Clustering localities in Bangalore, India based on Restaurants

Rashmika M G

August 19, 2019

1. Introduction

1.1 Business Problem

People visiting new cities would be highly interested in the localities with the best restaurants in the city. People might want to know how good a given restaurant is based on the ratings the restaurant has received and would like to know the price range the Restaurant falls under so that they can make informed budget decisions. Also, they would like to know the best localities where they could find these restaurants. The information of ratings and price range of various restaurants in the city and their localities in form of graphs, charts and maps would help people decide which restaurant to choose amongst the many restaurants in the city. And also which locality to visit. Also combining the location of the restaurants in the city with their price and rating information would help visitors make easy decisions about the locations they should visit. A map of the restaurants and another map of the localities with specific color attributes will be plotted to highlight their position. Further, we will classify the various locations into different clusters using a Machine Learning Algorithm, the K-means clustering Algorithm. This enables any visitor to take a quick glance and decide what place to visit.

1.2 Interested Audience

The target audience for this project are people interested in exploring restaurants in Bangalore and startups who would like to harness this information and present it to consumers. Firstly, it is for any person who is visiting Bangalore for the first time and would like to choose a location of stay based on his restaurant preferences or for anyone who would like to explore different restaurant options in various localities in Bangalore. They can use the plots and maps from this project to quickly select restaurants that suit their budget and rating preferences. Secondly, any startup can use this information to create a website or a mobile application, to allow individuals to the explore restaurants in various localities in the city using a map view.

2. Data

2.1 Data Sources

To get location and other information about various venues in Bangalore, two APIs were used. The Foursquare API and the Zomato API.

The Foursquare's explore API was used to fetch venues up to a range of 35 kilometers from the center of Bangalore. The names, categories and locations (latitude and longitude) of these venues were collected.

Using the name, latitude and longitude values obtained from the Foursquare API, we used the Zomato search API to fetch data from its database. The Zomato API allows to find only restaurants based on a search criteria using the name, latitude, longitude, etc.

The data from the two APIs do not match completely because Foursquare API retrieves all venues in Bangalore and the Zomato API retrieves only restaurants in Bangalore. So, we combine the two datasets to get only Restaurants from the Foursquare API and the corresponding ratings and price information from the Zomato API. We use various techniques of Data cleaning to get the final dataset.

From Foursquare API (https://foursquare.com/city-guide), the following for each venue was retrieved:

- **Name:** The name of the venue.
- **Category:** The category type as defined by the API.
- **Latitude:** The latitude value of the venue.
- **Longitude:** The longitude value of the venue.

From Zomato API (https://developers.zomato.com/api), the following for each restaurant was retrieved:

- **Name:** The name of the restaurant.
- **Locality:** The locality of the restaurant.
- **Rating:** The average rating of the restaurant given be users.
- **Price range:** The price ranges the restaurant belongs to as defined by Zomato.
- **Price for two:** The average cost for two people dining at the restaurant.
- **Latitude:** The latitude value of the restaurant.
- **Longitude:** The longitude value of the restaurant.

2.2 Data Cleaning

From the Foursquare API, we get the following data:

	name	categories	lat	lng
0	JW Marriott Hotel Bengaluru	Hotel	12.972362	77.595051
1	UB City	Shopping Mall	12.971709	77.595905
2	Cubbon Park	Park	12.977042	77.595277
3	Truffles - Ice & Spice	Burger Joint	12.971802	77.601031
4	Toscano	Italian Restaurant	12.971980	77.596066

Using the above name, latitude and longitude values, we perform a search query using the Zomato API and get the following data:

	Unnamed: 0	venue	latitude	longitude	locality	price_for_two	price_range	rating
0	0	ROYCE' Chocolate	12.972469	77.595103	JW Marriott Bengaluru, Lavelle Road	1000	3	3.5
1	1	Shiro	12.971758	77.595922	UB City	3000	4	4.4
2	2	Mathsya Darshini	12.975296	77.588858	Lavelle Road	350	1	3.4
3	3	Truffles	12.971769	77.601137	St. Marks Road	900	2	4.4
4	4	Shiro	12.971758	77.595922	UB City	3000	4	4.4

We initially source data from the Zomato API and then we save the sourced data into a csv file. To run the notebook later we source data from the csv file directly instead of calling the API again and again. We do this because Zomato limits its API to be called only 1000 times a day. We will perform the following steps to clean the above dataframe.

