Jesse Noppe-Brandon(jn2934), Weijun Huang(wh2531)
Yiyang Zhou(yz10590), Mike Zaharchenko (mz2433)
Professor Rodriguez
CS-GY 6513
Due: 5/13/2025

Predicting Fire Risk in NYC Based on Heating Complaints

Introduction

In this project, our group attempted to predict fire risk in New York City based on historical fire incidents and housing maintenance code complaint data.

Our hypothesis was that buildings with a high volume of inadequate heating complaints are more likely to experience fires in the near future as, when heating is insufficient, tenants may turn to unsafe alternatives like space heaters or ovens. The use of these methods can increase the risk of a fire incident taking place. Our objective was to create a data pipeline to efficiently process and analyze NYC Open Data's available datasets, train a machine learning model on them, segment New York City into grid-like regions using geolocation data, and create a heatmap representation of regions with the highest fire risk.

Process

For the data processing and preparation section of this project, we pulled the Fire Incident Dispatch Data dataset and the Housing Maintenance Code Complaints and Problems from NYC Open Data.

The former dataset contains information about incidents that the FDNY has responded to over a 24 year time frame (2001 to 2025). The information in this dataset is generated by the Starfire Computer-Aided Dispatch System that the FDNY uses when responding to 911 calls. It contains information such as the time and date the incident occurred, the nature of the incident, the borough, district, zip code, and intersection where it took place. In addition to this, the dataset contains incident response data, which was not deemed relevant to the scope of this project and thus filtered out.

The latter dataset contains information about complaints of conditions that violate the New York City Housing Maintenance Code (HMC) or the New York State Multiple Dwelling Law (MDL) made to the Department of Housing Preservation and Development (HPD). This dataset contains information pertaining to the date the complaint was received, the nature of the complaint, and the specific location where the violation took place.

In total, we processed and analyzed approximately 26 million records. Approximately 11 million of those were from the Fire Incident Dispatch Data dataset and around 15M were from the Housing Maintenance Code Complaints and Problems dataset.

To process this data, we used the following stack:

- Python: our base language
- Spark: to build the pipeline and process data concurrently
- SciKit-Learn/PySpark: to train our machine learning models
- Pandas: to clean, analyze, and visualize the data
- NYC Planning's Geoservice API: to geocode the intersection location

Fire Incident Dispatch Data (FIDD)

Within the fire incident dispatch dataset, the following steps were taken to clean and pre-process the dataset in Spark:

1. We first dropped rows that contained null values in the critical columns. We define a critical column as a column that is required to be populated for us to perform analysis. Such columns included:
    a. address_2 (data necessary to form an intersection)
2. We then filtered the dataset to only include entries within the ten-year date range 01/01/2015 to 04/30/2025
3. Afterwards, the dataset was filtered to only include incidents that were classified as dwelling fires. Rows included contained one of:
    a. Private Dwelling Fire
    b. Multiple Dwelling 'A' Fire
    c. Multiple Dwelling 'B' Fire
4. Address information was then normalized to the following format: {street 1, street 2}

All in all, this condensed the dataset from about 11.1M records and 3.12GB to approximately 130,000 records.

After preprocessing, we had to geocode the data to convert intersections into coordinates that could be used to fit the approximate incident locations into a grid. The Geoservice API takes intersection information and produces a set of coordinates where that intersection occurs. Since Fire Incident data is depersonalized, it does not provide exact coordinates or a house number- just an approximate location where the incident occurred. Though not completely precise, this type of data fit well with the grid scheme that we wanted to achieve in the final result, as a grid would encompass multiple such intersections. However, to attain usable coordinates from this, we had to build a pipeline that would use Spark to concurrently process the data and call the Geoservice API to transform intersection information to coordinates.

Initially, we had some big concurrency issues. We were pulling all of the rows into a single driver process memory, which completely bypassed distributed processing. In addition to this, we were making HTTP requests one by one and waiting for each request to return before moving on to the next row. This blocked execution and greatly limited how quickly we could geocode the given data. To improve processing, we implemented several optimizations:

Instead of processing rows sequentially, we used mapPartitions to enable concurrent processing by each partition.

This allowed us to cut down our processing time from approximately 6 hours for the entire Fire Incident Dispatch Data dataset to approximately an hour and 45 minutes- an improvement of 70.83%.

