Scripting for Data Analysis
IST 652

Prerequisites:

For graduate students, no specific course is required, but some programming knowledge will be assumed. This may be acquired through courses or online resources.

Course Catalog Description:

Scripting for the data science pipeline. Acquiring, accessing, and transforming data in the forms of structured, semistructured, and unstructured data.

Additional Course Description:

The goal of this class is to teach students the tools and skills of scripting needed to solve problems of accessing and preparing data in a variety of formats and situations, sometimes known as *data wrangling*. The scripting will provide the skills needed to form data science pipelines, from acquiring and cleaning data to accessing data and transforming data for analysis or visualization.

The main content focus is on information access and processing tasks on the types of structured, semistructured, and unstructured data in current use in information applications. For these three types of data, the course includes the use of structured numeric and text data such as that from a spreadsheet or database, the use of data obtained through standard data exchange formats such as HTML or XML from web pages or JSON from web-based APIs, and the use of data obtained by pattern matching from text or log files. The scripting language Python was chosen because of its ease of use and available packages to work with data in many information applications. The skills learned in this class are intended to complement the analytical and visualization skills learned in other data science courses. The scripting language Python will be taught, but it will be assumed that students already have a programming background, either through course work or through online study.

Course Objectives:

Upon successful completion of this course, the student will be able to

- write scripts to access and amass data from fields in structured data, access fields in semistructured data, and define and find patterns of data in unstructured data;
- prepare and transform data to produce data summaries, lists, and networks;
- analyze and solve data access problems for the three types of data and to find and deploy appropriate software packages that can be integrated into the problem solution; and
- frame real-world data questions and show how they can be answered from data.

Textbook and Course Materials:

Python for Everybody: https://www.py4e.com/book (Python version)

Available free online or for purchase as a (paperback) book. Other readings will be assigned from online resources.

Course Organization:

Class sessions are designed around developing solutions for the main focus application tasks. Each section of the course contains a smaller number of daily problems that lead up to focus on one overall application design challenge in each of the three types of data. The instructor will then demonstrate how to solve an example of the design challenge, and then students will be asked to solve a similar problem. The design challenges will focus on the use of structured numeric and text data such as that from a spreadsheet or database, the use of data obtained through standard data exchange formats such as HTML or XML from web pages or JSON from web-based APIs, and the use of data obtained by pattern matching from text or log files.

Coursework and Assessment:

The coursework will consist mainly of small weekly lab programming and analysis problems, three larger design and programming problems for homework, and the final project. There will also be two quizzes, where half the quiz will be multiple choice questions and the other half will be programming questions. The class attendance and participation grade includes both asynchronous (completing asynchronous exercises) AND live session attendance and participation. The final project will require an additional report on a data science analysis question appropriate for the problem data.

3 lab problems *(grading scale is 100 points)*:	20%
2 quizzes *(grading scale is 100 points)*:	20%
Participation:	10%
2 homework assignments *(grading scale is 100 points):*	30%
Final Project:	20%
(grading breakdown: Proposal 5%, Presentation 7%, Report 8%)	

Grading Scale:

Grades*	Grade Points/ Credit*	Percentage Range	Total Points
A	4.0	90–100	
A-	3.66	85–89.9	
B+	3.33	80–84.9	
B	3.0	75–79.9	
B-	2.66	70–74.9	
C+	2.33	65–69.9	
C	2.0	60–64.9	
C-	1.66	55–59.9	
F	0	below 45	

Lab Problems:

ONE:

For the NBAfile.py program, for each line, create a string using string formatting that puts the team, attendance, and ticket prices into a formatted string. Each line should look something like:

'The attendance at Atlanta was 13993 and the ticket price was $20.06'

Your program should then print these strings instead of the lines. Submit your code and the output of your program.

DUE 24 hours before the live session in Week 3.

TWO:

Dictionaries: You may wish to write the code for parts a–d in one Python file.

