Sean O'Grady
CSE 332 Lab 5 Report

Introduction

The objective of this lab was to compile multiple different data visualizations for each student's dataset used through the semester into one dashboard. All visualizations have to be visible on one page, and must have different brushing and linking interactions. The intention of having all of these on one page with interactivity between them was to be able to better investigate "data stories", or comprehensive insights into what the data can tell us. The requirements for the dashboard are to have at least four plots, one of which must not be from previous labs, and to include at least two different brushing/linking interactions between the plots. The dashboard is to make use of the D3.js library in order to create the visualizations, as well as the Flask Python web framework.

Description

My dataset used throughout the semester regards COVID-19 cases, hospitalizations, and deaths in New York City, as well as data for just Brooklyn. My dashboard was made using Python's Flask web framework, as well as HTML, Javascript, and CSS.

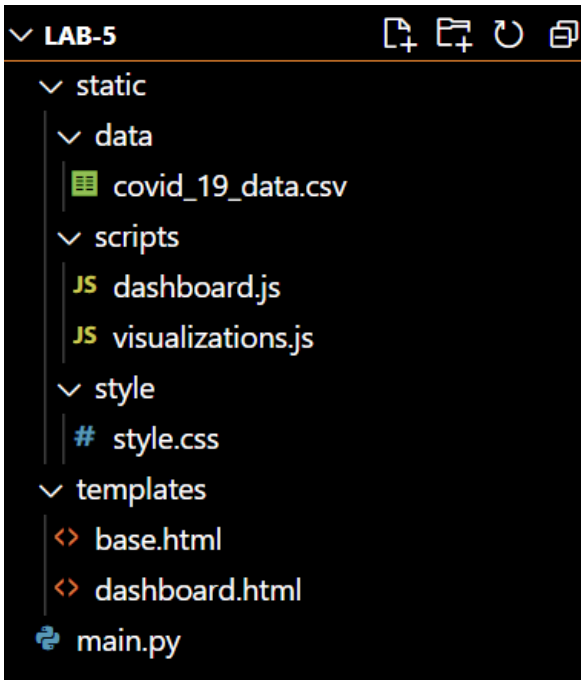

Fig. 1. The file structure for the dashboard project. The page can be launched by running "main.py", then checking the terminal for the address that it launched at.

The visualizations from previous labs that I chose for my dashboard include a parallel coordinates plot, a correlation matrix, and a scatterplot matrix. The new visualization that I chose to include was an area plot.

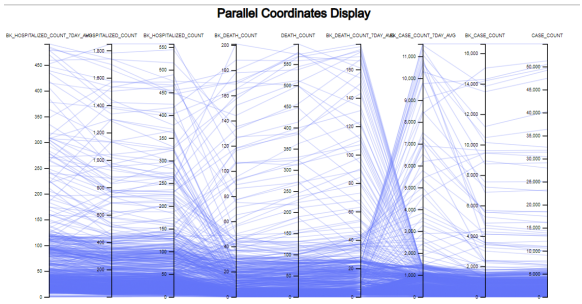

Fig. 2. My parallel coordinates display.

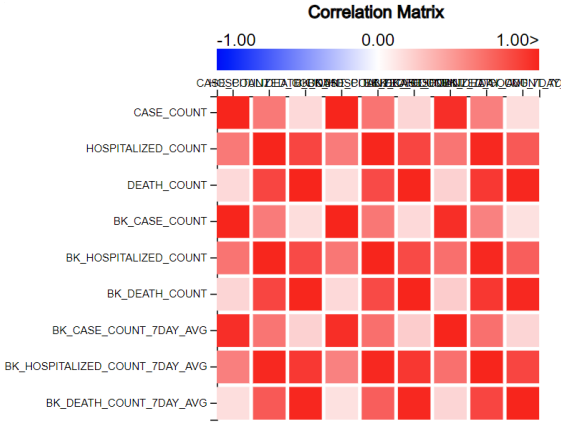

Fig. 3. My correlation matrix.

