Ideal Location for Vending Machines in Boston Using Data Science

1. Introduction

1.1 Background

There are approximately 4.6 million vending machines in the United States, producing over $64 billion in profits in one year alone. Not only has the Vending Machine marketplace been steadily growing, but it's relatively easy to start your own. Some advantages to starting a vending machine business, aside from profitability, are low setup costs, flexibility in location, and less risk with cash only vending machines (no accounts receivable).

1.2 Problem

Once you decide to invest in a vending machine business opportunity, the most critical factor in whether your business will be lucrative is the location of your vending machines. According to the Vending Group, the most popular places to install vending machines include apartment complexes, hotels, retail stores, and auto shops. Using Data Science, we can pinpoint locations that would be ideal for installing a vending machine. Let's say that you live in Massachusetts and want to install five vending machines in Boston, MA.

1.3 Interest

Anyone that wants to start a food related business, especially installing vending machines, would have interest. This project looks at locations that are ideal because there isn't easy accessibility to other alternatives for snacks and drinks. Installing vending machines in areas that already attract large numbers of people is a strategic way to increase revenue. Furthermore, this project looks into locations of high foot traffic and ones that lack businesses that sell snacks and drinks within walking distance.

2. Data Acquisition and Cleaning

2.1 Data Sources

The data related to venues is exclusively from the Foursquare API. While the Foursquare API is not the most comprehensive list, it breaks up Boston into neighborhoods that are consistent with the neighborhoods listed on another website I use for this project called Analyze Boston, a government-funded website. The longitude and latitude of each neighborhood, and Boston, MA are from latlong.net, a website recommended on NASA's website. The polygons illustrated in the choropleth map were downloaded from Analyze Boston, self-described as "City of Boston's open data hub" in GeoJSON file.

2.2 Data Cleaning

In order to analyze venues in close proximity to each other, I found that dividing the city by its neighborhoods was most effective. While researching Boston's neighborhoods, I found discrepancies in the number of neighborhoods, so I decided to use a government website. I

used Analyze Boston's CSV file for Boston neighborhoods, keeping the column for neighborhoods, and removing the other columns. Then, using latlong.net, I searched each neighborhood and listed the corresponding paired values (longitude and latitude) on an Excel Spreadsheet. After converting the spreadsheet to a csv file, I uploaded it on my GitHub repository. Using pandas for Python, a library used for data analysis, I created a get method which downloads the csv file into a pandas dataframe.

In another part of the project, after uploading the list of all venues for Boston, I had to remove the categories related to food. The first step was to drop all of the rows that had venues in food related categories. One issue I ran into was returning the column with the header “Venue Category”. When calling objects, there are no quotation marks, so I needed to remove the space character between the words, so “Venue Category” was renamed “VCategory”. The second issue was filtering out food-related venues. Since there were 707 different venues, and under different Foursquare API venue headers, the most efficient way to ensure each food-related venue was removed was to manually remove each one, by creating a function that would match the venue category with the text in column, and if it matched the text in my code, it would rewrite the DataFrameby removing that row only. While this process was tedious, I was certain that each venue was filtered out. Once the DataFrameno longer contained food-related venues, I dropped every column but the Neighborhood column.

3. Methodology

3.1 Exploratory Data Analysis

Due to the number of possible locations for vending machines, to be able to pinpoint specific locations, we should segment the city into sizeable areas. Fortunately, Boston is already divided into neighborhoods. The first step was to visualize the information I had available.

My next task in gathering data was to find the longitude and latitude of each Boston neighborhood. I first used Nominatim, from geopy which converts addresses to longitude and latitude values. Using latlong.net, a Nasa.gov recommended website, I searched each neighborhood and listed the corresponding paired values on my GitHub repository. (See link) Using pandas for Python, a library used for data analysis, I downloaded my csv file on GitHub into Python (Figure 1).

To visualize Boston's neighborhoods on a map, I used folium, a map rendering library, and created a map with blue markers on each neighborhood. While I now had a sense of where each neighborhood is located, I needed a way to figure out the most ideal area to install vending machines. Using Foursquare API, I was able to generate a list of venues locations near the location of Boston. First, I created the API request URL, made a get request, and created a method that returns only relevant information for each nearby venue. I also created a new pandas DataFramewhich returns the neighborhood for each venue; there were over 700 venues listed. Since I only wanted venues that include apartment complexes, hotels, retail stores, and auto shops, I needed to filter out restaurants, and other food venues. To visualize the data I just cleaned, I created a simple horizontal bar chart. As you can see from the bar

chart, Back Bay, Longwood Medical Area, and Downtown are heavily populated with these ideal venues (Figure 3).