Consider the following two dictionaries:

stock = {"banana": 6, "apple": 0, "orange": 32, "pear": 15}

prices = {"banana": 4, "apple": 2, "orange": 1.5, "pear": 3}

a. Show the expression that gets the value of the stock dictionary at the key 'orange'. Show a statement that adds an item to the stock dictionary called 'cherry' with some integer value and that adds 'cherry' to the prices dictionary with a numeric value. (Or pick your own fruit name.)

b. Write the code for a loop that iterates over the stock dictionary and prints each key and value.

c. Suppose that we have a list:
groceries = ['apple', 'banana', 'pear']
Write the code that will sum the total number in stock of the items in the groceries list.

d. Write the code that can print out the total value in stock of all the items. This program can iterate over the stock dictionary and for each item multiply the number in stock times the price of that item in the prices dictionary. (This can include the items for 'cherry' or not, as you choose.)

Due 24 hours before the live session in Week 4.

THREE:

Problem 1:

What will the following Python program print out?

```
def fred():
  print "Zap"

def jane():
  print "ABC"

jane()
fred()
jane()
```

a) Zap ABC jane fred jane
b) Zap ABC Zap
c) ABC Zap jane
d) ABC Zap ABC
e) Zap Zap Zap

Problem 2:

Rewrite your pay computation with time-and-a-half for overtime and create a function called computepay that takes two parameters (hours and rate).

```
Enter Hours: 45
Enter Rate: 10
Pay: 475.0
```

Submit your answer to Problem 1, along with your code and output from the running of your code in Problem 2, in a Word document.

Due 24 hours before the live session in Week 7.

Homework Assignments:

Homework 1: Structured Data
Due: 24 hours before the live session in Week 5

You can choose to complete this assignment by yourself or with a group of at most two total participants. Each person must turn the assignment in for grading, and each person must contribute to the development of the program. Use the file Donors_Data.csv.

Structured Data Processing

For purposes of this writeup, we will use examples from the Donors data file.

The main outline of your assignment is to write a program that will read in the data from a file, such as the .csv file saved from Excel. This will be in a format that is structured with lines of data representing one type of unit (i.e., one donor in the donors file). Your program will represent the data as Python data structures. You may choose for the overall structure to be one or both of the following:

- A list of dictionaries, or some combination of lists, dictionaries, and NumPy arrays
- A pandas dataframe

You will do data exploration and cleaning on this data.

The program will do some processing to convert the data to a form that will answer at least two questions as described below, and write files with the data suitable for answering each question. Graphing is optional.

Data:

You may choose a data set to work with. As a guideline, data sets should have somewhere between 500 and 4,000 lines of data with some number of columns between 4 and 50. These guidelines are not exact limits, just guidance for selecting data.

If the data comes in an Excel spreadsheet with a lot of columns, it is okay to first edit the file to remove columns that you do not need for your processing. For example, in the Donors data, you might wish to create a separate spreadsheet with only a few columns of data.

Questions:

For this assignment, at least one question that you choose to answer should look at the data in a different unit of analysis than is present in the data file. For example, instead of looking at individual donors, you could look at the donors of each of the nine income or wealth types.

Simplest example question (you should do one more complex than this):
For each wealth type, what is the average home value of all the donors of that type?

- Unit of analysis: wealth types
- Comparison: for each wealth type, compute the average home values of the neighborhoods of all the donors of that type
- Output: should be in a file with nine rows of data (you may also produce header and label rows), where each row has an income type (1–9) and the average home values

One way to increase the complexity of this particular question would be to add more items to be compared to the income types, e.g., add columns to the output with average total gifts or values of the last gifts. Another way is to introduce a more detailed unit of analysis; for example, suppose that for each income level you reported by gender, giving the average home value for both men and women in each category.

Other ideas:
Compare donors in the various zip codes with various types or amounts of giving.
Compare donors by the number of promotions with the total amount of donations and the frequency of donations.
Compare the number of months since the last donation to the donation amounts.

What to Submit:

In addition to the program that you write, you should write a small report. In it you should provide:

- Data and its source
- Description of your data exploration and data cleaning steps
- At least two clearly stated comparison questions with the unit of analysis, the comparison values, and how they are computed
- Brief description of the program
- Description of the output files

For your program, you may use any of the code developed in class as a template, but it is absolutely essential that you use appropriate variable names and that you write original comments for what your program does. Recall that good comments demonstrate your understanding of the code that you write and the problem that you are trying to solve.

Group Work:

If you choose to work in a group (of two), you may write and submit a single program, but you must process the data for two additional comparison questions. Each member of your group should write some part of the program, even if edited together later. Your report should describe the roles of the group members and who did what parts of the assignment, possibly including the data exploration, data cleaning, formulating questions to answer, and debugging.