However, we knew that by introducing asynchronous code and improving parallel processing, we could speed the process up further.

To accomplish this, we created 25 Spark partitions to handle making API requests, allowing parallel execution. Within each partition, we configured up to 150 concurrent TCP connections per host. We also introduced a semaphore to cap concurrent requests at 150 to ensure stability and prevent server overload.

As a result of these changes, the geocoding time dropped dramatically, from approximately an hour and 45 minutes to just 38 minutes. This was another improvement of about 71.43%.

In total, our optimization steps allowed us to improve total processing and geocoding time for the Fire Incident Dispatch Data dataset by about 89.5% from our first version of the algorithm.

Housing Maintenance Code Complaints and Problems (HMCCP)

Within the house heat complaint dataset, the following steps were taken to clean and pre-process the dataset in Spark:

1. We first dropped rows that contained null values in the critical columns. Such columns included:
    a. Longitude
    b. Latitude
2. We then filtered the dataset to only include entries within the ten-year date range 01/01/2015 to 04/30/2025- the same as the range we used for the Fire Incident Dispatch Data dataset
3. Afterwards, the dataset was filtered to only include complaints related to heat. Namely, only the following codes were included:
    a. NO HEAT AND NO HOT WATER
    b. NO HEAT.
4. In the end, we filtered out summer data due to insufficient data.

Joining FIDD and HMCCP

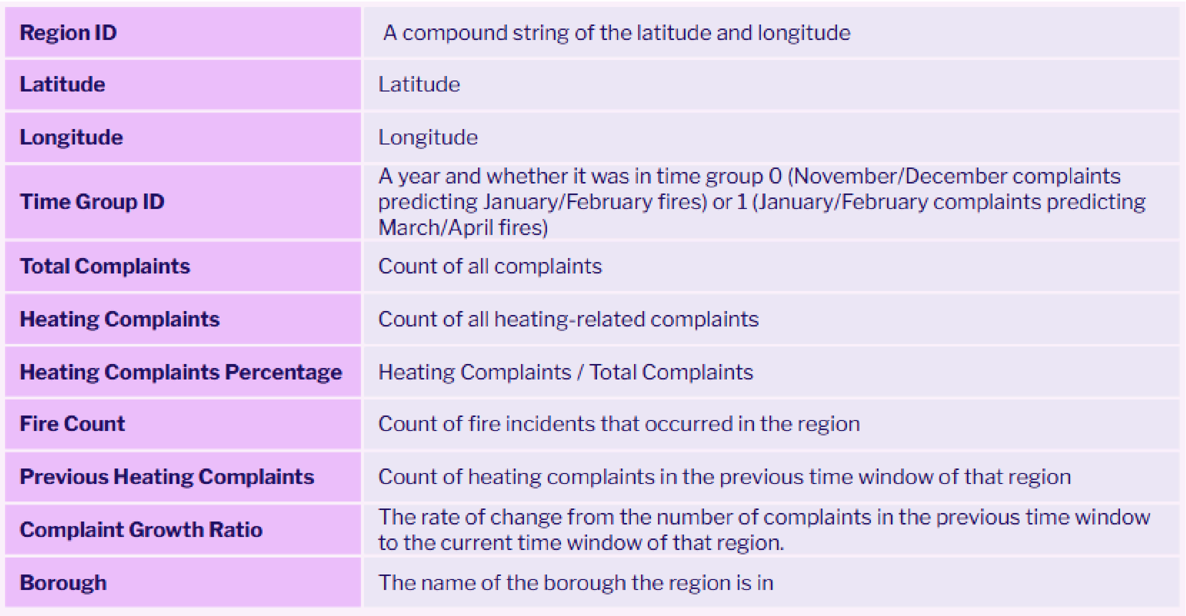
After pre-processing, we joined the fire and complaint rows based on time and regions.

Time: we matched complaints in a two-month window to fires in the next two-month window. Focusing only on the winter months, our windows were November/December complaints predicting January/February fires, and January/February complaints predicting March/April fires. We spoke about the idea of different length windows, but since many of our regions were already very sparse in terms of fires, we worried that smaller windows would leave too many rows with no fires, so we decided to stick with two-month windows.