1. **Remove unwanted columns:** We will remove the unwanted column Unnamed:0 which was sourced as an index from the csv file. Since we already have numeric index we will not require this column.
2. **Remove Duplicate rows:** We see that the 2nd and the 5th rows are duplicates, both containing the Restaurant Shiro, hence we will remove duplicate rows. There can a shopping mall at a particular location and inside the mall there could be a restaurant and a lounge. So, all these 3 venues the shopping mall, the restaurant and the lounge will have the same latitude and longitude values. When we try to search for these location values using the Zomato API, the query with the shopping mall would result with restaurant the mall has. Therefore, we drop these duplicate values.
3. **Remove Redundant features:** We will remove price_range column since it is redundant. The columns price_for_two and price_range both represent similar features.
4. **Clean Locality column:** We will clean the locality column to contain only the locality and strip off any unnecessary data such as the block information, hotel information etc. For example: the 1st row contains the Royce Chocolate restaurant which is located inside the Hotel, JW Marriott which is on Lavelle Road. We will strip off the Hotel name so that only Lavelle Road is present in the locality column.

The final dataframe will look like the one shown below:

	venue	latitude	longitude	locality	price_for_two	price_range	rating
0	ROYCE' Chocolate	12.972469	77.595103	Lavelle Road	1000	3	3.5
1	Shiro	12.971758	77.595922	UB City	3000	4	4.4
2	Mathsya Darshini	12.975296	77.588858	Lavelle Road	350	1	3.4
3	Truffles	12.971769	77.601137	St. Marks Road	900	2	4.4
4	Smoke House Deli	12.971659	77.598318	Lavelle Road	1600	3	4.7

3. Methodology

The first step in the Project is data sourcing. We retrieve the venues in Bangalore from the Foursquare API and corresponding restaurants at those venues from the Zomato API. We will extract the location data from the Foursquare API for all venues up to a distance of 35

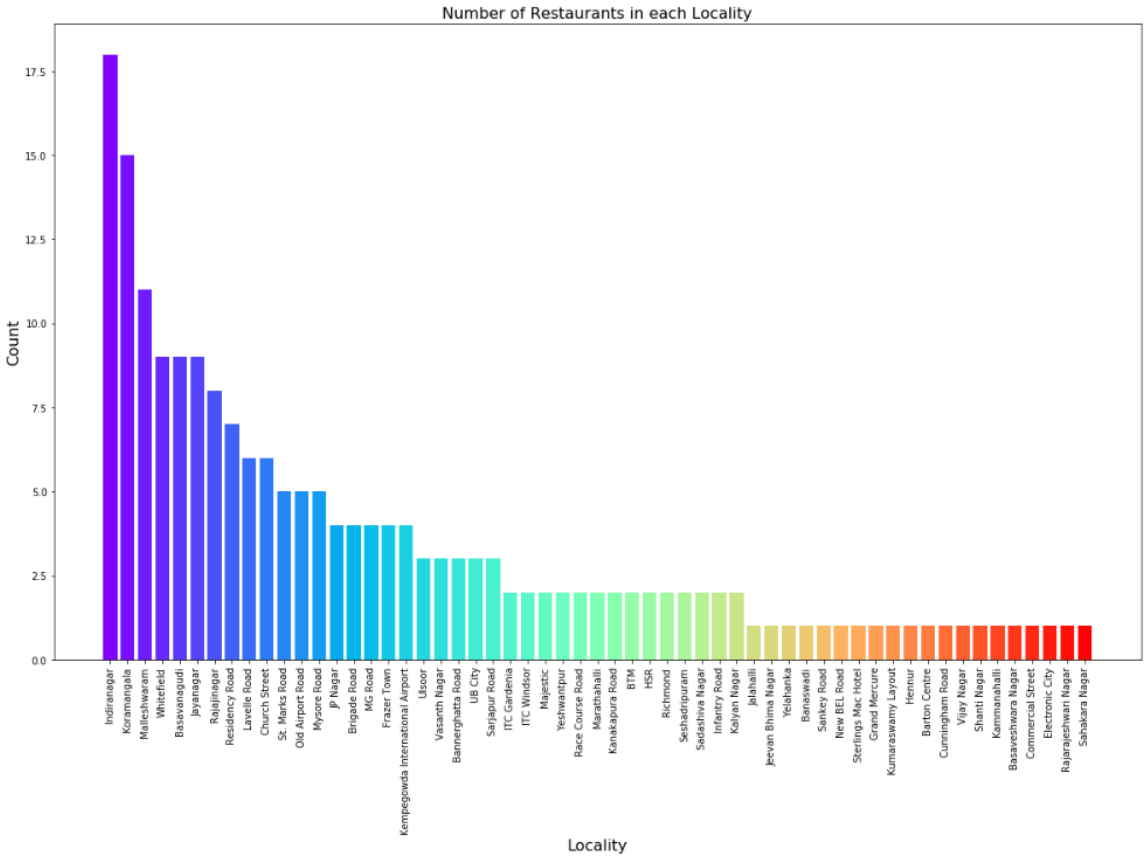
kilometers from the center of Bangalore. Using this, we will fetch the restaurant information including price for two and average ratings from Zomato API.