Submit the following in a single submission (for each person):

- Report
    - I must be able to find the data that you utilized (link to the website, etc.)
- Program(s)
    - Must be submitted as stand-alone Python files that I can execute on my own machine
- Output files

Homework 2: Semistructured Data

Due: 24 hours before the live session in Week 8

For this homework assignment, you may work on your own or you may work in a group of two people.

Semistructured Data Processing

The main outline of your assignment is to write a program that will read in JSON formatted data from a Mongo DB collection or from a file. This will be in a format that is structured with lines of data representing one type of unit, for example, one tweet for Twitter or one post from Facebook. Your program will contain the data as lists of JSON structures, which are just Python dictionaries and lists. Your program may also contain pandas dataframes for processed data.

The program will do some processing to collect data from some of the fields that will answer one or more questions as described below, and write a file with the data suitable for answering each question. Remember that some fields may be optional or have null values, so you may need to test for those conditions. Graphing is definitely optional.

Questions:

Types of questions:

- Process one collection of data and summarize information from a number of fields. This is similar to the example programs for Twitter hashtags or Facebook counts but must access different and more fields than in those examples.
- Process one collection of data and separate it into different categories and give some summary statistics on those categories. For example, bin the Tweets by day or by hour and report on the number of tweets per day or hour.

- Process two or more collections of data and compare some summary data about the two collections. For example, collect Twitter user timelines from different political candidates and compare the number of retweets of their tweets.

You may use the programs twitter_lang.py as an example, but you must use different fields. You may also use twitter_hashtags.py or facebook_counts.py, but in these programs you must add a part to write a file. In all cases, you must change the comments to reflect your individual understanding of the program. If you only do one question, then it must be more complex than these simple examples; otherwise, you may choose additional questions.

Data:

You may collect data from Twitter, Facebook, or some other URL that returns JSON data. (If you want to use another format, such as XML, please ask.) If you collect your own, please collect at least several hundred data items, if possible.

You may request me to collect data for you, and if so, please make that request as soon as possible.

What to Submit:

In addition to the program that you write, you should write a report. In it provide:

- The data and its source, including any preprocessing
- A clearly stated question that describes whether it is a summary or comparison question and what fields are being used in the data
- A brief description of the program
- A description of the output files

For your program, you may use any of the code developed in class as a template, but it is absolutely essential that you use appropriate variable names and that you write original comments for what your program does.

Submit your report, your program and your output file(s).

Group Work:

If you choose to work in a group (of two), you may write and submit one program, but you must process the data for (at least) two comparison questions. Each member of the group should write some part of the program, even if edited later together. Your report

should describe the roles of the group members and who did what parts of the project, possibly including data, formulating questions, and debugging.

Final Project

Final Project Proposal
Scripting for Data Analysis
Due: 24 hours before the live session in Week 6

Planning for the Project

For this assignment, you are to make an initial plan for a project. In the final project you will demonstrate your ability to write Python scripts to access and amass data from fields in one or more of the three types of data studied in the course and to prepare and use data to produce data summaries, lists, and other structures.

1. Choose whether to work individually or to work in a team of two or three people. If you wish to work in a team, specify the people that you have talked with to form a team.

2. Pick a topic of investigation and the data that you will use, ideally from more than one source. The topic could focus on one main data set but also have supporting data. Your topic may focus on a single target topic or person, combinations of them, a comparison of more than one target topic and person, or comparisons over time.

The data may come from any source: those that you have found online, collected from social media, or obtained through other means.

3. Pick several possible methods of analysis in order to give some initial idea of what analysis you will try. This analysis will be to answer the types of questions that you have worked on for the homework assignments. Since we are not focused on visualization, the results of your analysis can be reported as structured tables with a unit of analysis and collected, summarized, or computed values for those units.

The scale of the final project must be larger in scope than the homework assignments in at least one of the following dimensions:

- Incorporating multiple data sets, possibly combining structured, semistructured, or text data
- Conducting additional related analysis questions (either more complex questions or more questions)
- Including additional types of analysis or collecting data, e.g., using another API or social network analysis

4. If you know of places where you may need help with development, try to list that now. This can range from big things (I want to get information from FourSquare

comments) to small things (I'd like a program that helps me to get dates from the documents in my collection and be able to compare dates).

