



# Intro to Programming and Data Science (Fall 2022)

## Course Information [edit]

**COM 674: Introduction to Programming and Data Science**

**Location:** BRNG 2273

**Class Hours:** Tuesdays and Thursdays, 12–1:15 PM

## Instructor [edit]

**Instructor:** Jeremy Foote (<https://jeremyfoote.com>)

**Email:** [jdfote@purdue.edu](mailto:jdfote@purdue.edu)

**Office Hours:** Tuesdays, 2–4 pm; BRNG 2156 or by appointment

## Course Overview and Learning Objectives [edit]

This is an exciting time to be a social scientist and especially a Communication scholar! An increasing amount of our lives—and our interactions—are stored digitally. Social scientists are increasingly using that data to ask and answer questions about how the social world works. I firmly believe that computational tools have created a new frontier in the social sciences which those who develop computational skills can explore.

This class is an introduction into that world. The course is intended to give students an introduction to programming principles, the Python programming language, and data science tools and approaches. However, this is not a computer science class or a statistics class, and **no prior programming experience is required or expected**. We will focus on gaining access to data and basic data manipulation rather than complex statistical methods.

The main goal of the class is to help you to complete a preliminary, independent, data-centric project. As part of this project, you (on your own or in a team) will write software to collect data from web APIs, process and clean that data, and produce statistics, hypothesis tests, and graphical visualizations that address questions you are interested in.

I will consider this class a complete success if, at the end, every student can:

- Write or modify a program to collect a dataset from a publicly available data source.
- Read web API documentation and write Python software to parse and understand a new and unfamiliar web API.
- Understand and follow basic version control practices.
- Use digital data to effectively answer a substantively interesting question and to present this data effectively in the context of both a formal presentation and a written report.
- Identify multiple ways that computational approaches are being used for social science research.
- Feel comfortable taking more advanced computational methods courses or learning new techniques on your own.

## Required resources and texts [edit]

### Laptop [edit]

### Contents

- 1 Course Information
  - 1.1 Instructor
- 2 Course Overview and Learning Objectives
- 3 Required resources and texts
  - 3.1 Laptop
  - 3.2 Readings
- 4 Course logistics
  - 4.1 Note About This Syllabus
  - 4.2 Lectures
  - 4.3 Office hours and email
- 5 Assignments
  - 5.1 Research project
    - 5.1.1 Project idea and dataset identification
    - 5.1.2 Project planning document
    - 5.1.3 Project presentation and report
      - 5.1.3.1 The project report
      - 5.1.3.2 The presentation
  - 5.2 Coding Challenges
  - 5.3 Paper Discussions
  - 5.4 Reflection papers
- 6 Grades
- 7 Schedule
  - 7.1 Week 1: Introduction to Python and Computational Thinking (August 23)
  - 7.2 Week 2: Variables, conditionals, and functions (August 30)
  - 7.3 Week 3: Iteration, strings, and lists (September 6)
  - 7.4 Week 4: Reading from and writing to files (September 13)
  - 7.5 Week 5: Dictionaries and Tuples (September 20)
  - 7.6 CATCH UP Week (September 27)
  - 7.7 Week 6: Dataframes and Visualization (October 4)
  - 7.8 Week 7: Dataframes and visualization (continued) (October 13)
  - 7.9 Week 8: Collecting Data with APIs (October 18)
  - 7.10 Week 9: Collecting Data with APIs (continued) (October 25)
  - 7.11 Week 10: Introduction to Computational Text Analysis (November 1)
  - 7.12 Week 11: Data cleaning and operationalization (November 8)
  - 7.13 Week 12: Organizing and storing computational projects (November 15)
  - 7.14 Week 13: Statistical summaries and tests (November 29)
  - 7.15 Week 15: Final Project Presentation (December 6)
  - 7.16 Week 16: Final Paper Due (December 15)
- 8 Additional Resources
- 9 Administrative Notes

Windows, Mac OS, and Linux are all fine but an iPad or Android tablet won't work. We're going to install software during the class that requires about 5GB of extra space and you'll be working on projects for homework. For the classroom assignments, nothing is terribly intensive, but your own data collection could tax the resources of an older machine.

If for some reason your laptop dies mid-course, please contact me so we can get your new one up to speed.

