# How do I learn R?

*Alexander Coppock*

*7/19/2017*

"How do I learn R" is the wrong question because it suggests that there is some state of "Knowing R" that exists. You might revise and say "How do I learn *enough* R?" That's also wrong because there is no state of "Knowing Enough R." No matter how good you are at R, your code can always become more beautiful, expressive, fast, and clean.

OK, but really, you have to start somewhere, so how do you start learning R?

## Start with the tidyverse

The tidyverse is a set of packages by the folks at Rstudio. It's got a unified approach to data analysis and visualization. You may have heard of `dplyr` or `ggplot2` – these are the core `tidyverse` packages.

This point is contentious, but I think that beginners should learn the tidyverse before learning base R. Base R is confusing, filled with inconsistencies, and very computer-sciency. The tidyverse is internally consistent and great for the sorts of tasks that empirical social scientists tend to do. For a perspective on learning the tidyverse first that agrees with mine, see this blog post: http://varianceexplained.org/r/teach-tidyverse/

There are dozens and dozens of introductions to R on the internet. Most don't start with the tidyverse. Instead, they begin with the building blocks like object types (scalars, vectors, lists, etc.) and data types (numeric, character, factor, logical etc.). These things are important and I'm sure you'll learn about them eventually.

But when kids are learning about science, they make the volcano **first** and **then** they learn about why the combination of sodium bicarbonate and acetic acid is so unstable.

The volcano in this analogy is `ggplot2`. It's really fun to make statistical graphs. Work through Chapters 1 - 3 of R for Data Science as a very first introduction, and move on from there. For helpful cheatsheets, check out the resources here: https://www.rstudio.com/resources/cheatsheets/.

## Errors

In your first few weeks coding, you will make 1000's of errors.

- R is [probably] not broken. It could be that you need a package you don't have or more likely, your code has an error.
- Get better at typing (h/t @jennybryan). The computer needs precise instructions.
- Someone on the internet has had your problem and asked about it on stackoverflow. Still, you might not yet know the words to google in order to find the stackoverflow question. And even if you find it, the answer may or may not be polite, understandable to beginners, or the best way to solve the problem.

Persevere.

## Things to remember a month from now

I promise that a month from now, you will have had some successes and you will start to think of R as a useful tool. It's in this period that people start to develop bad habits.

- You will be tempted to do some tasks in a more comfortable environment (like Stata or Excel), then switch over to R for other things. Don't. Just figure out how to do the whole workflow in R.
- You'll say things like, "I don't care if it's pretty so long as it works." This is a **bad attitude**. Develop an eye for and an appreciation of clean code. Adopt and follow a style guide: http://style.tidyverse.org
- You are collaborating with your future self (h/t @hadleywickham). Strive to write scripts that are transparent and reproducible.

## You need a project

When you start coding, you're usually working on things other people give you like tutorials and problem sets. That's all fine, but you don't really start to take command of your own learning until you start working on your own project. You need to care about the answer so much that you just **have** to figure out how to solve the problem in R.