The second step is to clean the data we obtain from the Zomato API. We will remove unwanted columns, redundant features, duplicate rows and clean the locality column. The final data will include the Restaurant name, latitude, longitude, locality, price for two, and rating.

3.1 Data Analysis

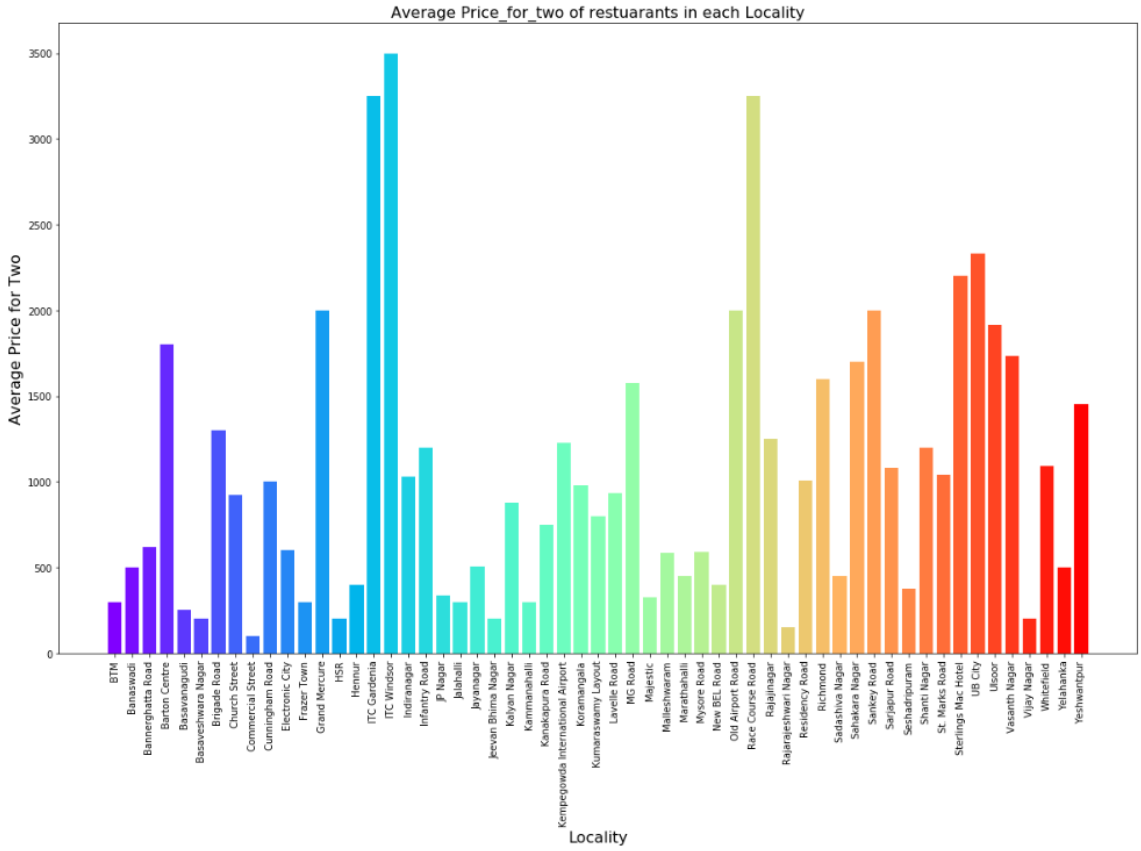
Using this dataset, we begin by analyzing the top localities in Bangalore that have the most number of restaurants types. This will allow us to better understand the places where many venues co-exist and are worth visiting. I'll also explore the venues based on the ratings and price range of various venues. The venues will be plot using rainbow color coding such that a simple glance at the chart would reveal the location of the maximum number of restaurants. We will also analyze average price of the restaurants in various localities and also their ratings. The aim to identify localities which can be recommended to visitors based on their price, rating and variety preferences.

