Lab #4 – Reading and plotting CSV data

Goals of the lab:

In this lab, we'll

- Learn another fun trick for string processing, string.split(), and use it to read CSV data
- Reinforce our skills in drawing plots and in using them to mislead 😊.

Overview of the lab

We'll be reading Covid infection data from the NY Times county-by-county Covid database for the state of Massachusetts. We'll pull out only Middlesex county (which includes Medford) and plot it. We can then compare it with the Tufts data from last week to find any interesting conclusions.

Details of the lab

Here is what your program should do (first, in normal English):

- You should write a function *read_NYTimes_data*() to read the data from the the NY Times data file. The function will return a list that contains the cumulative case count for Middlesex county on each day. It will also return two strings: the first and the last date that the file has data for.
- Using the cumulative data, compute the new cases on each day for the entire date range (similar to last week's lab).
- Plot the new cases per day (using the same methodology as in the previous lab).
- Create an array of *running-average* data. The running average is just the average of the previous seven days. So, in general, *running_avg*[*i*] is the average of the seven numbers *student_cumulative_cases*[*i*-6:*i*+1]. The first six elements *running_avg*[0:6] will be a special case; they are just the average of *all* the previous elements.
- Plot a seven-day running average on the same figure.

The NY Times data file

We'll be reading data from just one file this time. It came from the New York Times Covid database (https://github.com/nytimes/covid-19-data). You can find it at http://www.ece.tufts.edu/es/2/data/NYT/NYT_mass_counties.txt.

It has many lines, where each line looks similar to this:

```
2020-04-02,Middlesex,Massachusetts,25017,1870,29
```

This is the *CSV* format, and is yet another fairly standard method of describing data via text. *CSV* stands for Comma-Separated Values, and this format is simply a comma-separated list of values. While there are prepackaged CSV readers easily available on the web (e.g., the *csvreader* package), we'll spin our own so that we get a bit more practice with strings.

What do the fields of our particular file mean? The first field is a date. In fact, the dates in the file should always be in ascending order and include every day (i.e., we never skip from March 3 to March 5).

The next field is a county in Massachusetts (Tufts is in Middlesex county). Next comes the state, which is always Massachusetts for our file.

Finally come three numbers. The first is a geographic locator that you may ignore. The next two are the cumulative numbers for cases and deaths for that date and county (in this example, 1870 cases and 29 deaths).

Metacode

Here's metacode for the top-level code flow:

```
(cases, first_day, last_day) = read_NYTimes_data ()
print ("The data runs from", first_day, "to", last_day)

# Just like lab #3
create a list new_cases_per_day by subtracting counts like last time;
but this time, force any negative numbers up to zero.

# Create a running-7-day-average list. Make sure this one is the same
# length as the new_case_per_day list!
running_avg = an empty list
for i from 0 to (the index of the last element of new_cases_per_day):
    # usually first would just be i-7, but it will be special near
    # the beginning of the list.
    first = the index of the *first* element of the 7-day average
    avg = average of the numbers in new_cases_per_day[first:i+1]
    append avg to the running_avg list
plot both lists (new_cases_per_day and running_avg) on the same figure
```

Like last time, there will be one occasion (Sept. 2nd) when the cumulative case count actually decreases (see the *Questions* section below). To keep our graphs pretty, we force the resulting negative per-day new-case number up to zero.

While most of the lines above are reasonably close to Python, the *read_NYTimes_data*() function is still quite vague. So here's the metacode for it:

```
def read_NYTimes_data ()
      initialize first_date and last_date to something 😊
      initialize cases to an empty list
      open the NYTimes data file
      for each line in the file:
            fields = the fields from line
            if this line is for Middlesex county:
                  grab the date and case_count fields
                  update first_date and last_date appropriately
                  append case_count to cases
      return (cases, first_date, last_date)
```

That's the high-level *metacode*, which tells you "roughly" how to code everything. But I've purposely left some of the details for you! Specifically, "grab the *date* and *case_count* fields" and "update *first_date* and *last_date* appropriately."

How should you "grab the *date* and *case_count* fields?" I.e., how do you parse a single line and extract the fields that you want? Perhaps the easiest method is to use the function *string.split*(). So if *line* is a string that holds one line of text, then *line.split* (",") will return a list of individual strings (six strings in our case). Now you can work with the fields one by one however you like!

