Alpha Release

Anna Jurgensen

April 12, 2021

1 Overview

Lake Merritt is one of my favorite places in the Bay Area. For several years I rowed and volunteered with the Lake Merritt rowing club, running or biking the lake is my husband's typical form of exercise, and we live in the neighborhood nearby, just 200m up a street from Lakeshore Avenue. While the area has been a pleasant place to live and recreate for years, during the pandemic the use of the lake has increased and the behavior of visitors has gotten worse. This has led to a year of complaints by frustrated residents about crushing traffic, blasting music, littering, and worse
(https://oaklandside.org/2021/03/23/oakland-closer-to-adopting-regulations-for-vending-and-crowds-at-lake-merritt/,
https://oaklandside.org/2021/03/23/oakland-closer-to-adopting-regulations-for-vending-and-crowds-at-lake-merritt/,
https://oaklandside.org/2020/08/21/im-fighting-to-be-able-to-live-in-my-own-home-lakeshore-residents-share-serious-concerns/). I wanted to look at the City of Oakland's "311 request" data that reflected issues like littering, illegal dumping, vandalism, and abandoned cars in the immediate Lake Merritt area to better understand if and how these problems have changed over time, particularly over the course of the pandemic.

I am using data from the City of Oakland's 311 request system, which allows members of the public to report issues like graffiti, illegal dumping, and damaged infrastructure to the city via a phone call, a web form, or phone apps
(https://www.oaklandca.gov/services/oak311,
https://apps.apple.com/us/app/oak-311/id1316091489,
and https://seeclickfix.com/oakland). I accessed the publicly available data via the Socrata Open Data API
(https://dev.socrata.com/foundry/data.oaklandca.gov/quth-gb8e) via its python client, sodapy (https://github.com/xmunoz/sodapy). I cleaned the data in python using the pandas module, and using python module geopandas, I clipped the data to a polygon that included the lake, the parks that flank the lake, and a swathe of the adjacent neighborhood approximately 500 meters deep, which I generated using Wicket
(http://arthur-e.github.io/Wicket/sandbox-gmaps3.html).

Using python I saved the cleaned data as a JSON file to use with D3, and also as a CSV file to use with R for my initial, static visualizations.

2 Completed Features

I have completed the static visualizations I had planned to for the alpha release. These comprised

- an interleaved bar chart comparing pre-pandemic to pandemic era total volumes of requests for different issues types
- line charts for different request issue types comparing the monthly volumes of requests across 2020 and the 3 preceding years
- line charts for different request issue types comparing the volumes of requests across the days of the week for 2020 and the 3 preceding years
- a violin plot comparing distributions in the time it takes the city to fully address and close requests of different types

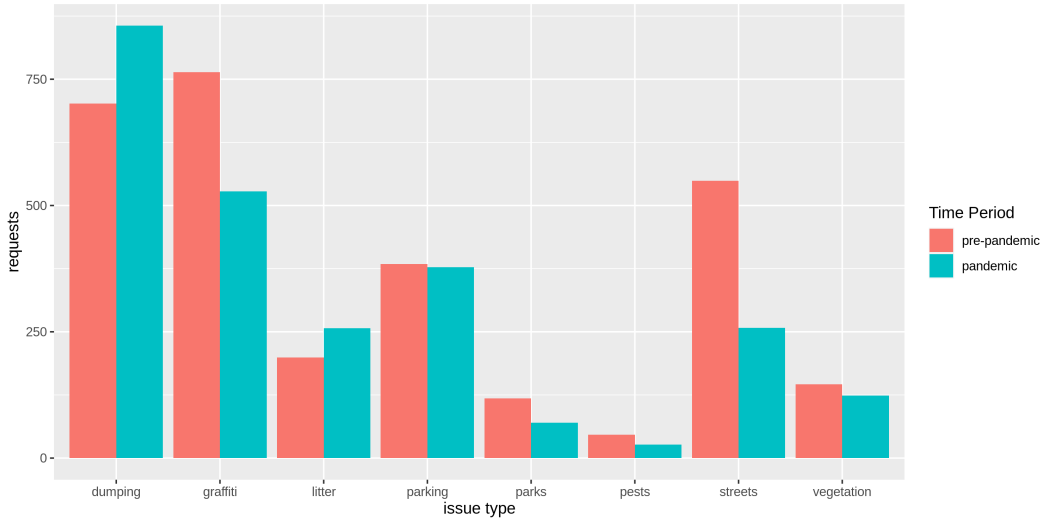
I've also complete a short history of the lake and mention of the controversy that motivated my project.