Regions: New York (latitude 40.490 to 40.920 and longitude -74.270 to -73.680) is divided into small grids with a resolution of 0.005 degrees.

As mentioned before, the idea was to split New York into small, granular grids that incidents and complaints could be grouped into, then allow the machine learning model to predict the outcomes for each specific grid.

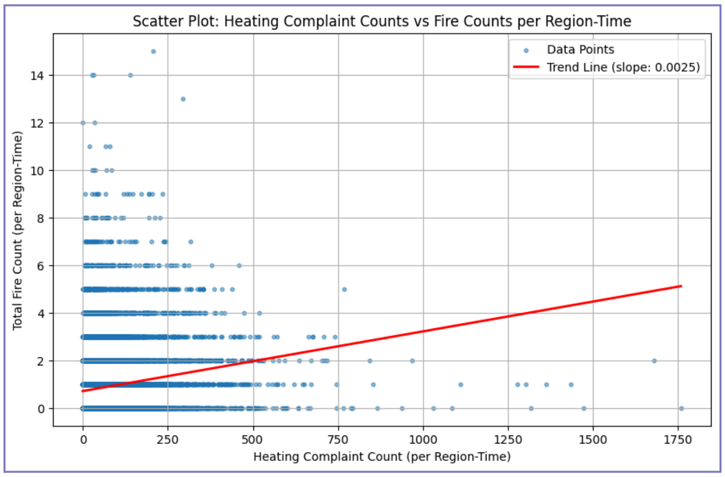
Below is a schema for the final output. This result is used in machine learning later. The motivation is to see if it is possible to predict fire risk within one region based on complaints from previous months.



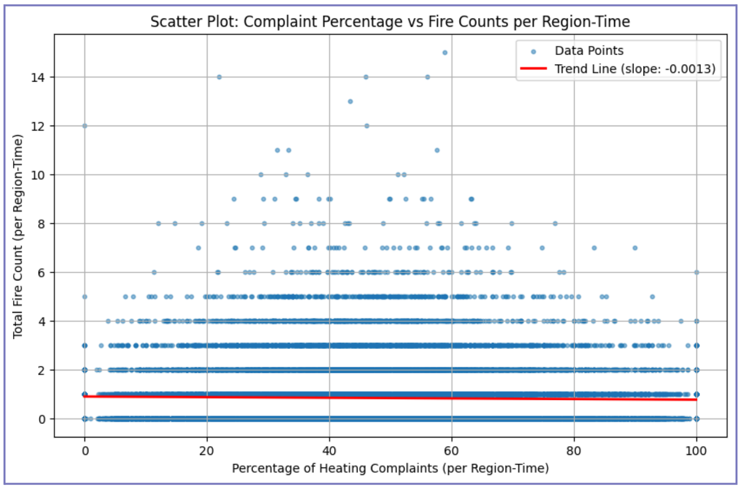
Region ID	A compound string of the latitude and longitude
Latitude	Latitude
Longitude	Longitude
Time Group ID	A year and whether it was in time group 0 (November/December complaints predicting January/February fires) or 1 (January/February complaints predicting March/April fires)
Total Complaints	Count of all complaints
Heating Complaints	Count of all heating-related complaints
Heating Complaints Percentage	Heating Complaints / Total Complaints
Fire Count	Count of fire incidents that occurred in the region
Previous Heating Complaints	Count of heating complaints in the previous time window of that region
Complaint Growth Ratio	The rate of change from the number of complaints in the previous time window to the current time window of that region.
Borough	The name of the borough the region is in

Visualizations

In order to test our hypothesis, we visualized the data in a couple of ways. We started with a scatter plot that showed “Number of Heating Complaints vs. Number of Fires” in each region/time window. The data ended up pretty clustered in the lower-left corner. We plotted a linear regression trend line, but only had a slope of 0.0025. With an R-squared value of 0.0256, we were shown that heating complaints alone are not a strong predictor of fires.



We worried that denser regions like those in Manhattan and Brooklyn might have more complaints but a similar number of fires, so we decided to also check “Percentage of Heating Complaints VS Number of Fires”. We assumed that this percentage might be a more normalized count, where if a larger percentage of the complaints were heating complaints, it meant there were more in that region/time. We plotted this as well, but the results were heavily clustered and showed no meaningful trend with fire occurrences.



