STA 3100: Programming with Data in R

Spring 2024
Course Meetings: M, W, F, Period 7 (1:55 PM-2:45 PM): Turlington 2319
Course Website

Instructors

- **Lead Instructor:** Josh Miles
  - **Email:** miles.j@ufl.edu or Canvas message
  - **Office:** Griffin-Floyd Hall 100
  - **Office hours**: Monday, 3 PM-4 PM — Thursday, 9:30 AM-11:30 AM, or by appointment
- **Teaching Assistant:** Animesh Mitra
  - **Email:** a.mitra@ufl.edu or Canvas message
  - **Office:** Griffin-Floyd Hall 209
  - **Office hours**: 3-4 F, 2-4 Tues, or by appointment

Course Objectives

This course is intended to introduce students to computer programming in the R language. In particular, the course will cover the following concepts in the R language:

- Data types, objects, and classes
- Libraries, functions, and environments
- Reading, writing, and manipulating data
- Producing numerical and graphical output from code
- Producing various types of reports with integrated programming output

Prerequisites

Any one of the following is required to ensure basic statistical and mathematical background:

- STA 2023 with a minimum grade of B
- STA 3024 with a minimum grade of B-
- STA 3032 with a minimum grade of B-
- an AP statistics score of 4 or 5.

In addition, the course assumes no prior experience with R (nor any other programming language), and it assumes no knowledge of computer science concepts more broadly.

A note about learning to program

In my experience, learning to program is considerably different than the way we learn most academic subjects. Most of the time, we learn in a "bottom up" manner: we begin with the most basic things and gradually build up to more complex and specific subjects. For example, in algebra, you first learn the most basic things, like adding, subtracting, multiplying, etc., and then you learn to do more complex operations like logarithms and trigonometric functions. Programming, on the other hand, is practically impossible to learn without first taking for granted several principles in order to focus on specific concepts. This is more of a "top-down" approach, and it adds an extra challenge to learning. As such, it's very normal to feel overwhelmed and uncapable in the beginning. I encourage you to remain patient and trust the process. Everything will make more sense with time. As we progress, we'll be able to understand more of the basic things that we have to take for granted at the beginning.

Materials

All student materials used in this course are free. The course will not use a textbook, instead following notes prepared by the instructor and other faculty. The course will also make heavy use of free software and online materials. The volume of how-to guides for all sorts of projects in R is quite extensive, and the course will sample many of these as is seen fit. Some of the most popular, streamlined, and thorough sources for learning R programming are listed below, but many other very good sources are available.

- RDocumentation
- R Markdown: The Definite Guide
- R for Data Science

Computational Requirements

Students will need to have frequent and reliable access to a computer capable of running R code. All computers supplied by UFIT Academic Technology classrooms and laboratories, including those at the university libraries, have R and RStudio installed and perform well enough to run any program of interest to this course without any trouble. However, because of data security policies, it's somewhat tedious (though possible) to efficiently manage an R installation on university computers. For these reasons, students with personal computers are encouraged to use them for programming tasks.

Students who wish to use personal computers will likely have no trouble with the computational requirements necessary for R and RStudio. You will need the following to install R and RStudio:

- For Windows users, Windows 10 or later
- For macOS users, macOS 10.13 (High Sierra) or later
- Other hardware requirements (e.g. RAM, processor speed) will be satisfied if you meet the operating system requirements.

On Windows, R requires about 179 MB of space, and RStudio requires about 861 MB. I can't readily check the storage requirements on other operating systems, but I suspect that each software will be similar in size to the sizes on Windows.

You will also need a handful of storage space to hold downloaded packages, datasets, and output that you produce. I suggest having at least 5 GB of storage available once you have downloaded R and RStudio.

Please contact the instructor if you have any concerns about the technology requirements of the course.