3 Static Visualizations

3.1 Pre-Pandemic vs. Pandemic Request Volume

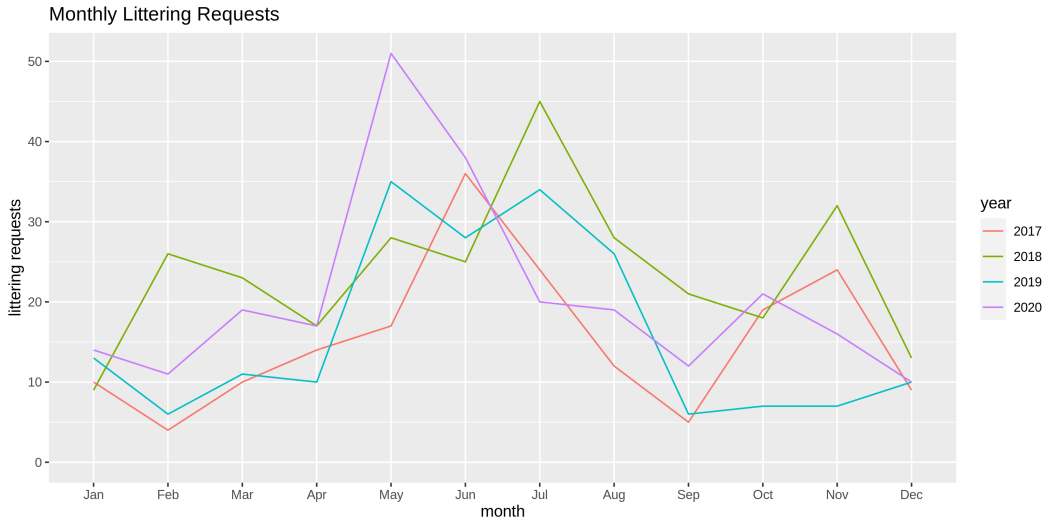
I thought that if the conditions at the lake have deteriorated significantly, we would see many more requests of all types during the pandemic (which I defined

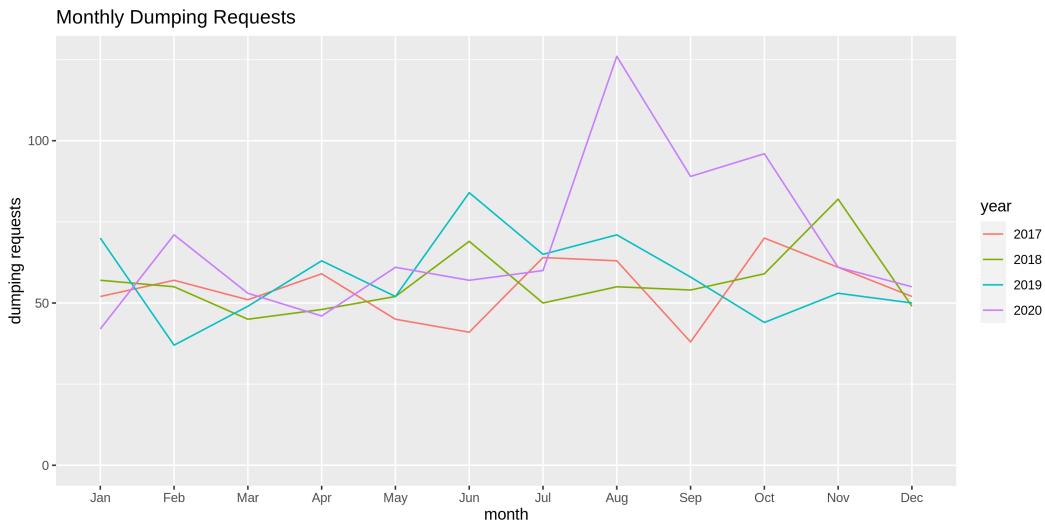
as March 1, 2020 through February 28, 2021) to the year preceding it (March 1, 2019 through February 29, 2020). Comparing the total volumes for these two periods across a number of the request issue types, however, it's clear that very few of the issue types saw any amount of increase during the pandemic from the prior year.