9.1	Attendance Policy
9.2	Incomplete
9.3	Academic Integrity
9.4	Nondiscrimination
9.5	Students with Disabilities
9.6	Emergency Preparation
9.7	Mental Health
10	Acknowledgements

## Readings [edit]

- Required text: **Python for Everybody** (<https://www.py4e.com/book>) [HTML Version (<https://www.py4e.com/html3/>)] [PDF version ([http://do1.dr-chuck.com/pythonlearn/EN\\_us/pythonlearn.pdf](http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf))] by Charles R. Severance. The book is freely licensed (<https://creativecommons.org/licenses/by/3.0/us/>) and available online for free. You can also buy the book if you prefer a hard copy.

I will list required chapters in the schedule below. In general, you should expect to spend far more time working on programming tasks than reading. Much like math or other technical courses, this course will build on itself every day. You should make every effort to cover the reading and exercise material every day in preparation for the next day.

- Other readings: Throughout the module we will read and discuss examples of computational social science that I find particularly well done or interesting. Many are available through the Purdue library. I will also make all of them available on Brightspace. If you come across additional examples that you think the class would benefit from, please suggest them to me.
- Optional readings: Matthew Salganik's book 'Bit by Bit: Social Research in the Digital Age' is a wonderful introduction to computational social science. We will not be discussing much of it in class but I highly recommend it.

## Course logistics [edit]

### Note About This Syllabus [edit]

Although the core expectations for this class are fixed, the details of readings and assignments may shift based on how the class goes. As a result, there are three important things to keep in mind:

1. Although details on this syllabus will change, I will not change readings or assignments less than three days before they are due. If you plan to read more than three days ahead, contact me first.
2. Check your email. Because this is a wiki, you will be able to track every change by clicking the *history* button on this page. I will also summarize these changes in announcements that will be sent via Brightspace email.
3. I will ask the class for voluntary anonymous feedback frequently. Please let me know what is working and what can be improved.

## Lectures [edit]

Our class time will follow the "flipped" classroom model. I will provide asynchronous materials (readings, recorded lectures, assignments, etc.) which you will work on before class and we will use our class time to review concepts, identify confusion, and synthesize.

The class sessions will typically follow the same format.

On Tuesdays, we will review the Coding Challenges that were due. Those who were assigned will explain how they solved (or didn't solve!) each problem.

On Thursdays, we will begin by discussing the reading for the day, with discussions generally led by a student. We will then have a discussion about the concepts that are still confusing. Next, we will have by optional co-working time to start on the next assignment.

I try to make the classes useful and to tailor them to the needs of the students, and I highly encourage you to attend as many of our sessions as possible. My teaching style is very conversational and relies on students being willing to speak up to express confusion, seek clarification, or make a well-informed point, so please participate!

## Office hours and email [edit]

- I will hold office hours from 2-4 on Tuesdays and by appointment. If you come with a programming question, I will expect that you have already tried to solve it yourself in multiple ways and that you have discussed it with a classmate. This policy lets me have time to help more students, but it's also a useful strategy. Often just trying to explain your code ([https://en.wikipedia.org/wiki/Rubber\\_duck\\_debugging](https://en.wikipedia.org/wiki/Rubber_duck_debugging)) can help you to recognize where you've gone wrong.
- I am also available by email. You can reach me at [jdfoote@purdue.edu](mailto:jdfoote@purdue.edu) (<mailto:jdfoote@purdue.edu>). I try hard to maintain a boundary between work and home and I typically respond only on weekdays during business hours (~9-5) but during the week I will generally respond within 24 hours.

## Assignments [edit]

There will be three main types of assignments. Each is discussed in detail below but here is a brief summary:

1. **Research Project:** The main outcome of this course will be a research project exploring a social science question using Python, and the bulk of your grade will be based on that project. Submit these via Brightspace
2. **Coding Challenges:** There will be weekly programming assignments that I will ask you to turn in on Brightspace but which will only be graded as complete/incomplete. I will also randomly assign someone to present their solution to each of the problems during our synchronous sessions.
3. **Paper Discussion:** Each week we will read and discuss a paper which uses computational approaches to address social science questions.

## Research project [edit]

As a demonstration of your learning in this course, you will design and carry out a quantitative research project, start to finish. This means you will:

- **Design and describe a plan for a study** — The study you design should involve quantitative analysis and should be something you can complete at least a first pass on during this module.
- **Find a dataset** — You should very quickly identify a dataset you will use to complete this project.
- **Analyze, visualize, report, and interpret your data** — You will do this in both a short paper and a short presentation.
- **Ensure that your work is replicable** — You will need to provide code and data for your analysis in a way that makes your work replicable by other researchers.