[Potential remaining topics: Social network analysis, geographic locations and maps, getting all Facebook comments. You may request to add to this list.]

5. State in what way you intend for your project to be larger in scope than either of the homework assignments.

6. Based on your plans, you may want to start collecting data.

Assignment Result

Hand in a short document with your initial project plan describing your team, your topic, and your potential methods of analysis, including an assessment of the scope of the project.

Ideas for Data or Projects

Many websites where people have done analysis also give the sources of their information. For example:

- Nate Silver's 538 website has many examples of analysis. One is this article by Rob Arthur and Jeff Asher on gun violence in Chicago. They say that they got their crime data from the City of Chicago open data portal https://data.cityofchicago.org/.
- Data journalist Yue Qiu has a website with several projects reporting data on workers, trains carrying crude oil, and other statistics from various government websites.

Examples of data sets used by students for the first homework:

- Baseball hall of fame data
- Airbnb data from Kaggle
- Used car test data from the EPA website
- Somerville surveys for sense of safety
- Victim crime data from the Bureau of Justice Statistics
- Data sets from the UPI website: faculty use of Wikipedia data, forest fires, red wine quality, bike rentals

Comparing social media content with real-world events or other items. Examples:

- "Tweet the Debates": Shamma et al. collected tweets associated with political debates and reported on tweet volume over time and social networks of people tweeting.
- "Information Flows in Events of Political Unrest": Nahon and Hemsley compared tweet volume over time with the blogosphere and news stories of events.

- "Toward Predicting Popularity of Social Marketing Messages": Yu et al. selected restaurants with the most Facebook fans in different categories and analyzed popularity of the posts based on the number of likes and then analyzed the different types of posts by significant (most frequent) words.
- Comparison of different rock bands: Authors collected tweets and Facebook posts over time, looking at user timelines, retweets, likes, number of comments, number of entities, and most frequent words from the text.
- Analysis of tweets around an event: Collect tweets from event hashtags, showing significant words over time, network of people tweeting, and user locations on a map.

Examples of student Final Projects (not all questions are reported, so these do not necessarily reflect the full scope of the student work, and some are multistudent):

Data sources: EPA car review data, Edmund's Car Reviews and Dealership Reviews.
Questions: What are the car dealerships located in the vicinity of Syracuse, and how far away are they? How would you rate American-made automobiles according to mileage, horsepower, fuel efficiency, and cost?

Data sources: Twitter collection from Dave Matthews Band and Phish, including the user profiles and the last 2,000 tweets from their user timelines.
Question: Compare the popularity of the two bands by comparing follower and favorite counts from each profile, average numbers of retweets, and retweets and favorites per followers.

Data source: Tweets collected April 18, 2016, around #parisattacks OR #bataclan with 32K tweets.
Questions: What are the demographics of the tweets? Who are the most influential users (using SNA and retweets)? What are the demographics of the information (looking at the URLs)? What are the sentiments expressed in the tweets?

Data source: Tweets about Boston Red Sox and NY Yankees, and Facebook posts and comments from the two teams' Facebook fan pages.
Questions: Updated analysis of many questions based on earlier article by Bialik in Five-thirty-eight for 2014.

Data sources: Airbnb data set, collected tweets about Airbnb.
Questions: What factors influence the customer review scores? How much money can each host make in a particular time period? Can we use tweets about airbnb to discover recent popular travel trends?

Data sources: MovieLens data set with 100K reviews and selected movie reviews downloaded from IMDb in HTML.
Questions: Do movie ratings differ according to gender and genre? Do movie reviews differ by gender for movies with male or female protagonists?

Data sources: Tweets around the topic #techno, user profiles from SoundCloud in the genre techno, from the API.
Question: Compare the demographics of the Twitter users and the SoundCloud users.

