FINAL REPORT

Capstone Project - The Battle of Neighbourhoods

-SUHAS SOMASHEKAR

Strategic Location for Establishing a Restaurant

TABLE OF CONTENTS

Finding the optimal location in opening an eatery in Bangalore, India using k-Means Clustering

BACKGROUND

Bangalore is the third-most populous city in India with a population of 10 million.It is referred as the "Silicon Valley of India" (or "IT capital of India") because of its role as the nation's leading information technology (IT) exporter.
Indian technological organisations ISRO, Infosys, Wipro and HAL are headquartered in the city. Bangalore is the second fastest-growing major metropolis in India. Recent estimates of the economy of Bangalore's metropolitan area have ranged from $45 to $83 billion.With its population growing rapidly, it has high requirements for quality Restaurants.

The success of establishing a new restaurant depends on several factors: demand, brand loyalty, quality of food, competition, and so on. In most cases, a restaurant's location plays an essential determinant for its success. Hence, it is advantageous and of utmost importance to determine the most strategic location for establishment in order to maximize business profits.

Business Problem

A client seeks to establish a new restaurant in a Bangalore city neighborhood.

1. Which neighbourhood would appear to be the optimal and most strategic location for the business operations?
2. How will the analysis from this report help the new businesses strategically target the market and help in a high return on investment so is low risk?
3. What are the neighbourhoods that lack good number of restaurants?
4. What are the neighbourhoods that contain more number of restaurants?
5. Which area should a person visit to have a quality food experience?

- The objective of this capstone project is to locate the optimal neighborhood for operation.
- Our foundation of reasoning would be based on spending power, distribution of ethnic group , and competition, across each neighbourhood.
- We will mainly be utilizing the Foursquare API and the extensive geographical and census data from Bangalore's Open Data Portal.