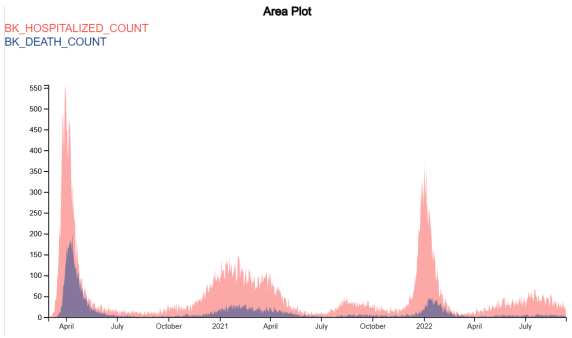

Fig. 4. My area plot (time series with two attributes).

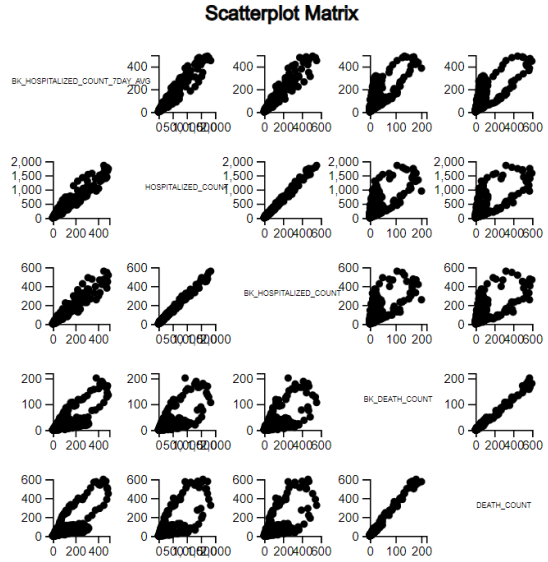

Fig. 5. My scatterplot matrix.

I have also included a few brushing and linking techniques on the dashboard. When you hover over a square on the correlation matrix, it shows the two attributes that the square represents, as well as the correlation between them. When you click on a square, it will update the area plot to display those two attributes. The area plot itself is brushable on the x-axis, narrowing the date range that the data displays. This will update the parallel coordinates display to show only data from that range. The user can double click the area plot to return to the full date range, which also restores the parallel coordinates display to show all data.

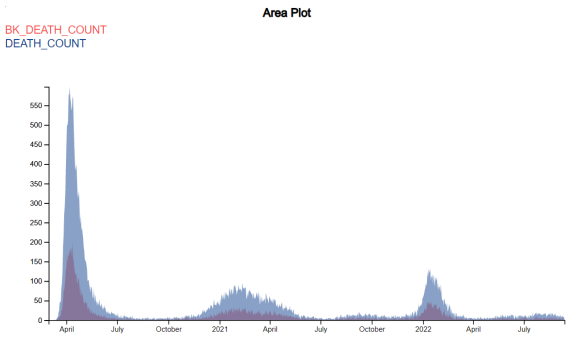

Fig. 6. My area plot after clicking the BK_DEATH_COUNT/DEATH_COUNT square on the correlation matrix.

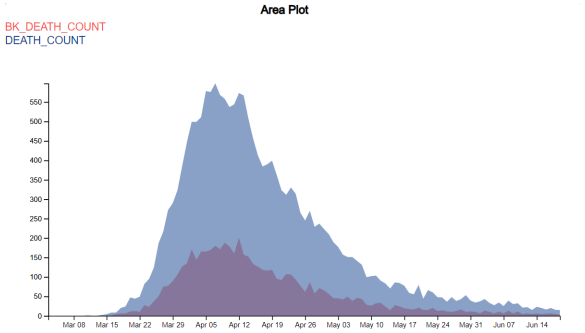

Fig. 7. The same plot as Fig. 7 after brushing the date range.