Finally, to test our hypothesis, we turned our hypothesis into conditional statements where “High_C->High_F” represented our hypothesis and “High_C->Low_F” represented our null hypothesis. For every row, we looked at the number of heating complaints, and if that number was above the 75th quartile for all rows, we marked High_C as true in a new column (false otherwise). For High_F, if the number of fires in that row was greater than or equal to 1, then we marked High_F as true in a new column (false otherwise). Each row then had a final column added, which represented which quadrant it was in of the four possible combinations.

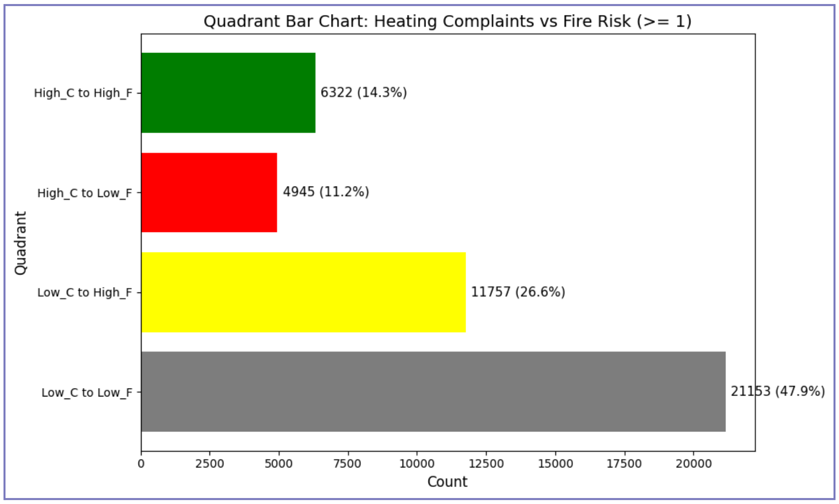
We then plotted a quadrant bar chart, which counted the number of occurrences for each of the four quadrants among all rows. The results showed a very similar amount between our hypothesis and our null hypothesis, with most landing in “Low_C->Low_F” which we considered an unremarkable baseline.

Green = Hypothesis

Red = Null hypothesis

Yellow = Unexpected cases

Gray = Baseline low-risk



We also played around with different thresholds for High_F, such as setting it to 2, but this just further decreased the green and yellow lines, since most rows had 0 or 1 fires. This also reinforced the conclusion that our hypothesis was not statistically significant.

Machine Learning

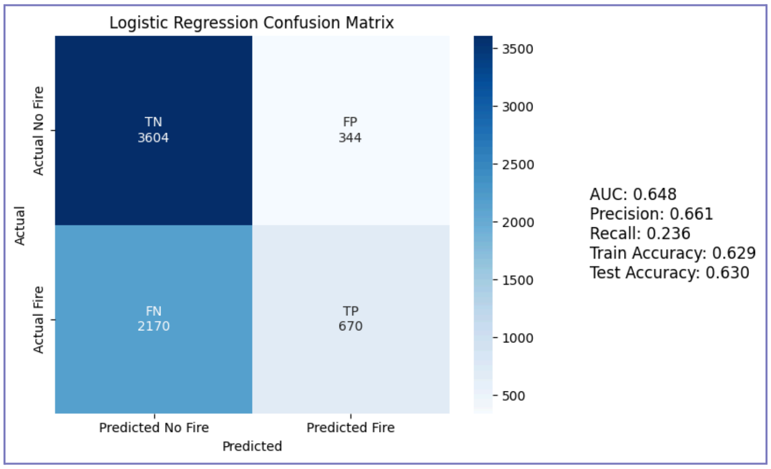
Following our conclusion from our visualization, we decided to test our hypothesis a little further with some machine learning models. Our goal was to predict whether a fire would occur in a given region during the next 2-month period. To do this, we used six features:

- Total complaints
- Heating complaints
- Percentage of heating complaints
- Previous period’s heating complaints
- Growth ratio in heating complaints
- Borough (one-hot encoded)