*I strongly urge you* to work on a project that will further your academic career outside of the class. There are many ways that this can happen. Some obvious options are to prepare a project that you can submit for publication, that you can use as pilot analysis that you can report in a grant or thesis proposal, and/or that fulfills a degree requirement. I prefer that you do projects on your own but it may be possible to work as a small team (maximum 3 people). Team projects are expected to be more ambitious than individual projects. Preliminary assignments will help you to develop your idea and to get feedback from me and others.

There are several intermediate milestones and deadlines to help you accomplish a successful research project. Unless otherwise noted, all deliverables should be submitted via Brightspace.

## Project idea and dataset identification [edit]

### Due date

September 6

### Maximum length

500 words (~1-2 pages)

Early on, I want you to identify and describe your final project. Your description should be short and can be either paragraphs or bullets. It should include the following:

- An abstract of the proposed study including the topic, research question, theoretical motivation, object(s) of study, and anticipated research contribution.
- An identification of the dataset you will use and a description of the columns or type of data it will include. If you do not currently have access to these data, explain why not and when you will have access (If you need ideas, this page lists some open datasets).
- A short (several sentences) description of how the project will fit into your career trajectory.

## Project planning document [edit]

### Due date

October 18

### Maximum length

~4-5 pages

The project planning document is a basic shell/outline of an empirical quantitative research paper. The planning document should focus around three big questions:

- Why are you planning to do this analysis? Make sure to introduce any background information about the topic, the community, your business, or anything else that will be required to properly contextualize your study.
- How will you get the data to analyze? Describe the data sources will you collect and how they will be collected.
- How will you analyze the data? Describe the visualizations, tables, or statistical tests that you will produce.

One approach that I have found helpful is outlined on this wiki page.

## Project presentation and report [edit]

### Report due date

December 14

### Maximum length

4000 words (~15 pages)

### Presentation due date

December 6 and December 8

### Maximum length

8 minutes

## The project report [edit]

You will write a document or a Jupyter Notebook that will ideally provide the foundation for a high quality short research paper that you might revise and submit for publication. I do not expect the report to be ready for publication, but it should contain polished drafts of all the necessary components of a scholarly quantitative empirical research study. In terms of the structure, please see the page on the structure of a quantitative empirical research paper.

The great thing about a Jupyter Notebook is that it allows you to provide data, code, and any documentation sufficient to enable the replication of all analysis and visualizations. If you choose to write the report as a Word document, then you will need to include the code in a separate file.

Because the emphasis in this class is on methods and because I'm not an expert in each of your fields, I'm happy to assume that your paper, proposal, or thesis chapter has already established the relevance and significance of your study and has a comprehensive literature review, well-

grounded conceptual approach, and compelling reason why this research is important. As a result, you do not need to focus on these elements of the work in your written submission. Instead, feel free to start with a brief summary of the purpose and importance of this research followed by an introduction of your research questions or hypotheses. If you provide more detail, that's fine, but I won't give you detailed feedback on these parts and they will not figure prominently in my assessment of the work.

Jupyter Notebooks do not have all of the tools for citations that Word or LaTeX or even Google Docs have, so while I expect you to cite related work your references section does not need to be as polished as citation management software would make it.

## The presentation [edit]

The presentation will provide an opportunity to share a brief summary of your project and findings with the other members of the class. However, don't treat it as a comprehensive overview of your paper: I would rather you tell a subset of the story well than the whole story in a rushed fashion. For instance, you can give a completely successful presentation by describing the motivation and walking through one plot in your paper. Since many of you will give many research presentations throughout your career, I strongly encourage you to take the opportunity to refine your academic presentation skills.

I anticipate that most people will either create a PowerPoint presentation or will walk us through a simple Jupyter Notebook. All presentations will need to be a *maximum of 8 minutes long*. Concisely communicating an idea in the time allotted is an important skill in its own right. Presentations should be uploaded to the Discussion forum on Brightspace created for this purpose.

## Coding Challenges [edit]

Nearly every week you will have set of coding challenges to complete. These coding challenges will be turned in on Brightspace but will not be graded. I encourage you to work together on these challenges but to make sure that you understand the concepts yourself.