Final Project Presentations

To be presented in Week 10 and 11 live sessions

- Length of presentation slots
    - 1 person: 10 minutes
    - 2 people: 15 minutes
    - 3 people: 20 minutes
- Leave time for questions (2 minutes per presentation)
- Practice!
- What to include:
    - Data sources
    - Single analysis question that was answered
        - Can build upon this section if you have multiple group members
    - Overall description of the program you created

Final Project Report—DUE: 24 hours before the live session in Week 10

In addition to carrying out the project, you must write a final project report. In this report, you should

- Describe the data and its source(s), including any preprocessing
- Describe your methods of analysis, including the questions that will be answered, in what fields the data will be used, and what the resulting output will be
- Include an overall description of the program
- If your project is a group project, describe the tasks and roles of each member of the group
- (Grad students) Draw conclusions from your results about your data

What to Submit

- The project report
- The Python program, documented with comments
- The output(s) of the program

Course-Specific Policies

Attendance is required and cannot be made up except for excused absences for illness and for other school-sanctioned activities. For assignments, late work will be accepted but will be penalized with the points equivalent to ½ of a letter grade for one day late and on a sliding scale for additional lateness of up to a full letter grade.

Academic Integrity

Syracuse University's Academic Integrity Policy reflects the high value that we, as a university community, place on honesty in academic work. The policy defines our expectations for academic honesty and holds students accountable for the integrity of all work they submit. Students should understand that it is their responsibility to learn about course-specific expectations, as well as about university-wide academic integrity expectations. The policy governs appropriate citation and use of sources, the integrity of work submitted in exams and assignments, and the veracity of signatures on attendance sheets and other verification of participation in class activities. The policy also prohibits students from submitting the same work in more than one class without receiving written authorization in advance from both instructors. Under the policy, students found in violation are subject to grade sanctions determined by the course instructor and nongrade sanctions determined by the School or College where the course is offered as described in the Violation and Sanction Classification Rubric. SU students are required to read an online summary of the University's academic integrity expectations and provide an electronic signature agreeing to abide by them twice a year during preterm check-in on MySlice.
For more information about the policy, see http://academicintegrity.syr.edu.

Disability-Related Accommodations

Syracuse University values diversity and inclusion; we are committed to a climate of mutual respect and full participation. If you believe that you need accommodations for a disability, please contact the Office of Disability Services (ODS), disabilityservices.syr.edu, located at 804 University Avenue, room 309, or call 315.443.4498 for an appointment to discuss your needs and the process for requesting accommodations. ODS is responsible for coordinating disability-related accommodations and will issue "Accommodation Authorization Letters" to students as appropriate. Since accommodations may require early planning and generally are not provided retroactively, please contact ODS as soon as possible. Our goal at the iSchool is to create learning environments that are useable, equitable, inclusive, and welcoming. If there are aspects of the instruction or design of this course that result in barriers to your inclusion or accurate assessment or achievement, please meet with me to discuss additional strategies beyond official accommodations that may be helpful to your success.

Religious Observances Notification and Policy

SU religious observances notification and policy, found at http://hendricks.syr.edu/spiritual-life/index.html, recognizes the diversity of faiths represented among the campus community and protects the rights of students, faculty, and staff to observe religious holidays according to their tradition. Under the policy, students are provided an opportunity to make up any examination, study, or work requirements that may be missed due to a religious observance provided they notify their

instructors before the end of the second week of classes for regular session classes and by the submission deadline for flexibly formatted classes.

For fall and spring semesters, an online notification process is available for students in **My Slice / StudentServices / Enrollment / MyReligiousObservances / Add a Notification.** Instructors may access a list of their students who have submitted a notification in My Slice Faculty Center.

Educational Use of Student Work

Student work prepared for University courses in any media may be used for educational purposes, if the course syllabus makes clear that such use may occur. You grant permission to have your work used in this manner by registering for, and by continuing to be enrolled in, courses where such use of student work is announced in the course syllabus.

I intend to use academic work that you complete this semester in subsequent semesters for educational purposes. Before using your work for that purpose, I will either get your written permission or render the work anonymous by removing all your personal identification.

Course Evaluation

There will be an end-of-course evaluation for you to complete this semester, described below. This evaluation will be conducted online and is entirely anonymous. You will receive a notification from the Syracuse University Office of Institutional Research & Assessment (OIRA) department in your e-mail account with the evaluation website link and your passcode.

- End-of-semester evaluation will be available for completion approximately Week 10. This evaluation is slightly longer, and it is used to gauge the instructor performance and make adjustments to the course to ensure it meets our student needs.

We faculty work hard to do the best possible job when preparing and delivering courses for our students. Please understand that not only does the school use the course evaluations to make decisions about the curriculum in order to improve where necessary, but it also uses them to make decisions about faculty members. Please take the time and fill out this evaluation as your feedback and support of this assessment effort is very much appreciated.