Capstone #2 Project Milestone Report

Pawandeep Jandir

Introduction

The City and County of San Francisco operates a 311 program. It is the official service for residents, visitors, and businesses to obtain information, report problems, or submit service requests to the city (non-emergencies only). This system is available 24 hours a day and is meant to be a fast and efficient way to interact with the local government. Requests can be made online, over the phone, though the mobile app, amongst other ways.

The Problem

The 311 program offers a glimpse into the local government and how it operates. The interactions represent, to some degree, the needs and desires of San Francisco. This presents an opportunity to ask the question: can we predict future interactions, specifically, service request volumes? Even further, do certain service requests come from certain areas or neighborhoods of San Francisco? By answering these questions we can try to optimize the government, via the 311 program. This would allow the city to be more proactive and responsive to the needs of people through predictive modeling. To help with the prediction, socio-economic and demographic data (from the US Census Bureau) of San Francisco.

The Client

The potential target for this project is the San Francisco local government. The analysis can help the city understand how certain city services are used and requested throughout different parts of the city. A predictive model providing count or volume data on future service requests can help the local government in becoming more effective for its citizens and visitors. By becoming more efficient and proactive, the San Francisco local government can improve the city and make a large positive impact on its communities.

The Data

The main data source is the 311 request data available at San Francisco's open data portal. In addition, San Francisco's socio-economic, demographic and other types of data is available from the US Census Bureau FactFinder website. Lastly, San Francisco's open data portal also provides a shapefile, in the form of a GeoJSON, outlining the city. This is needed during the data cleaning phase and will be discussed in further detail in that section, below.

Data wrangling

This section details the data wrangling and cleaning of the various data stores to create a single cohesive dataset.

Acquisition

After identification of data sources, the next step is to obtain it somehow. For the 311 dataset, we can go to San Francisco's open data portal website for the 311 program. From here, it is as simple as downloading the dataset as a CSV file. Additional export methods are also available including JSON and XML. An API also exists which makes streaming the data a possibility as well. For the demographic

data, we go to the FactFinder website maintained by the US Census Bureau. From here, we submit the correct location: City and County of San Francisco and request a breakdown by Census Tract. We also specify the American Community Survey dataset. From here we select the relevant socio-economic and demographic data columns. See Tab. 1 for the full list of these variables.

Cleaning

First we read-in the 311 data with *pandas* which has over 2.8 million rows. A warning is given about the last column of the input CSV file. It is a column with a string for a hyperlink to an associated image or picture to the service request. Not all of the rows have this information, so the first thing we can do is simply drop this column and replace it with a boolean column showing wether or not there was a string. The information contained in the links is perhaps useful but is outside the scope of this project and can perhaps be investigated in another project. Next, because this is a time series dataset we want to ensure the time-based columns are recognized as such, so we convert them to datetime.

Now we can see the dataset starts midway through 2008. Because we want to capture seasonality and fully model the data, we can throw out the rows from this partial first year. Similarly, we can throw away data from this year (2018). We can question whether this is good idea; however, it is probably a good idea. Due to computational limits on my personal machine, this full dataset would take too long to process. The key component to this analysis is connecting this 311 data to the US Census' tract. The US Census provides an API where the tract number is returned given a street address, which we do have in the 311 data. The problem is the time it takes to make this call. Even with batch submission of addresses, it takes around a half second to process each. Given the 2.8 million rows, that would mean it would take over 16 entire days to complete the whole dataset! This limitation forces us to pare down the dataset to keep this time to more reasonable levels.

With this restriction in place, we can start slowly reducing the size of our dataset. First up, there are around 300,000 rows with no listed address: listed as "Not associated with a specific address." Along with the other handful of missing addresses, these rows are dropped from the dataset. The API also has strict rules on the input addresses: they must have street numbers. This means intersections are not accepted and thus we also have to drop rows that do not have any street number in the address. We also remove rows whose address appears to have more commas than we generally expect from a normal street address. There are certainly ways to parse this, but again since we are motivated to remove rows, this is another negative filter for the data.

We can turn our attention to the actual types of requests in this dataset. The top sixteen categories cover nearly 90% of the entire dataset. Again, because we want to reduce the dataset, we can keep only the top 8. These eight category types represent about 74% of the dataset. The list below details these eight categories.

