

Department of City and Regional Planning, U.C. Berkeley

CYPLAN255: Urban Informatics and Visualization

Spring 2022

Instructor Max Gardner / magardner@berkeley.edu
Office hours: Tue 1-3pm / sign-up [here](#)

GSI Irene Farah Rivadeneyra / irenef@berkeley.edu
Office hours: Thu 2-4pm / sign-up [here](#)

Details Meeting times: Mon/Wed 9:30-11am (“Berkeley time”)
Meeting location: 102 Bauer Wurster Auditorium / [Zoom](#) (as needed)
Course website: <https://bcourses.berkeley.edu/courses/1511685>
Course GitHub repository: <https://github.com/mxndrwgrdnr/CYPLAN255>

Prerequisites: CP201A, CP204C, or equivalent experience
Grading: out of 100 pts – attendance (10%) / assignments (15%) / project (75%)

Overview

The goal of this course is to train students to analyze urban data, derive insights, and create effective visualizations using open source software tools and public data. The course will first introduce the fundamentals of programming in Python before moving on to a survey of data analysis/visualization tools and technologies. Sessions will include lectures and practice exercises. Assignments will reinforce the skills and topics being presented. A final project will provide an opportunity for students to use these skills to complete an end-to-end data analysis of their own design, the results of which will be published on GitHub and presented in class.

This is a “hands-on” course. It requires some tolerance for experimentation, self-directed trial and error, and an interest in learning to write code. If you are willing to roll up your sleeves and embrace some uncertainty, you'll learn the fundamentals of urban data analysis and visualization, and might discover an entirely new lens through which to study, plan, and design neighborhoods, cities, and regions.

Course materials and Attendance

All required readings will be provided via bCourses or hyperlinks on this electronic syllabus. Lecture slides, example code, and demos/exercises will all be made available via GitHub.

We'll write code in Jupyter notebooks using the Anaconda Python distribution plus some additional software libraries. In some cases you may want to use a Berkeley service called [DataHub](#) instead of your own computer – but in general we encourage you to get comfortable installing Python and Python tools on your own computer. You'll get far more comfortable with it that way, and know that whatever you learn, and whatever you install, you can take with you when the class is over. We will use only open source, free software in this course. You'll be surprised how far you can go with it.

You should plan to bring a laptop** to all class sessions.

**NOTE TO WINDOWS USERS: Most exercises and lectures will be OS-agnostic, but command-line tools will be demonstrated in a Unix-like terminal. Windows users can use the Windows command prompt if they choose, but instructor support will be limited. Instead, I recommend installing one of the following to gain access to a Unix-like shell: [Windows Subsystem for Linux](#), [Cygwin](#), [Git SCM](#), or [PyCharm](#).

Assignments (15 pts)

Students will develop skills gradually through exercises paced over the semester. These will typically involve writing some code and documenting it, using Jupyter Notebooks that can be shared and interactively run inside a web browser, and providing a writeup discussing the assignment and its results.

Assignments will be posted on the course GitHub repository, and students will need to pull them down from there. Assignments will generally be due one week from the day they are assigned, by 11:59pm PST. Students will submit their completed assignments by opening a pull request on the course repo.

Assignments are designed to build a degree of mastery of skills and will be used as a means of ensuring that students are keeping up with the material and not falling behind. All assignments will be marked down 10% for each day late, so please submit on time.

Readings and Other Assignments (ungraded)

This course has readings associated with nearly every class meeting. These are *suggested* readings, unless otherwise specified. You will not be quizzed on them, and they may or may

not be referenced in class. They are, however, **strongly recommended**. They have been thoughtfully compiled over the many years this course has been taught, and are designed to help you get the most of this course and make your final projects a success.

In addition to the readings, the Course Schedule (see below) specifies several other assignments which, unless explicitly stated, will not be collected. They will, however, be used for the following: 1) to facilitate discussion/break-out groups in class; 2) to inspire final project ideas; and 3) to ensure that you make steady progress on your final projects throughout the course of the semester. In some cases there will be class time designated for working on these ungraded assignments, but not always. It is in your own best interest, and that of your fellow students, that you keep up with them.

