Applied Data Science Capstone

Clustering Bangalore Neighborhoods

Introduction

Background

Bangalore (Bengaluru) is a bustling metropolis of India. People from all walks of life find their way into this diverse city. Due to a large tech industry that creates thousands of jobs a year, there is a huge influx in population.

Problem

Every person that moves in is faced with the question of where to stay. One major factor in determining that is what facilities and attractions a neighborhood offers. If we analyzed different neighborhoods in Bangalore clustered them on their most popular attractions, it could serve as a good starting point in a search for accommodation.

When business owners determine the location of their store/business they need to look into what other attractions offer. Lack of similar businesses may offer a competitive advantage but might indicate a lack of consumer interest. Similarly a lot of similar businesses can mean either there is a lot of demand and more businesses are welcome or that the market has become saturated.

Interest

Neighborhoods analysis is can be an important tool for businesses in determining where to open their shop. It can also help people moving into a new area find similar neighborhoods that they can choose from

We will be clustering the different neighborhoods in Bangalore based on their 10 most popular attractions.

.

Data

Acquisition, Cleaning and Feature Selection

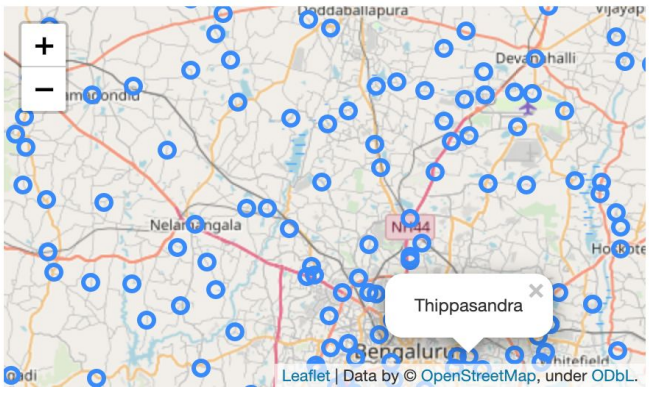
Pincodes and their corresponding post office were obtained from data.gov.in(https://data.gov.in/resources/all-india-pincode-directory-contact-details-along-latitude-and-longitude).

Only the 'officename' and 'regionname' columns were chosen from this dataset. This dataset was then filtered to contain only pincodes in the region Bangalore. The post office name was taken as the neighborhood name, and trailing characters were cleaned. Each neighborhood name was geocoded using Nominatim. Then the most popular attractions from each neighborhood were obtained using the Foursquare Places API. The API returned the top 100 results. Only the venue category was used.

The resulting dataset was filtered to contain just the top ten attractions of each neighborhood (based on frequency of occurrence). This data was used for further analysis and clustering.

Analysis and Preprocessing

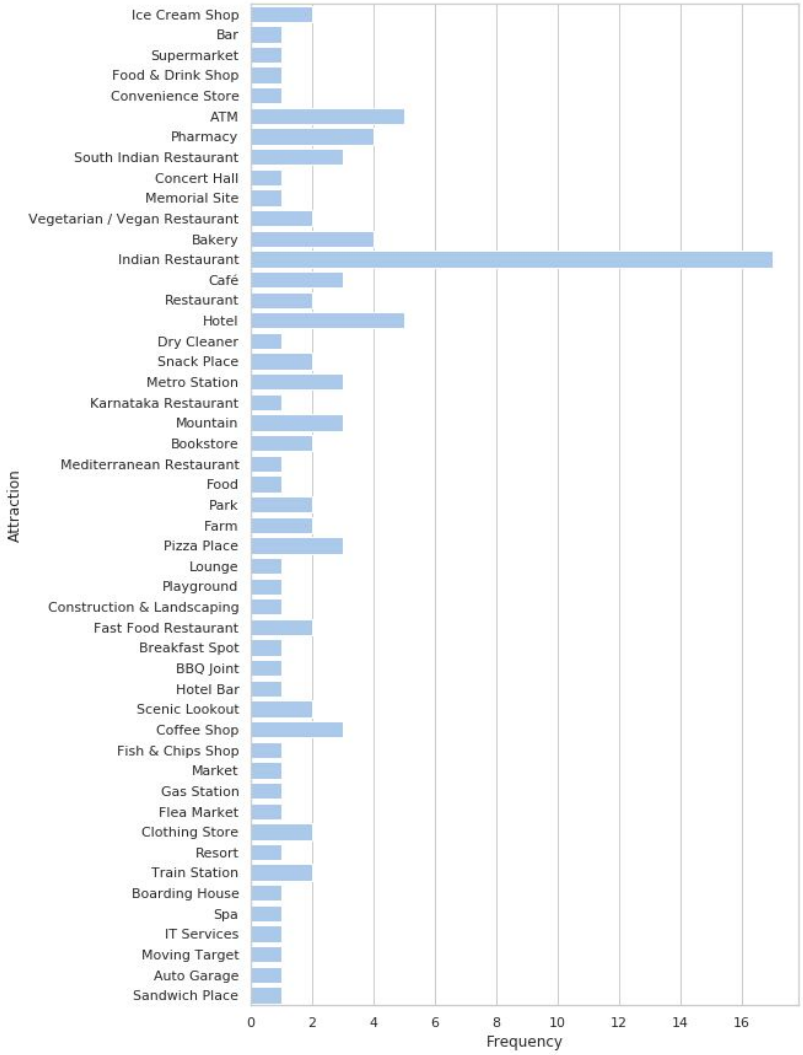
The neighborhoods were initially plotted on a map



