Introduction to Computational Tools for Social Science

This course will provide graduate students the technical skills necessary to conduct research in computational social science and digital humanities, introducing them to the basic computer literacy, programming skills, and application knowledge that students need to be successful in further methods work.

The course is currently divided into four main sections. In the first section, students learn how their computers work and communicate with other computers using git and bash. In the second, we turn our attention to the structure, analysis, and visualization of data, with an emphasis in R. In the third, students learn applications to collect new data (e.g., using APIs and webscraping). In the fourth, students learn additional means of analyzing and visualizing data, including tools like text analysis in Python and machine learning.

Objectives

- Understand basic programming terminologies, structures, and conventions
- Navigate and operate effectively in a UNIX environment
- Master basic Git and GitHub workflows
- Write, execute, and debug R code for novel data collection, cleaning, analysis, and visualization
- Write and execute basic code for text analysis in Python
- Be familiar with the concepts and tools of a variety of computational social science / digital humanities applications
- Be familiar with the basic guidelines around reproducible research, good scientific computing practices, and ethics/privacy/legal quandaries
- Learn independently and train themselves in a variety of computational applications and tasks through online documentation (we will have access to Datacamp's courses for the duration of the semester).

Logistics

Instructor

Jae Yeon Kim

jaeyeonkim@berkeley.edu

Section Assistant

Julia Christensen

jbchristensen@berkeley.edu

Time and Location

Lecture: Monday 10am - 12pm (Dwinelle 209)

Section: Wednesday 4pm - 6pm (Barrows 118)

Office Hours

By appointment with Julia Christensen (email or bCourses to set up), 715 Barrows.

bCourses

We will use bCourses for communication (announcements and questions) and turning in assignments. You should ask questions about class material and assignments through the bCourses website so that everyone can benefit from the discussion. We encourage you to respond to each other's questions as well.

GitHub

All course materials will be posted on Github at https://github.com/jaeyk/PS239T, including class notes, code demonstrations, sample data, and assignments. Students are required to use GitHub for their final projects, which will be publically available, unless they have special considerations (e.g. proprietary data).

Accessibility

This class is committed to creating an environment in which everyone can participate, regardless of background, discipline, or disability. If you have a particular concern, please come to me as soon as possible so that we can make special arrangements.

Course Requirements and Grades

This is a graded class based on the following:

- Completion of assigned homework (50%)
- Participation (25%)
- Final project (25%)

Assignments

Assignments will be assigned at the end of every session. They will be due at the start of the following class unless otherwise noted. The assignments will be frequent but each of them should be fairly short.

You are encouraged to work in groups, but the work you turn in must be your own. Group submission of homework, or turning in copies of the same code or output, is not acceptable. Remember, the only way you actually learn how to write code is to write code.

Unless otherwise specified, assignments should be turned in as **pdf documents** via the bCourses site.

Class Participation

The class participation portion of the grade can be satisfied in one or more of the following ways:

- attending the lecture and section (note that section is non-optional)
- asking and answering questions in class
- contributing to class discussion through the bCourse site, and/or
- collaborating with the campus computing community, either by attending a D-Lab or BIDS workshop, submitting a pull request to a campus github repository (including the class repository), answering a question on StackExchange, or other involvement in the social computing / digital humanities community.

Because we will be using laptops every class, the temptation to attend to other things during slow moments will be high. While you may choose to do so, I do request that you think of your laptop screen as in the public domain for the duration of classtime. Please do not load anything that will distract your classmates or is otherwise inappropriate to a classroom setting.

Final Project

The final project consists of using the tools we learned in class on your own data of interest. First- and second-year students in the political science department are encouraged to use this as an opportunity to gather data to be used for other courses or the second-year thesis. Students are required to write a short proposal by March 25 (no more than 2 paragraphs) in order to get approval and feedback from the instructors.

During sections in April we will have **lightning talk sessions** where students present their projects in a maximum 5 minute talk, with 5 minutes for class Q&A. Since there is no expectation of a formal paper, you should select a project that is completable by the end of the term. In other words, submitting a research design for your future dissertation that will use skills from the class but collects no data is not acceptable, but completing a viably small portion of a study or thesis is.

Class Activities and Materials

Lecture

Classes will follow a “workshop” style, combining lecture and lab formats. The class is interactive, with students programming every session. During the “skills” parts of the class, we will be learning how to program in Unix, HTML, and R by following course notes and tutorials. During the “applications” sections, we will follow a similar structure, with occasional guest speakers.

Section

The Wednesday "lab" section will generally be a less formal session dedicated to helping students with materials from lecture and homework. It will be mostly student led, so come with questions. If there are no questions, the lab turns into a "hackathon" where groups can work on the assignments together. Section is required unless prior permission to miss it is obtained from both the instructor and one's groupmates. Attending office hours is not a substitute for attending section.

Computer Requirements

The software needed for the course is as follows:

- Access to the UNIX command line (e.g., a Mac laptop, a Bash wrapper on Windows)
- Git
- R and RStudio (latest versions)
- Anaconda and Python 3 (latest versions)
- Pandoc and LaTeX

This requires a computer that can handle all this software. Almost any Mac will do the job. Most Windows machines are fine too if they have enough space and memory.

You must have all the software downloaded and installed PRIOR to the first day of class. If there are issues with installation on your machine, please contact the section assistant, Julia Christensen, for assistance.

See B_Install.md for more information.

Books and Other Resources

There are no official textbooks for this class. Please see the references for advanced subjects and the style guides for efficient programming and project management. Note that these references will be updated throughout the semester. For the semester, we will have access to all of Datacamp's premium course materials (many thanks to Datacamp!).

Curriculum Outline / Schedule

The schedule is subject to change based on the class's rate of progress.

- **Jan. 23:** Introduction and Setup ("Installfest")
- **Jan. 28/30:** Unix, Bash, and Git
- **Jan. 4/6:** Data Structure in R
- **Feb. 11/13:** Data Analysis in R
- **Feb. 18:** Presidents' Day
- **Feb. 20:** Data Visualization in R (group formation due)
- **Feb. 25/27:** Intro to Python
- **Mar. 4/6:** APIs (project proposal draft due, see the 2018 examples)
- **Mar. 11/13:** HTML/CSS/Javascript
- **Mar. 18/20:** Web scraping (guest lecture by Jaren Haber, Sociology & Computational Text Analysis Working Group)
- **Mar. 25/27:** SPRING BREAK [no class] (final project proposal due)
- **Apr. 1/3**: Online Sampling, Survey, and Field Experiments [Note: final presentations occur in lieu of regular section in April]
- **Apr. 8/10:** Text Analysis in R (guest lecture by Marla Stuart, Social Work & BIDS Data Science Fellow)
- **Apr. 15/17:** Unsupervised Machine Learning in R
- **Apr. 22/24:** Supervised Machine Learning in R (guest lecture by Chris Kennedy, Biostats & BIDS Data Science Fellow)
- **Apr. 29/30:** Wrap-up and Package Development in R