Assignments Overview

Overview & Goals

This is the hands-on section of the course. The *practice* of data science largely boils down to writing code. So although this is not a programming class *per se*, the majority of the sections and assignments will center around using Python to *do* data science.

Not everything you are expected to do will be explicitly mapped out, step by step. Not only would that level of direction be prohibitively long to prepare, it would ultimately be dishonest and unhelpful; data science is often about figuring things out. If clear, step-by-step instructions could be written down that always worked, data science would be automated and this class would be moot. The human level of figuring things out when the answers are not entirely clear and there is uncertainty in how to proceed is the actual “practice” part of “data science in practice.” You are going to have to be okay with trying things out that don't work, with getting stuck, with not quite knowing what is going on all the time. That's the job.

What we don't want you to do is get (or even feel!) totally lost. If you do find yourself wandering through a problem that you really have no idea about, and you’re 17 layers deep into stack overflow questions, and Jupyter is running at a sluggish pace due to an uncountable number of browser tabs... pause, and ask someone for help. Most of the time a quick intervention from someone who can identify the problem, give you an overview as to what is happening, and who can suggest where to look can save you *tons* of time (and frustration).

As instructors, we are here to help you, but keep in mind that the # students >>> # instructors. Thus, we encourage you to help one another. If you are unsure about something, ask your fellow students. Everyone has different levels of experience in different domains. This is how {data science, programming, science, research, industry} works: teams of people with different backgrounds and expertise share knowledge and ideas and work toward a common goal. The best way to learn something, or to check if you really know it, is to teach it. So if you do think you know something, offer help! Sections and office hours are meant to be collaborative experiences.

Also, *please* ask questions on piazza and come to office hours. Be aware that we may not know the answer to every question, at least not right away. Data science and programming are broad and dynamic, and no single person can be on top of everything. If nothing else, we will try and give you pointers on what might be happening, and whether it is likely to be tractable, or potentially very messy and worth circumventing. Knowing which problems you can solve, and which you can't (at least in the context of a given project) is an important skill.

The goal of this class is not to teach you (all of) data science; we can’t possibly do that in 10 weeks. The goal is to give you a meaningful introduction and hands-on experience of what data science is, and to provide you with a basis from which you can continue to learn data science. There are an amazing amount of resources out there for this topic. The difficulty, as a newcomer, is to figure out what it is you’re looking for, and how to find it; that is, figuring out what you don’t know, what you need to learn, and where you can find those answers. Technical experts don’t know all the answers, they just know where to find the answers and how to implement them. A major goal of these sections and assignments is to show you what is available so that you know where to look if you want to keep going in data science.

Section Topics & Hands-On Materials

Section topics will follow the topics outlined for the lectures. For each lecture topic outlined in the syllabus, there will be some materials in the Tutorials folder covering that topic. We may not cover all the tutorial materials explicitly in section, but we will prioritize covering material that is needed for the assignments. That is, sections will serve as hands-on sessions targeted at fulfilling the assignments.

The tutorials are not intended, or written, as in-depth tutorials covering everything about the topics. Instead they are more like an index, or a map. For each topic, we aim to give a cursory overview and simple demonstration of what the topic under investigation is, and guide you to bigger and better resources to really dive into it. You are not expected to, and will not be able to, follow every link for every topic; pick the ones you are most interested in and/or are the most helpful ones to get you unstuck for the assignments and/or project.

Section Attendance & Switching

Section attendance is not strictly mandatory but is recommended. Sections will be used for hands-on tutorials and overviews of the assignments. Sections are the best way to get detailed instructions on how to do the assignments. You should make sure you are enrolled in a section that you can attend, and plan to attend that section each week. If something comes up, and you have a conflict, you may instead attend another section. Each section, in a given week, will cover the same material.

Supported Tools

Officially, we will be using, assignments will require, and we will support the use of:

- python3 with the anaconda platform
- Jupyter Notebooks
- git & Github (optionally using the SourceTree GUI)

The assignments must be completed using these tools. You are welcome to explore other tools as you explore these topics, and to use different tools for the project. Note however, that we offer no guarantees that we can help with other languages / modules / tools, etc.

Assignments

Assignments will be done in Jupyter Notebooks. They will be released on Github (https://github.com/COGS108) and submitted to TritonED.

Assignment Schedule

The (tentative) schedule for assignments is as follows:

Name	**Due Date**
A1 - Set Up & Github (12%)	11:59 pm, Sunday, January 21st
A2 - Data Exploration (12%)	11:59 pm, Sunday, February 4th
A3 - Data Privacy (12%)	11:59 pm, Sunday, February 11th
A4 - Data Analysis (12%)	11:59 pm, Sunday, February 25th
A5 - Name TBD (12%)	11:59 pm, Sunday, March 4th

Naming Conventions

We will be tracking assignments and files using a unique identifier. **You must follow the naming specifications we outline in the assignments.** This is necessary due to the size of the class - even your full names may not be unique. All files that you submit should include a unique ID of the form ‘‘$####’, which is comprised of the first letter of your last name, followed by the last 4 numbers of your student ID. For example, hypothetical student Michael Jordan, student number A012345678 would use the code ‘J5678’ for all his assignments.

Assignments are auto-graded and these scripts may fail with improperly named files. Triple check your submitted files meet the specified naming conventions. If you fail to do so, it is likely your submission won’t make it through the grading procedure and you will not receive a grade.

Using Jupyter Notebooks for Class Assignments

We will be using a system that allows for automatically grading notebook submissions. Notebooks will be released with step-by-step instructions on what code to enter. Follow these instructions for working with these notebooks:

- Whenever you see ‘# YOUR CODE HERE’, replace it with code to answer the question.
    - Also, remove the ‘raise’ line, or the notebook will raise an error
- Do not edit or delete and cell that has ‘assert’ lines in it.
    - These lines are used to check your code. Editing them will be flagged as attempted cheating, and they will be reset to the original versions before grading.
- You can add new cells, and write extra code, as long as you follow what is written above.
- Your grade is partly graded on the public tests (the ‘asserts’) that are released with the assignment, and partly on a hidden set of tests.

Assignment Questions & Using Piazza

Please use Piazza for all general questions. For example, if you find a question unclear, or are totally lost and need some direction, ask on Piazza! When answering questions on Piazza, you may answer with suggestions on what topics / ideas / lectures to look into, and/or vague pseudocode but do **not** provide code that answers assignment questions. For questions that are tangential or unrelated to answering assignments, you may post minimal code segments. If you have specific questions about your assignment (for example, you are missing a grade), email the class email (COGS108 {at} gmail.com)

Grades

Grades will be released on TritonED a week after the submission date. We will try to send out automated alerts if we do not receive a submission and/or if it fails to be processed, but ultimately it is your responsibility to check your grades and get in touch if any are missing or you think there is a problem.

Assignment Regrades & Solutions

After grading is complete, we will release the assignment solutions at the same time that we release the grades (1 week after the submission deadline). These solutions will be our solutions and the full test-suite that was used for grading. Note that there may be multiple possible solutions, and variations may receive full credit, provided they pass the test criteria that are specified in the question set up. If you think there is a mistake or ambiguity (for example, your different solution meets the question specifications, but fails on an unexpected test) email the course email (COGS108 {at} gmail.com), and we will look into it.

Late Submissions

You may submit late assignments, up until assignment solutions are posted, but for 75% credit.