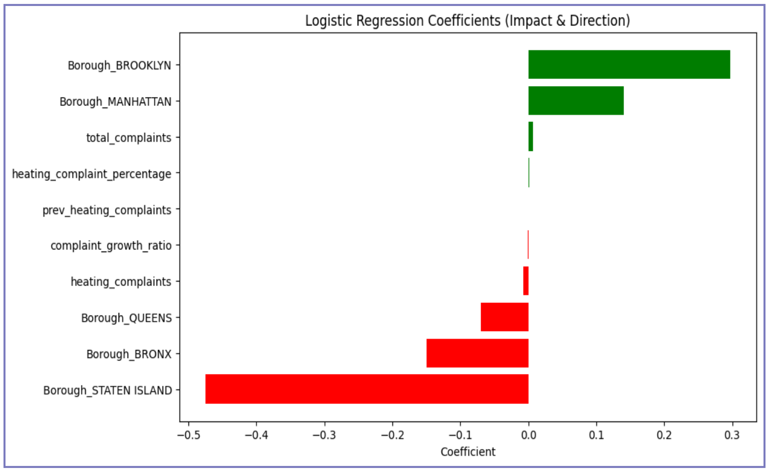
We added total complaints, previous period’s heating complaints, and growth ratio in heating complaints, to add some volume and change over time-related features we hadn’t explored in the visualizations. We added Borough to add some location-related information to see if it could give the model some information we hadn’t explored yet. Our label column marked whether or not a fire occurred, which was true if there had been at least one fire, and false otherwise.

We settled on two models: Logistic Regression and Random Forest. We chose Logistic Regression for easy interpretation and to be able to see the positive and negative effects of each of the features. We chose Random Forest to be able to find some of the non-linear relationships that Logistic Regression didn’t pick up. Despite only having around 40,000 rows, all of the machine learning was done in PySpark to allow for scalability.

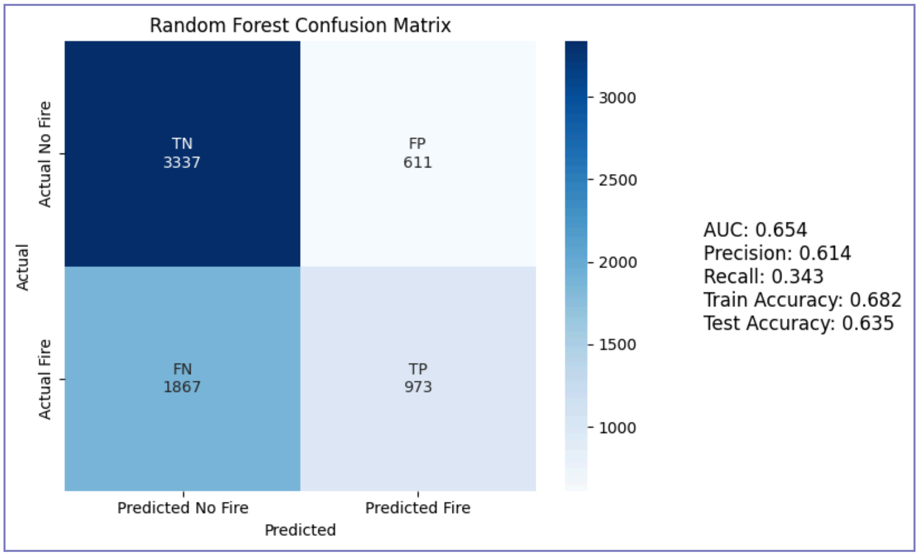
Our Logistic Regression model was only able to achieve a test accuracy of 63% and an AUC of 0.648, reinforcing that our features lacked strong predictive power. It's very low recall (23.6%), indicated it missed many fire events — a serious issue for a risk prediction task. Its precision score was decent, but that is not surprising when having a fire is way less common than not



We looked at the coefficients and found that borough played a strong role in the model's prediction. Denser boroughs increased the probability of the prediction, which makes sense. Interestingly, heating complaints had a slightly negative coefficient, meaning they were associated with a lower likelihood of fire, while total complaints showed a modest positive effect. We tested a model without borough as well, and the results for the 5 remaining categories were similar.

