Syllabus - CSE 6040x: Intro to Computing for Data Analysis

#teaching/cse6040 #teaching/omsa

Instructor: Professor Richard (Rich) Vuduc
Co-creators: Vaishnavi Eleti and Rachel Wiseley

Course description. This course is your hands-on introduction to programming techniques relevant to data analysis and machine learning. Most of the programming exercises will be based on Python and SQL.

What will you learn? You will build, "from scratch," the basic components of a data analysis pipeline: collection, preprocessing, storage, analysis, and visualization. You will see many examples of high-level data analysis questions, concepts and techniques for formalizing those questions into mathematical or computational tasks, and methods for translating those tasks into code. Beyond programming languages and best practices, you'll learn elementary data processing algorithms, notions of program correctness and efficiency, and numerical methods for linear algebra and mathematical optimization.

Philosophy and approach. The basic philosophy of this course is that you'll learn the material best by a combination of reading, thinking, and most importantly, actively doing. Therefore, you should make an effort to complete all assignments, including any "optional" parts.

Honor code. All course participants—you, the teaching assistants, and me—are expected and required to abide by the letter and the spirit of the edX Honor Code and the Georgia Tech Honor Code. In particular, always keep the following in mind:

- Ethical behavior is extremely important in all facets of life. Honest and ethical behavior is expected at all times.
- You are responsible for completing your own work.
- Any learner found in violation of the edX Honor Code will be subject to any or all of the actions listed in the edX Honor Code.
- For Georgia Tech students, all incidents of suspected dishonesty or violations of the Georgia Tech Honor Code will be reported to and handled by the Dean of Students office. Penalties for violating the collaboration policy can be severe; alleged violations are adjudicated by the Dean of Students office and not by the instructor.

Prerequisites. You should have at least an undergraduate-level understanding in the following topics:

- Programming proficiency in Python or similar language
- Basic calculus
- Probability and statistics
- Linear algebra

What does "programming proficiency" mean? For context, this course aims to fill in gaps in your programming background, in preparation for other programming-intensive courses in the MS Analytics program. If you have a significant programming background already, you will probably get less out of this course than you might have otherwise.

The formal prerequisite for this course, as it appears in OSCAR, is: Undergraduate Semester level CS 1371 Minimum Grade of D.

In more human terms, you should be familiar with basic programming ideas at the level of the Python Bootcamp, which most on-campus MS Analytics students would have taken.

Assignments and grading. Your grade will be based on a combination of "lab notebooks" (programming homework) and three exams. Passing grade is 60%.

- Notebooks: 50% (each notebook is weighted equally)
- Midterm 1: 10%
- Midterm 2: 15%
- Final exam: 25%

Due date convention. For all assignments with due dates are due at 11:59 UTC. Here is a handy online tool, the Time Zone Converter.

Late policy. For your lab notebooks, you get three “late passes.” That is, you may submit your lab notebook up to 48 hours after the official due date **without penalty.** Any assignment submitted after you run out of passes or after 48 hours (with or without a pass) will get zero credit.

You do **not** need to ask before "using" a late pass; just turn in any three assignments in the 48-hour late pass window. On our end, we will simply apply the late passes at the end of the semester when we compute your final grade. (And if more than three assignments are late, you'll get zeroes on the all but the first three late assignments.)

Late passes do **not** apply to exams. They are due on the posted date, following the same due date convention explained above.

Collaboration policy. You may collaborate on the lab notebooks at the “whiteboard” level. That is, you can discuss ideas and have technical conversations with other students in the class, which we especially encourage on the online forums. However, each student must write-up and submit his or her own notebooks.

You must do all exams completely on your own, without any assistance from others.

School supplies. The main pieces of equipment you will need are a pen or pencil, paper, an internet-enabled device, and your brain!

There is **no** required textbook; however, the following may be a handy resource.

William McKinney. Python for Data Analysis: Data wrangling with Pandas, NumPy, and IPython. O'Reilly Media, October 2012. ISBN-13: 978-1449319793. Buy on Amazon

Note: The edition above is the first, which is pretty old at this point (2012). A new edition is scheduled to come out in early Fall 2017, which is why we did not require it for the course.

Accommodations for individuals with disabilities. If you are a student with learning needs that require special accommodation, please contact the Office of Disability Services at (404) 894-2563 or http://disabilityservices.gatech.edu/, as soon as possible, to make an appointment to discuss your special needs and to obtain an accommodations letter. Please also e-mail me as soon as possible in order to set up a time to discuss your learning needs.

Course Topics and Schedule for Fall 2017

The topics are divided into roughly three units, as outlined below. A more detailed schedule will be posted when the class begins, but the typical pace is 1 or 2 topics per week and 1 notebook (homework assignment or exam) per week.

Module 0: Fundamentals.

- Topic 0: Course and co-developer intros
- Topic 1: Python bootcamp review + intro to Jupyter
- Topic 2: Pairwise association mining
    - Default dictionaries, asymptotic running time
- Topic 3: Mathematical preliminaries
    - probability, calculus, linear algebra
- Topic 4: Representing numbers
    - floating-point arithmetic, numerical analysis

Module 1: Representing, transforming, and visualizing data.

- Topic 5: Preprocessing unstructured data
    - Strings and regular expressions
- Topic 6: Mining the web
    - (Notebook only) HTML processing, web APIs
- Topic 7: Tidying data
    - Pandas, merge/join, tibbles and bits, melting and casting
- Topic 8: Visualizing data and results
    - Seaborn, Bokeh
- Topic 9: Relational data (SQL)

Module 2: The analysis of data.

- Topic 10: Intro to numerical computing
  - NumPy / SciPy
- Topic 11: Ranking relational objects
  - Graphs as (sparse) matrices, PageRank
- Topic 12: Linear regression
  - Direct (e.g., QR) and online (e.g., LMS) methods
- Topic 13: Classification
  - Logistic regression, numerical optimization
- Topic 14: Clustering
  - The k-means algorithm
- Topic 15: Compression
  - Principal components analysis (PCA), singular value decomposition (SVD)
- Topic 16: Putting it all together
  - (Notebook only) Eigenfaces