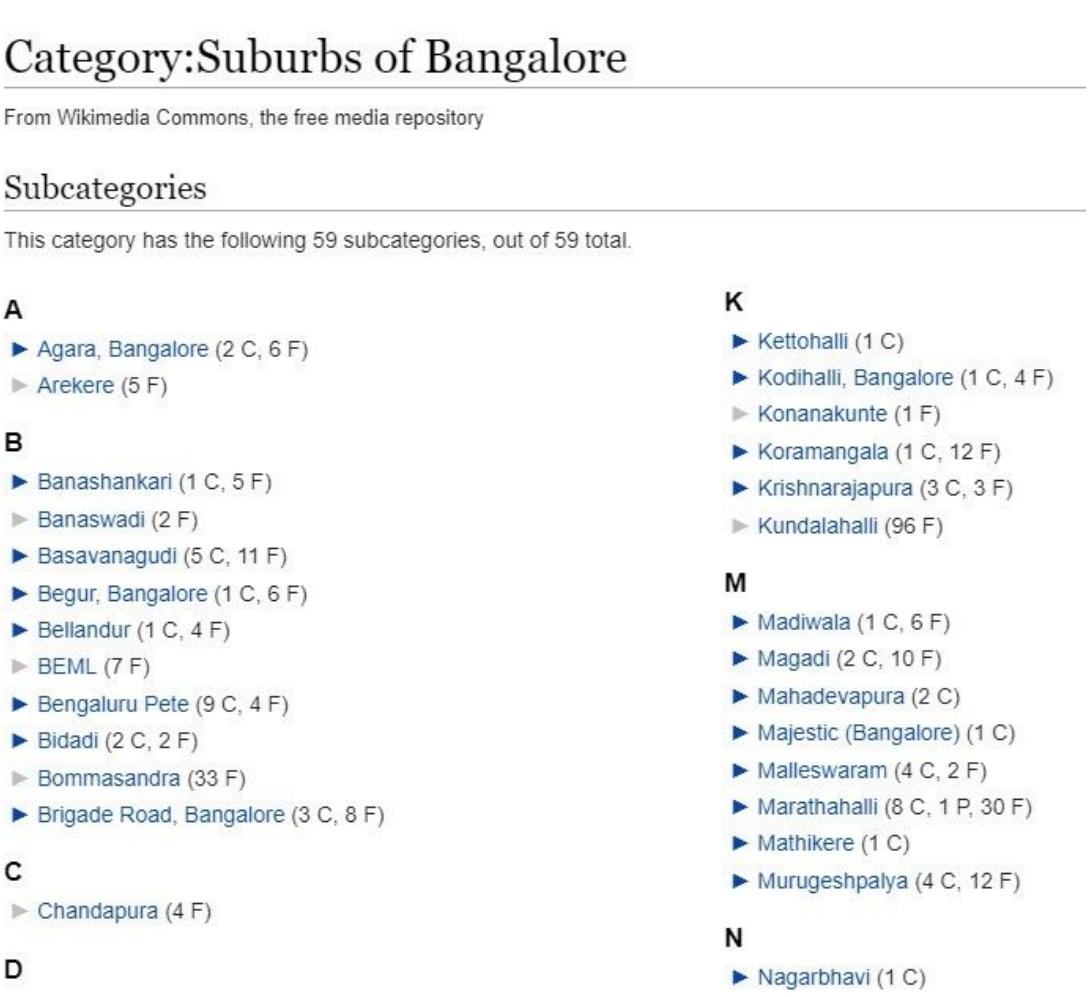
Stake Holders

- Fellow entrepreneurs seeking to either establish a new restaurant of a certain niche or have plans to expand their franchised restaurants would be very interested in the competitive advantages and business values this finding can potentially reap.
- The Neighbourhood will benefit directly from the opening of a new Eatery providing them better choices for their food
- Government which will benefit indirectly from opening of a new Restaurant through collection of taxes.
- People planning to settle in neighbourhoods which have more number of restaurants
- Visitors looking forward to have a good Food Experience.

Data Sources

1. The neighbourhoods of Bangalore alongside their respective postal codes and boroughs were scraped from Wikipedia.

Geographical coordinates for each neighbourhood were extracted from https://commons.wikimedia.org/wiki/Category:Suburbs_of_Bangalore.
We will find the latitude and longitude of each neighborhood and cluster them according to the restaurants present in each neighborhood fetched from foursquare location data. Then we will make a decision examining each cluster of neighborhoods.



Category:Suburbs of Bangalore

From Wikimedia Commons, the free media repository

Subcategories

This category has the following 59 subcategories, out of 59 total.

A
- Agara, Bangalore (2 C, 6 F)
- Arekere (5 F)

B
- Banashankari (1 C, 5 F)
- Banaswadi (2 F)
- Basavanagudi (5 C, 11 F)
- Begur, Bangalore (1 C, 6 F)
- Bellandur (1 C, 4 F)
- BEML (7 F)
- Bengaluru Pete (9 C, 4 F)
- Bidadi (2 C, 2 F)
- Bommasandra (33 F)
- Brigade Road, Bangalore (3 C, 8 F)

C
- Chandapura (4 F)

D

K
- Kettohalli (1 C)
- Kodihalli, Bangalore (1 C, 4 F)
- Konanakunte (1 F)
- Koramangala (1 C, 12 F)
- Krishnarajapura (3 C, 3 F)
- Kundalahalli (96 F)

M
- Madiwala (1 C, 6 F)
- Magadi (2 C, 10 F)
- Mahadevapura (2 C)
- Majestic (Bangalore) (1 C)
- Malleswaram (4 C, 2 F)
- Marathahalli (8 C, 1 P, 30 F)
- Mathikere (1 C)
- Murugeshpalya (4 C, 12 F)

N
- Nagarbhavi (1 C)

2.For identifying the number of restaurants in the vicinity of each neighbourhood, we will be utilizing Foursquare API, more specifically, its explore function. One has to register for a Foursquare developer account to access their API credentials.