You will probably also want to convert the "numbers" from strings to integers. The Python code **int** ("13") converts from the string "13" to the integer 13.

Challenge problem

1. Like last time, you can use actual dates for the *x* axis of your graph.

Questions

For questions #1 and #2, you may want to refer to the NY Times explanation of their Massachusetts data at https://www.nytimes.com/interactive/2020/us/massachusetts-coronavirus-cases.html)

1. You should see a decrease in the cumulative case count on September 2, 2020. Do you have any explanation for this?
2. Can you explain the data on Nov. 26th, Dec. 25th and Jan. 1st? (These correspond to days #266, 295 and 302).
3. Compare the data from roughly Jan 14th through Feb 28th in this lab vs. what you saw in the previous lab. What conclusions would you draw? Are the Tufts infections roughly following the same pattern as Middlesex county? If not, why might that be?

What to turn in:

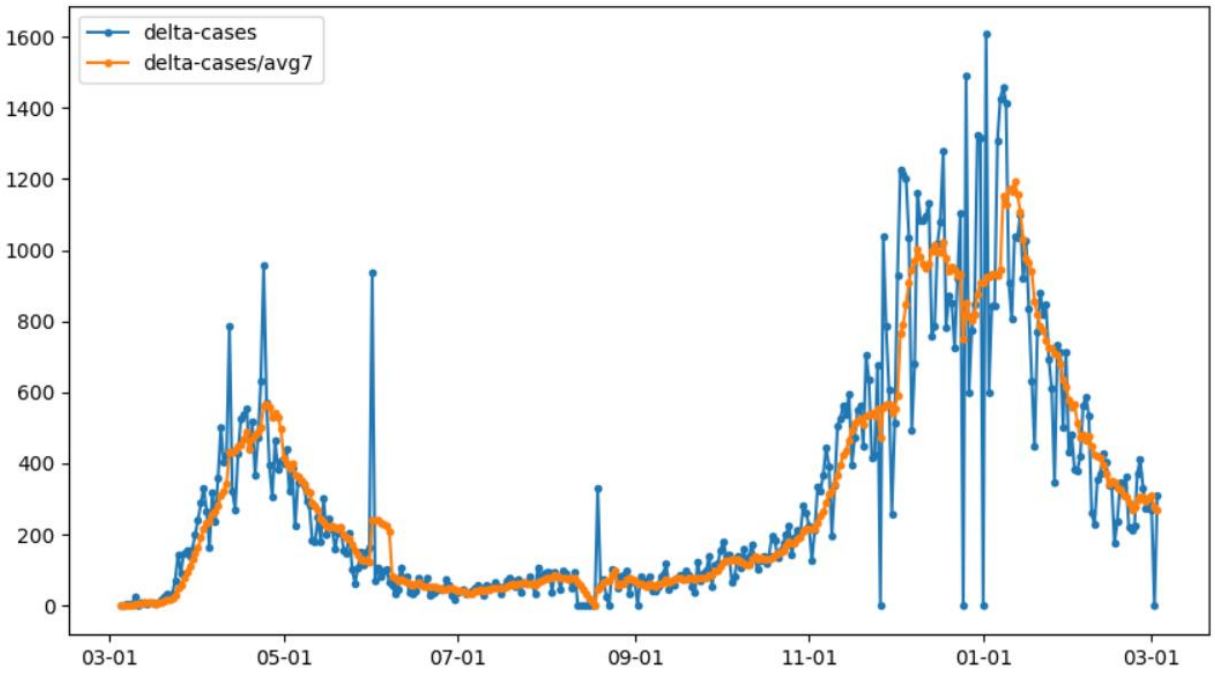
- Your program, *lab3_read_NYTimes.py*
- A .pdf with your one plot and the answers to the questions.

***Grading*:**

- Code & plot correctness: 60 pts
- Code clarity: 10 pts
- Questions: 10 pts each

Example plots

Here is what the plot should look like. However, note that this is the "extra challenge" version that has dates rather than numbers for the *x* axis (also, yours probably won't have all the dots).


delta-cases
delta-cases/avg7
1600
1400
1200
1000
800
600
400
200
0
03-01
05-01
07-01
09-01
11-01
01-01
03-01