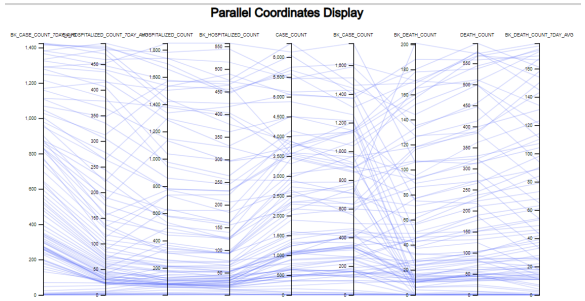

Fig. 8. The parallel coordinates display after brushing the area plot's date range.

There is also a refresh button to return everything back to its default state. For reference, the defaults are set so that the area plot displays the relationship between CASE_COUNT and HOSPITALIZED_COUNT, and the parallel coordinates display shows the full range of data.

Data Story 1

As I was investigating what insights I could gain from the dashboard about my data, I came across something that was potentially useful. I was clicking through the correlation matrix to see if the area plot would show anything interesting, which it did when I selected BK_HOSPITALIZED_COUNT/BK_DEATH_COUNT (the 7-day average version of these attributes is easier to look at). It can be seen that while highly correlated (0.75), the relationship between them is consistently staggered with the peak of the deaths happening after the peak of the hospitalizations. After looking closer at the peaks by brushing the area plot, it can be seen that the peak in deaths typically started about 1-2 weeks after the peak in hospitalizations. This could explain the interesting shape of the scatterplot between these two variables, where it shows two separate trends, with almost no data between the two trends. This is likely because the deaths tended to be ramping up when the hospitalizations were peaking, and the hospitalizations were winding down when the deaths were peaking. Regardless, this has given a good insight into the time offset between hospitalizations and deaths, and could potentially be used by hospitals during an influx of patients in order to better plan resources.

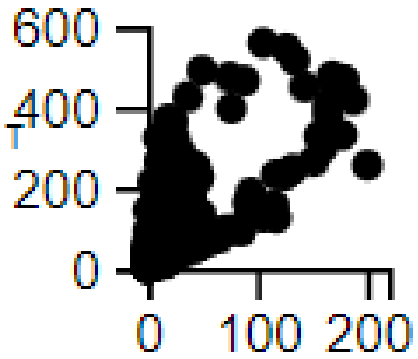

Fig. 9. The scatterplot between BK_HOSPITALIZED_COUNT and BK_DEATH_COUNT.

Data Story 2

Another interesting insight to note is the relationship between the 7-day average death count and 7-day average case count, both from Brooklyn, on the parallel coordinates display. When there are some of the highest death counts, there are some of the lowest case counts, and vice versa. Looking at the correlation matrix, these two variables have one of the lowest correlations, at 0.15. This shouldn't make sense however, as it's logical that more cases means more deaths. However, the scatterplot matrix can help clear up this seemingly illogical finding, as there are two separate trends that err closer and closer to the x and y axes respectively the lower the correlation between the two variables go. It's not that they're not correlated in reality, but the staggered, yet parallel nature of the variables makes the Pearson's correlation less useful as the strength gets closer to 0. It can be seen on the area chart that there is a visible difference between the peaks in cases and the peaks in deaths. These findings are revealing of the nature of the COVID-19 pandemic in New York city, as the cases, hospitalizations, and deaths remain fairly level until there are brief, yet severe flare-ups.

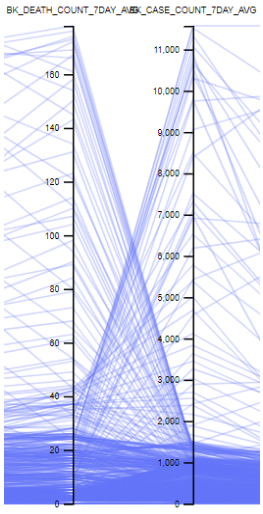

Fig. 10. Section from the parallel coordinates display showing the relationship between the 7-day average variables discussed.