The number of venue categories for each neighborhood was displayed. It was found that there were 172 unique categories.

Neighborhood	
Adugodi	5
Agram	100
Akkur	28
Alahalli	1
Amruthahalli	3

The most popular attractions were then visualized using seaborn.

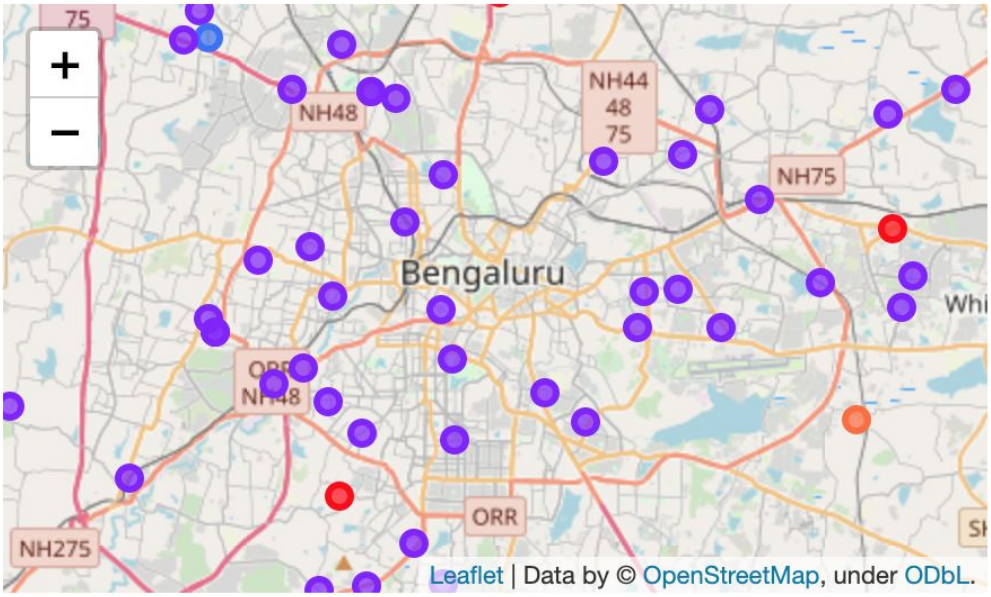


The categories were then one-hot encoded as clustering is not meaningful when done on categorical data.

Clustering

K Means Clustering

K means clustering was applied to the dataset with varying values of k and the mean squared errors were plotted. There were 104 values in the dataset so the value of k was iterated between 2 and 11 (square root of 104 is 10.2). The resulting elbow plot was inconclusive so the silhouette score was computed for various values of k. k=8 was determined to be the best choice as it had the highest silhouette score. The resulting clusters were visualized using folium.



Each individual cluster was then analyzed to check for patterns.

K Medoids Clustering

K Medoids clustering aims to minimize the sum of dissimilarities between the centroid of each cluster and points labelled to be in a cluster. It was implemented using the sklearn.extra.clustering library.

The optimal value of k was determined using the elbow method. There were 2 distinct elbows at k=2 and k=6, so clustering was done using both these points. k=6 makes more sense in this context as it splits the database into a larger number of clusters which is more meaningful.

DBSCAN Clustering

DBSCAN stands for Density Based Spatial Clustering of Applications with Noise. It is a density based clustering algorithm, hence the number of clusters are not explicitly inputted. It takes two inputs - epsilon, which is the radius of the circle it considers, and minpts, which is the minimum number of points that need to be inside the circle with radius epsilon for it to be considered a 'core point'. It was implemented using the sklearn.clustering library.

Despite varying the values of epsilon and minpts over a wide range DBSCAN never clustered the data into more than 2 clusters. Thus, the results were not very useful

Agglomerative Clustering

Agglomerative Clustering is a hierarchical clustering algorithm. The data was clustered into 8 different clusters. The clusters were observed for similarity

Results

Evaluation Metrics

Internal and external metrics were used to compare the various clustering algorithms and evaluate the quality of the clusters.