Final Projects (75 pts)

Final projects will require harnessing the skills practiced in the exercises and developing a more independent work plan to accomplish an analysis of data. More details will be provided later in the semester.

Project components and due dates:

1. Project proposal + initial analysis (10 pts)
 - Due Sunday, Mar 13
2. Final presentation (20 pts)
 - Slides, etc. presented in class on Apr 20, 25, or 27
3. github.io project page (45 pts)
 - Due Monday, May 9 (first day of Finals week)

Self-Reliance, Collaboration, and Academic Integrity

This course requires a lot of experimentation and trial-and-error. Google and [StackOverflow](#) will be your best friends! Google your questions, Google any error messages, and if you can't find an answer, talk to your classmates, and if you still can't sort it out, e-mail Max and Irene. When you e-mail us, tell us what you've searched and what you've discovered, and include screenshots, links, and error messages. 99% of the time, somebody else has encountered the exact issue you are having and has documented the solution.

That being said, you are welcome — in fact, encouraged — to work on the homework exercises and your semester project together with other students. Discussing code is a great way to understand it better, and can make tracking down bugs less frustrating. If you copy an entire substantive piece of code (i.e., several lines or more) from the internet or from another student, we ask that you indicate this in a code comment. Otherwise, we will expect everything

you submit to be your own original work. Details of the U.C. Berkeley Academic Honor Code can be found [here](#).

Campus Policies and Guidelines

<https://teaching.berkeley.edu/campus-policies>

Accommodations for Students with Disabilities

UC Berkeley is committed to creating a learning environment that meets the needs of its diverse student body. If you anticipate or experience any barriers to learning in this course, please feel welcome to discuss your concerns with me.

If you have a disability, or think you may have a disability, you can work with the Disabled Students' Program (DSP) to request an official accommodation. The Disabled Students' Program (DSP) is the campus office responsible for authorizing disability-related academic accommodations, in cooperation with the students themselves and their instructors. You can find more information about DSP, including contact information and the application process here: dsp.berkeley.edu. If you have already been approved for accommodations through DSP, please meet with me so we can develop an implementation plan together.

Students who need academic accommodations or have questions about their accommodations should contact DSP, located at 260 César Chávez Student Center. Students may call 510-642-0518 (voice), 510-642-6376 (TTY), or email dsp@berkeley.edu.

Department Climate Statement

The Department of City and Regional Planning in the College of Environmental Design is committed to an equitable and inclusive educational environment for all. As students, staff, and faculty, we strive to foster a community in which we celebrate our diversity and affirm the dignity of each person by respecting the identities, perspectives, and experiences of those with whom we work. As a member of the UC Berkeley community, the Department of City and Regional Planning is committed to a safe work environment for all.

The following campus-wide resources are available to assist with this effort:

- [Gender Equity Resource Center](#)
- [Path to Care: Sexual Violence and Sexual Harassment](#)
- [Office for the Prevention of Harassment and Discrimination \(OPHD\)](#)
- [OPHD info for students, staff, and faculty](#)
- [University Health Services: Counseling and Psychological Services](#)

- [Centers for Educational Justice and Community Engagement](#)

Additional Resources

The following books and websites may be helpful resources, and we will draw material from many of them during the semester. (All readings assigned for class will be available online or as PDFs in bCourses.) Each piece of software we'll use also has official documentation online.

Adhikari, Ani and John DeNero, Computational and Inferential Thinking, 2019

(<https://inferentialthinking.com>)

Online textbook developed for Berkeley's Foundations of Data Science class.

Downey, Allen, Think Python, 2nd Edition, O'Reilly Media, 2015

(<https://learning-python.com/about-lp5e.html>)

Introduction to programming using Python. All the material is online.

Foster, Ian, et al., Big Data and Social Science, CRC Press, 2017

A practical guide to gathering data and working with it in various ways.

***Lutz, Mark, Learning Python, 5th Edition, O'Reilly Media, 2013**

(<https://learning-python.com/about-lp5e.html>)

Much more depth than you need for this class, but a great reference.

McKinney, Wes, Python for Data Analysis, 2nd Edition, O'Reilly Media, 2017

More depth about Pandas than Python Data Science Handbook, but less readable.