Once I had a list of venues that fit the criteria for ideal venue categories, my next task was locating them on a map. To use the data, I utilized one hot encoding. Each instance of a venue creates a new row in the DataFrame, and produces a 1 if the instance is a part of a category, and produces 0 for each of the other columns in the row (Figure 4).

Since I only have five vending machines, and I want to maximize profits, I need to make sure that the area is a popular location. Since I know that there are other businesses nearby, I use Foursquare as a tool to indicate the top five venues for each Boston neighborhood. I then created a new DataFramethat has the top 10 venues for each neighborhood. (Figure 5)

3.2 Machine Learning

The next step is to cluster the data to plot on a map. In order find the optimal number of clusters, k, I use the Elbow method. As you can see from the elbow method, the optimal k is 4 clusters (Figure 6). I use k-means clustering because it is an unsupervised machine learning algorithm. Referring back to Figure 5, you can see there are repeating venue categories. K-means clustering uses an algorithm to create initial estimates for the k (4 in this project) centroids and iterates between solving for the Euclidean distance, as well as recomputing centroids by taking the average of the centroid's cluster. Since I am trying to find "hubs" of highly frequented areas, k-means clustering is the best suited for this machine learning,

I then made a new map that shows the clusters, using folium (Figure 7). There are four cluster, each in a different color (red, purple, blue, and yellow/green). When examining the clusters, we can see that cluster 2 contains the greatest number of instances. Clusters are formed from bases of my final map.

The most intriguing part of this map is that Folium uses meters as its units. Since there are approximately 400 meters in 0.25 miles (walking distance), I created a radius of each circle marker to be 400. So, if the vending machine is placed at the center of the circle, any person located within the shaded region of the circle will be within walking distance of the vending machine.
Even though, from Figure 7, we see the clusters, it's important to consider its surroundings. While the clusters are formed based on non-food venues, a vending machine located near numerous restaurants will have a reciprocal effect.

I used Analyze Boston's geoJSON file for the Boston neighborhoods, and imported it in folium. I created a new DataFramefor food-related venues only, in the same manner as non-food venues. (Figure 8).

4. Results

As you can see from the map, each neighborhood is a separate color. The darker the area, the more restaurants there are in that neighborhood. Ideally, we'd want to pick an area than is yellow or orange, and contains multiple circles, since that means those are areas with highly frequented venues than are not near many food locations. The next step would be to go to each location and find local businesses that would be willing to let you install a vending machine in or outside their store.

5. Conclusion

Using a variety of data science techniques including data visualization, data analysis, machine learning, and Python coding, I was able to pinpoint specific locations that would make for ideal vending machine locations.

Sources:

- https://www.thebalancesmb.com/starting-a-vending-machine-business-4138408
- https://blog.vendinggroup.com/6-best-places-to-install-vending-machines
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3377942/

Figure 1.

	Borough	Neighborhood	Latitude	Longitude
0	Boston	Jamaica Plain	42.310871	-71.125061
1	Boston	Leather District	42.347960	-71.056410
2	Boston	Back Bay	42.350266	-71.080978
3	Boston	Bay Village	42.350150	-71.065190
4	Boston	Downtown	42.355300	-71.055280
5	Boston	Roxbury	42.317982	-71.158508
6	Boston	Fenway	42.332670	-71.097910
7	Boston	Chinatown	42.347960	-71.056410
8	Boston	West Roxbury	42.278870	-71.159390
9	Boston	Beacon Hill	42.360291	-71.068680
10	Boston	Roslindale	42.317982	-71.158508
11	Boston	North End	42.365528	-71.060883
12	Boston	East Boston	42.389130	-71.007050
13	Boston	Brighton	42.317980	-71.158510
14	Boston	Mission Hill	42.334000	-71.097908
15	Boston	Dorchester	42.299780	-71.078840
16	Boston	Mattapan	42.252159	-71.124947
17	Boston	Longwood Medical Area	42.358990	-71.058630
18	Boston	South End	42.332670	-71.097910
19	Boston	South Boston	42.332670	-71.097910
20	Boston	Charlestown	42.377760	-71.067320
21	Boston	South Boston Waterfront	42.326810	-71.004650
22	Boston	Harbor Islands	42.349010	-71.034700
23	Boston	Hyde Park	42.255810	-71.124130
24	Boston	West End	42.365650	-71.067270
25	Boston	Allston	42.351140	-71.131440

Figure 2.