5. Use the Foursquare API to explore the neighborhoods

In [98]:
```
# define Foursquare Credentials and Version
CLIENT_ID = 'A2VC5XGM0A0G3EUICIQTHSN4KKKX14QLG2DQKSZ5AHBSIYVR' # your Foursquare ID
CLIENT_SECRET = 'X2IBNGDICER2NWFOGW0VGKD3DPWBWG3SQPWC5UG32OHGC1BQ' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: A2VC5XGM0A0G3EUICIQTHSN4KKKX14QLG2DQKSZ5AHBSIYVR
CLIENT_SECRET:X2IBNGDICER2NWFOGW0VGKD3DPWBWG3SQPWC5UG32OHGC1BQ
```

Data Clearing

Data downloaded or scraped from multiple sources will be combined into one table. If there are a lot of missing values for certain neighbourhoods, due to lack of record keeping,such values will be removed. Few assumptions will be made to construct the dataframe:

- Only the cells that have an assigned borough will be processed; boroughs that were not assigned will be ignored.
- Neighbourhoods missing more than two census data value will be dropped.
- A column that features the group rows by neighborhood and by taking the mean of the frequency of occurrence of each category will be constructed.

METHODOLOGY

- **Web scraping Wikipedia page for neighbourhoods list**

Firstly, we need to get the list of neighbourhoods in the city of Bangalore. Fortunately, the list is available in the Wikipedia page (https://en.wikipedia.org/wiki/Category:Suburbs_in_Bangalore). We will do web scraping using Python requests and beautifulsoup packages to extract the list of neighbourhoods data. However, this is just a list of names.

2. Scrap data from Wikipedia page into a DataFrame

In [198]:
```
# send the GET request
data = requests.get("https://commons.wikimedia.org/wiki/Category:Suburbs_of_Bangalore").text
```

In [202]:
```
# parse data from the html into a beautifulsoup object
soup = BeautifulSoup(data, 'html.parser')
```

In [203]:
```
# create a list to store neighborhood data
neighborhoodList = []
soup
```

```
min*/mw.user.tokens.set({"patrolToken":"+\\","watchToken":"+\\","csrfToken":"+\\"});
});});</script>
<link href="/w/load.php?lang=en&amp;modules=ext.categoryTree.styles%7Cext.tmh.thumbnail.styles%7Cext.uls.pt%7Cext.visualEditor.desktopArticleTarget.noscript%7Cext.wikimediaBadges%7Cjquery.makeCollapsible.styles%7Cmediawiki.action.view.categoryPage.styles%7Cmediawiki.helplink%7Cskins.vector.styles.legacy%7Cwikibase.client.init&amp;only=styles&amp;skin=vector" rel="stylesheet"/>
<script async="" src="/w/load.php?lang=en&amp;modules=startup&amp;only=scripts&amp;raw=1&amp;skin=vector"></script>
<meta content="" name="ResourceLoaderDynamicStyles"/>
<link href="/w/load.php?lang=en&amp;modules=ext.gadget.Long-Image-Names-in-Categories&amp;only=styles&amp;skin=vector" rel="stylesheet"/>
<link href="/w/load.php?lang=en&amp;modules=site.styles&amp;only=styles&amp;skin=vector" rel="stylesheet"/>
<meta content="MediaWiki 1.35.0-wmf.30" name="generator"/>
<meta content="origin" name="referrer"/>
<meta content="origin-when-crossorigin" name="referrer"/>
<meta content="origin-when-cross-origin" name="referrer"/>
<link href="/w/index.php?title=Category:Suburbs_of_Bangalore&amp;action=edit" rel="alternate" title="Edit" type="application/x-wiki"/>
<link href="/w/index.php?title=Category:Suburbs_of_Bangalore&amp;action=edit" rel="edit" title="Edit"/>
```

- **Get latitude and longitude coordinates using Geocoder**

Geocoder API was used to retrieve the coordinates (latitude and longitude of each town centers). This Geocoder converts address into coordinates .

3. Get the geographical coordinates

```
address = 'Bangalore, India'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Bangalore City are {}, {}.'.format(latitude, longitude))
```

```
The geograpical coordinate of Bangalore City are 12.9791198, 77.5912997.
```

- **Use Foursquare API to get venue data-**

We use Foursquare API to get the top 100 venues that are within a radius of 2000 meters.We need to register a Foursquare Developer Account in order to obtain the Foursquare ID and Foursquare secretkey. We then make API calls to Foursquare passing in the geographical coordinates of the neighbourhoods in a Python loop. Foursquare will return the venue data in JSON format and we will extract the venue name, venue category, venue latitude and longitude.With the data, we can check how many venues were returned for each neighbourhood and examine how many unique categories can be curated from all the returned venues.

