scales/versions/0.4.1>

Looks like the Financial Times is going to start sharing R tutorials: <https://github.com/ft-interactive/R-tutorials> The first one up is on joining data to shapefiles. FYI, shapefiles are one of the ways mapping information is stored. So, for instance, we have shapefiles for all the counties in Illinois that can be used to show average income by county or median age.

Other resources - advanced

I took this one on doing interactives with R at NICAR 2017. I would say it's a little advanced, but it used things we use here at the DH for our interactive maps and graphics.

<https://paldhous.github.io/NICAR/2017/r-to-javascript.html>

This is a weekend project - take some time on a weekend and just go through this as an exercise: Andrew Ba Tran examines police stun gun use in Connecticut. He works the data, and examines it visually with charts and maps all generated in R. This was for a NICAR session. <https://trendct.github.io/data/2016/06/stun-guns/>

This is amazing: Washington Post shared all the data and R scripts they used to break this major story: "Jared Kushner and his partners used a program meant for job-starved areas to build a luxury skyscraper" https://github.com/wpinvestigative/kushner_eb5_census

Final thoughts

Open source and fair use

The open-source ecosystem is pretty unique. Millions of people (not an exaggeration) participate in building and maintaining software that's free to use and share. In some cases, we've used some of their code in our scripts. That's OK, it's how open source coding is supposed to work.

For instance, say you want to load in a comma-separated file. You search the internet and find a chunk of code someone posted:

```
df <- read_csv("filename.csv")
```

It's fine to use the code - no one owns the code itself. And part of the open-source ethos is sharing knowledge so that others can learn. They do that because others did that for them.

However, taking someone's data and their script, running it to produce their work and then using their result in your work - that could be a problem.

But if you're just looking to learn and improve your own skills, don't be afraid to look at other peoples' work.