- Street and Sidewalk Cleaning
- Graffiti
- Abandoned Vehicle
- Homeless Concerns
- Encampments
- Sewer Issues
- Streetlights
- Damaged Property

Note that some of these are related, but are categorized separately. We will also analyze these correlations in full, later in this document. With this choice, we get to keep much of the dataset while still dropping rows.

Then we can make sure all rows in the dataset are actually in San Francisco. After a visual check, this is not the case. As mentioned earlier, we can get a shapefile containing the outline of the San Francisco.

Using this GeoJSON file, we can cut out all rows outside the city limits. After performing this step, we can plot where the requests have been made. This dot graph is shown below in Fig. 1.

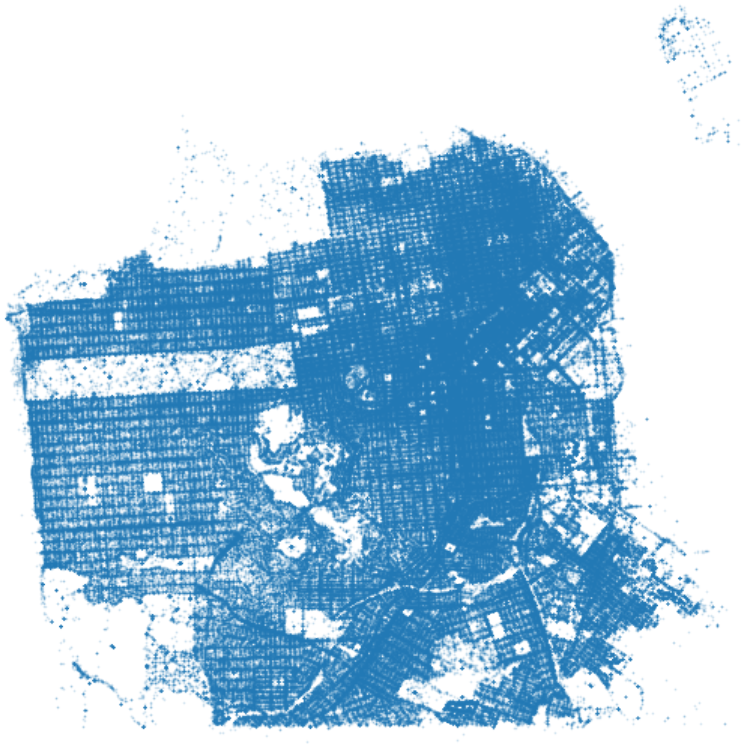

Figure 1: 311 service requests in San Francisco

This graphic gives us a nice outline of the city, including Golden Gate Park (the sparse bar in the upper left). Lastly, we look at the time component of the dataset again. It represents nine full years, from 2009 to 2017. By keeping only the last three years (2015 to 2017), we reduce the dataset to about 550,000. Three years of data should be enough to be able to capture possible seasonality and fit the data well.

Columns

At the end of this process we have many variables or columns in our dataset. Many are not needed or otherwise are not within the scope of the analysis. Such columns include status notes, request details, and responsible agency. It should be noted that there may be textual information in these columns that might yield predictive information. However, this type of feature engineering is outside the goal of this project so these columns, amongst several others, are dropped from the dataset. The table below, Tab. 1, lists all kept columns with descriptions and any related notes.

