

PPHA 30536: Data and Programming for Public Policy II

Jeff Levy
levyjeff@uchicago.edu
Keller 3101

Winter Quarter, 2021

Course Information

January 11th - March 20th, 2021

Asynchronous lecture posted every Monday and Wednesday AM

Labs (on Zoom, all times CST)			
Monday	10:50	AM	Jeff Levy
Tuesday	2:00	PM	Molly Bair
Wednesday	10:50	AM	Jeff Levy
Thursday	4:00	PM	Merritt Smith
Friday	8:00	AM	Molly Bair
Friday	9:00	PM	Merritt Smith

Office Hours

Please email me to schedule meetings over Zoom. I am available most days.

Teaching Assistants

Merritt Smith - merrittsmith@uchicago.edu
Molly Bair - mgbair@uchicago.edu

Prerequisites

The course PPHA 30535, Data and Programming for Public Policy I, is required to take this course. If you did not take PPHA 30535, or took PPHA 30535

sections 3 or 4 (taught in Python), you should email me for approval, and may be required to demonstrate proficiency with R.

Course Objectives

This course will build directly on the material covered in PPHA 30535. We will assume a grasp of the R skills from the previous class at the start, so that we can focus on practical applications to research. Whereas the goal of the first class was to introduce R and ggplot as tools for data analysis, and to prepare students for internship-level policy research positions, the goals of this course will be to:

1. Go from applying R to structured questions with clearly-defined answers, to using R to solve broad research questions
2. Deepen existing skills; for example, we will spend more time using dplyr and functions
3. Broaden into new skills that require a higher level of R proficiency
4. Prepare for the post-graduation job market

Software and Resources

Most of the software and resources for this class are identical to the R sections of PPHA 30535 in the spring.

I will be using R Studio throughout the course, though you may write in any platform that supports R code. Homework and projects will be retrieved and submitted through GitHub.

For the three lectures and one homework covering Python, you will need to install the latest version of Anaconda Python. If you have older versions of Python already installed, it may be easier to uninstall it and reinstall a new version to ensure library compatibility.

Attendance

You do not have to be on campus to take this class.

Remote attendance to a minimum of one lab per week is mandatory, and graded. The labs run by the TAs and by me are interchangeable for the purposes of this grade, and are listed at the top of the syllabus. You may also, of course, attend more than one in a given week if you choose.

If you experience issues with attending class or completing work, please speak with me directly so we can find an accommodation.

Academic Integrity

All code you turn in must be your own. Do not share your code with your classmates, or ask others for theirs. That said, the practice of writing code is very often a collaborative one. To avoid academic dishonesty, and a potential failing grade, please follow these guidelines:

1. You MAY search for help online (e.g. StackOverflow)
 - You must always cite the source by leaving a link to it in the comments of your code
 - You MAY NOT copy verbatim - find inspiration and then rewrite it
 - You MAY NOT take solutions to problem sets from online
2. You MAY work with your classmates
 - You must always cite the individuals you collaborate directly with by including their names in the comments at the top of your program
 - You MAY NOT share or look at each other's code
 - You MAY share output (e.g. plots or error messages)
 - You MAY discuss concepts and theory (e.g. using a whiteboard)
3. You MAY participate in discussions on Piazza
 - You MAY share generic or pseudo code, and ideas
 - You MAY NOT share specific code from your own work
4. When explicitly allowed, you MAY work in groups
 - If groups are optional, you must declare your group the day the assignment is given
 - You will collaborate, share code, and submit only one assignment

It is very important that you use proper citations. If you turn in an assignment that the grader deems to be too unoriginal (e.g. your solutions too closely follow a solution found online, or another classmate), but you have cited all the sources, then you may be allowed a chance to redo your work. If the same thing happens but you have not cited the sources, you will receive a failing grade and possibly be subject to other sanctions under the university's Academic Integrity guide.

The above rules apply to interactions with your classmates and the internet. You may present your code and questions to the professor or the TAs at any time.

Homework, Exams, Due Dates, and Grading

There are no exams for this class. Your grade will consist of assignments, lab attendance, and a project. All assignments and projects are due on the specified day before midnight, over GitHub Classrooms.

- Homework 1: *Syllabus, Project, and Irregular Data* - Jan 11th-Jan 24th
- Homework 2: *Plotting* - Jan 25th-Feb 7th
- Homework 3: *Text and NLP* - Feb 8th-Feb 21st
- Homework 4: *Python and Other Languages* - Feb 22nd-Mar 7th
- Final Project: Due March 17th

Your final grade will be calculated as 50% assignments, 40% final project, and 10% weekly lab attendance. A minimum of 60% is required to pass this course. Among those who pass, final grades will use the following curve: 1/3 A, 1/4 A-, 1/4 B+, 1/12 B, 1/12 B-.

Late Policy

Each student gets three “late tokens” to cover the entire quarter. **Each token grants a 12 hour extension to the due date of any assignment or the final project, and can be used at your discretion and without any need for explanation.** Turning in an assignment late will result in the automatic use of as many of your three tokens as are needed to cover the time. Any late time that cannot be covered by these three tokens will result in a 10% decrease in score per 12 hour block. Note that there is no fractional usage; turning something in 13 hours late will result in the loss of two tokens, or one token and 10%, or no tokens and 20%.

The only possible exception to this late policy is if an issue is serious enough to involve Student Affairs, in which case your situation will be handled on an individual basis after consultation.

Course Outline

This exact outline is tentative, and may vary with class pacing and feedback.

Week 1 - Homework 1 Material

- January 11th - Introduction, project discussion
- January 13th - Difficult and irregularly-shaped data 1

Week 2

- January 18th - No class for Martin Luther King Day
- January 20th - Difficult and irregularly-shaped data 2

Week 3 - Homework 2 Material

- January 25th - Spatial data 1
- January 27th - Spatial data 2

Week 4

- February 1st - Principles of dataviz
- February 3rd - Interactive plotting

Week 5 - Homework 3 Material

- February 8th - Parsing PDF documents
- February 10th - Text and language processing 1

Week 6

- February 15th - Text and language processing 2
- February 17th - Text and language processing 3

Week 7 - Homework 4 Material

- February 22nd - Intro to Python 1
- February 24th - Intro to Python 2

Week 8

- March 1st - Intro to Python 3
- March 3rd - Other common data languages

Week 9 - Final Topics

- March 8th - Introduction to big data and other R topics
- March 10th - Code samples and job prep

Week 10

- Finals week - no classes
- March 17th - Final project due