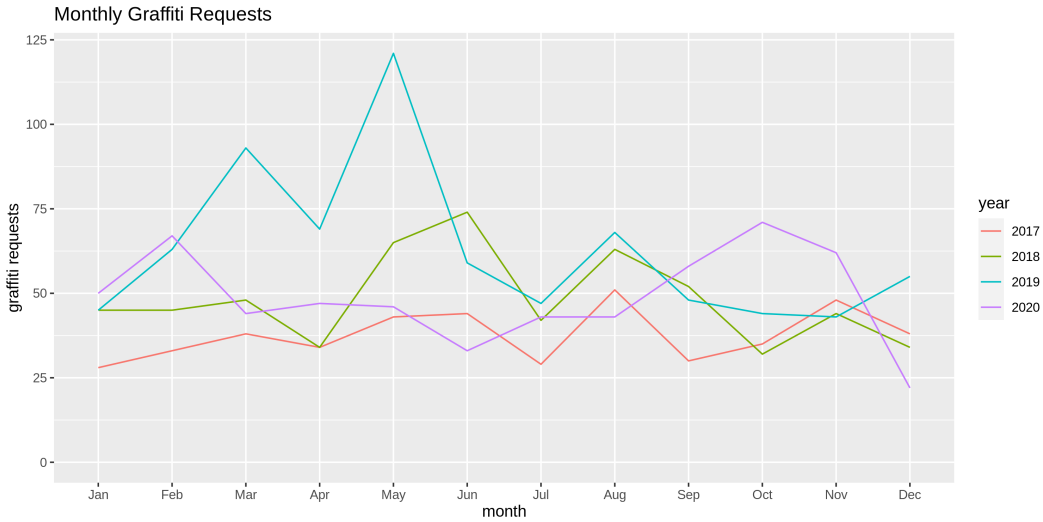


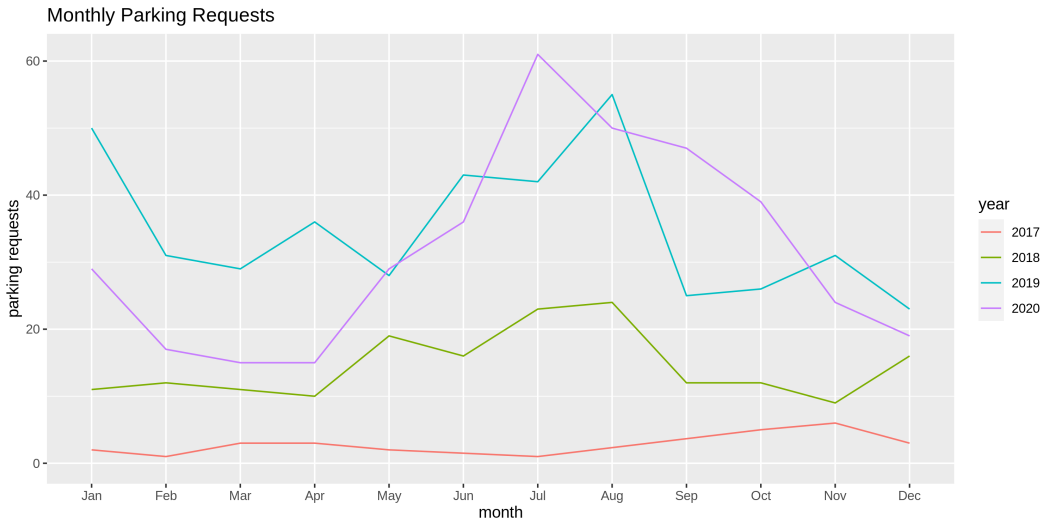
3.2 Request Volume by Month of Year

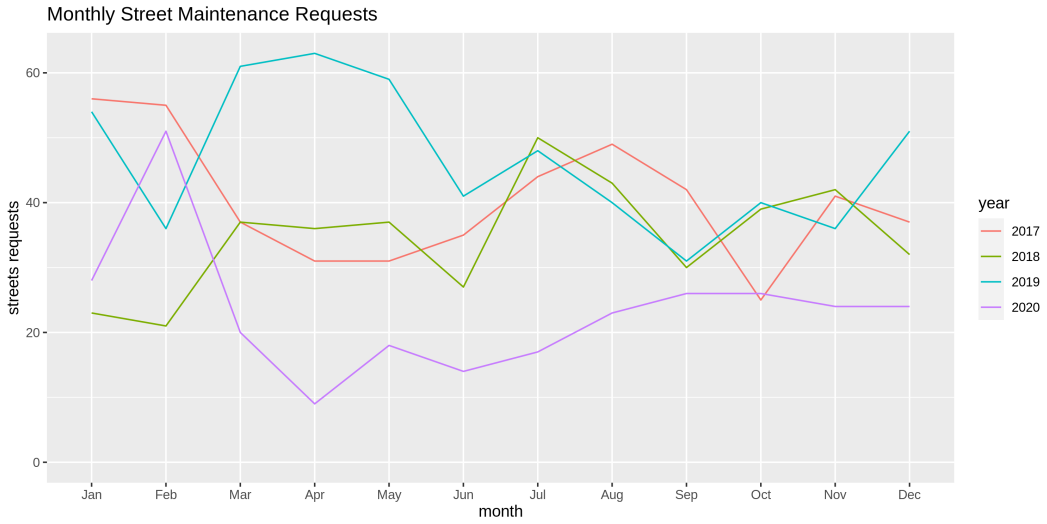
I believed that looking at request volume across the months of the year, comparing 2020 to the preceding years might be more insightful than just bulk yearly volume of requests. Would there be higher volumes in the summer of 2020 than previous years (as would be expected given residents' complaints) or not? To examine the data for these patterns I created line charts for different request issue