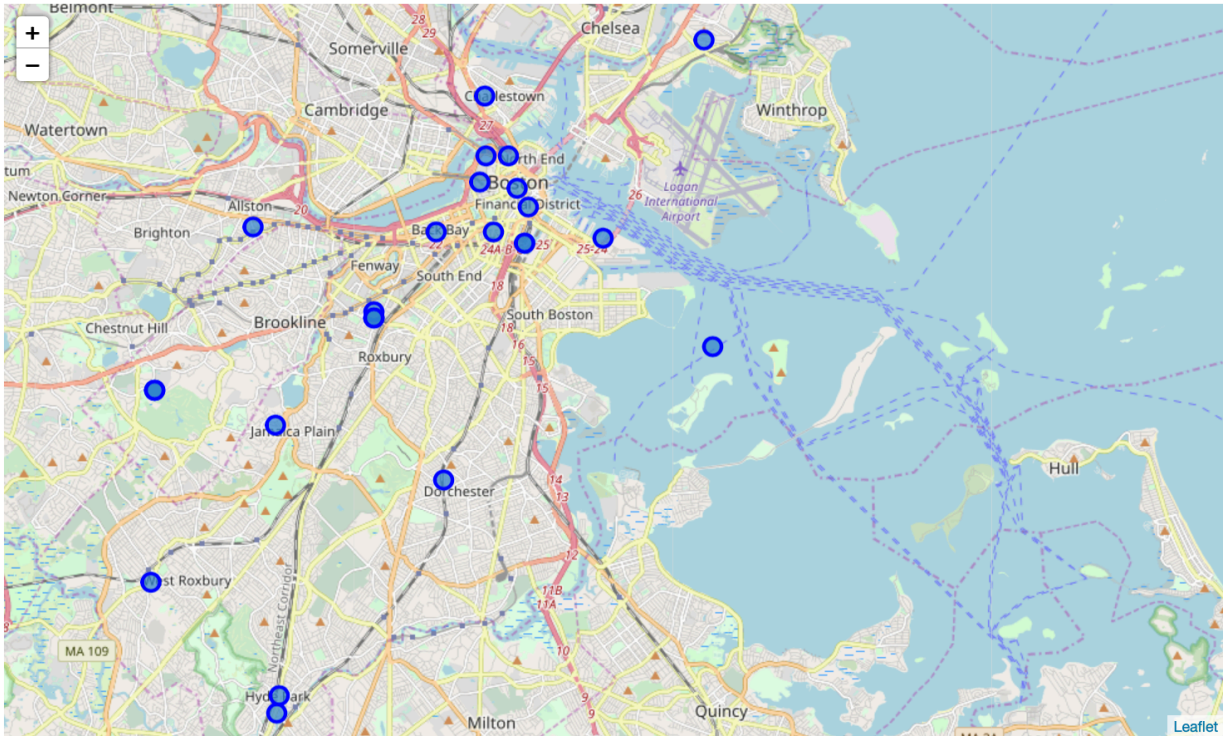

Figure 3.