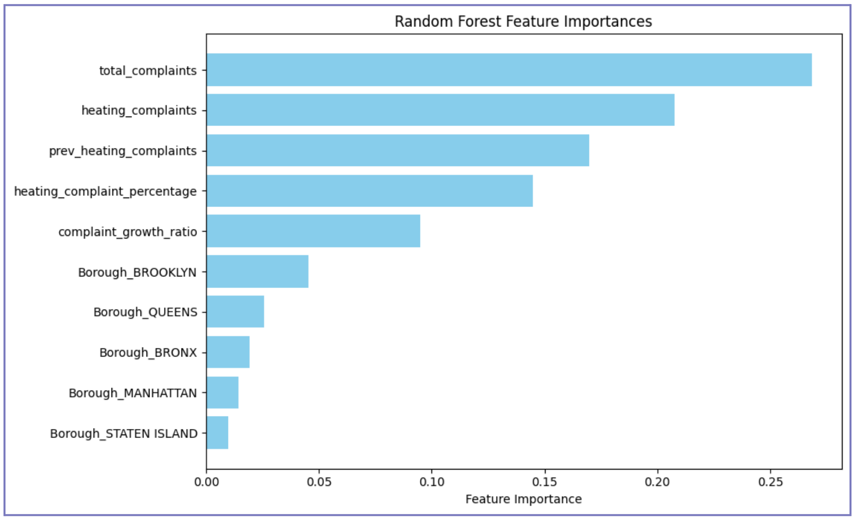


Our Random Forest Model achieved a test accuracy and AUC nearly identical to the Logistic Regression model. Its recall (34.3%) was slightly higher than Logistic Regression’s (23.6%), but still too low for effective risk detection. Its precision score was very similar as well.



AUC: 0.654
Precision: 0.614
Recall: 0.343
Train Accuracy: 0.682
Test Accuracy: 0.635

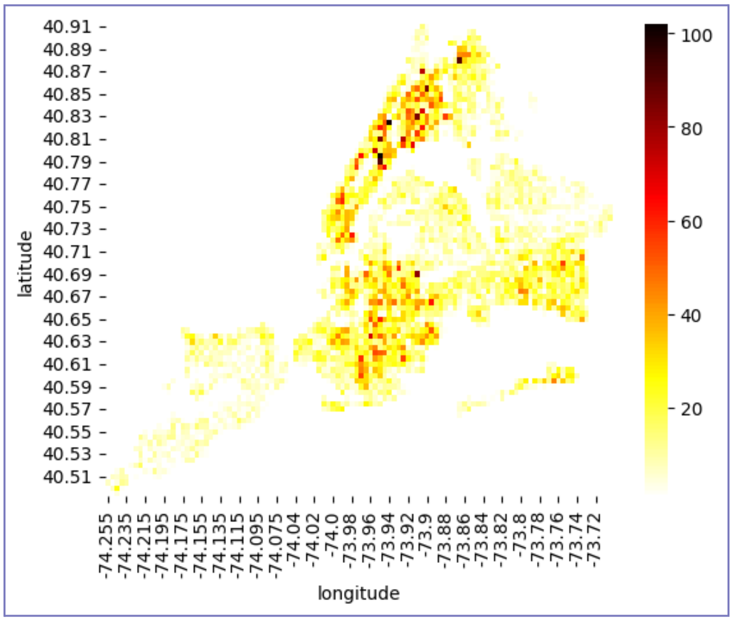
Looking at the feature importances for the model, we saw very different results from the Logistic Regression coefficients. All feature importances are positive, but Borough features played a much smaller role here, likely due to how Random Forest handles one-hot encoded categorical data. Here, total complaints were also the highest predictor of the numerical categories, though heating complaints increased to second place. Overall, both models performed similarly, though Random Forest placed greater emphasis on complaint volume over geographic location. Neither model was able to use the features to become an accurate predictor of future fires.