Table 1: Full list of features in the cleaned dataset

Name	Description	Notes
Date	Date of the service request	-
Category	Service request type	Eight categories
Supervisor_District	SF Supervisor District	Twelve categories
Police_District	SF Police District	Ten categories
Latitude	Latitude of service request	-
Longitude	Longitude of service request	-
Source	Ways to make service request	Seven categories
Has_Media_Link	If media was included in service request	Boolean
Is_Closed	If service request is now closed	Boolean
Tract	Given US Census Tract	199 categories
Poverty	Poverty ratio in percent	Range from 0 to 100
Income	Median income in dollars	Range from 19,440 to 250,000
Unemployment	Unemployment ratio in percent	Range from 0 to 100
Population	Total population	Range from 89 to 13,057
Percent_Male	Percentage of male population	Range from 0 to 100
Percent_Pop_0_14	Ratio of population between 0 and 14 years old	Range from 0 to 1
Percent_Pop_15_24	Ratio of population between 15 and 24 years old	Range from 0 to 1
Percent_Pop_25_64	Ratio of population between 25 and 65 years old	Range from 0 to 1
Percent_Pop_65_up	Ratio of population over 65 years old	Range from 0 to 1
Percent_Pop_White	Ratio of population identifying as white	Range from 0 to 1
Percent_Pop_Black	Ratio of population identifying as black	Range from 0 to 1
Percent_Pop_Hispanic	Ratio of population identifying as hispanic	Range from 0 to 1
Percent_Pop_Asian	Ratio of population identifying as asian	Range from 0 to 1
Percent_Pop_Other	Ratio of population identifying as other	Range from 0 to 1

The full code detailing these data wrangling steps can be found on my GitHub. A link to this *JupyterNotebook* is given here.

Data Exploration

Initial and exploratory data analysis of the cleaned dataset is performed using various visuals and inferential statistics. The purpose is to gain some insight into how the time series component of the data is distributed and influenced by the various service request categories.

Visual Analysis

We can start exploring the data by looking at its time component. As we previously did, there are three years worth of service requests. How many requests are in each year? This is answered in Tab. 2 below.

Table 2: Request count per year

Year	2015	2016	2017
Count	131,522	186,867	227,737

The increasing trend is pretty clear. We can try to see what the dependence is on the month, regardless of the year. Fig. 2 shows this below.

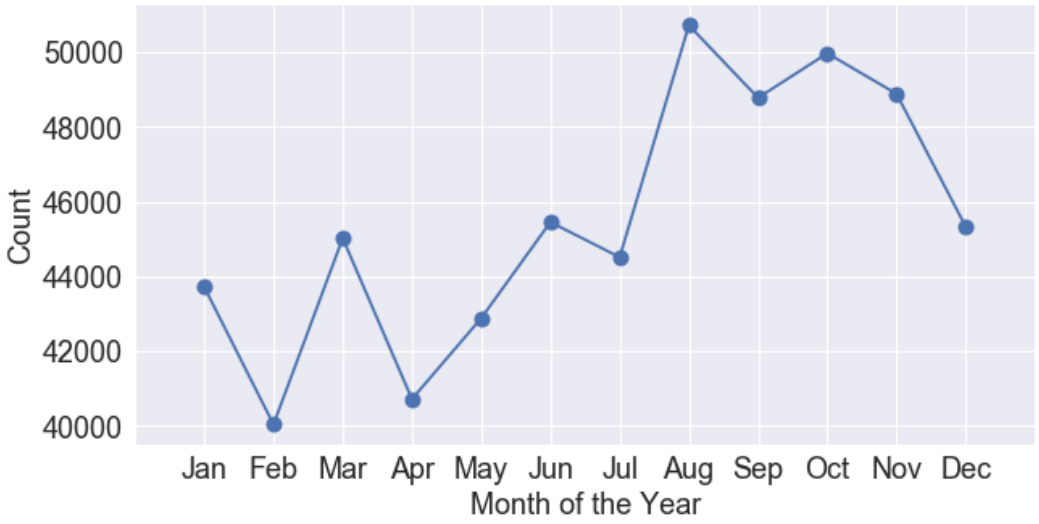

Figure 2: Request count per month

It is interesting to see that there is quite a bit of variation in any particular month. It is clear the back half of the year gets more requests, though it is not immediately clear why this is true. Perhaps there are PR campaigns or other advertisement for the 311 service during that time frame? At this point we can only speculate, though it is an intriguing result. We can also make a similar plot, in Fig. 3, for the days of the week to see how service requests fare across the seven days.