types comparing the volumes of requests across the months of the year for 2017-2020.


Monthly Littering Requests
littering requests
50
40
30
20
10
0
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
month
year
2017
2018
2019
2020


Monthly Dumping Requests
dumping requests
100
50
0
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
month
year
2017
2018
2019
2020

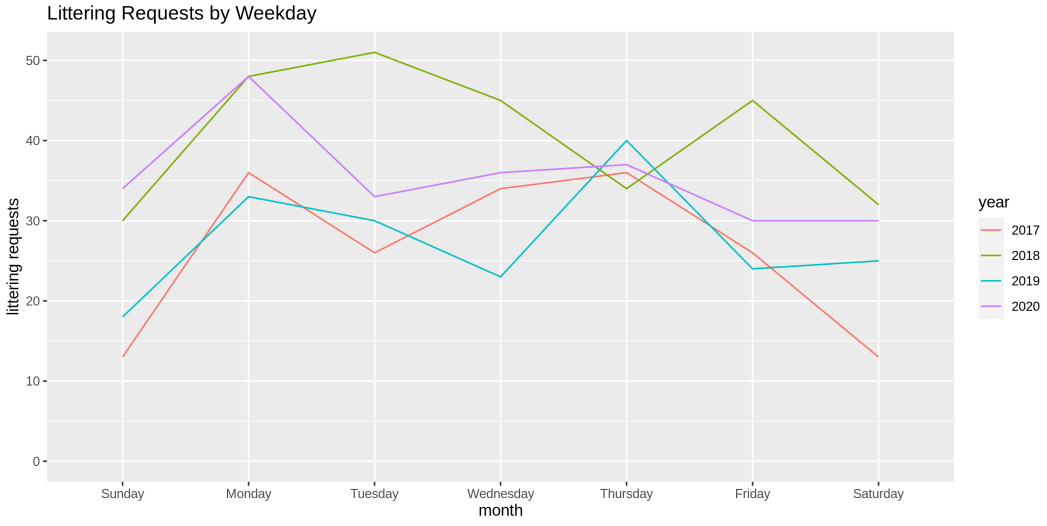

Monthly Graffiti Requests
graffiti requests
125
100
75
50
25
0
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
month
year
2017
2018
2019
2020