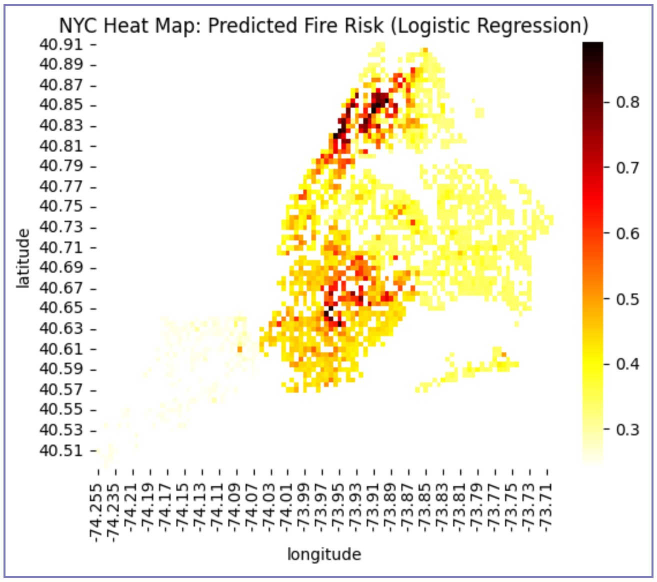
Heat Maps

To further demonstrate our result, we plotted 3 heatmaps.

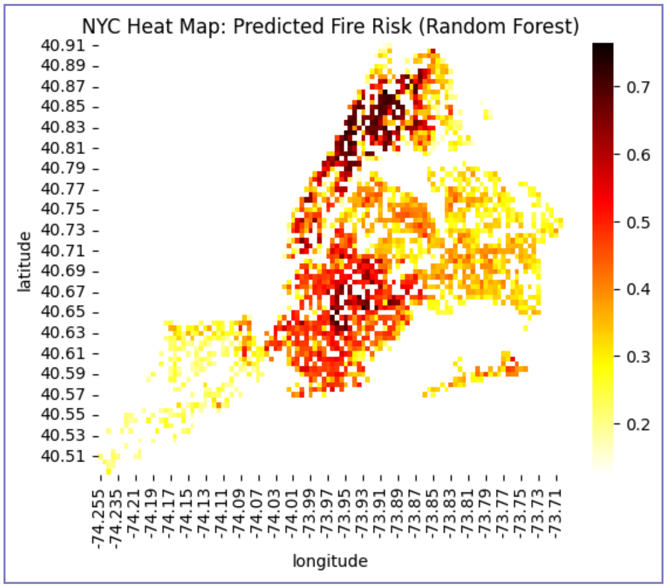
Our first heatmap plotted the number of fire incidents from the last 10 years aggregated over all time windows for each region. We started with this since all of our tests had shown our hypothesis wasn't correct. This would just be a general show of where the most fires had occurred. Manhattan and Brooklyn have the most concentrated amount of fires, similar to what our logistic regression model showed for coefficients.



Our second heat map plotted the fire risk based on the prediction of the logistic regression model. The model predicted a 0 or 1 for each column, but we also added the probabilities before it took the greater of the two. This heatmap shows the mean of all of the predictions for each region over all its time windows.



The final heat map plotted the fire risk based on the prediction of the random forest model. It plots the same probability feature that we used with the logistic regression model. It has much higher concentrations of high probability areas that match less with the fire count map.



All three of these maps are somewhat similar, and it seems that Manhattan and Brooklyn are much more vulnerable than other regions, which makes sense considering their higher population density. In contrast, Staten Island is relatively safe from fire incidents.

Conclusion

Overall, all of our data exploration showed that the number of Heating Complaints is inadequate to predict fires by itself. In future work, we would try adding more features such as demographic information and building data to see if they could increase our predictive power. This could still support our original goal of identifying areas at greater fire risk due to neglect.

Interestingly, the heat maps created from the models showed a somewhat similar pattern to the fire count heat map. The models didn’t receive the latitude or longitude, so it seems like there was some predictive power, despite all of our other visualizations showing that there wasn’t. However, this mostly correlated with denser areas like Manhattan and Brooklyn, which also had more fires overall, so it seems to affirm that while heating complaints are not a primary indicator of whether or not there will be a fire, they might be a small piece of the puzzle.

If more features led to more robust predictions, this project could be extended to a streaming pipeline. Housing Complaint and Fire Incident data is updated regularly (new data arrived while we were working on this project), and a pipeline could continually process this incoming data and use it to improve predictions. Our processing pipeline and machine learning are already Spark optimized, so if a streaming pipeline were set up, our system could easily be extended.

While our hypothesis didn’t hold, heating complaints still seem to be a small but relevant piece of the puzzle. This landlord neglect affects tenants’ safety and well-being, and whether or not it directly causes fires, better care of buildings and tenants leads to a safer city.