Exploring the Data Locality of Restaurants in Boston

by

Kody Richardson

I. Introduction

Covid-19 has taken its toll on many industries including the restaurant industry as well. This has caused many businesses to shut down. Now with the regulations lifting we are seeing restaurants re-open. Boston is a urban area with a growing population. With the regulations lifting many people will start to rebuild their businesses.

II. Problem

I am looking to re-open my Chinese restaurant in Boston however I do not know what the best location is to open my new restaurant. I want to find a place that does not have as much competition not only from other Chinese restaurants but from other types of restaurants as well while being able to bring in plenty of customers.

III. Solution

We will gather the data of all the restaurant locations of Boston as well as the Neighborhoods. We will then cluster them together to select the best location based on latitude and longitude and neighborhood to build the new restaurant.

IV. Data Requirements

We will require 2 main components to find the location of a new restaurant. The first being data on the longitude and latitude of all the neighborhoods for clustering purposes. I have created an external csv file that I will use for this data requirement on each neighborhood. The second requirement will be the longitude and latitude of all the restaurants in Boston where we will use foursquare to acquire said data.

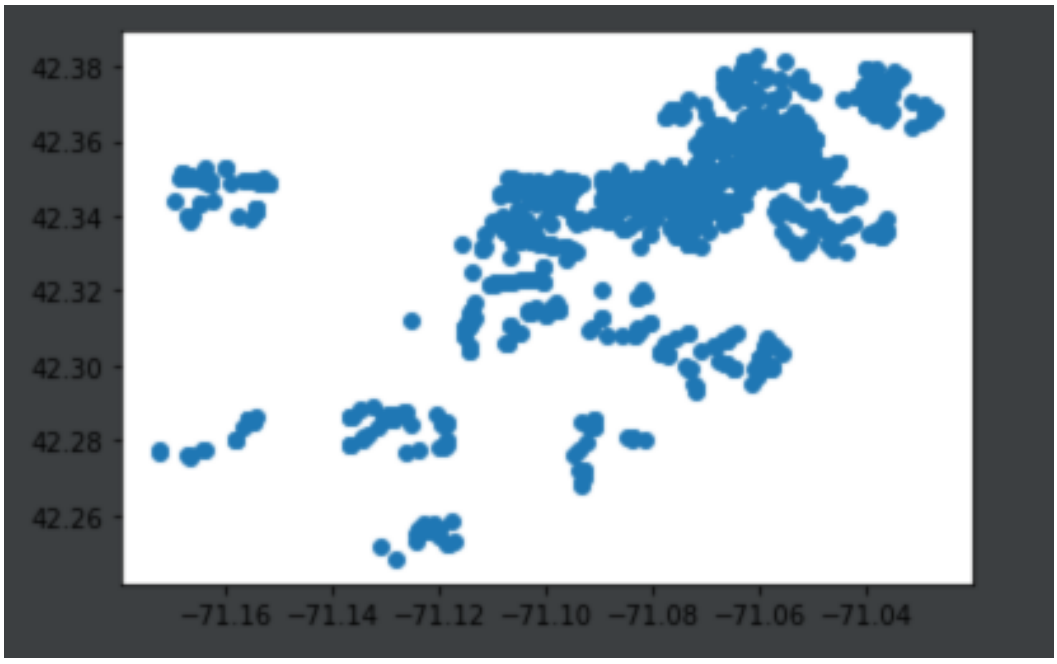
	Neighborhood	Latitude	Longitude
0	Allson	42.352900	71.132100
1	Back Bay	42.351294	-71.080356
2	Bay Village	42.349100	-71.068000
3	Beacon Hill	42.358300	-71.066100
4	Brighton	42.346400	-71.162700

A snippet of our Data Frame showing some of the neighborhoods and their locations

For collecting our data of all the restaurants, we will be using foursquare to loop through each type of restaurant such as American, Asian, Mexican, Italian, etc... After we will converge all of this data into one data frame for further analysis.

V. Methodology

After collecting the data listed above and putting the data into data frames using pandas. It is time to create a clustering algorithm based on density of the restaurants. The algorithm will sort our data into different clusters for us to view.