Now, let's get the top 100 venues that are within a radius of 2000 meters.

```
radius = 2000
LIMIT = 100

venues = []

for lat, long, neighborhood in zip(kl_df['Latitude'], kl_df['Longitude'], kl_df['Neighborhood']):

    # create the API request URL
    url = "https://api.foursquare.com/v2/venues/explore?client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}".format(
        CLIENT_ID,
        CLIENT_SECRET,
        VERSION,
        lat,
        long,
        radius,
        LIMIT)

    # make the GET request
    results = requests.get(url).json()["response"]['groups'][0]['items']

    # return only relevant information for each nearby venue
    for venue in results:
        venues.append((
            neighborhood,
            lat,
            long,
            venue['venue']['name'],
            venue['venue']['location']['lat'],
            venue['venue']['location']['lng'],
            venue['venue']['categories'][0]['name']))
```

- **Group data by neighbourhood and taking the mean of the frequency of occurrence of each venue category**

We will analyse each neighbourhood by grouping the rows by neighbourhood and taking the mean of the frequency of occurrence of each venue category.By doing so, we are also preparing the data for use in clustering.

Next, let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

```
kl_grouped = kl_onehot.groupby(["Neighborhoods"]).mean().reset_index()

print(kl_grouped.shape)
kl_grouped
```

```
(57, 115)
```

	Neighborhoods	ATM	Accessories Store	Airport	Arcade	Art Gallery	Art Museum	Arts & Crafts Store	Athletics & Sports	Auto Garage	Auto Workshop	Badminton Court	Basketball Court	Big B Sto
0	Arekere	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.0000	0.000000	0.000000	0.00000	0.0000
1	BEML	0.000000	0.0000	0.000000	0.010000	0.000000	0.00	0.000000	0.010000	0.0000	0.000000	0.000000	0.00000	0.0000
2	Banashankari	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.0000	0.000000	0.000000	0.00000	0.0000
3	Banaswadi	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.020408	0.020408	0.0000	0.000000	0.000000	0.00000	0.0000
4	Basavanagudi	0.000000	0.0000	0.000000	0.010000	0.010000	0.00	0.000000	0.000000	0.0000	0.000000	0.000000	0.00000	0.0000
5	Begur	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.0625	0.000000	0.000000	0.00000	0.0000
6	Bellandur	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.0000	0.000000	0.011364	0.00000	0.0000
7	Bengaluru Pete	0.000000	0.0000	0.000000	0.010000	0.010000	0.00	0.010000	0.010000	0.0000	0.000000	0.000000	0.00000	0.0000
8	Bidadi	0.000000	0.0000	0.000000	0.010000	0.010000	0.00	0.010000	0.010000	0.0000	0.000000	0.000000	0.00000	0.0000
9	Bommasandra	0.000000	0.0000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.0000	0.000000	0.000000	0.00000	0.0000

- **Filter venue category by Eatery**

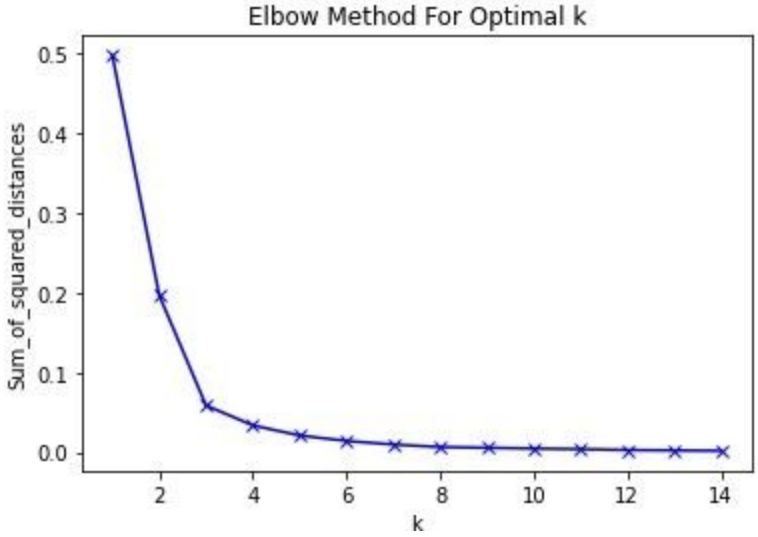
Since we are analysing the “Eatery” data,we will filter the “Eatery” as venue category for the neighbourhoods.