1. **Number of Restaurants in each Locality:**



From the above bar graph, we see that the locality with the greatest number of restaurants is **Indiranagar**, followed by **Koramangala** and **Malleshwaram**. As a tourist we can plan to visit above localities for a maximum variety in the number of different restaurants to visit.

2. **Average price for two of restaurants in each locality:**



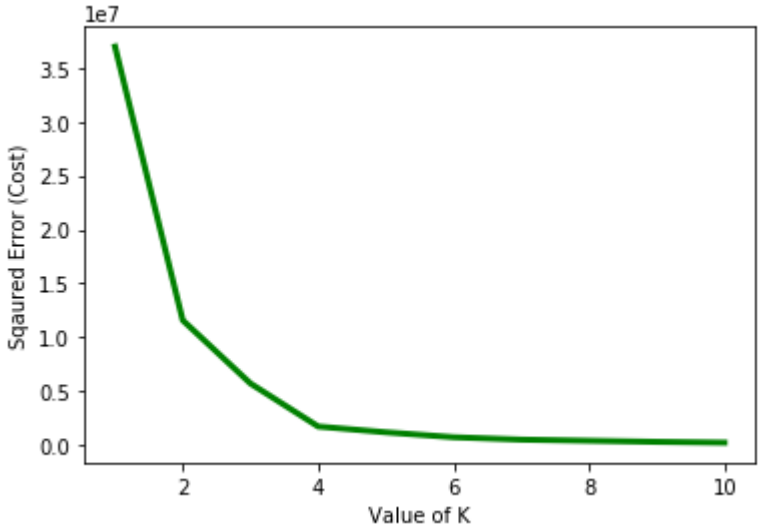
From the above bar chart, we can see the localities with the most expensive restaurants, ITC Gardenia and ITC Winsdor to the localities having the least expensive restaurants. This can help visitors to plan according to their budget.
A similar analysis can be done on the restaurant ratings in different localities in Bangalore.

3.2 Machine Learning Algorithm – K-Means Clustering

We will cluster the various localities in Bangalore based on its restaurants rating and price data. We will use the K-mean Clustering algorithm to cluster the localities into K different clusters. Restaurants in a cluster will be similar to each other, while restaurants in other clusters will be dissimilar to restaurants in another cluster. To find out the best value of K in K-means we will use the elbow method. We'll plot:

- values for K on the horizontal axis
- the distortion on the Y axis (the values calculated with the cost function).

When K increases, the centroids are closer to the cluster's centroids. The improvements will decline, at some point rapidly, creating the elbow shape. That point is the optimal value for K. In the plot below, K=2 and K=4 are optimal. We will choose K=4 for this project.