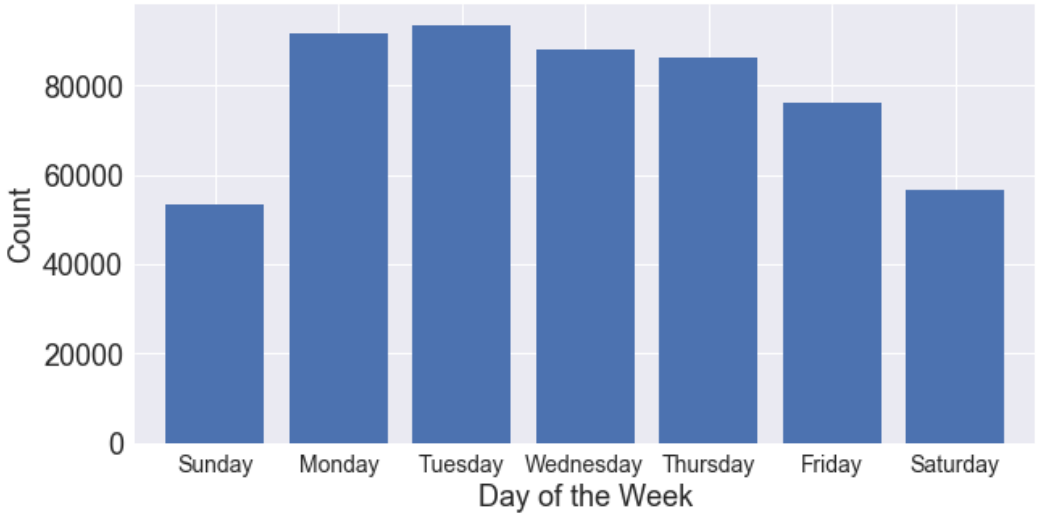

Figure 3: Request count per day of the week

This distribution also seems to make sense. There are less requests on the weekend, but is pretty constant on both weekends and week days, respectively.

Next, we can see how the different categories are distributed throughout the three years in the data. Here we up-sample the service requests to the order of weeks (from seconds) and count the requests. In

other words, we resample per week and count the requests for each category during that time frame. Then we can plot this weekly count data as a time series for the eight categories. This plot is shown below in Fig. 4.