Attendance

Attendance at lectures will not be recorded nor graded. Students can miss lectures for any reason (except for your final exam presentation, for which you must be present in the lecture during which your presentation is scheduled). Course notes will be posted on Canvas, along with any other supplementary documents or websites used in the lectures. The lectures, which will not be recorded, will follow these notes closely but will often include extra detail and examples that are not in the posted notes. The lectures will therefore be more helpful than the notes alone, and as such, attendance at lectures is encouraged.

Lectures will usually involve both a slide presentation and demonstrations of code writing and execution. Students who have laptops are encouraged, though not required, to bring them to follow along with the code writing, so that they can take their own notes in their code files and experiment with the code beyond what is shown in lecture. The code files will be posted after lecture for students' reference, though they likely will be of limited help for students who did not attend the corresponding lecture.

This course will also abide by the university attendance policies.

Rough Schedule

The following is a *rough* outline of subjects that the course will cover, by week. The time spent on certain subjects will probably often be substantially more or less than what is estimated here, and certain subjects will be reiterated and expanded upon throughout the course.

- Week 1: Syllabus, RStudio and R basics, data types
- Week 2: Objects, functions and libraries, data structures
- Week 3: Basics of data manipulation, basic statistics on datasets
- Week 4: Data manipulation and organization
- Week 5: Function writing, graphical output
- Week 6: Introduction to RMarkdown and LaTeX
- Week 7: RMarkdown and LaTeX
- Week 8: Tidyverse data operations
- Week 9: Tidyverse data operations
- Week 10: Spring break
- Week 11: Regular expression operations
- Week 12: Writing code with natural language AI (e.g. ChatGPT), statistical modeling
- Week 13: Statistical modeling
- Week 14: Simulation
- Week 15: Data fetching operations
- Week 16: Project presentations

Assignments and Grading:

The course will have 5 homework assignments, 5 brief practice assignments, and 2 projects. All assignment submissions will be completed via Canvas and will require access to R and RStudio.

Homework Each assignment will be worth 10% of the final grade. Students will be given roughly 10 days to complete each assignment. You are allowed to consult any online resources that you can find, and you may consult with other students for help with the homework. You may not, however, copy answers from other students, nor have them write any code for you at any point while completing homework. Accordingly, all students will need to submit their own copy of each homework. If you have questions about the resources allowed for homework, please direct them to the instructor. If you have questions about the grade or feedback that you receive on homework, please direct them to the TA (who will grade and provide feedback for the homework).

Homework assignments will not be assigned concurrently with projects.

Practice assignments Each practice assignment will be worth 2% of the final grade. These will be similar to the homeworks content-wise, though substantially smaller and less challenging. Their purpose is to allow you to practice concepts discussed in lecture before the homework is assessed. You will be allowed approximately 3 days to complete each practice. The rules concerning resources and help for these are the same as those for the homework assignments Please direct questions about grades and feedback on practice assignments to the TA.

Projects There will be 2 projects assigned in the semester: one to be turned in just before spring break, worth 15% of the final grade, and the other to be presented before the class in the final week of lectures, worth 25% of the final grade. Projects can be done individually or in groups of 2 or 3 students, at the discretion of students. For students working in groups, only one submission per group is required. Groups will also need to include a brief statement of each member's contributions. Each member will be graded individually based on their contributions.

For the final project, each group will need to submit a proposal for their project that specifies the expected content of the project. The projects need not follow their proposals strictly, but please have your projects roughly follow the outline that you give. If you are forced to make a large revision to your project, please consult the instructor.

Please direct questions about grades and feedback on the projects to the instructor (who will grade them).

A note to new programmers You might think you're being clever by copying a friend's code and then changing up a few variable names, comments, and other small details. Think twice here. The instructor and TA have seen hundreds of files of student code, and we can see through those tricks pretty easily.