Each day I will randomly select a set of students to share their solutions to a selected exercise. This will involve putting your solution on Piazza at least one hour before the next day's lecture starts, and being prepared to walk us through the solution. If you can't figure out the problem that's been assigned to you, then explain where you got stuck and what you tried. I encourage you to also use Piazza to ask and answer each other's questions as you work on the assignments. We will use some of our lecture time to review the problems and I will make sure that a correct solution is posted by the end of that day. As you will see over the course of the module, there are many possible solutions to many programming problems and my own approaches will often be different than yours. That's completely fine! Coding is a creative act!

While I won't be grading these assignments, I will review a sample of them to look for common difficulties or problems. If you want me to provide specific feedback for an assignment, please let me know and I will be happy to do so.

## Paper Discussions [edit]

Every Thursday we will review a paper that uses computational methods. On the first day, I will ask you to sign up to lead the discussion for one or more of these papers. When leading the discussion, you will prepare a presentation as though you were presenting the paper at a conference and then lead a discussion about it.

I am eager to update the papers that we read in the class. If you would like to propose a different paper that you would prefer to present on, I will almost certainly say yes.

When you are not presenting, I do not expect you to read the week's paper thoroughly. However, you should read it closely enough that you are prepared to discuss it.

## Reflection papers [edit]

As discussed in more detail below, two times during the course I will ask you to respond to a set of reflection questions. These questions are intended to help you to think about what you have learned and accomplished and to craft goals for the remainder of the course. They are also an important way for me to gather feedback about how the course is going so that I can adjust.

## Grades [edit]

This course will follow a "self-assessment" philosophy. I am more interested in helping you to learn things that will be useful to you than in assigning grades. The university still requires grades, so you will be leading the evaluation of your work. At the beginning of the course, I will encourage you to think about and write down what you hope to get out of the course. Three times during the course you will reflect on what you have accomplished thus far, how it has met, not met, or exceeded expectations, based both on rubrics and personal goals and objectives. At each of these stages you will receive feedback on your assessments. By the end of the semester, you should have a clear vision of your accomplishments and growth, which you will turn into a grade. As the instructor-of-record, I maintain the right to disagree with your assessment and alter grades as I see fit, but any time that I do this it will be accompanied by an explanation and discussion. These personal assessments, reflecting both honest and meaningful reflection of your work will be the most important factor in final grades.

I suggest that we use the following rubric in our assessment:

- 25%: class participation, including attendance, participation in discussions and group work, and significant effort towards weekly assignments.
- 5%: Final Project Idea.
- 10%: Final Project Proposal.
- 40%: Final Project paper/Jupyter notebook.
- 20%: Final Presentation including your slides and presentation.

My interpretation of grade levels (A, B, C, D/F) is the following:

A: Reflects work that exceeds expectations on multiple fronts and to a great degree. Students reaching this level of achievement will:

- Do what it takes to learn the programming principles and techniques, including looking to outside sources if necessary.
- Engage thoughtfully with an ambitious research project.
- Take intellectual risks, offering interpretations based on synthesizing material and asking for feedback from peers.
- Sharing work early allowing extra time for engagement with others.

- Write reflections that grapple meaningfully with lessons learned as well as challenges.
- Complete all or nearly all assignments at a high level.

B: Reflects strong work. Work at this level will be of consistently high quality. Students reaching this level of achievement will:

- Be more safe or consistent than the work described above.
- Ask meaningful questions of peers and engage them in fruitful discussion.
- Exceed requirements, but in fairly straightforward ways
- Compose complete and sufficiently detailed reflections.
- Complete nearly all of the programming assignments at a high level

C: This reflects meeting the minimum expectations of the course. Students reaching this level of achievement will:

- Turn in and complete required assignments on time.
- Be collegial and continue discussion, through asking simple or limited questions.
- Compose reflections with straightforward and easily manageable goals and/or avoid discussions of challenges.
- Not complete programming assignments or turn some in in a hasty or incomplete manner.

D/F: These are reserved for cases in which students do not complete work or participate. Students may also be impeding the ability of others to learn.

## Schedule [edit]

**NOTE:** This section will be modified throughout the course to meet the class's needs. Check back in often.

There are links to each day's slides. Note that these are slides from an earlier version of the class and will typically be updated the day of each class.

