

Syllabus for CS&SS 508

Introduction to R for Social Scientists

University of Washington

Instructor: Breon Haskett (mailto:breonh@uw.edu)

Class Locations

Lecture - SAVERY 117

Wednesdays 3:30-5:20

Lab (Optional but highly recommended) - SAVERY 117

Mondays 3:30-5:20

Breon's Office Hours - ZOOM Link (<https://washington.zoom.us/j/94740190466>)

Tuesdays 09:00-11:00

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:

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

By the end of this course you should feel *confident* approaching any data you encounter in the future. We will cover almost 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 prerequisites 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 (https://breonh.github.io/csss_508/)

breonh.github.io/csss_508/ (https://breonh.github.io/csss_508/)

The course website is accessible without a UWnetID and features lecture 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 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. Beginning in week 5, we will walk through homeworks in lab. This gives students an opportunity to be guided through more challenging tasks.

Extra Help

This course features both a Slack channel and a mailing list for obtaining extra help. As you develop your skills, you will find that learning *how* to ask questions well makes finding answers much easier. Please note that this course has a large number of students but no teaching assistant, so students are encouraged and expected to assist one another with technical problems, both in and out of class. Diagnosing problems in others' code is a very effective way to improve your understanding of programming.

Mailing List (mailto:csss508a_au21@uw.edu)

This course features a mailing list for asking questions. I encourage you to use this list as your primary means of answering *long technical questions*. To use the mailing list, address your email to csss508a_au20@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 (breonh@uw.edu (<mailto:breonh@uw.edu>)) that may be useful to others in the class will be answered with a response to the class mailing list.

Slack Channel (<https://uwcsss508.slack.com/>)

We will also use a Slack channel for collaborative communication. You may ask and answer questions in the Slack channel instead of the mailing list. Slack is best-suited for *short technical questions* or *extended conversations*. Your @uw.edu email must be used to register for the channel.

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 (https://breonh.github.io/csss_508/homework.html): 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

(https://breonh.github.io/CSSS508/docs/peer_review.html): 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* (<http://r4ds.had.co.nz/index.html>) by Garrett Grolemund and Hadley Wickham, a great general introduction to R programming for data management and analysis.
- *Advanced R* (<http://adv-r.had.co.nz/>) by Hadley Wickham, a deeper look into programming in R.
- *The R Inferno* (<http://www.burns-stat.com/documents/books/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* (<http://shop.oreilly.com/product/0636920028574.do>) by Garrett Grolemund, an introduction to R focused on learning to program.
- *The Art of R Programming* (<https://www.nostarch.com/artofr.htm>) by Norman Matloff, a highly readable general text on programming in R.
- *R Graphics* (<https://www.crcpress.com/R-Graphics-Second-Edition/Murrell/p/book/9781439831762>) 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.
- RStudio Cheat Sheets (<https://www.rstudio.com/resources/cheatsheets/>): Handy reference sheets covering a variety of common R tasks and packages.
- R-Bloggers (<https://www.r-bloggers.com/>): A blog of news and tutorials about R with multiple updates per day and thousands of archived posts.
- StackOverflow (<https://stackoverflow.com/>), 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 emphasize the `tidyverse` (<https://tidyverse.org>) family of packages for manipulating and displaying data. You can acquire R from the Comprehensive R Archive Network (CRAN) (<https://cran.rstudio.com/>) and RStudio from the RStudio home page (<https://www.rstudio.com/products/rstudio/download/>) (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

1. RStudio and Markdown
2. Visualizing Data
3. Manipulating and Summarizing Data
4. Understanding R Data Structures
5. Importing, Exporting, Cleaning Data
6. Using Loops
7. Writing Functions
8. Working with Text Data
9. Working with Geographical Data
10. Reproducibility and Model Results

Religious Accomodation

Washington state law requires that UW develop a policy for accommodation of student absences or significant hardship due to reasons of faith or conscience, or for organized religious activities. The UW's policy, including more information about how to request an accommodation, is available at Religious Accommodations Policy (<https://registrar.washington.edu/staffandfaculty/religious-accommodations-policy/>). Accommodations must be requested within the first two weeks of this course using the Religious Accommodations Request form (<https://registrar.washington.edu/students/religious-accommodations-request/>).

Disability Resources

Your experience in this class is important to me. If you have already established accommodations with Disability Resources for Students (DRS), please communicate your approved accommodations to me at your earliest convenience so we can discuss your needs in this course. If you have not yet established services through DRS, but have a temporary health condition or permanent disability that requires accommodations (conditions include but not limited to; mental health, attention-related, learning, vision, hearing, physical or health impacts), you are welcome to contact DRS at 206-543-8924 or uwdrs@uw.edu (uwdrs@uw.edu) or disability.uw.edu (disability.uw.edu). DRS offers resources and coordinates reasonable accommodations for students with disabilities and/or temporary health conditions. Reasonable accommodations are established through an interactive process between you, your instructor(s) and DRS. It is the policy and practice of the University of Washington to create inclusive and accessible learning environments consistent with federal and state law.