Capstone Project 2: Milestone Report 1

Problem Statement

When people require non-emergency municipal services, the number to call is 311. In New York, all the data about 311 calls received in New York city is available online. A potential problem that can be addressed is the number of late resolutions. For each call that gets placed, an estimated due date is made as to when the issue will be resolved. In this project, a model will be built that will predict if a new call that is placed will be resolved on time or not. This information would be useful to various New York public agencies in order to improve their performance and address the needs of the people in a more timely manner.

In this case, the client would be NYC 311. What they could do with the information learned from the project is, they could provide better service to individuals living in areas where wait times are longer than other areas.

Data Wrangling

The dataset that was used is the “NYC Open Data 311 Service Requests from 2010 to Present”. The data was acquired by using the Socrata Open Data API (SODA) which provides programmatic access to the dataset. The original dataset contains 22.1 million rows and 41 columns. To make the project more manageable in python as well as relevant to today, a 10,000 row subset of the data from January 1, 2019 to the present was taken. Because subsets of the data were taken as well as the feed being

updated daily, different versions of the dataset were saved based on the date of access. In the future, the models built on the different days of access could have their performance evaluated and compared.

Once the dataset was read into python, cleaning and transformation of the data was required to prepare the dataset for analysis. With 45 columns in the original dataset, we need to cut down the number of features the model will be built upon. First step, many columns containing redundant information were dropped. For instance, longitude and latitude describes the location of the incident, but so do the following columns: intersection 1, intersection 2, cross street 1, cross street 2, location zip, location address, location, bridge highway name, road ramp, bridge highway direction, bridge highway segment, x coordinate, y coordinate, landmark, street name, incident zip, and incident address. Next, rows containing null values were dropped shrinking out dataset from 50,000 to 16,586.

A new feature called ‘late’ was also created by comparing the”closed_date” of a row with its estimated “due_date”. For those rows where the closed date was greater than the due date, a label of “True” was applied. For those rows that failed the argument above, a “False” label was applied. These labels were then multiplied by “1” in order to get boolean values of “0” or “1”, so that we could use the late feature as the target variable. It was found that approximately 16.36% of all the rows in the dataset were labeled as ‘late’.

Continuing to clean the data, I decided that the day of the week the call was placed on could contain interesting information, so the variable “created_date” was converted to datetime and the day of the week extracted.

In previous version of the project, after converting categorical variables to dummy variables through one hot encoding, a total of 850 columns were produced. As this was too many features to make any conclusions off of, a mapping dictionary was created that transformed 67 different “complaint_types” into 8 groups. Those groups being: water, plants, animals, cleaning, human complaint, maintenance, noise, and traffic. The same method was applied to the “location_types” feature where 53 unique location types was converted into 3 groups of business, residential, and public.

Finally, since the variable late is a combination of the features “closed_date” and “due_date”, both those features were dropped from the dataframe. Thus the final variables selected to go into the predictive model were:

Agency: which New York City agency was assigned to take care of the incident

Borough: which New York City borough the incident took place in

Latitude: latitude of the address where the incident took place

Longitude: longitude of the address where the incident took place

Open_data_channel_type: how the incident was reported

Late: whether or not the incident was resolved on time or not

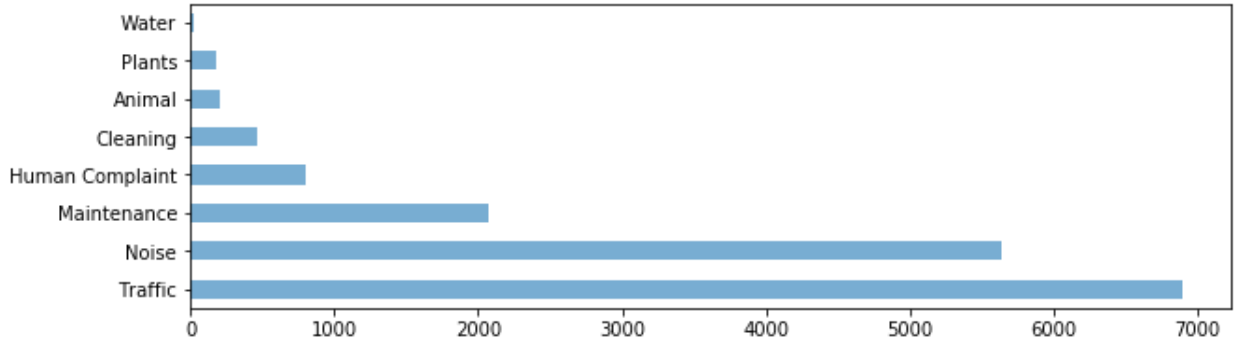
Created_day_of_week: what day of the week the incident was reported

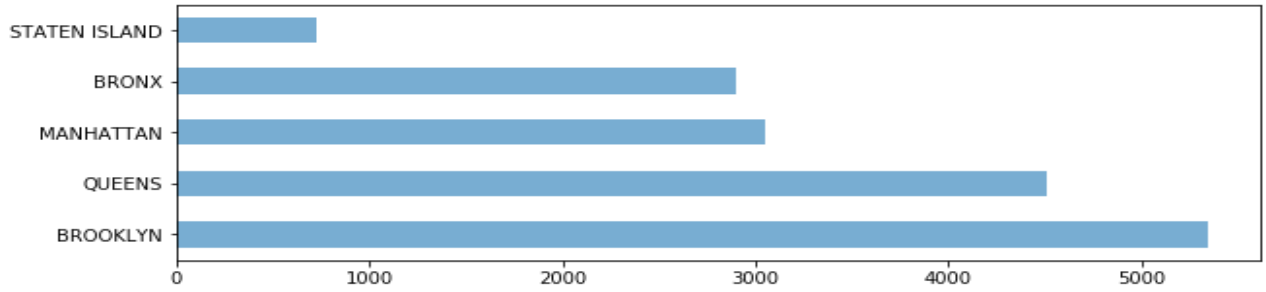
Complaint Grouping: the 8 different types of complaints

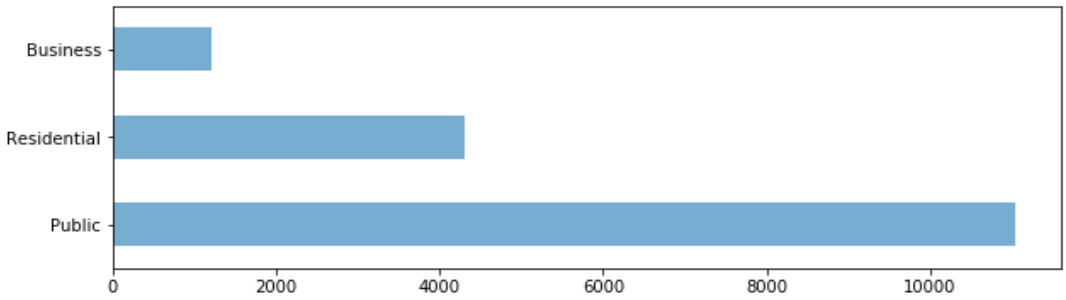
Location_Grouping: the 3 different types of location groups