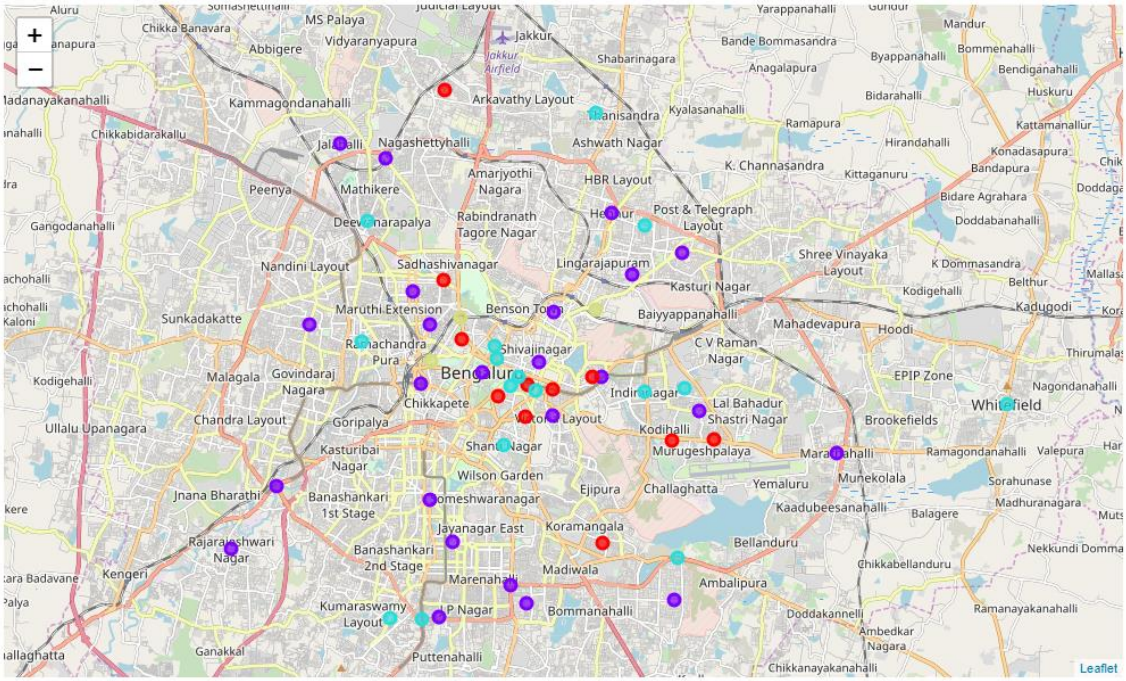


Now that we have clustered various localities based on their restaurants, we will plot these localities on the map of Bangalore with different colors for different clusters.

To plot these localities we will need their latitude and longitude values which we will obtain from the Geopy API. We will get a dataframe as follows:

	Cluster Labels	locality	latitude	longitude
0	1	BTM	12.911276	77.604565
1	1	Banaswadi	13.014162	77.651854
2	1	Bannerghatta Road	12.916659	77.599842
3	0	Barton Centre	12.975367	77.605053
4	1	Basavanagudi	12.941726	77.575502

In the below map of Bangalore, we plot the different localities color coded according to their cluster groups.



Analysis of Cluster:

1. Cluster 0 -

```
locality_merged.loc[locality_merged['Cluster Labels'] == 0]
```

	venue	latitude	longitude	locality	price_for_two	price_range	rating	Cluster Labels
1	Shiro	12.971758	77.595922	UB City	3000	4	4.4	0
11	Cafe Noir	12.972126	77.596441	UB City	1500	3	4.2	0
12	Skyye	12.971632	77.596371	UB City	2500	4	4.3	0
14	McDonald's	12.976243	77.598372	MG Road	500	2	3.8	0
15	Rim Naam - The Oberoi	12.972776	77.618641	MG Road	3000	4	4.6	0
17	Toast & Tonic	12.966665	77.608927	Richmond	2000	4	4.6	0

