Link to Shiny Application: https://github.com/k-you28/NYPD_Shiny_App/tree/main

In my Shiny App, it displays information regarding the weekly arrests that are made in New York among all five boroughs: Manhattan, Brooklyn, Queens, Staten Island, and the Bronx. The app allows the user to select their interested boroughs, a date range, and the type of offense they want to look up. A blue histogram is then displayed along with a red smooth trend curve with a fill. The x-axis represents the date while the y-axis represents the number of arrests made on said date. The core UI features include being able to choose a starting and ending date for the data that you want displayed, which boroughs you want to display, and the type of offense you are interested in visualizing. This is my NYPD Arrests - Interactive Visualization.

As I was working through my app, I did learn some interesting things about this data. But something that piqued my interest the most is that when looking at arson offenses, in the last two months of 2014, there seems to be a huge spike in arson arrests. The typical number of arson arrests lies underneath five per week, but this one specific week had almost thirty. I did try to look up what kind of events occurred that could cause such a huge spike, but couldn't find anything aside from a couple of apartment fires. It most likely is just a coincidence that there were so many arson incidents in the same week, but I am still not completely certain. Something else I found worth mentioning is that crime always seems to dip right before the year begins (aka November and December), so it seems lawbreakers celebrate Christmas.

I began by determining which data I found the most interesting in the dataset and settled on the amount of crimes in relation to time. I also decided that adding boroughs would give more flexibility for the user so I made borough selections a feature too. In terms of data preparation, there was a little bit. Removing N/A rows was one step and another was renaming all boroughs to their full names rather than working with their abbreviations. After all, trying to read M

instead of Manhattan is confusing to say the least. The offenses that had less than five hundred counts were taken out as there is not enough data to make any meaningful conclusions.

The reactive graph structure of my application is quite simple. There are three different inputs: offense, borough, and dateRange. Then there are two reactive expressions: weeklyData() and filteredData(), which combined, accept all three inputs mentioned above. Finally with these two reactive expressions, they are used as input for all two different outputs: arrestPlot and summaryText.