## Week 1: Introduction to Python and Computational Thinking (August 23) [edit]

**Assignment Due:**

- None

**Required Readings:**

- Intro to Jupyter Notebooks video ([https://www.youtube.com/watch?v=5pf0\\_bpNbkw](https://www.youtube.com/watch?v=5pf0_bpNbkw))

**Agenda:**

- Class overview and expectations – We'll walk through this syllabus.
- Week 1 Coding challenge - Includes installing Python and going through a number of exercises.
- Today's slides ([https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day\\_1/lecture/day\\_1.html](https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day_1/lecture/day_1.html))

**By the end of class you will:**

- Have a working python environment on your personal laptop.
- Have written your first program in the python language.

## Week 2: Variables, conditionals, and functions (August 30) [edit]

**Assignments Due:**

- Finish Week 1 exercises and tutorials
- Fill out this short survey ([https://docs.google.com/forms/d/e/1FAIpQLSfUiGogs2jDXIHaxz1ooVBZFkRF2NdMaf00lgZvk7f69rby9w/viewform?usp=sf\\_link](https://docs.google.com/forms/d/e/1FAIpQLSfUiGogs2jDXIHaxz1ooVBZFkRF2NdMaf00lgZvk7f69rby9w/viewform?usp=sf_link))
- Sign up to be a discussant here (<https://docs.google.com/spreadsheets/d/1uSo-Ya5DghaLu1BYk94EVU2kBVmExRWwOa1586GbFUU/edit?usp=sharing>)
- Week 2 Coding Challenge (turn in on Brightspace)

**Readings (before class):**

- Bit By bit, Introduction (<https://www.bitbybitbook.com/en/1st-ed/introduction/>)
- Python for Everybody, chapters 1-4
- Today's Jupyter Notebook ([https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day\\_2/day\\_2.ipynb](https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day_2/day_2.ipynb)) (Right-click, save, and open in Jupyter)
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014636/View>)

**Agenda:**

- Review Week 1 and Week 2 Exercises
- Today's slides ([https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day\\_2/lecture/day\\_2.html](https://jeremydfoote.com/Intro-to-Programming-and-Data-Science/day_2/lecture/day_2.html))
- Introduce wordplay project

## Week 3: Iteration, strings, and lists (September 6) [edit]

**Assignment Due:**

- Final project dataset and idea (turn in on Brightspace).
- Week 3 Coding Challenge

#### Readings:

- Python for Everybody

```
chapters_to_read = [5, 6, 8]
```

- Today's Jupyter Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_3/day\\_3.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_3/day_3.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014637/View>)
- Foote, J., Shaw, A., & Hill, B.M. (2017). Computational analysis of social media scholarship ([https://jeremyfoote.com/files/foote\\_computational\\_2017.pdf](https://jeremyfoote.com/files/foote_computational_2017.pdf)). In Burgess, J., Poell, T., Marwick, A. (Eds.), *The Sage Handbook of Social Media*. Sage.
  - Discussant: Christina

#### Agenda:

- Programming principles (iteration, strings, and lists)
- Go over last day's assignment
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_3/lecture/day\\_3.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_3/lecture/day_3.html))

## Week 4: Reading from and writing to files (September 13) [edit]

#### Assignment Due:

- Week 4 Coding Challenges

#### Readings:

```
book = open('Python for Everybody', 'r')
for chapter in book:
    if chapter == '7':
        read(chapter)
book.close()
```

- Today's Jupyter Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_4/day\\_4.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_4/day_4.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014638/View>)
- Nelson, Laura K. 2017. "Computational Grounded Theory: A Methodological Framework (<https://doi.org/10.1177%2F0049124117729703>)."  
*Sociological Methods and Research*.
  - Discussant: Chiara

#### Agenda:

- Reading from and writing to files
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_4/lecture/day\\_4.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_4/lecture/day_4.html))

## Week 5: Dictionaries and Tuples (September 20) [edit]

#### Assignment Due:

- Week 5 Coding Challenges
- Do the Twitter Authentication Setup

#### Readings:

- Python for Everybody, chapters 9 and 10
- Today's Jupyter Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_5/day\\_5.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_5/day_5.ipynb))
  - Video walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014639/View>)
- Margolin, D. B., Hannak, A., & Weber, I. (2018). Political Fact-Checking on Twitter: When Do Corrections Have an Effect? (<https://doi.org/10.180/10584609.2017.1334018>) *Political Communication*, 35(2), 196–219.
  - Discussant: Alisa

#### Agenda:

- Dictionaries
- Tuples
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_5/lecture/day\\_5.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_5/lecture/day_5.html))

## CATCH UP Week (September 27) [edit]

#### Readings:

- Shen, C., Monge, P., & Williams, D. (2014). Virtual brokerage and closure: Network structure and social capital in a massively multiplayer online game (<https://libkey.io/libraries/228/articles/5013123/full-text-file>). *Communication Research*. 41(4): 459–480.