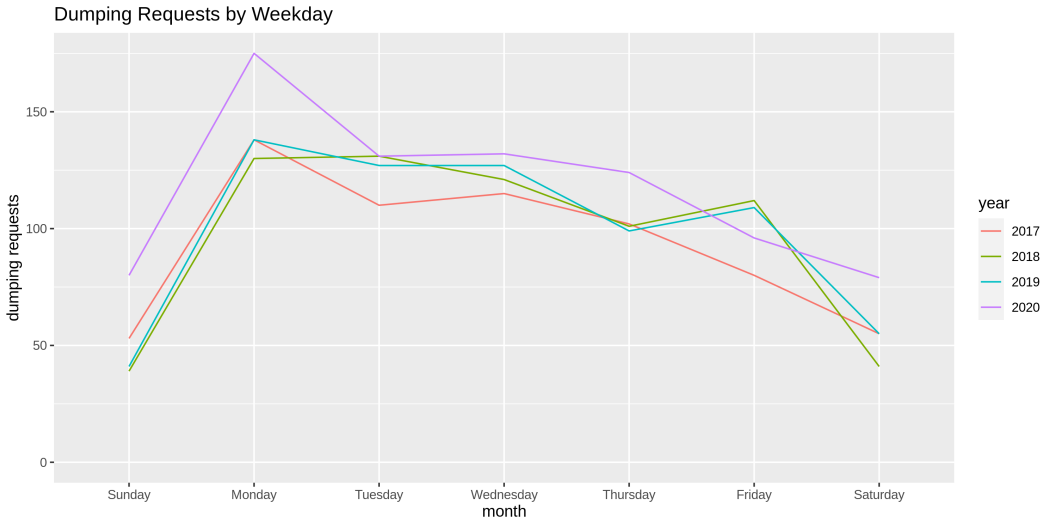

Monthly Parking Requests
parking requests
60
40
20
0
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
month
year
2017
2018
2019
2020

