<div align="center">

INFO/CS 3300: Data-driven Web Applications
INFO 5100: Visual Analytics for the Web
Spring 2018
Prof. David Mimno

</div>

Location: Klarman KG70
Time: MWF 11:15--12:05
Credits: 3
Websites: http://mimno.infosci.cornell.edu/info3300/, CMS, Piazza
Prerequisite: CS 2110 (or familiarity with object-oriented programming), INFO 2300 (or experience developing interactive web pages with HTML, CSS, and Javascript)

Course Description

The web has become an outstanding environment for telling stories with data. This course will cover technologies for representing, modeling and displaying data in the context of interactive web pages. Practical skills for building web pages will be mixed with data mining algorithms and theoretical approaches to graphics. We will use the D3 Javascript library to build both static and dynamic visualizations. Students will also learn functional programming style in Javascript and scalable vector graphics (SVG) as necessary to use D3. We will introduce a number of popular data mining models and algorithms such as Naïve Bayes classifiers, k-means clustering, and network layout methods, which we will incorporate into web visualizations.

Contact Information

Email Prof. Mimno for course administrative questions. Post to Piazza about course content questions.

Professor:
David Mimno, Gates 205. 607-255-8919. Best contact method is email mimno@cornell.edu. I will reply within 24 hours.

Grad TAs:
Moontae Lee (moontae@cs.cornell.edu).
Sharifa Sultana (ss3634@cornell.edu)
Eric Porter (ecp94@cornell.edu)

TAs:

TBA

Grading

Grades will be based on attendance (10%), weekly homework (40%), two open-ended group projects (15% each), and a take-home final exam (for undergraduates in 3300) OR a third group project (for masters students in 5100) (20%). Homework will be assigned on Mondays, due at midnight the following Monday, and returned the following Wednesday. Work will be turned in through CMS. Regrade requests should go in writing to mimno@cornell.edu. The first project will be to design a static (non-interactive) web visualization. The second project will be to design an interactive visualization. Each group will be assigned to a TA. You will send weekly progress reports to your TA listing what each team member is responsible for and what has been accomplished in the past week, and flagging any problems or questions.

Absences and late/missing work

[The following rules apply under normal circumstances. If you have experienced a personal crisis or a medical condition contact Prof. Mimno as soon as possible. The earlier we hear from you, the more we can help. *It is ok to not be ok*.]

Class time will mix lecture, discussion, and hands-on programming exercises, so attendance is important. If you will be absent, there is a web form linked from the course website. Absences due to reasonable circumstances that are reported *before* class will not be penalized. You will be informed within 24 hours if we think you have missed a class.

If you do not submit work by the deadline, or if you submit a file that does not contain the correct work, we will record a zero, with no exceptions. This policy is harsh but necessary for a class of this size. In order to protect you your two lowest homework grades will be discarded. Students sometimes use these "homework passes" during unusually busy weeks (often due to interview or event travel), but you are always encouraged to submit as much as you can.

Check carefully that you have submitted the correct file in the correct format. We will make an effort to open any file you submit. In rare circumstances we are able to accept work that was not submitted correctly if a you can provide a version of the file that is timestamped in a way that you do not control, such as a Github checkin or an email attachment. Your laptop's "last modified" date is *not* acceptable. In the unlikely event that you are having difficulty with CMS, you may submit work by email before the deadline without penalty.

In-class work

Web programming is a complicated skill, and there's no way to learn without practice. Each class will involve a daily programming problem that we will work on together. Template HTML files will be made available before class.

Laptops

In order to facilitate interactive in-class work, you are allowed to bring a laptop. If you don't have one or don't choose to bring one, work with someone sitting near you. If you have a laptop, you will be expected to use it for relevant work. "Multitasking" is a myth. Distractions limit your ability to learn, *and the ability of those near you*. If your laptop is open, expect to show the results of your in-class work, or to have thoughtful questions.

Academic Integrity

We will follow university policies as outlined in the Academic Integrity Handbook. You are encouraged to discuss homework, but each student will complete assignments alone.

Using other people's code is an important part of programming, but for group projects the code should be substantially the work of the group members except for standard libraries such as D3, lodash, and jQuery. Any code used in projects that was not written by the group members should be placed in separate files and clearly labeled with their source URLs. If you have benefitted from online resources such as examples or StackOverflow answers, list the URLs in comments in your own code, even if you did not directly copy anything.

Project work that relates to your other classes or research is encouraged, but you may not recycle assignments. There must be no doubt that the work you turn in for this class was done for this class.

Students with Disabilities

We will make every effort possible to ensure that the class works for all students. Contact Prof. Mimno if there is anything we should know about. If there is a specific event such as an exam that you are concerned about, please inform us at least two weeks in advance so that we have time to make arrangements.

SONA Credits

Many researchers on campus need participants for user studies and other types of experiments. The SONA system allows you to register for such studies. You will get 0.5 percent extra for this course, up to a maximum of 2.0 percent, for each 30 minute study (or equivalent). Participating in studies is a great way to find out what real research looks like. To register, go to this URL:

  https://cornell-comm.sona-systems.com

Course Outline

The following is a tentative course outline, subject to change.

Week 1, Jan 24 (WF)   Course objectives. Web environments. Javascript, JSON

Week 2, Jan 29   SVG and D3, Grammar of Graphics
Week 3, Feb 5   D3: circles, text, scales; naive Bayes classifiers
Week 4, Feb 12   D3: lines; Linear regression; Color theory
  [Winter break]
Week 5, Feb 21   D3: transitions; k-means
Week 6, Feb 26   [Project 1 due Thursday 3/1] Self-positioning layouts and networks
Week 7, Mar 5   Network sampling
Week 8, Mar 12   D3: paths, mouseovers; maps and geographic data
Week 9, Mar 19   More paths: Time series smoothing
Week 10, Mar 26   Perceptrons, SVMs
  [Spring break]
Week 11, Apr 9   Locality-sensitive Hashing
Week 12, Apr 16   [Project 2 due Thursday 4/12]
Week 13, Apr 23
Week 14, Apr 30   Randomization, permutation testing, bootstrapping
Week 15, May 7 (MW)   Summary and review

Masters (5100) have project 3, Undergrads (3300) have take home final. Due dates for projects and finals are specified by the registrar.