Stat 610: Statistical Computing

Meeting time: Mondays and Wednesdays, 11:30am-12:45pm Meeting location: HU 217
Website: jfukuyama.github.io/teaching/stat610

Instructor: Prof. Julia Fukuyama jfukuyam at iu dot edu
Office hours: Tuesdays and Thursdays 12-1pm TBA

Associate Instructor: Mr. Yue Yu yyu3 at iu dot edu
Office hours: TBA TBA

Occasional lab: Fridays 3-4:15pm, LH 135.

Course Overview

As a statistician, you will need to manipulate data, optimize, and simulate. You will also need to know enough about how the methods you use work to diagnose problems when they arise and to be able to implement modified versions when the standard implementations don't suit your purposes.

You also need to write accurate, clean, maintainable, demonstrably correct code. To that end, the first half of the class will be devoted to how to program well, with statistical tasks giving us the computational problems.

Once we have the software engineering down, we will move on to the algorithms used in applied statistics. These can be roughly broken up into optimization methods and stochastic simulation methods.

A couple times throughout the semester, there will be a lab. This will involve a script that you can go through with the TA involving some more complicated material.

Textbooks

The primary textbook for the first half of the course with be *The Art of R Programming*, by Norman Matloff. *The R Cookbook*, by Paul Teetor, will also be useful. The primary textbook for the second half of the course will be *Numerical Analysis for Statisticians*, by Kenneth Lange. Additional readings will be posted on the course website.

Class Schedule

Dates and topics subject to change.

Week 1

- Data types and data structures

Week 2 (Labor day on Monday)

- Flow control and looping

Week 3

- Regular expressions
- Text representations and data frames

Week 4

- Writing and calling functions
- More functions

Week 5

- Split/apply/combine I
- Split/apply/combine II

Week 6

- Shape changing/merging/transformations
- Debugging

Week 7

- Testing
- Top-down design

Week 8

- Version control/git I
- Version control/git II

Week 9

- Performance enhancement and code profiling
- Midterm exam

Week 10

- Fitting and using statistical models
- Newton's method

Week 11

- Newton's method II/IRLS
- EM

Week 12

- Generating random deviates
- Monte Carlo integration

Week 13

- ABC
- Markov chains

Week 14

- Metropolis Hastings
- MCMC Applications

Week 15

- Final project presentations/code review

Assessment

Assessment will be based on a combination of homework, an in-class midterm, and a final project. Final grades will be based on:

- 15% homework
- 35% midterm exam
- 40% final project
- 10% participation

Full points for participation can be obtained by participating in class or by sending me mail (3x over the course of the semester) about mistakes in the notes.

- Showing up to class consistently is worth 6 of 10 participation points, with the remainder being more active participation (asking/answering questions).
- If you send me mistakes in the notes, for them to count towards participation you will need to include the keywords "610", "participation", and "correction" in the email, preferably in the subject line.

For homework, the plan is:

- There will be homeworks assigned most weeks.

- Homeworks will be graded out of 2 points, with 2 points being a good-faith effort, 1 point being "something was turned in but it is embarrassing," and 0 points for not turning anything in. Homework has been downweighted relative to previous years due to the relative ease of cheating on coding assignments.
- Homeworks will generally be due on Wednesdays.
- At the time the homework is assigned, we will often not have covered all the material needed to complete the homework, but we will have covered everything by the week before the due date. The idea is to give you the homework early enough that you can think about it while the material is being covered in lecture. Therefore, it will generally be a good idea to take a look at the homework when it is assigned even if you aren't able to complete all the problems yet.
- If you have any concerns about the format for the homework, feel free to email them or to submit them (anonymously) at the form here.

Course Policies

Late Policy

Each student has five “free” late days to use on assignments. After that, homework is penalized at one point (out of five, remember!) per 24 hours. Special accommodations may be granted if you ask very early or if there are extenuating circumstances.

Academic Integrity

You are expected to abide by the guidelines of the IU Code of Student Rights, Responsibilities, and Conduct (http://studentcode.iu.edu/responsibilities/academic-misconduct.html) regarding cheating and plagiarism. Any ideas or materials taken from another source must be fully acknowledged and cited.

For several assignments, you will be asked to make a good-faith effort at completing the assignment without the use of generative AI and then to compare with code an AI provides. However, if not specified in the assignment text, you should not use generative AI to write code for you. Doing so can make sense in other contexts, but in a course, where the purpose is to learn to code better in R, it is better to struggle a bit without an AI to smooth things over.

Disability Accommodation

Please contact me if you require assistance or academic accommodations for a disability. You should establish your eligibility for disability support services through the Office of Disability Services for Students in Wells Library W302, 812-855-7578.