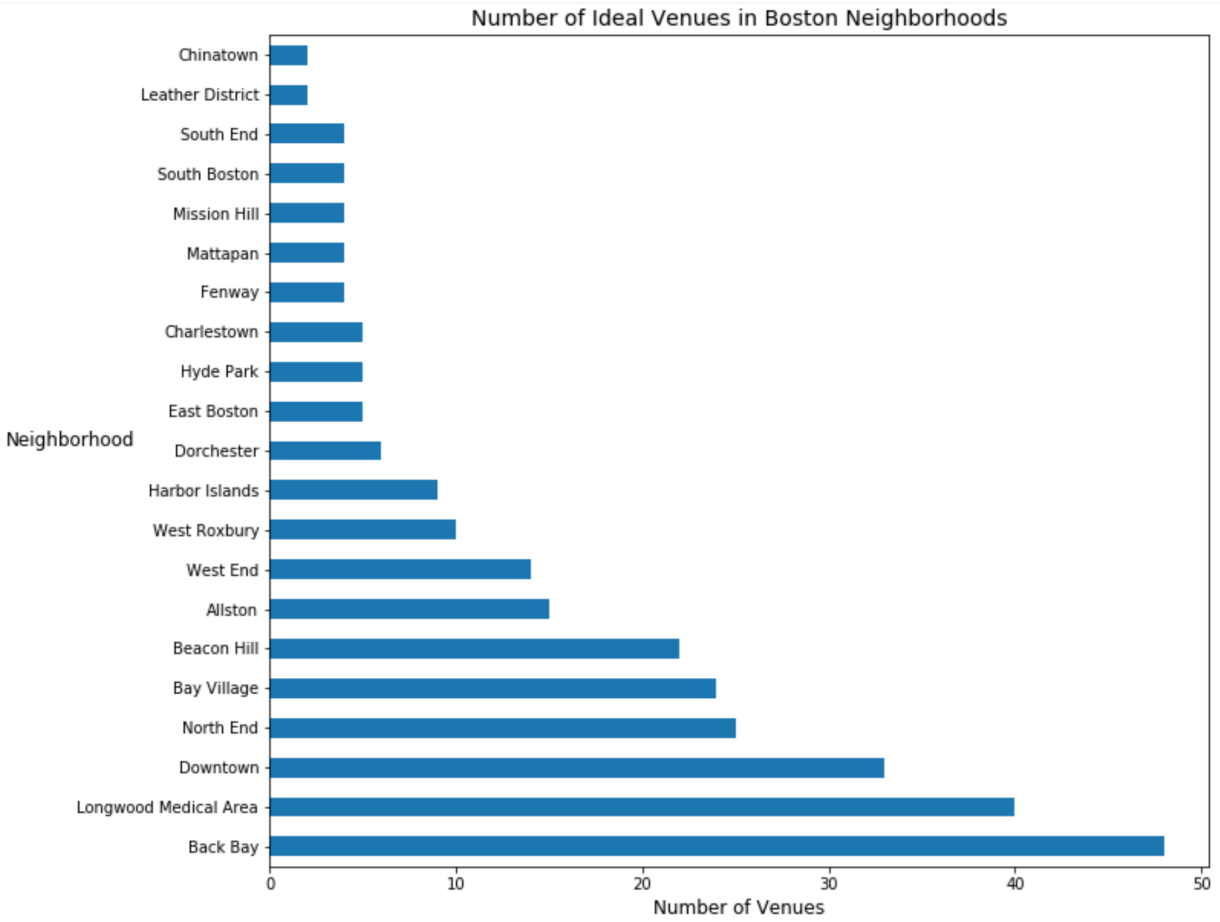

Figure 4.

	Neighborhood	Accessories Store	Art Museum	Athletics & Sports	Automotive Shop	Bank	Bar	Beach	Bed & Breakfast	Beer Bar	Beer Garden	Bike Rental / Bike Share	Board Shop	Boat or Ferry	Boutique	Boxing Gym
3	Leather District	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Leather District	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	Back Bay	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	Back Bay	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	Back Bay	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 5.

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Allston	Bar	Gym / Fitness Center	Rock Club	Department Store	Board Shop	Pharmacy	Dive Bar	Clothing Store	Liquor Store	Scenic Lookout
1	Back Bay	Sporting Goods Shop	Hotel	Cosmetics Shop	Clothing Store	Salon / Barbershop	Pet Store	Furniture / Home Store	Men's Store	Women's Store	Plaza
2	Bay Village	Theater	Hotel	Hotel Bar	Performing Arts Venue	Comedy Club	Smoke Shop	Gym	Lounge	Movie Theater	Event Space
3	Beacon Hill	Hotel Bar	Museum	Gift Shop	Kids Store	Clothing Store	History Museum	Health & Beauty Service	Gym	Optical Shop	Other Repair Shop
4	Charlestown	Yoga Studio	Bank	Pharmacy	Pet Store	Shopping Mall	Zoo Exhibit	Furniture / Home Store	Dive Bar	Dry Cleaner	Electronics Store

Figure 6.