Pilgrim, Mark, Dive into Python 3, Apress, 2009 (<https://diveintopython3.net>)

Nice tutorials and reference for aspects of core Python syntax and programming concepts, but missing some topics that are in Think Python. All the material is online.

***VanderPlas, Jake, Python Data Science Handbook, O'Reilly Media, 2016**

(<https://jakevdp.github.io/PythonDataScienceHandbook>)

Excellent – working with data, making graphs and charts, machine learning. All the material is online.

Real Python (<https://realpython.com>) — Great Python tutorials on numerous topics.

Rey, Sergio, et al., Geographic Data Science with Python, 2020

(<https://geographicdata.science/book/intro.html>)

Great resource for geospatial data analysis in Python from the creators of PySAL.

Software Carpentry (<https://software-carpentry.org/lessons>) — Tutorials about scientific computing.

Stack Overflow (<https://stackoverflow.com>) — Best website for user-contributed coding Q&As.

Topics + Course Schedule

The topics covered by this course are organized into the following seven (7) modules:

1. Fundamentals of Programming
2. Intro to Data Analysis in Python
3. Intro to Data Visualization
4. APIs + Open Data
5. Working with Geospatial Data
6. Visualizing Spatial Data
7. Statistical Analysis + Machine Learning

MODULE 1: FUNDAMENTALS OF PROGRAMMING

Weds, Jan 19

Course Introduction

Overview of the course, expectations, prerequisites, learning objectives, assignments and projects.

Assignments

- Install the Anaconda distribution of Python on your computer, and verify that it is working. Find your download [here](#).
- Create a personal GitHub account using your Berkeley e-mail address.
- Via bCourses, submit links to three (3) examples of interesting public/open datasets that you think you or your fellow students might be interested in exploring in the context of this class. In 2-3 sentences, describe their relevance to topics in transportation, housing, land use, urban design, etc. 2-3 sentences. Show us something we haven't seen before – the American Community Survey doesn't count!

Readings

- Why Python:
 - <http://lorenabarba.com/blog/why-i-push-for-python/>
 - <https://www.codefellows.org/blog/5-reasons-why-python-is-powerful-enough-for-google>
- Chapter 1 of [Think Python](#)
- Wilson et al. "Best practices for scientific computing," PLOS Biology, 2014. <https://doi.org/10.1371/journal.pbio.1001745>

Mon, Jan 24

Intro to the Command-line

Using a command-line interpreter; common syntax, programs, and arguments; accessing and navigating the file system; Python interpreters; conda environments; starting/stopping a Jupyter server; using Git; text editors

Assignments

- Create a CYPLAN255 GitHub repo and push your first commit.

Readings:

- Command-line guides for [Windows](#) or [Unix-like](#) (MacOS/Linux)
- <https://docs.github.com/en/get-started/using-git/about-git>

Weds, Jan 26

Git and GitHub

Principles of distributed version control; repositories; commits; branches; forks; making a GitHub pages website

Assignments

- Create your own github.io website by following [this](#) tutorial.

Readings

- <https://docs.github.com/en/get-started/using-git/about-git>

MODULE 2: INTRO TO DATA ANALYSIS IN PYTHON

Mon, Jan 31

Intro to Python with Jupyter Notebooks

Notebooks and Python kernels; variables; expressions; built-in data types and structures

Assignments

- Assignment 1 released (due Sun, Feb 6)

Readings

- Chapter 3 of [Python for Data Analysis](#)
- Chapters 2, 4 of [Dive Into Python 3](#)
- Chapters 8, 10, 11, 12 of [Think Python](#)

Wed, Feb 2

Programming Logic

Control flow in Python (conditional logic, loops, functions)

Readings

- Chapters 2, 3, 5-7 of [Think Python](#)
- Chapters 1, 7 of [Dive Into Python 3](#)

Mon, Feb 7

Data Analysis in Python

NumPy arrays and matrices; Pandas Series and DataFrames; loading, displaying and exporting data; descriptive statistics; indexing and filtering;

Assignments

- Assignment 2 released (due Sun, Feb 13)

Readings

- Chapters 4, 5, and 6 of Wes McKinney's [Python for Data Analysis](#)

Wed, Feb 9

More Pandas

Vectorized operations; merge, join, concatenate; group by and aggregations; cleaning and imputing missing data

Readings

- Chapter 7 in [Python for Data Analysis](#)

MODULE 3: INTRO TO DATA VISUALIZATION

Mon, Feb 14 🍷

Data Visualization - Pt. I

Data viz. for good and evil; use Matplotlib and Seaborn to create static images; dimensionality of data; continuous vs. categorical data; univariate distributions

Assignments

- Find three (3) examples of interesting data visualizations and describe in 2-3 sentences what makes each of them good, bad, or misleading. Be prepared to talk about them in class.

