Flashlight: Property Assessment Visualization for the City of Boston

Tyler Brown, Sicheng Hao, Nischal Mahaveer Chand, Sumedh Sankhe

Summary

As a new home-buyer, it's easy to find out about your home but hard to get an understanding of your neighborhood. Flashlight makes it easier for you to see your potential neighborhood in Boston. This discrepancy is because current real estate websites emphasize individual properties rather than individual neighborhoods. Our group communicates the differences between Boston neighborhoods using an interactive data visualization called Flashlight.

Our dataset includes Property Assessment history from 2014-2017 [**?**] using Boston's Open Data Hub. We enriched the property assessment data with coordinates from Open Addresses [**?**], and neighborhood boundaries from Zillow [**?**]. These combined datasets provide unique insights to new home-buyers in Boston. As open data becomes more prevalent in cities across the United States, we can scale our insights and models.

Methods

We used methods for collecting, preparing, modeling, and presenting our data. Each step of the process is detailed here.

0.1 Data Collection

We started with Boston's Property Assessment data from 2014-2017 [**?**]. This dataset "[g]ives property, or parcel, ownership together with value information, which ensures fair assessment of Boston taxable and non-taxable property of all types and classifications."[**?**]. We wanted to use this information because it helps us capture changes in Boston properties over time. For example, a remodeled property would change it's property tax assessment value we have this variable available to us.

After starting with the Property Assessment dataset, we brought in additional datasets to increase the value of our data collection. Neighborhoods in Boston were not named or geographically demarcated in the Property Assessment dataset so we brought in Neighborhood

Boundaries from Zillow [?] to make this distinction. Additionally, geographic coordinates for each assessed property's address were occasionally not coded correctly or included at all for 2017 so we had to bring in those values using Open Addresses [?]. Once neighborhood names, boundaries, and missing coordinates were available, we were able to proceed to data preparation.

0.2 Data Preparation

There were a number of steps involved in data preparation.

0.2.1 Data Audits

The purpose of a data audit is to answer questions related to data quantity and quality. We started with our Property Assessment dataset by checking for the quantity of populated items within each variable. About 73% of variables were less than 70% populated. We were able to disregard these variables for our modeling purposes. Data quality checks are not as easily automated but value added.

Analyzing data quality helped us understand data problems up front such as having a non-unique primary key about 0.3% of the time, latitude and longitude were missing or corrupted about 35% of the time, and the ratio of unique addresses concatenated with coordinates to unique coordinates were about $3:1$. We also found that the Property Assessment data did not map to defined neighborhoods in Boston. Understanding these shortcomings with a data audit allowed our group to plan remediation steps early in our analysis.

0.2.2 Geocoding Missing Addresses

During the data audit we found about 35% of addresses within the Property Assessment data were not matched to any coordinate pair. Our team work to mitigate this deficiency by leveraging the a "free and open global address collection" called the OpenAddresses [?] project. There are several strategies one can take when Geocoding (mapping address strings to coordinate pairs) addresses. The simplest would be to use Google Maps Geocoding API [?].

The Google Maps Geocoding API has a free usage tier which maxes out at 2,500 requests per day [?]. Our data required geocoding an order of magnitude more coordinates so this approach was out of the question. We resolved the issue by creating 'address hashes' for each address in the OpenAddress and Property Assessment dataset with missing coordinates. An 'address hash' was a string concatenated of concatenated values for street number, street name, city, and zip code. We excluded unit number as a simplifying assumption because we expected all unit numbers to be at the same property. Once 'address hash' had been computed for both datasets we performed an inner join.

In some cases, this join generated a many-to-many relationship between addresses in both datasets due to our exclusion of unit numbers. We resolved this issue by grouping and taking the first coordinate pair. Subsequent iterations of our analysis can be improved by adding robustness to our geocoding procedure.

0.2.3 Working with GeoJSON and Python

To create the graphical interface for the map, we used Leaflet - a library to show interactive maps. The team looked at a number of ways to plot the boundaries of each region on the map, and decided that the best way would be to use GeoJSON - a format for encoding a variety of geographic data structures. We used Zillow's shapefiles to use as the base to extract region boundaries and convert them to GeoJSON. Unfortunately, R did not have a robust GeoJSON library and did not offer the features we had in mind. We switched to Python to convert the shapefile to GeoJSON files. We then realised that there was no way to map the original data to the region we had. To counter this, we mapped each point on the map to a region by checking if the latitude and longitude of a point were inside a perticular region polygon. The mean interior score of each region was used to fill the polygons, and multiple points were clustered in higher zoom levels to reduce overplotting. Finally, depending on the zoom level, either the regions or the points are shown.

0.3 Data Modeling

The purpose of our data modeling was to create a representation of how prices for properties would change in the next year, all though physical features of the property play an important role in the value of a property increase on property value largely depends on the market conditions, surrounding neighborhoods. Hence to take into account the missing features about market it was decided upon to create a simple linear model at first using previous evaluations to predict for the year 2017. A more important prediction would be to predict how the price per square feet changes per region, hence the same linear model applied to predict price per sqft. First the price per sqft was calculate for every PID, it was grouped by the region and the mean price per sqft was calculated for that particular region for all years. Next a linear regression model was created with 3 variables and 1 response variable, the variables included the valuations for the year 2014 through 2016 which predicted the evaluations for the year 2017. With an RMSE of 5.45/$sqftandaMAEof$4.477/sqft the model performed unexcitingly. Considering all the variables were linearly related to the response variable. The above model was applied to predict the price per sqft for the year 2018 taking into considerations the values from 2015 through 2017. Since there was no way to gauge the performance of this model with real data as of now it would be interesting to see the performance. An interesting problem would be to gauge how housing evaluations are done and how neighborhoods end up having such diverse scores even when they are next to each other.

0.4 Data Presentation

We use Shiny Dashboard to present our data. The dashboard has two parts, a interface to filter data and the map itself. The filters can be used to look at specific data points on the map. The map, rendered using Leaflet, shows the regions upon startup. Each region is filled using the mean interior score for that region. Hovering over any region brings up a small dialog showing statistics, like mean, max, and min interior scores, property count, and the projected value change for 2018 in price per square feet. Zooming in hides the polygon layer and show marker clusters and the ploygon boundaries. The clusters help with the overplotting problem. Each cluster shows the number of properties in the highlighted ploygon when hoving over the cluster. When a cluster is clicked, or the map is zommed, the clusters split to show a higher level of detail and exposing individual property markers. When a property marker is clicked, it brings up a popup with property details and statistics for that property.
Filtering will help users pin down on the type of properties to view on the map, and then the map will help users determine which neighborhood would most suit their needs.

Results

We had some results.

Discussion

Let's discuss what we did.

Statement of Contributions

Together everyone achieves more.

- **Tyler Brown:**
- **Sicheng Hao:**
- **Nischal Mahaveer Chand:**
- **Sumedh Sankhe:**

Appendices

Appendices here.