Discussant: Ajay

## Week 6: Dataframes and Visualization (October 4) [edit]

### Assignment Due:

- Week 6 Coding Challenges

### Readings:

- Week 6 notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_6/day\\_6.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_6/day_6.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014640/View>)
- Shaw, A., & Hill, B. M. (2014). Laboratories of oligarchy? How the iron law extends to peer production. *Journal of Communication*, 64(2), 215–238. <https://doi.org/10.1111/jcom.12082>

Discussants: Miriam

### Agenda:

- Dataframes and visualization
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_6/lecture/day\\_6.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_6/lecture/day_6.html))

## Week 7: Dataframes and visualization (continued) (October 13) [edit]

### OCTOBER BREAK ON OCTOBER 11

### Assignment Due:

- Week 7 Coding Challenges

### Readings:

- Week 7 notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_7/day\\_7.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_7/day_7.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014641/View>)
- Lazer, D., & Radford, J. (2017). Data ex Machina: Introduction to Big Data (<https://doi.org/10.1146/annurev-soc-060116-053457>). *Annual Review of Sociology*, 43(1), 19–39.
  - Discussant: Paxton and Luhao

### Agenda:

- Visualizations in Seaborn
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_7/lecture/day\\_7.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_7/lecture/day_7.html))

## Week 8: Collecting Data with APIs (October 18) [edit]

### Assignment Due:

- Week 8 Coding Challenges.
  - Video instructions to install tweepy (<https://youtu.be/TASX3evcgG4>)
- First self-assessment reflection is due (on Brightspace).
- Project Planning Document Due

### Readings:

- Intro to APIs Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_8/day\\_8.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_8/day_8.ipynb))
  - (Long) walkthrough of notebook (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014642/View>)
- Twitter V2 Academic API Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/extra\\_topics/twitter\\_v2\\_example.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/extra_topics/twitter_v2_example.ipynb))
  - Video walkthrough (<https://www.youtube.com/watch?v=rQEsls9LERM&lc=UgzEMuqQB8UpkOQr0Hd4AaABAq>)
- Kieran Healy and James Moody (2014). "Data Visualization in Sociology (<https://doi.org/10.1146/annurev-soc-071312-145551>)." *American Review of Sociology*. 40: 105-28.
  - Discussant: Keqin

### Agenda:

- Introduce the requests (<https://2.python-requests.org/en/master/>) library
- Discuss the main kinds of online data gathering: downloading, scraping, and APIs.
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_8/lecture/day\\_8.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_8/lecture/day_8.html))

## Week 9: Collecting Data with APIs (continued) (October 25) [edit]

### Assignment Due:

- Week 9 Coding Challenges

### Readings:

- Week 9 Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_9/day\\_9.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_9/day_9.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014643/View>)
- Python for Everybody, Chapter 13
- Christopher A. Bail et al. 2018. Exposure to opposing views on social media can increase political polarization (<https://doi.org/10.1073/pnas.1804840115>). PNAS 115(37): 9216-9221
  - Discussant: Ryan
- If you are interested in doing web scraping, then look at this incredible mini-course on the topic (<https://github.com/CU-ITSS/Web-Data-Scraping-S2019>). It is all done with Jupyter Notebooks and you have all of the prerequisite knowledge to understand it.
- Very brief lecture on web scraping from Spring 2020 (<https://youtu.be/daUuC-PMZc4>).

### Agenda:

- A workflow for doing work with APIs
- Ethics of digital trace data
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_9/lecture/day\\_9.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_9/lecture/day_9.html))

## Week 10: Introduction to Computational Text Analysis (November 1) [edit]

### Assignment Due:

- Week 10 Coding Challenges

### Readings:

- Today's Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_10/day\\_10.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_10/day_10.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014644/View>)
- Sara Klingenstein, Tim Hitchcock, and Simon DeDeo. 2014. The civilizing process in London's Old Baily (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4084475/>). Proceedings of the National Academy of Sciences 111(26): 9419-9424.