Exploratory Analysis

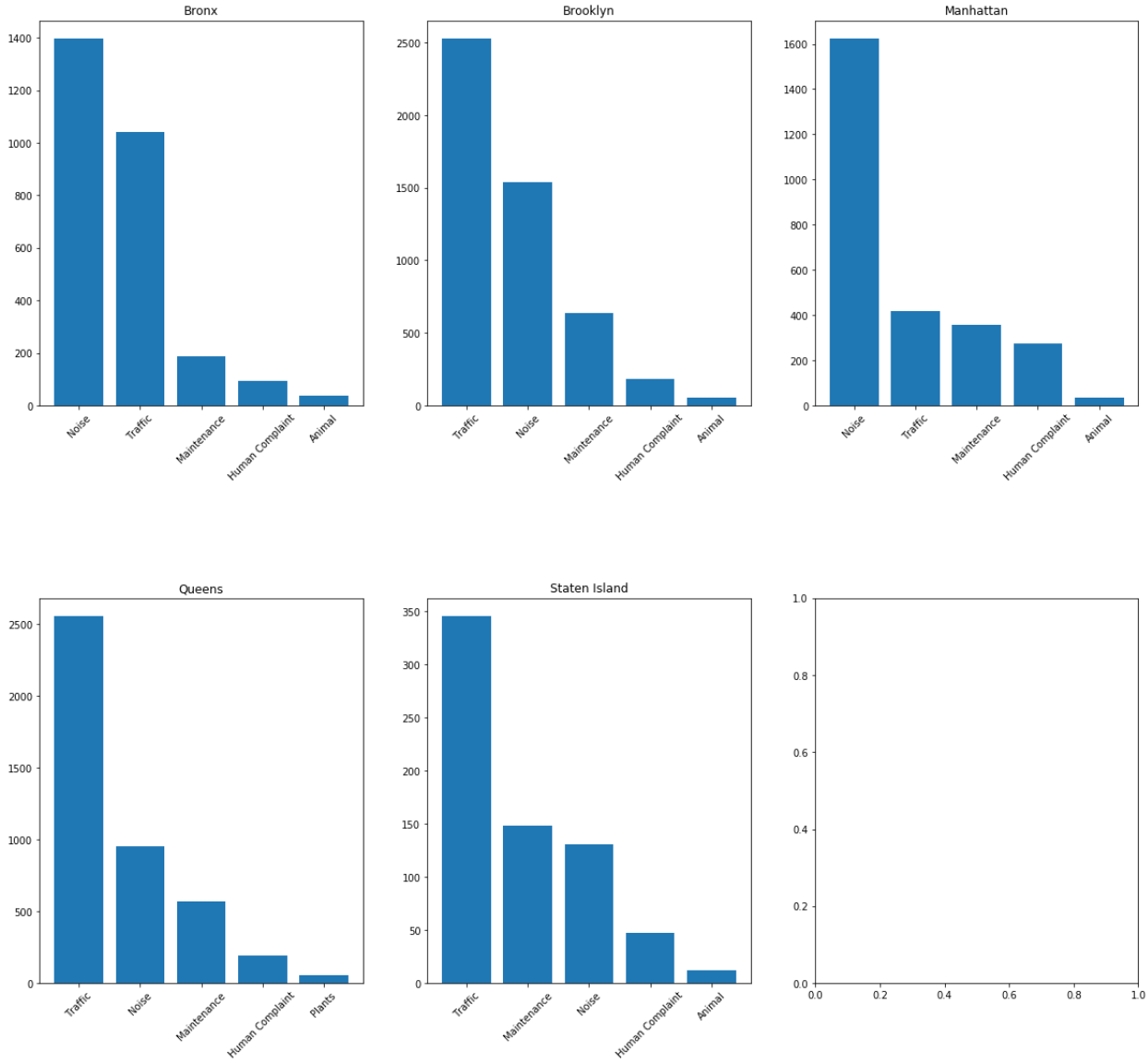
With data preprocessing finished, exploratory analysis could be conducted to see how our data looked. After creating bar charts of each of the features, it was seen that there were some unwanted labels for some of the columns. For instance, there were six unique boroughs in the “boroughs” column of the dataset, when there are only five New York City boroughs. Thus, all rows where the borough was labeled as unspecified were discarded. After this was done,a few of the bar charts appeared as follows:



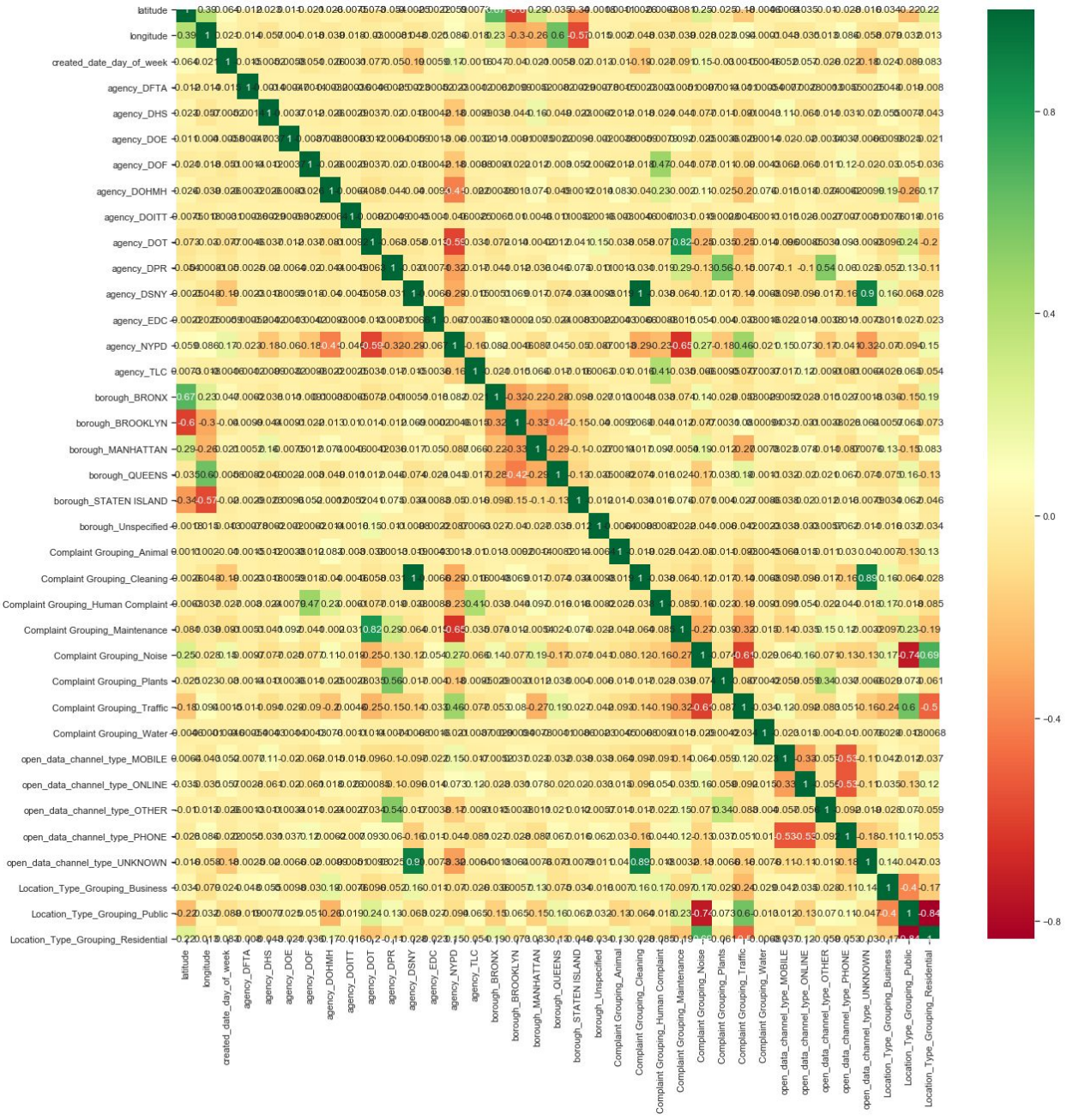




Bar charts were also created for the complaint types per borough.



Dummy variables were then created to allow the use of machine learning algorithms from sci-kit learn. The dummy variables created were then placed in a correlation matrix to show correlation coefficients between variables.



Although a few of the features seem to be correlated, overall there is not much correlation between the features. Now the dataset is ready to be used for predictive modeling.