Readings

- Chapter 1 of *Envisioning Information* (Tufte, 1990)
- Chapter 9 in [Python for Data Analysis](#)
- [Seaborn](#) tutorial/documentation

Wed, Feb 16

Data Visualization - Pt. II

Interactive plots, widgets, and apps.

Assignments

- Assignment 3 released (due Tue, Feb 22)

Readings

- Interactive Bokeh [tutorials](#)
- [Plotly](#) tutorial/documentation
- Streamlit [tutorial](#)

Mon, Feb 21 *NO CLASS (President's Day)*

MODULE 4: OPEN DATA AND APIs

Wed, Feb 23 **Intro to APIs**

What's in an API; performing queries; authentication; Socrata;

Readings

- Red Hat guide to APIs: [here](#)
- Read the Socrata [getting started](#) page
- Look over some of the resources that companies and public agencies have put together for third-party software developers:
 - [BART](#)
 - [Twitter](#)
 - [Google Maps](#)
 - [Urban Institute](#)

Mon, Feb 28 **APIs and Beyond**

Geocoding; web scraping; parsing XML

Readings

- <https://guides.library.illinois.edu/Geocoding>

MODULE 5: WORKING WITH GEOSPATIAL DATA

Wed, Mar 2 **Intro to Geospatial Data Analysis**

Vector vs. raster; coordinate reference systems and projections; spatial data types and file formats; spatial indexing; common spatial transformations

Assignments

- Get familiar with OpenStreetMap by reading about it [here](#), and contribute at least one (1) change to a Humanitarian OpenStreetMap (HOTOSM) project by following the HOTOSM [quickstart guide](#).

Readings

- [mapschool.io](#)
- Part I of [Geographic Data Science with Python](#)
- https://wiki.osmfoundation.org/wiki/Main_Page
- <https://tasks.hotosm.org/learn/quickstart>

Mon, Mar 7

FOSS tools for Geospatial Data Analysis

Survey of open source tools for manipulating geospatial data from the command-line, a Python session, a browser, or your desktop.

Assignment

- Project proposal assignment released (Due Sun, Mar 13)

Readings

- [This](#) great blog post from Development Seed (you can try to follow along with the tutorial on your computer but I think some of the links they use are not longer active)
- <https://www.osgeo.org/about/what-is-open-source/>
- <https://macwright.com/2012/10/31/gis-with-python-shapely-fiona.html>
- Chapters [6](#), [8](#), and [9](#) of Geographic Data Science with Python

Wed, Mar 9

Intro to Network Analysis

Graph theory; GTFS; Python tools for working with networks

Readings

- Boeing, Geoff. "OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks." *Computers, Environment and Urban Systems* 65 (2017): 126-139.
<https://doi.org/10.1016/j.compenvurbsys.2017.05.004>
- Blanchard SD, Waddell P. Assessment of Regional Transit Accessibility in the San Francisco Bay Area of California with UrbanAccess. *Transportation Research Record*. 2017;2654(1):45-54.
<https://doi.org/10.3141%2F2654-06>

- Foti, Fletcher, Paul Waddell, and Dennis Luxen. "A generalized computational framework for accessibility: from the pedestrian to the metropolitan scale." *Proceedings of the 4th TRB Conference on Innovations in Travel Modeling*. Transportation Research Board. 2012. <http://onlinepubs.trb.org/onlinepubs/conferences/2012/4thITM/Papers-A/0117-000062.pdf>
- <https://www.mapzen.com/blog/animating-transitland/>
- Li, Yang, and Wei "David" Fan. "Modeling and evaluating public transit equity and accessibility by integrating general transit feed specification data: Case study of the City of Charlotte." *Journal of Transportation Engineering, Part A: Systems* 146.10 (2020): 04020112. Available [here](#).