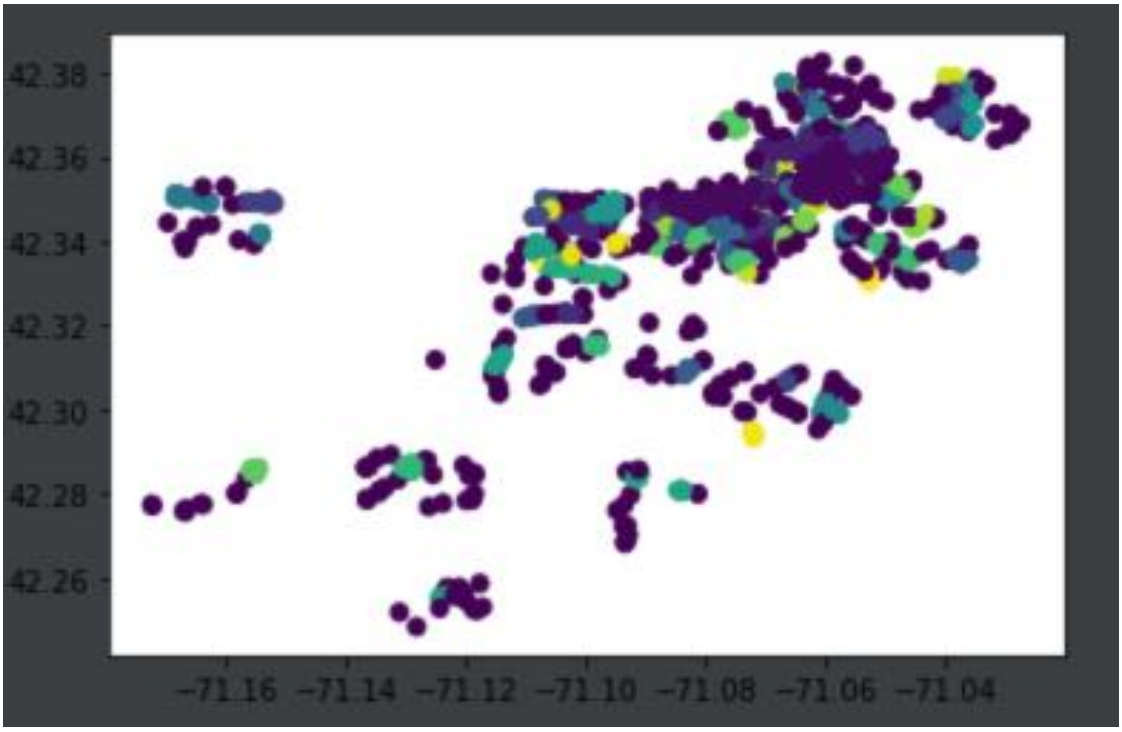


In the picture above we get a view of the different restaurant which have formed groupings. We will want to further delve into each of these groupings to find where the lowest density of Chinese Restaurants will be located.

Next, we will be looking to find clusters of our restaurant data but first we will need to find the epsilon value as well as the minimum samples value. The epsilon value will be used to determine how dense the cluster will need to be unique. While the minimum sample value will be needed for the minimum number to form a cluster.

```
# Epsilon dictates how tight the density groups can be.
# Min samples says how few data points are needed for a cluster to form
# Cluster list is a numpy array containing the X and Y coordinates of every Venue
dbScan = DBSCAN(eps=0.001, min_samples=5).fit_predict(clusterList)    clusterList:
```

We will be using the above code to fit our requirements above in the db scan to cluster our restaurants.



With the code run, we have shown what the clusters look like in the graph above. As we can see our groupings of restaurants have formed and we are able to better see the density of restaurants and clusters. Next, we will be needing to find the number of Chinese restaurants with in each of those clusters.

```
numberOfResterauntsPerCluster = []
for c in clusterList:
    counter = 0
    for k in clusterResultList:
        if k == c:
            counter+=1
    numberOfResterauntsPerCluster.append(counter)
print(numberOfResterauntsPerCluster)
```

```
[10, 7, 110, 8, 10, 10, 11, 9, 16, 18, 14, 9, 60, 11, 13, 8, 14, 21, 10, 6, 246, 12, 11, 5, 4,
```

```
numberOfChinesePerCluster = []

for c in clusterList:
    counter = 0
    for idx in range(len(clusterResultList)):
        if c == clusterResultList[idx]: # This means that the district and catagory algin
            if labels[idx] == labelResteraunt:
                counter+=1
    numberOfChinesePerCluster.append(counter)
print(numberOfChinesePerCluster)
```

```
[0, 0, 8, 0, 1, 0, 0, 1, 0, 0, 1, 2, 0, 2, 2, 0, 0, 1, 0, 2, 4, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0
```

We will use the above code to find first the number of restaurants per cluster, before breaking it further down into the number of Chinese restaurants per cluster.

Now that we have that data, we will need to find the balance between the density of restaurants and density of Chinese restaurants. We would not want to build our new restaurant in a place that has too many other types of restaurants or a place that has too many Chinese restaurants. We are looking for a unique location with the right balance of restaurants to Chinese restaurants ratio.

To achieve our goal, we will use the following code in the picture belove to find the right ration between our densities. The code will then give us a longitude and latitude of general area of where to build our new restaurant.

```
for idx in range(len(clusterList)):  clusterList: <class 'list'>: ['78', '59', '3
    weight = numberOfResterauntsPerCluster[idx]/totalVenues  numberOfResterauntsP
    percentX = numberOfChinesePerCluster[idx]/numberOfResterauntsPerCluster[idx]
    value = weight*percentX      weight: 0.010632911392405063  percentX: 0.0
    if percentX !=0:  percentX: 0.0
        finalResults.append( (clusterList[idx],value) )  finalResults: <class 'li
```

```
for idx in range(len(clusterResultList)):  clusterR
    if clusterResultList[idx] == maxName:  clusterR
        counter+=1  counter: 0
        longCounter = longCounter + longList[idx]
        latCounter  = latCounter  + latList[idx]  l
```

VI. Results

Now that we have our entire methodology down its time to run our code. The code has looked at all of our data and selected a longitude and latitude for us. With them being -71.05890045602365 and 42.35555054151881 respectively. We now know the general location of where we should build a new Chinese Restaurant.

VII. Discussion

Through out the project of finding the best location we were able to achieve our goal. However, through the process I had noticed that we could include more data to make the accuracy of the prediction model more precise. If I were to continue with this project, I would include things such as the population of each district in Boston as well as the average income of households for each neighborhood. I believe we could look at how the population density would affect the overall density or restaurants. With the household income we could take a look at what kind of Chinese restaurant we would want to open such as fine dining or street food. It would also affect our pricing strategy of the restaurant.

VIII. Conclusion

We have created our model based on the density of restaurants and Chinese restaurants per district in Boston. We have found a general location to build or new restaurant, however our data set is not complete and we can further refine our options. We have taken the first step in creating a predication model and we have many more steps to go in the future to make everything more precise. We can further tweak our DB Scan algorithm and introduce new data as stated above. As always there is more work to be done in the pursuit of data science and I look forward to exploring more of this project.