2. Cluster 1 -

```
locality_merged.loc[locality_merged['Cluster Labels'] == 1]
```

	venue	latitude	longitude	locality	price_for_two	price_range	rating	Cluster Labels
0	ROYCE' Chocolate	12.972469	77.595103	Lavelle Road	1000	3	3.5	1
2	Mathsya Darshini	12.975296	77.588858	Lavelle Road	350	1	3.4	1
3	Truffles	12.971769	77.601137	St. Marks Road	900	2	4.4	1
4	Smoke House Deli	12.971659	77.598318	Lavelle Road	1600	3	4.7	1
5	Hard Rock Cafe	12.976034	77.601567	St. Marks Road	2500	4	4.5	1
6	Corner House Ice Cream	12.973186	77.599967	Lavelle Road	350	1	4.4	1
8	Harima	12.967536	77.599901	Residency Road	2000	4	4.3	1

3. Cluster 2 -

```
locality_merged.loc[locality_merged['Cluster Labels'] == 2]
```

	venue	latitude	longitude	locality	price_for_two	price_range	rating	Cluster Labels
18	Brahmin's Coffee Bar	12.954032	77.568948	Basavanagudi	100	1	4.8	2
26	Hari Super Sandwich	12.932848	77.582555	Jayanagar	200	1	4.4	2
30	CTR	12.998270	77.569455	Malleshwaram	150	1	4.7	2
38	Upahara Darshini	12.939350	77.571491	Basavanagudi	150	1	4.2	2
39	Fishland	12.975600	77.578557	Majestic	500	2	4.1	2
42	S R & Sons Bakery	12.983502	77.606561	Commercial Street	100	1	3.5	2
45	Mavalli Tiffin Room (MTR)	12.955176	77.585622	Basavanagudi	250	1	4.5	2
47	Cookie Man	13.011356	77.555020	Malleshwaram	150	1	3.6	2

4. Cluster 3 -

```
locality_merged.loc[locality_merged['Cluster Labels'] == 3]
```

	venue	latitude	longitude	locality	price_for_two	price_range	rating	Cluster Labels
7	Masala Klub - The Taj West End	12.984113	77.583968	Race Course Road	4000	4	4.4	3
9	Edo Restaurant & Bar - ITC Gardenia	12.967392	77.596392	ITC Gardenia	4000	4	4.3	3
34	Dum Pukht Jolly Nabobs - ITC Windsor	12.994669	77.585355	ITC Windsor	5000	4	4.3	3
56	Cubbon Pavilion - ITC Gardenia	12.967401	77.596393	ITC Gardenia	2500	4	4.3	3
60	Blue Bar - The Taj West End	12.984111	77.583966	Race Course Road	2500	4	4.0	3
126	The Raj Pavilion - ITC Windsor	12.994645	77.585229	ITC Windsor	2000	4	4.2	3

5. Results and Discussion

After collecting data from the Foursquare API, we got a list of 242 venues. However, not all venues have restaurants. We then collected a list of all restaurants from these 242 venues from the Zomato API. We got a list of 195 restaurants after removing the duplicates. We grouped the restaurants based on locality. We obtained a total of 57 localities. The locality with the highest number of restaurants is Indiranagar, followed by Koramangala and then Malleshwaram. ITC Gardenia and ITC Windsor are the localities with restaurants which are very high priced.

Finally, through clusters we identified that following:

1. Cluster 0 has localities with its Restaurants being expensive for 2 people.
2. Cluster 1 has a slightly less expensive restaurant localities.
3. Cluster 2 has the cheapest restaurants, mostly ranging from Rs.100 to 1000 for 2 people.
4. Cluster 3 has the most expensive restaurants in the most expensive localities of Bangalore and all the restaurants has a rating of more than 4.

Conclusion

The purpose of this project was to cluster the localities of Bangalore based on their Restaurants price and ratings. The restaurants have been identified using Foursquare and Zomato API and their localities have been clustered and plotted on the map. The map shows us the most high rated, expensive localities and the least expensive ones based on their restaurants.