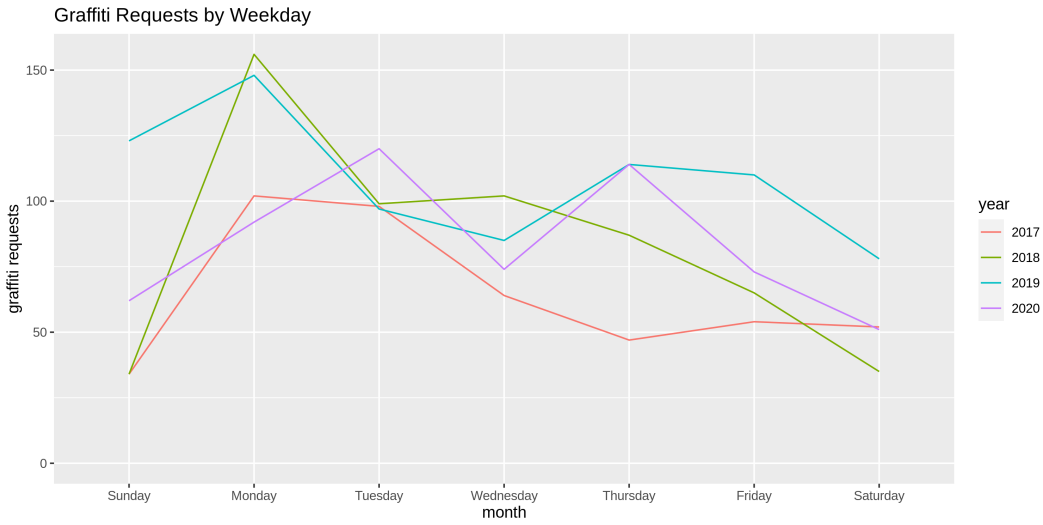


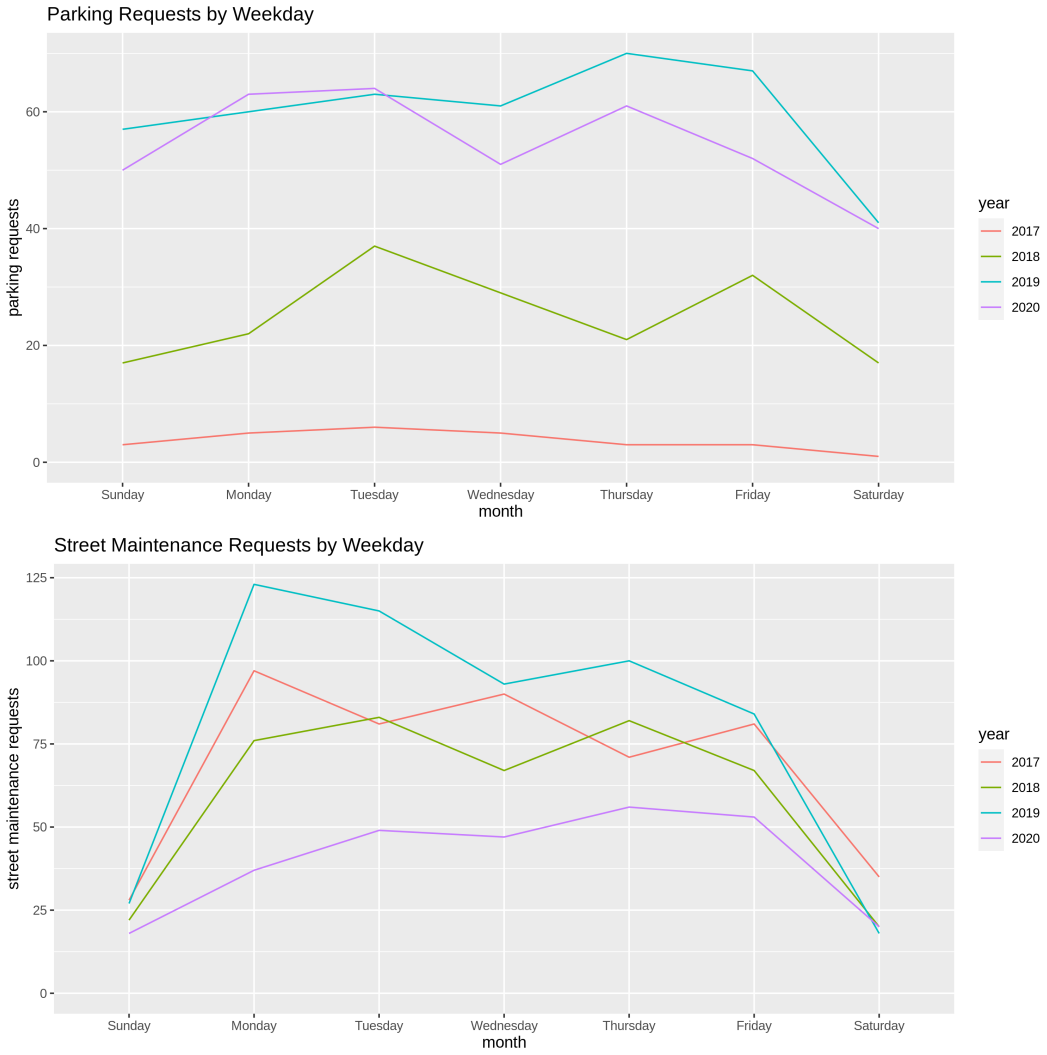
3.3 Request Volume by Day of Week

Further, I thought when the requests were being made during the week might be interesting, too. Were requests mostly made during the weekend (when the lake has the most visitors), or would they be in the aftermath of the weekend on Monday when residents noticed the issues, or even midweek, perhaps signalling the city isn't addressing issues like overflowing garbage bins quickly enough. To examine the data for these patterns I created line charts for different request issue types comparing the volumes of requests across the days of the week for 2017-2020.



