CMSC140 Intro to Programming Blaheta

Data Project

Fall 2017

During the second half of the course, you'll work on a data processing project to answer a quantitative question of interest. The purpose of the project is to help you connect this course content directly with a professional or personal interest of your own, and to perform a complete start-to-finish piece of data analysis using the tools and techniques we've been developing (and will continue to develop).

The question

So, what makes a good question? For our purposes, a good question is one that you

- are interested in,
- have data about (preferably in CSV format), and
- can answer by applying a straightforward algorithm to that data.

What makes an algorithm straightforward is, to a first approximation, appearing in Chapter 4 or 6 in the book. We'll talk more about that in the next week or two.

What's likely to be the more important issue is finding data. Two excellent sources of relatively clean data are CORGIS[1] and data.gov[2]. You might also ask some of your professors in your major if they have any interesting data sets to work with—I'm happy to try to help convert them into a format that you'll be able to process. Browse through these data sources and find something that appeals to you. It's great if it connects to your major, but it could also connect to a hobby or just plain sound interesting to you. Look at the columns of the data and think about what you might ask about.

[1] `https://think.cs.vt.edu/corgis/csv`

[2] `https://catalog.data.gov` , look for CSV data sets

Tell us about it

After you've got an idea of the question (or perhaps a few related questions) that you want to ask about your data, you need to write it up. Your initial proposal will include a link to the data set, explain how the data represents information of interest, and formulate your proposed question(s) in terms of the algorithm you intend to use to solve it. The document should be about a page long (12pt, double spaced, 1" margins). There should be enough background that a reader can understand why the question is interesting, and at least a little detail on how the data will be processed to answer it.

I'll respond to the proposal and (if necessary) help you refine it into something workable.

Then, at the end of the month, you'll give a very brief "elevator speech" to the class telling us what your question is and why it's interesting (to you, and maybe to us as well).

Answer it

The core of the assignment will be writing the actual program that implements your chosen algorithm to perform the data analysis you need. Your program will read in your data file and produce a value or values that can be used to answer the question(s). (The actual *answer* to the question(s) may require further interpretation.) The program should be written such that if the data set were updated or traded for a related data set in the same format, the program would still run (and produce an updated answer).

Tell us about it

Once your program is running reliably and you're sure it's running correctly, you'll need to wrap up the project by using the program's output to actually answer the question; an important step here is to understand enough about the problem to know whether your answer is plausible (i.e. not obviously the result of a bug!), and to situate the answer in the context of the original question.

In the lead-up to handing in the final writeup, you'll give two presentations to the class. The first is a "lightning talk" that explains to us in technical terms what your algorithm does and how it works to answer your question.

The second is another elevator speech, wherein you actually answer the original question.

The final writeup should include the content from the first writeup plus the content from both talks, lightly edited for flow, plus instructions on how to run your program (so I can duplicate your results). The format is, again, 12pt, double-spaced, with 1" margins. If you make statements that require citation, you should cite them (obviously!), but you can use APA, MLA, or whatever format you're most familiar with in your own discipline.

Your submission will be, via canvas, a .zip file containing

- The .py file you're running
- The data file(s) that you can run it on
- The writeup (probably a .doc or .docx file)

Elevator speech / Lightning talk

In the context of our course, "elevator speech" and "lightning talk" are near-synonyms. Both are very short talks, of 90 seconds *maximum*, that you'll be expected to give in a well-practiced way (no time for stalling—we'll do the whole class in one class period!). I will record them so that I can grade them and give feedback.

The difference between the two is in focus and tone. The idea of an elevator speech is that you serendipitously encounter someone of status in an elevator, and have the duration of the ride to pique their curiosity about your project. It's not technical so much as persuasive: you're trying to convince whoever's listening that your idea is interesting. In our case, your two elevator speeches should give background on your question, persuade us that it's interesting, and (in the last one) resolve the cliffhanger and answer the question.

The lightning talk, on the other hand, originally evolved at academic conferences as a way to convey material of technical interest in a very brief period. Here it is presumed that the audience is already basically interested, and wants to hear how your system works, but doesn't have a lot of time to devote to it. In our case, you'll give the briefest mention of your question and spend your time focusing on the data itself, how it's processed, and what your algorithm will do with it.

Scoring summary

Proposal writeup	12
Question speech	8
Program	40
Lightning talk	16
Answer speech	16
Answer writeup	8
Total	100

Deadline calendar

Note that the first two deadlines are a change from the original syllabus.

October 20 (Friday)	Proposal writeup
November 1 (Wednesday)	Question speech
November 27 (Monday)	Lightning talk
December 1 (Friday)	Answer speech, writeup, program