Create a new DataFrame for Eatery data only

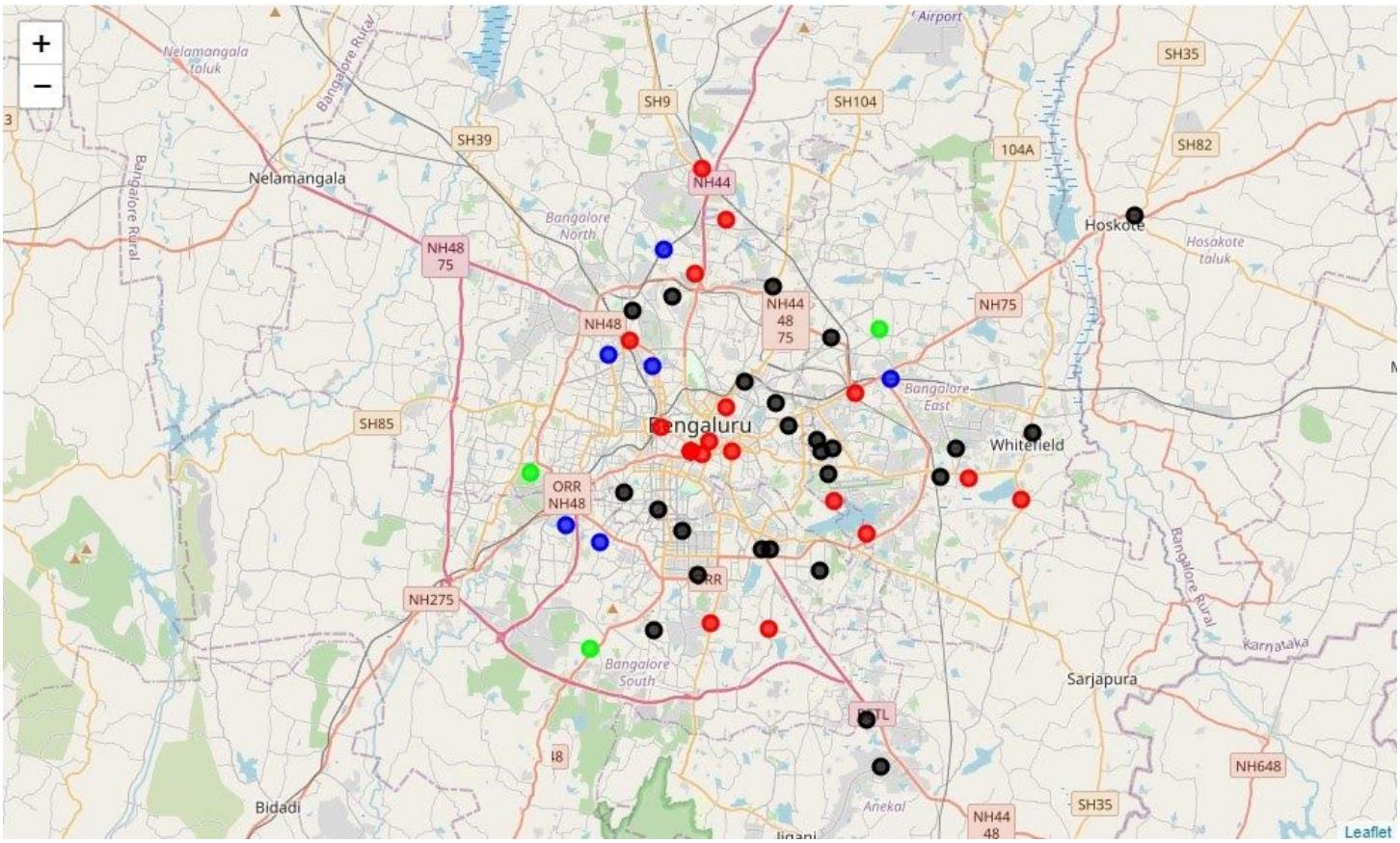
```
kl_mall = kl_grouped[["Neighborhoods","Eatery"]]
kl_mall
```