Silhouette Scores:

The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The silhouette ranges from −1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

K Means : 0.19610746739247847

K Medoids : 0.07733267946208815

DBSCAN : 0.1391318150805085

Agglomerative Clustering : 0.137552821016789

Davies-Bouldin Scores:

The score is defined as the average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances. Thus, clusters which are farther apart and less dispersed will result in a better score.

K Means : 1.1792534290426284

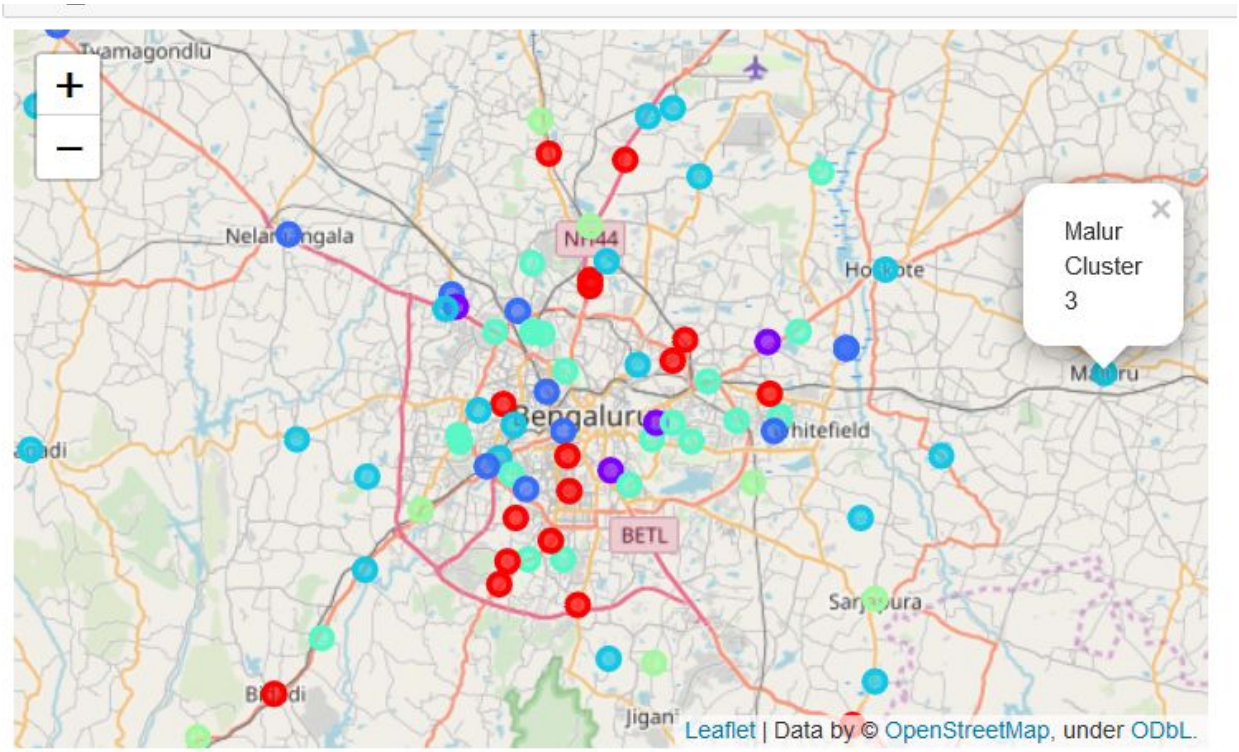
K Medoids : 3.916227564389721

DBSCAN : 3.4678885613218626

Agglomerative Clustering : 1.3956784752393547

Inference

DBSCAN seems to do best according to metrics as it has the best Silhouette Score as well as the second best Davies-Bouldin Score. However, it splits the dataset into 2 clusters which is too generalized. K Medoids offers the best Davies-Bouldin Score so it its results can be considered for further analysis.



Final Map Showing Neighborhoods Clustered by Attractions

Discussion

We can analyze the resulting clusters based on what our needs are. For aspiring entrepreneurs it can serve as a useful metric to predict the success of their venture based on what are the other popular attractions in the area. For people considering moving into a neighborhood they can check what other neighborhoods are similar to theirs, allowing them to account for other factors such as price or distance to workplace.

Conclusion

Neighborhood Analysis can be a valuable step in making decisions for business and personal reasons. There are several different clustering algorithms. There is no 'one-size-fits-all' approach to clustering. Instead we must experiment with various

clustering algorithms and use metrics to determine which algorithm yields the best results. Once we cluster neighborhoods we can try to identify patterns within the cluster or perform data analysis and visualization techniques to make better sense of the clusters. We can then use this information to guide our decision making process.