A statement about assignment and grading philosophy for this course This course is intended to give you a working knowledge of the real world process of writing code and producing useful output. It's also meant to get you comfortable using the huge wealth of resources at your side to help you succeed. Rarely, if ever, will you be asked in real life to write code that needs to be correct on the first try, with no troubleshooting, in a handful of minutes. Accordingly, I find it unnecessary to assess students in such a manner. This course is also not intended to make you struggle for a good grade. I intend it to function as a means for you to expand your skills and resources as a programmer and data analyst.

Late Work In an effort to simplify communications about extensions to assignment deadlines, this course will implement a "Late Days" policy for homework. This policy is as follows:

- Students will be given 3 days (72 hours) of late submission time. These hours act as a "bank" of time to "pay" for lateness on assignments, regardless of the reason for late submission.
- Late time is cumulative across assignments. For example, if you submit your first homework 24 hours late, you will have 48 remaining late hours to apply to any combination of your remaining homework assignments.

- Late hours will be calculated individually per assignment, and will be rounded to the nearest hour.
- If you have remaining late hours, they will be applied by default to a late assignment. If you wish not to use your late hour budget for a given assignment, please state so in a comment when you submit the assignment.
- Late hours will be visible to students in Canvas.
- If you have an **excused reason** for late submission, and provide appropriate documentation or otherwise receive approval from the instructor, you will be given an alternate deadline for which your late hours will apply.

If a student has used all of their late hours, or if the student chooses not to use late hours for a given assignment:

1. If the student does not have an excused reason for late submission:
    - Assignments that are 0-24 hours late will be given a 20% penalty.
    - Assignments that are 24-48 hours late will be given a 40% penalty.
    - Assignments that are more than 48 hours late will not be accepted and will be given a 0% grade.
2. If the student does have an excused reason for late submission, the instructor will arrange an alternate deadline for the student in consideration of their circumstances. Beyond this deadline, the above late penalties will apply.

Finally, any circumstances regarding late hours and/or late assignments not described here will be handled according to the discretion of the instructor. Please direct all inquiries or concerns about late work policy to the lead instructor.

Final Grades

- 5 homework assignments (10% each): 50%
- 5 practice assignments (2% each): 10%
- 2 projects (15% and 25%): 40%

The grading scale below will be used.

Final percentage grades will be rounded to the nearest percentage point when assigning letter grades. Any decision to curve grades will be made at the discretion of the instructor. In this case, grades would only be curved upward, and the scale above will be used for the curved grades.

Grade points will be assigned to letter grades according to the university's grading policy.

Important Dates:

1. Monday, January 15: NO CLASS (University holiday)
2. March 9-17: NO CLASS (Spring Break)
3. Wednesday, April 23: Last day of regular class
4. Wednesday, May 1, 10:00 AM-12:00 PM: the scheduled final exam time for this course. I am not planning to use this time, unless students prefer to present their final project during this then. We will discuss this possibility closer to the end of the semester.

A	94-100%
A-	90-93%
B+	87-89 %
B	83-86 %
B-	80-82%
C+	77-79%
C	73-76%
C-	65-72%
D+	57-64%
D	50-56%
F	$\leq$ 49%

Accommodations

Students with disabilities who experience learning barriers and would like to request academic accommodations should connect with the disability Resource Center. It is important for students to share their accommodation letter with their instructor and discuss their access needs, as early as possible in the semester.

Course Evaluation

Students are expected to provide professional and respectful feedback on the quality of instruction in this course by completing course evaluations online via GatorEvals. Guidance on how to give feedback in a professional and respectful manner is available at `https://gatorevals.aa.ufl.edu/students/`. Students will be notified when the evaluation period opens, and can complete evaluations through the email they receive from GatorEvals, in their Canvas course menu under GatorEvals, or via `https://ufl.bluera.com/ufl/`. Summaries of course evaluation results are available to students at `https://gatorevals.aa.ufl.edu/public-results/`.

Academic Honesty:

All instances of academic dishonesty will be handled according to the university's academic honesty policies (see pages 11-14, in particular).