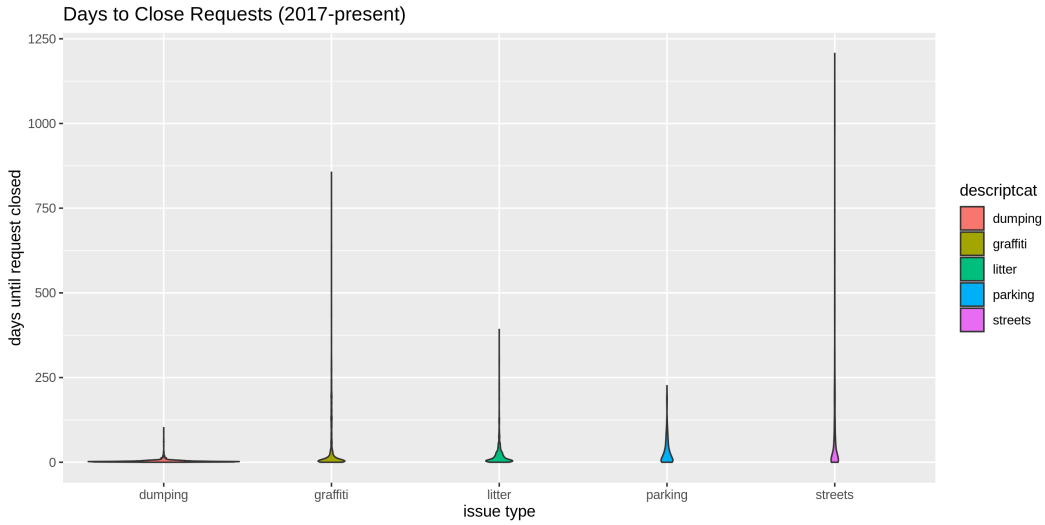

Dumping Requests by Weekday
dumping requests
150
100
50
0
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
month
year
2017
2018
2019
2020


Graffiti Requests by Weekday
graffiti requests
150
100
50
0
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
month
year
2017
2018
2019
2020



3.4 Time to Close Requests

I thought that residents' frustrations with problems around the like might not be entirely due to an increase of visitors; it might be due, at least in part, to the city being slow to address certain types of complaints and residents, spending more time around the lake since they are working from home during the pandemic, might just be noticing the city's inefficiency. To examine this, I generated a violin plot to compare the distributions in the number of days it takes the city to fully address and close requests of different types.



4 Upcoming Immediate Milestones

The most immediate milestone is setting up my page to use scrollama, to which I'll add my plots going forward, accompanied by my discussion of the plots and possible interpretations.

The next most immediate milestones to tackle will be adding the map features to the page. One map will be added which will display the areas and features of the lake. Another interactive map will be added that will allow the user to plot incident locations for different types of service requests.

Also of urgent concern is remaking the static plots as interactive plots in D3 in order collapse multiple static images into single interactive ones.

5 Roadblocks

The only true roadblock I have encountered is the way Oakland's system encodes issue types obscures whether requests for repairs to damage, whether in parks or on the streets, is due to vandalism or not, so I cannot actually draw any conclusions about vandalism as I had hoped[1].

The city's encoding of issue types resulted in more cleaning of the data than I had anticipated. At first glance, it looked as though I could use a field called "request category," which had values like "graffiti, "illegal dumping", and

[1]The Oakland Police Department's crime data probably would have been more useful for this, and could have shed light on some other areas of concern that residents have had, like an increase of violent crimes and property crimes, but even through the Socrata API the OPD crime data is only available for the previous 90 days (and only available in very aggregate form afterward), so it was impossible to get fine-grained data I needed to examine the cases throughout the pandemic, let alone compare the cases to any time period pre-pandemic.

"parks." However, on closer inspection I found that the true issues crosscut these labels. For example, graffiti-related requests were not only found with request category of "graffiti," they were also found with labels like "parks," "traffic," and "building maintenance." As a result, I had to manually sort through the 320-odd values of the "request description" field to group the related request issues together. Though this was annoying and slowed my progress, it was not an insurmountable problem.