Introduction to R
ICPSR Summer Program
May 13-15, 2019
University of Houston
Houston, TX

<table>
<thead>
<tr>
<th>Instructor</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ryan Kennedy, PhD</td>
<td>Office: Philip G. Hoffman Hall (PGH), room 447</td>
</tr>
<tr>
<td>Senator Don Henderson Chair</td>
<td>Email: <a href="mailto:ryan.patrick.kennedy@gmail.com">ryan.patrick.kennedy@gmail.com</a></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>Office Phone: 713-743-1663</td>
</tr>
<tr>
<td>Department of Political Science</td>
<td>Cell Phone: 713-855-4811</td>
</tr>
<tr>
<td>University of Houston</td>
<td></td>
</tr>
</tbody>
</table>

Course Overview

R has become the software of choice for a great deal of social science analysis. The fact that it is free and open source, along with the wide variety of libraries available for use in analysis, have contributed to this popularity. And a number of graduate programs, including my own, emphasize skills in R to the exclusion of traditionally popular statistical software (e.g. SPSS, Stata, SAS).

Yet, for many students, their first foray into using R is an exercise in frustration. Part of the problem is that R is usually taught in a cookbook-like manner. Students are shown how to do basic statistics using R by copying a series of commands – basically like using dropdown menus, but without the convenience of the dropdown menus. There is little emphasis on understanding how R works. For example, students are taught to run a regression using the command `lm(y ~ x, data = dataset)`, but are never taught that `lm()` is a function – much less, how they can write their own functions to do almost anything they want. In other words, R is often taught in a way that emphasizes its weaknesses (no dropdown menus, flexible command structure) and barely mentions its strengths (ease of programming, openness to new techniques, quality of replication files). Predictably, this results in frustrated students who learn to rely on copying formulas from cookbooks rather than developing their own unique projects.

This course takes a very different approach. Instead of teaching students a series of models or summary statistics commands (which they can easily look up online), it focuses on using R as an elegant and approachable programming language, through which students can do a variety of tasks, just one of which is running statistical models.

Moreover, the course presents a “tidy” version of R programming, utilizing the “tidyverse” group of tools developed by Hadley Wickham and others to make R programming more unified and easier to understand. These tools produce a “grammar of analysis” that will allow students
to quickly produce outstanding research. It also produces code that will be more easily understandable to others (and yourself) and allows more consistency in process. This is not to say we will be completely ignoring some aspects of base R, rather the goal is to get students working in a system that will be more comfortable, readable, and consistent. It will also be more in line with current publication expectations (e.g., producing plots using ggplot2, rather than the base R graphics).

The goal of the course is to produce students who are confident users of R in a range of situations and can easily expand their knowledge to new fields.

**Schedule**

Morning session: 9:00–12:00  
Lunch: 12:00–1:00  
Afternoon Session: 1:00–5:00

**Textbook**

There is no required textbook for this course. The instructor will provide detailed handouts (part of a book project) and labs. For those who would like a reference, and a book that is a good follow-up to this 3-day workshop, I recommend Hadley Wickham’s *R for Data Science*.

**Outline**

- **Day 1** – We will briefly go through setup and basic use of R and RStudio. We will also cover some of the basic parts of R: objects, functions, and libraries. We will also go over some of the contemporary tools provided by RStudio to assist with collaboration and replication. We will finish by going through the “tidyverse” tools for data setup, management and visualization. Students will work through examples using the 2016 NES.

- **Day 2** – Students will be introduced to basic programming, including conditional logic, loops, and functions. We will work through an application to webscraping, where students will be asked to design their own functions to assist in processing unstructured data. We will also work through a series of visualizations so students will gain experience with a variety to visualization situations, including generation of maps.

- **Day 3** – Students will be introduced to markdown language to share their work with others (and even automate some of the process of book/article/blog writing). They will also be introduced to tools for automatic generation of tables and figures to make creating and updating manuscripts easier. Finally, we will cover an example of application creation using Shiny, and we will cover topics specifically chosen by class participants to address their particular research needs.