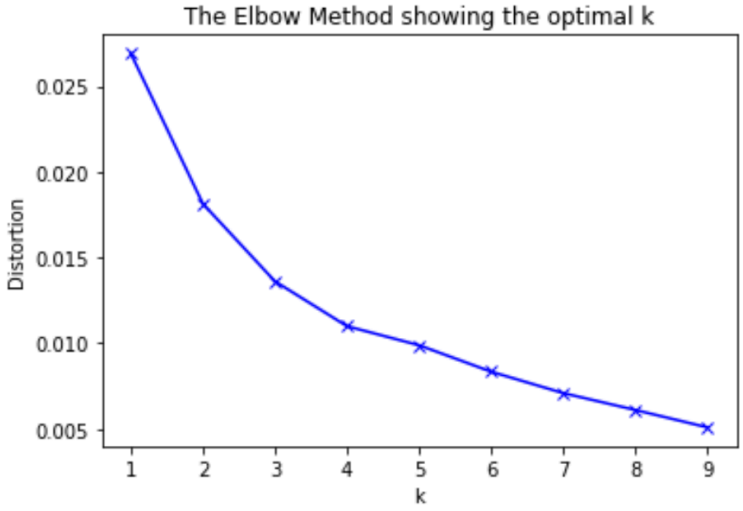

Figure 7.

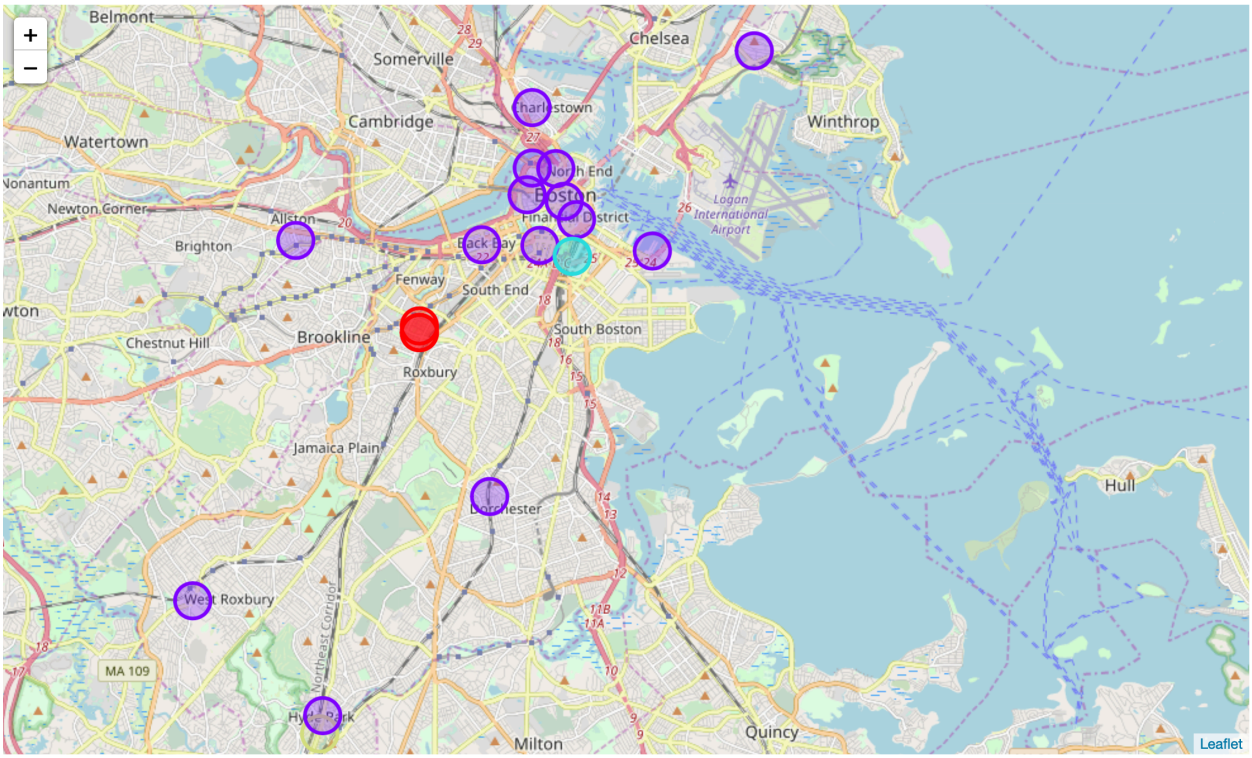

Figure 8.