	Neighborhoods	Eatery
0	Arekere	0.710526
1	BEML	0.730000
2	Banashankari	0.500000
3	Banaswadi	0.877551
4	Basavanagudi	0.820000
5	Begur	0.750000
6	Bellandur	0.727273
7	Bengaluru Pete	0.720000
8	Bidadi	0.720000
9	Bommasandra	0.857143
10	Brigade Road	0.730000
12	Devanahalli	0.833333
13	Dhobi Ghat	0.720000
14	Domlur	0.769231

- **<u>Perform clustering on the data by using k-means clustering</u>** -
  Neighborhood K-Means clustering based on mean occurrence of venue category : To cluster the neighborhoods into ‘n’ clusters we used the K-Means clustering Algorithm. K-means clustering aims to partition ‘’n” observations into k clusters in which each observation belongs to the cluster with the nearest mean. It uses iterative refinement approach. Here elbow method is used to find the value of K.



- **Visualize the clusters in a map using Folium**

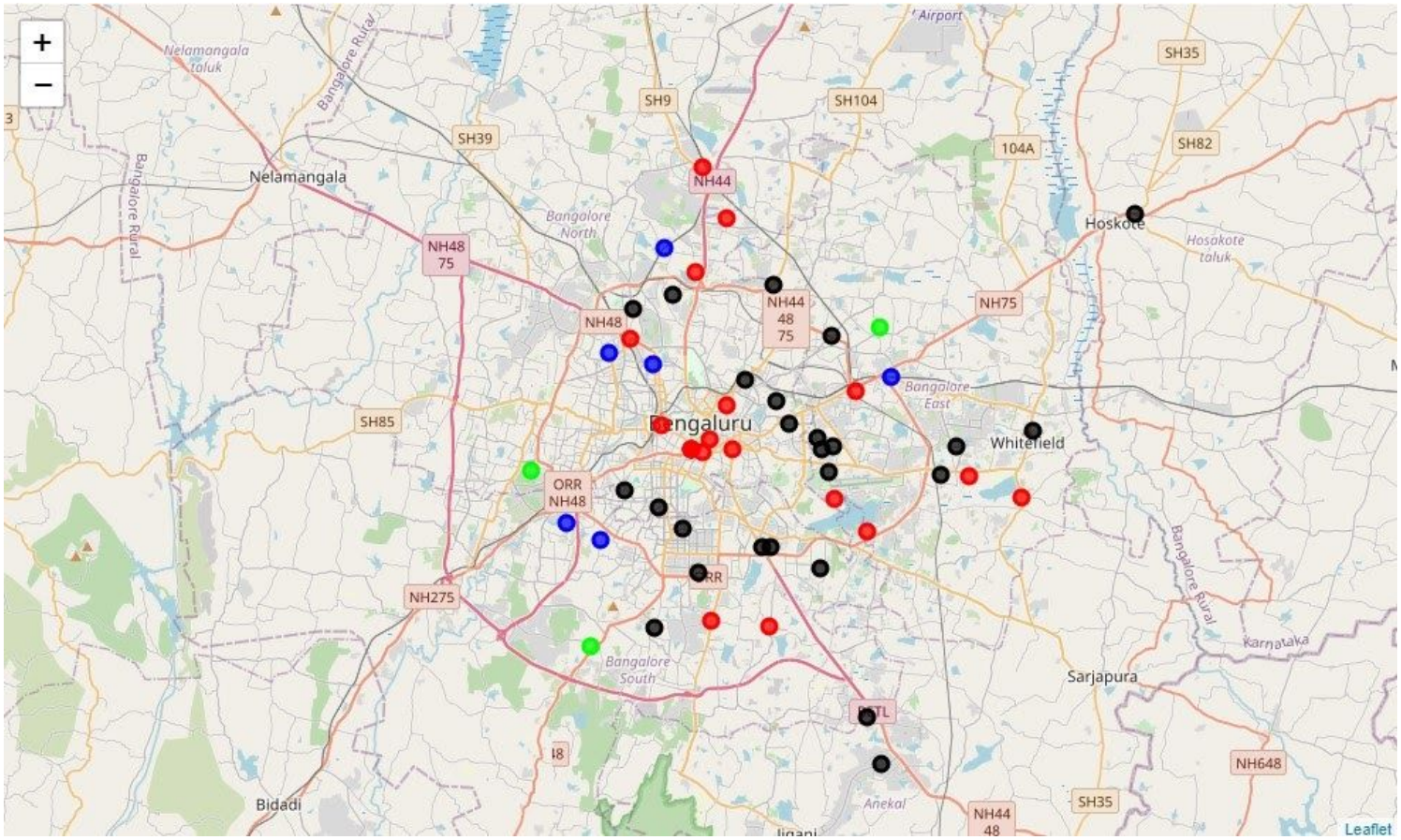


RESULTS

The results from the k-means clustering show that we can categorize the neighbourhoods into 4 clusters based on the frequency of occurrence for “Restaurants”:

- Cluster 0: Neighbourhoods with average concentration of Restaurants
- Cluster 1: Neighbourhoods with highest concentration of Restaurants
- Cluster 2: Neighbourhoods with moderately high concentration of Restaurants
- Cluster 3: Neighbourhoods with lowest concentration of Restaurants

The results of the clustering are visualized in the map below with cluster 0 in black colour, cluster 1 in red colour, cluster 2 green colour ,cluster 3 in blue color.



Discussion

- As the observations noted from the map in the Results section, most of the Restaurants are concentrated in the central area of Bangalore city, with the highest number in cluster 1 and moderately high number in cluster 2.
- On the other hand, cluster 3 has very low concentration in the neighbourhoods. This represents a great opportunity and high potential areas to open new Restaurants as there is very little to no competition from existing Restaurants.
- Meanwhile, Restaurants in cluster 1 are likely suffering from intense competition due to oversupply and high concentration of Restaurants.
- From another perspective, the results also show that the oversupply of Restaurants mostly happened in the central area of the city, with the suburb area have very few Restaurants.