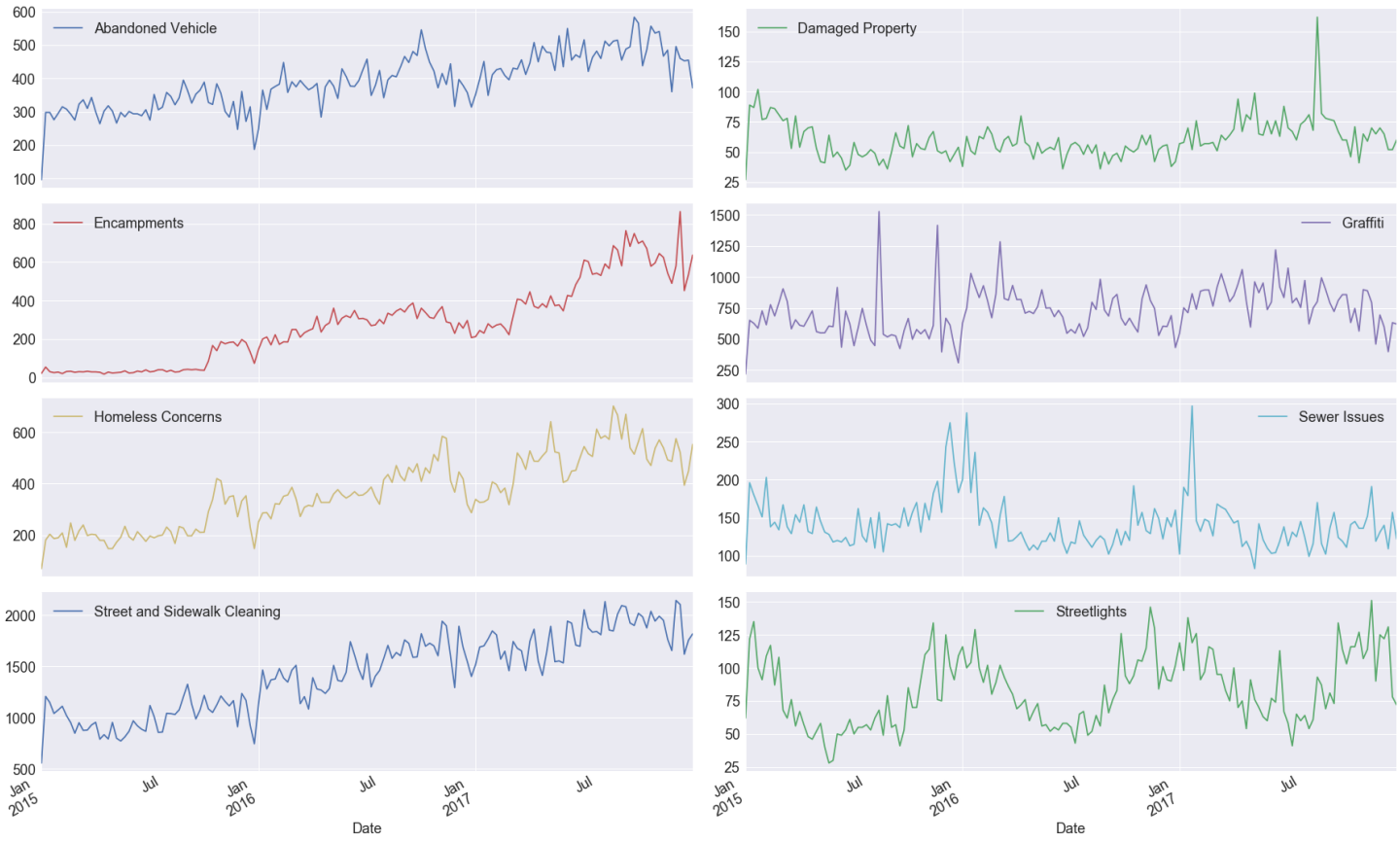

Figure 4: Weekly request count for each category

We can see how the eight categories have had requests over the past few years. Its interesting to see the large uptick in requests about Encampments starting in late 2015. We can presume that coupled with the overall upward trend in Homeless Concerns, San Francisco is actively looking into this problem. There is also an obvious seasonal pattern in Streetlight requests, which we perhaps should have guessed beforehand.

Statistical Inference

We can also look at how the eight categories are related to each other. As we have already surmised, these categories are potentially related. Some of these are obvious like Abandoned Vehicle and Street and Sidewalk Cleaning. How much are they related though and can we quantify it? We can make a heatmap showing the Pearson correlation. This triangular heatmap is shown below, in the top panel of Fig. 5. This plot does indeed some large correlations. How significant are they though? We can calculate the p-values for the correlations and plot the results, in the bottom panel of Fig. 5, in the same way (i.e. a triangular heat map).

By chance, any one of these comparisons can be very small (i.e. $p < 0.05$) potentially indicating significance where there might not actually be any. This is known as the multiple comparisons problem

or the look-elsewhere effect in particle physics. This issue cannot be ignored. The simplest, and most conservative, technique to deal with this problem is to divide the threshold, or α level, by the number of comparisons made. In this case there are 28 comparisons ($\frac{8*7}{2}$), so we will need to adjust with this number in mind. For visual purposes, we will actually multiply the p-values by 28, so that we can keep the usual α =0.05 threshold. The effect is the exact same though. In this plot the z-axis (or color bar) has an upper limit of 0.05 to show which correlations might be significant at the α=0.05 level. Note also that a lighter color indicates a small (adjusted) p-value, suggesting real significance of the correlation.

Figure 5: Correlations between weekly request counts for all eight categories. The top panel shows the Pearson correlation coefficient while the bottom panel shows the associated p-value.

Even with our correction, we can see that some p-values are (very) small, indicating real significance and correlation between categories. In fact, over half show significance. This makes some sense as many of these categories do share similarities in the underlying issue. For example, we can probably expect real correlation between Encampments and Homeless Concerns.

The full code detailing these data exploration steps can be found on my GitHub. A link to this *JupyterNotebook* is given here.

Conclusion

We were able to combine and clean different data sources to create a single cohesive dataset summarizing the 311 service requests to the local government of San Francisco. Because of computational concerns, we had to reduce the size of the dataset from about 2.8 million records to about 550,000 records. We were then able to learn more about the service requests made to the local government of San Francisco. We summarized the time aspect of the data to understand the distribution of service requests across different time scales. It is clear there are some strong correlations between some of these request categories.

This entire project can be found on GitHub here. The code and scripts can be run completely out of the box as the repository includes all related code, data, and reports, including this document.