Modern Data Structures  
(G5072)  

Time TBD  
Location TBD  
Instructor: Gregory M. Eirich  

Course Goals  
This course is intended to provide a detailed tour on how to access, clean, “munge” and organize data, both big and small. (It should also give students a flavor of what would be expected of them in a typical data science interview.) Each week will have simple, moderate and complex examples in class, with code to follow. Students will then practice additional exercises at home. The end point of each project would be to get the data organized and cleaned enough so that it is in a data-frame, ready for subsequent analysis and graphing. Therefore, no analysis or visualization (beyond just basic tables and plots to make sure everything was correctly organized) will be taught; and this will free up substantial time for the “nitty-gritty” of all of this data wrangling.  

Course Expectations  

Attendance and Class Participation. Your attendance and participation are necessary at every meeting. This class will work best when students ask a lot of questions.  

Exams. We will have an in-class final, which will require the students to generate code during class to perform common operations, just as they would find in a data science interview.  

Homework. Homework problems will be assigned on a weekly basis, and students are expected to work on them alone.  

Plagiarism and Academic Dishonesty: Students must do all their work within the boundaries of acceptable academic norms. See the Academic Honesty page of the CU website regarding college policy on plagiarism and other forms of academic dishonesty - http://www.columbia.edu/cu/history/ugrad/main/handbook/academic_honesty.html. Students found guilty of plagiarism or academic dishonesty will be subject to appropriate disciplinary action, which may include reduction of grade, a failure in the course, suspension or expulsion. This includes lab reports – if they are copied from another student, severe penalties may be applied. ** Note that plagiarism is also possible when writing code, so be careful to write your own code.  

Late Assignments. Students will lose points for handing in late assignments, at the discretion of the instructor and teaching assistant.  

Textbook. There is no textbook for this kind of course, but you will have weekly readings to refer to.  

Grade Distribution. The distribution of the parts for your grade is as follows:  

   Final Exam = 40%  
   Homework Assignments = 50%
Attendance and Participation = 10%

Changes: There may be adjustments in the scheduling of assignments, exams, and classrooms. Changes will be posted on Courseworks along with other announcements.

GSAS Disabilities Accommodations: If you have been certified by Disability Services (DS) to receive accommodations, please either bring your accommodation letter from DS to your professor’s office hours to confirm your accommodation needs, or ask your liaison in GSAS to consult with your professor. If you believe that you may have a disability that requires accommodation, please contact Disability Services at 212-854-2388 or disability@columbia.edu. Important: To request and receive an accommodation you must be certified by DS.

GSAS Academic Integrity Policy: Columbia's intellectual community relies on academic integrity and responsibility as the cornerstone of its work. Graduate students are expected to exhibit the highest level of personal and academic honesty as they engage in scholarly discourse and research. In practical terms, you must be responsible for the full and accurate attribution of the ideas of others in all of your research papers and projects; you must be honest when taking your examinations; you must always submit your own work and not that of another student, scholar, or internet source. Graduate students are responsible for knowing and correctly utilizing referencing and bibliographical guidelines. When in doubt, consult your professor. Citation and plagiarism-prevention resources can be found at the GSAS page on Academic Integrity and Responsible Conduct of Research. Failure to observe these rules of conduct will have serious academic consequences, up to and including dismissal from the university. If a faculty member suspects a breach of academic honesty, appropriate investigative and disciplinary action will be taken following Dean's Discipline procedures.
Proposed Schedule of Classes, for Tuesday’s Section

Week 1 - Introduction to the Class

Part I: Data Manipulation

Week 2 - Github; “Hello World,” by Github Guides; “An Intro to Git and GitHub for Beginners (Tutorial),” by Meghan Nelson

Week 3 - Basics of the tidyverse, especially dplyr and magrittr; “Why R is Hard to Learn,” by Robert A. Muenchen

Week 4 - Functions I: the basic logic and simple steps; “Functions” in “Advanced R” by Hadley Wickham

Week 5 - Functions II: nested operations and complex sets of commands; “Swirl – R Programming – Lesson 9 – Functions,” by Johnny Chan; “Writing An R Package From Scratch,” by Hilary Parker; “Instructions for Creating Your Own R Package” by Song Kim, Phil Martin and Nina McMurry

Week 6 - Functions III: free expression and strings; “Handling and Processing Strings” in R by Gaston Sanchez

Part II: Getting Data In

Week 7 - APIs; “Using Data.gov APIs in R,” University of Virginia Library; “Accessing APIs from R (and a little R programming),” by Christoph Waldhauser

Week 8 - json; “Using R to download and parse JSON: an example using data from an open data portal,” by ZevRoss; “Better handling of JSON data in R?,” by Rolf Fredheim

Week 9 - HTML; “Using rvest to Scrape an HTML Table,” by Cory Nissen, “How To Screen-scrape,” by Chris Bail

Week 10 - SQL; practicing using SQLZOO

Part III: Other “Big Data” Consideration

Week 11 - Big data considerations vis-a-vis algorithms; “Basic Introduction into Algorithms and Data Structures,” by Frauke Liers; “Introduction to Pseudocode,” by Carnegie Mellon’s Robotics Academy

Week 12 - Amazon Web Services and parallelization; “A comprehensive beginner’s guide to start ML with Amazon Web Services (AWS),” by Aarshay Jain; “Analyzing Your Data on the AWS Cloud (with R),” by Tal Galili; “Five ways to handle Big Data in R,” by Oliver Bracht

Week 13 - Wrapping everything up