- Therefore, this project recommends Hotel Owners to capitalize on these findings to open new Restaurants in neighbourhoods in cluster 3 with little to no competition.
- Entrepreneurs with unique selling propositions to stand out from the competition can also open new Restaurants in neighbourhoods in cluster 0 with moderate competition.

- Lastly, Entrepreneurs are advised to avoid neighbourhoods in cluster 2 which already have high concentration of Restaurants and suffering from intense competition..

Limitations and Suggestions for Future Research

- In this project, we only consider one factor i.e. frequency of occurrence of restaurants, there are other factors such as population and income of residents that could influence the location decision of a new restaurant. However, to the best knowledge of this researcher such data are not available to the neighbourhood level required by this project.
- Future research could devise a methodology to estimate such data to be used in the clustering algorithm to determine the preferred locations to open a Hotel.
- In addition, this project made use of the free Sandbox Tier Account of Foursquare API that came with limitations as to the number of API calls and results returned. Future research could make use of paid account to bypass these limitations and obtain more results.

Conclusion

- In this project, we have gone through the process of identifying the business problem, specifying the data required, extracting and preparing the data, performing machine learning by clustering the data into 4 clusters based on their similarities, and lastly providing recommendations to the relevant stakeholders i.e. Potential Restaurant Owners and investors regarding the best locations to open a new Restaurant
- Entrepreneurs can capitalize on these findings to open new Restaurants in neighborhoods in cluster 3 with little to no competition.
- Entrepreneurs with unique selling propositions to stand out from the competition can also open new Restaurants in neighborhoods in cluster 0 or cluster 2 with moderate competition.
- Lastly, Entrepreneurs are advised to avoid neighborhoods in cluster 2 which already have high concentration of Eateries and suffering from intense competition.

The findings of this project will help the relevant stakeholders to capitalize on the opportunities on high potential locations while avoiding overcrowded areas in their decisions to open a new Restaurant.

THANK YOU!