CSSS508: Introduction to R for Social Scientists

University of Washington

Charles Lanfear

Autumn 2017

Class Locations

Lecture

Savery 117: Wednesdays 3:30-5:20

Lab (Optional but highly recommended)

Savery 117: Mondays 3:30-5:20

My Office Savery 240: By appointment

Overview

This course is intended to give students a foundational understanding of programming in the statistical language R. This knowledge is intended to be broadly useful wherever you encounter data in your education and career. General topics we will focus on include:

- Organizing projects and creating reproducible research
- Exploring data with graphics and summaries
- Cleaning and preparing data for analyses
- Linking data sets together

By the end of this course you should feel *confident* approaching any data you encounter in the future. We will cover virtually no statistics, however it is the intention that this course will leave you prepared to progress in CSSS or STAT courses with the ability to focus on *statistics* instead of *coding*. Additionally, I hope the basic concepts you learn will be applicable to other programming languages and research in general, such as logic and algorithmic thinking.

Prerequisites

No specific courses are required and I anticipate this will be the first course in which most students encounter programming. Programming is difficult and can be learned effectively only through time, effort, and exposure. Trial and error is a key aspect of learning to code and you will *always* find yourself searching for answers or asking for help. Therefore the only prequisites for this course are the capability to work hard at something difficult and a willingness to ask for help and provide help when asked.

Course Website

clanfear.github.io/CSSS508/

The course website is accessible without a UWnetID and features all of the slides, R code, and assignments. It will be updated continually throughout the quarter, and available in full after the term has ended as a reference.

Labs

Lab time is scheduled in the same room as lectures (Savery 117) on Mondays from 3:30 to 5:20. Attendance is optional for labs and this time should be considered a form of extended office hours and collaborative work time. You are highly encouraged to attend and ask questions about homework, lectures, or programming in R in general.

Extra Help

Mailing List

This course features a mailing list for asking questions related to homeworks. I encourage you to use this list as your primary means of answering technical questions. As you develop your skills, you will find that learning *how* to ask questions well makes finding answers much easier. To use the mailing list, send your email to `cs_ss508a_au17@uw.edu` and it will be sent to the instructor and all students in the class. Unless you specifically request otherwise, emailed questions directed to me (clanfear@uw.edu) that may be useful to others in the class will be answered with a response to the class mailing list. Please note that this course has a large number of students but no teaching assistant, so students are both encouraged and expected to assist one another with technical problems, both in and out of class. Diagnosing problems inothers' code is a very effective way to improve your understanding of programming.

Tutorial Package

This course also features an R package containing interactive tutorials for practicing concepts from lecture. Completion of tutorials is recommended to check your understanding but is not required. The package is in an early stage of development, so please email me if you find bugs or problems. Installation instructions are found in the Week 1 slides.

Course Requirements

This course is graded as credit / no credit. To pass you need to receive at least 60% of the available points. There will be approximately 7 graded homeworks, one nearly every week. There are 4 points possible for each week that features a homework.

Homework: 75% (3 points)

These must be turned in as knitted markdown documents which we will learn to create and for which templates are provided. They will be graded on a 0 to 3 point scale based on a simple effort-focused rubric found on the course website. These are designed first and foremost to develop skills rather than "prove" you have learned concepts. I encourage you to communicate and work together, so long as you write and explain your code yourself and do not copy work wholesale. You can learn a lot from replicating others' code but you will learn nothing if you copy it without knowing how it works.

Peer Review: 25% (1 point)

Each week an assignment is due, students will be randomly assigned to grade another student's assignment following the 0 to 3 rubric. They will be expected to provide constructive feedback and commentary if something new was learned; reading others' code is an important skill and you will write better code knowing others will see it. These reviews will be due midnight prior to the following class meeting. Each peer review is worth 1 point and will be evaluated by the instructor on a binary "good" / "not good" scale.

Course Texts

There are no texts you are required to purchase for this course. The main recommended texts are available online:

- *R for Data Science* by Garrett Grolemund and Hadley Wickham, a great general introduction to R programming for data management and analysis.
- *Advanced R* by Hadley Wickham, a deeper look into programming in R.
- *The R Inferno* by Patrick Burns, an amusing look at unintuitive, frustrating, and confusing elements of R programming and how to overcome them with best practices. To quote, "If you are using R and you think you're in hell, this is a map for you."

Additional recommended readings will be posted on the website over the term. The following books or resources are recommended and may prove useful. If purchasing, remember to shop around and avoid previous editions as R is a constantly maturing language:

- *Hands on Programming with R* by Garrett Grolemund, an introduction to R focused on learning to program.
- *The Art of R Programming* by Norman Matloff, a highly readable general text on programming in R.
- *R Graphics* by Paul Murrell, the definitive text on graphics in R. Vital if you want to get deeper into `base` or `grid` (the system underlying `ggplot2`) graphics.
- R-Bloggers: A blog of news and tutorials about R with multiple updates per day and thousands of archived posts.
- RStudio Cheat Sheets: Handy reference sheets covering a variety of common R tasks and packages.
- StackOverflow, the internet's largest programming help community. Nearly any question you have about programming in R has probably already been asked and answered here, and if not, this is the place to ask.

Software

This is a course in R programming, so you will be expected to use R. You are welcome to use lab computers, however R and RStudio are free software, so you are strongly urged to use your own computer so that you will be more comfortable and have easy access. In terms of R packages, this course will ephasize the `tidyverse` family of packages for manipulating and displaying data. You can acquire R from the Comprehensive R Archive Network (CRAN) and RStudio from the RStudio home page(you want the free RStudio desktop version). The instructor can provide support with installing R or RStudio in office hours or over email.

Course Outline

All topics are subject to change, however a tentative outline is as follows, with the last two weeks reserved for topics requested by students.

Week

1. R, RStudio, and Markdown Documents
2. Plotting with `ggplot2`
3. Manipulating and Summarizing Data with `dplyr`
4. R Data Structures
5. Importing, Exporting, Cleaning Data

6. Loops
7. Vectorization and Writing Functions
8. Manipulating Text and Regular Expressions
9. Spatial Data and Mapping
10. Social Media Data