### Agenda:

- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_10/lecture/day\\_10.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_10/lecture/day_10.html))

### Resources:

- Two videos of me clumsily solving the Week 8 Problems (<https://youtu.be/FhxZdc1OaNU>)
- My answers to the Week 8 problems ([https://github.com/jdfoote/Intro-to-Programming-and-Data-Science/blob/master/resources/solutions/Twitter\\_answers.ipynb](https://github.com/jdfoote/Intro-to-Programming-and-Data-Science/blob/master/resources/solutions/Twitter_answers.ipynb))

## Week 11: Data cleaning and operationalization (November 8) [edit]

### Assignment Due:

- Week 11 Coding Challenges ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_11/day\\_11.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_11/day_11.ipynb))

### Readings:

- Today's Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_11/day\\_11.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_11/day_11.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014645/View>)
- Robert K. Merton. 1948. The Bearing of Empirical Research Upon the Development of Social Theory ([https://www-jstor-org.ezproxy.lib.purdue.edu/stable/2087142?sid=primo&origin=crossref&seq=1#metadata\\_info\\_tab\\_contents](https://www-jstor-org.ezproxy.lib.purdue.edu/stable/2087142?sid=primo&origin=crossref&seq=1#metadata_info_tab_contents)). American Sociological Review 13(5): 505-515.
- DellaPosta, D., Shi, Y., & Macy, M. (2015). Why Do Liberals Drink Lattes (<https://doi.org/10.1086/681254>)? American Journal of Sociology, 120(5), 1473–1511.
  - Discussant: Dyuti and Bakhti

### Resources:

- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_11/lecture/day\\_11.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_11/lecture/day_11.html))

## Week 12: Organizing and storing computational projects (November 15) [edit]

### Assignment Due:

- Week 12 Coding Challenges

### Readings:

- Video introducing a way to organize code and data ([https://youtu.be/-\\_mjC3lAKL4](https://youtu.be/-_mjC3lAKL4)) (from the Spring 2020 version of the class)
- Git & GitHub Crash Course For Beginners ([https://www.youtube.com/watch?v=SWYqp7iY\\_Tc](https://www.youtube.com/watch?v=SWYqp7iY_Tc)) - YouTube video (not by me) introducing Git



and Github

- Interactive git branching tutorial (<https://learngitbranching.js.org/>)
- Tan, C. (2018). Tracing community genealogy: How new communities emerge from the old (<https://aaai.org/ocs/index.php/ICWSM/ICWSM18/paper/view/17811>). Proceedings of the Twelfth International Conference on Web and Social Media (ICWSM '18), 395–404.
  - Discussant: Woojin

#### Agenda:

- Tour of Github
- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_12/lecture/day\\_12.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_12/lecture/day_12.html))

#### Resources:

### Week 13: Statistical summaries and tests (November 29) [edit]

#### Assignment Due:

- Week 13 Coding Challenges

#### Readings:

- Vitak, J., Shilton, K., & Ashktorab, Z. (2016). Beyond the Belmont Principles: Ethical Challenges, Practices, and Beliefs in the Online Data Research Community (<https://doi.org/10.1145/2818048.2820078>). Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, 941–953.
- (Optional) Williams, M. L., Burnap, P., & Sloan, L. (2017). Towards an Ethical Framework for Publishing Twitter Data in Social Research: Taking into Account Users' Views, Online Context and Algorithmic Estimation (<https://doi.org/10.1177/0038038517708140>): Sociology.
- (Optional) Salganik, M. Ethics (<https://www.bitbybitbook.com/en/1st-ed/ethics/>) chapter from Bit By Bit.
- (Optional) Crawford, K., & Finn, M. (2015). The limits of crisis data: Analytical and ethical challenges of using social and mobile data to understand disasters (<https://doi.org/10.1007/s10708-014-9597-z>). GeoJournal, 80(4), 491–502.
- Week 13 Notebook ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_13/day\\_13.ipynb](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_13/day_13.ipynb))
  - Notebook walkthrough (<https://purdue.brightspace.com/d2l/le/content/598734/viewContent/10014646/View>)

#### Agenda:

- Today's slides ([https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day\\_13/lecture/day\\_13.html](https://jeremyfoote.com/Intro-to-Programming-and-Data-Science/day_13/lecture/day_13.html))