MODULE 6: VISUALIZING GEOSPATIAL DATA

Mon, Mar 14

Effective Communication of Spatial Data

Types of geospatial visualizations; color theory; common pitfalls of cartographic representation

Readings

- <https://www.nytimes.com/interactive/2020/10/30/opinion/election-results-maps.html>
- <https://ai.googleblog.com/2019/08/turbo-improved-rainbow-colormap-for.html>
- Wong, David WS. "The modifiable areal unit problem (MAUP)." *WorldMinds: Geographical perspectives on 100 problems*. Springer, Dordrecht, 2004. 571-575. <https://doi.org/10.1016/B978-008044910-4.00475-2>

Wed, Mar 16

Building Static Maps in Python

Survey of Python libraries for plotting geospatial data on a map

Readings

- Norwood, Carla; Cumming, Gabriel (2012). Making Maps That Matter: Situating GIS within Community Conversations about Changing Landscapes. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 47(1), 2–17. [doi:10.3138/carto.47.1.2](https://doi.org/10.3138/carto.47.1.2)
- Chapter [5](#) of Geographic Data Science with Python

Mon, Mar 21

NO CLASS (Spring Break)

Wed, Mar 23 *NO CLASS (Spring Break)*

Mon, Mar 28 **Building Interactive Maps**

Survey of tools and technology for creating dynamic maps in Python and other open source frameworks

Readings

- <http://www.liedman.net/tiled-maps/>
- <https://www.axismaps.com/guide/what-is-a-web-map>
- <https://www.axismaps.com/guide/should-a-map-be-interactive>
- <https://mapbrief.com/2017/04/06/few-interact-with-our-interactive-maps-what-can-we-do-about-it/>

Wed, Mar 30 **Misc. Maps**

Visualizing big geo-data; dot-density maps;

Readings

- <https://blog.mapbox.com/mapping-millions-of-dots-77eead9bd663>
- http://andrewgaidus.com/Dot_Density_County_Maps/

MODULE 7: STATISTICAL ANALYSIS + MACHINE LEARNING

Mon, Apr 4 **Statistical Analysis**

Choosing the right algorithm; identification vs. prediction; sampling from univariate probability distributions; OLS multiple regression with statsmodels

Readings

- <https://pdf4pro.com/view/introduction-to-multiple-regression-biddle-23f2dd.html>
- http://onlinestatbook.com/2/regression/multiple_regression.html

Wed, Apr 6 **Advanced Stats/Machine Learning in Python**

Random forest-based regression in scikit-learn; clustering algorithms; discrete choice theory and models

Readings

- Chapters 1-3 in [Discrete Choice Models with Simulation](#)

Mon, Apr 11 Causal Inference Methods in Urban Science

Deep dive into two examples from the recent literature

Readings

- Currie, Janet, et al. *Do housing prices reflect environmental health risks? Evidence from more than 1600 toxic plant openings and closings*. No. w18700. National Bureau of Economic Research, 2013. Available [here](#).
- *TBD*

Wed, Apr 13 Special Topics I: Visualizing Mobility Data

Guest speaker: *TBA*

Mon, Apr 18 Special Topics II: Geospatial Data for Social Justice

Guest speaker: *TBA*

Wed, Apr 20 Presentations I

Mon, Apr 25 Presentations II

Wed, Apr 27 Presentations III

UC Berkeley sits on the territory of xučyun (Huichin), the ancestral and unceded land of the Chochenyo speaking Ohlone people, the successors of the sovereign Verona Band of Alameda County. This land was and continues to be of great importance to the Muwekma Ohlone Tribe and other familial descendants of the Verona Band.

We recognize that every member of the Berkeley community has, and continues to benefit from, the use and occupation of this land, since the institution's founding in 1868. Consistent with our values of community, inclusion and diversity, we have a responsibility to acknowledge and make visible the university's relationship to Native peoples.

It is vitally important that we not only recognize the history of the land on which we stand, but also, we recognize that the Muwekma Ohlone people are alive and flourishing members of the Berkeley and broader Bay Area communities today.

Read more on the Centers for Educational Justice & Community Engagement [website](#).