### Week 15: Final Project Presentation (December 6) [edit]

#### Assignment Due:

- Final project presentations
- Prepare a presentation and post it on Brightspace

#### Readings:

- NONE

#### Agenda:

- We will listen to and respond to each other's projects

### Week 16: Final Paper Due (December 15) [edit]

#### Assignment Due:

- Final paper due
- Final self reflection due

## Additional Resources [edit]

These are some topics we touched on in class covered in more depth

- Using Tweepy to do full historical search on Twitter (<https://youtu.be/rQEsls9LERM>)
- Regular Expressions (<https://www.youtube.com/watch?v=K8L6KVGg-7o>)
- List Comprehensions (<https://www.youtube.com/watch?v=3dt40GnU5sM>)
- Network Analysis ([https://youtu.be/flwcAf1\\_1RU](https://youtu.be/flwcAf1_1RU))
- Getting data from Reddit (<https://youtu.be/KBDJhhz4oXA>)
- Classes and Object-oriented programming (<https://www.youtube.com/watch?v=ZDa-Z5JzLYM>) (This is a set of videos)
- Tutorial on syntax parsing in Python (<https://bbengfort.github.io/snippets/2018/06/22/corenlp-nltk-parses.html>) (It's complicated!)

## Administrative Notes [edit]

### Attendance Policy [edit]

Attendance is very important and it will be difficult to make up for any classes that are missed. It is expected that students communicate well in advance to faculty so that arrangements can be made for making up the work that was missed. It is your responsibility to seek out support from classmates for notes, handouts, and other information.

### Incomplete [edit]

A grade of incomplete (I) will be given only in unusual circumstances. The request must describe the circumstances, along with a proposed timeline for completing the course work. Submitting a request does not ensure that an incomplete grade will be granted. If granted, you will be required to fill out and sign an "Incomplete Contract" form that will be turned in with the course grades. Any requests made after the course is completed will not be considered for an incomplete grade.

### Academic Integrity [edit]

While I encourage collaboration, I expect that any work that you submit is your own. Basic guidelines for Purdue students are outlined here (<http://www.purdue.edu/odos/osrr/academic-integrity/index.html>) but I expect you to be exemplary members of the academic community. Please get in touch if you have any questions or concerns.

### Nondiscrimination [edit]

I strongly support Purdue's policy of nondiscrimination (below). If you feel like any member of our classroom—including me—is not living up to these principles, then please come and talk to me about it.

Purdue University is committed to maintaining a community which recognizes and values the inherent worth and dignity of every person; fosters tolerance, sensitivity, understanding, and mutual respect among its members; and encourages each individual to strive to reach his or her own potential. In pursuit of its goal of academic excellence, the University seeks to develop and nurture diversity. The University believes that diversity among its many members strengthens the institution, stimulates creativity, promotes the exchange of ideas, and enriches campus life.

### Students with Disabilities [edit]

Purdue University strives to make learning experiences as accessible as possible. If you anticipate or experience physical or academic barriers based on disability, you are welcome to let me know so that we can discuss options. You are also encouraged to contact the Disability Resource Center at: [drc@purdue.edu](mailto:drc@purdue.edu) or by phone: 765-494-1247.

### Emergency Preparation [edit]

In the event of a major campus emergency, I will update the requirements and deadlines as needed.

### Mental Health [edit]

If you or someone you know is feeling overwhelmed, depressed, and/or in need of mental health support, services are available. For help, such individuals should contact Counseling and Psychological Services (CAPS) at 765-494-6995 during and after hours, on weekends and holidays, or by going to the CAPS office of the second floor of the Purdue University Student Health Center (PUSH) during business hours.

### Acknowledgements [edit]

This course is heavily based on earlier courses taught by Tommy Guy (<https://commlead.uw.edu/team/guy/>) and Mako Hill (<https://mako.cc/>) at the University of Washington as well as a course taught by Laura Nelson (<http://www.lauraknelson.com/p/about.html>) at Northeastern University.

Retrieved from "[https://wiki.communitydata.science/index.php?title=Intro\\_to\\_Programming\\_and\\_Data\\_Science\\_\(Fall\\_2022\)&oldid=251185](https://wiki.communitydata.science/index.php?title=Intro_to_Programming_and_Data_Science_(Fall_2022)&oldid=251185)"

Content is available under Attribution-Share Alike 3.0 Unported unless otherwise noted.