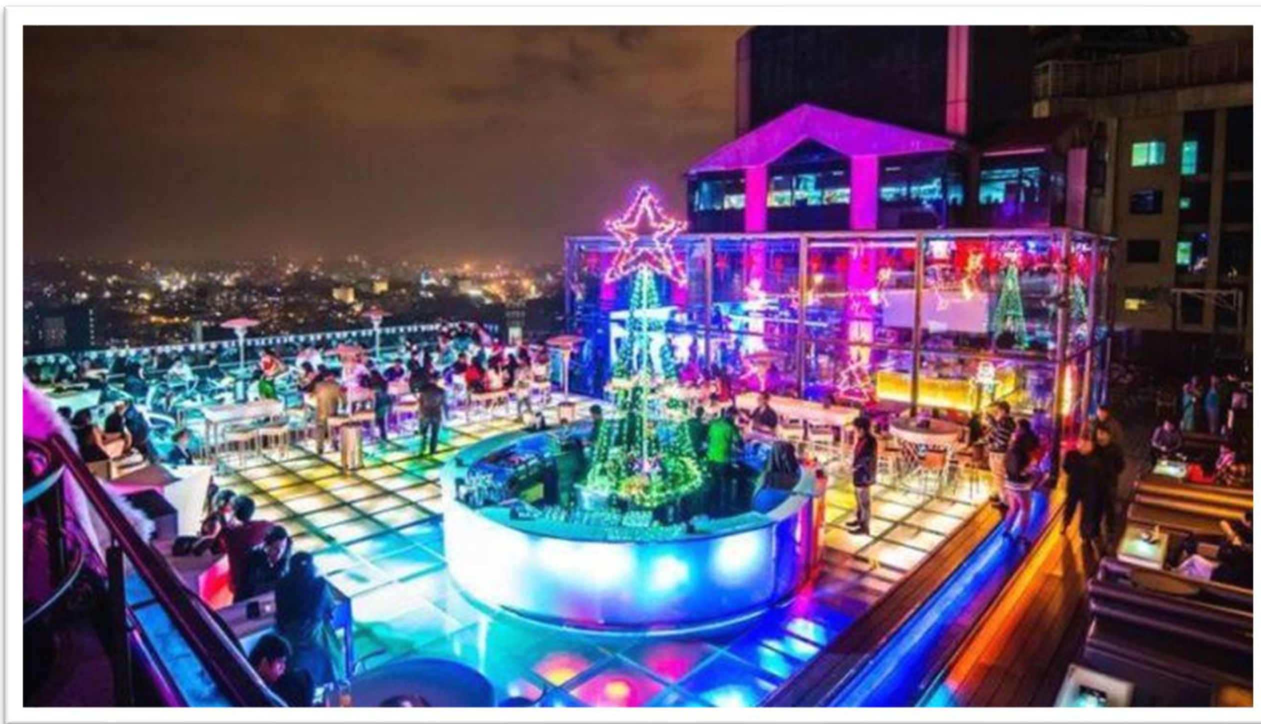
COURSERA CAPSTONE

IBM Data Science Professional Certificate

Clustering Areas of Bangalore on the basis of number of Night Life Venues in the Area

By: Manash Pratim Borah



1. *Introduction:*

Bangalore, the capital of the state of Karnataka, is not only known as the ‘Silicon Valley of India’ but it is also the ‘Pub Capital of India. Ever since the IT boom in India, Bangalore has been flooded with people from all of India and abroad and with the advent of this young energetic group of individuals, the night life of Bangalore grew, with the establishment of numerous clubs, pubs and restaurants.

With the hope of riding the wave of young people’s advent into the city, most of the venues were constructed near the areas where population of young and office going people were high. But with time there has been a saturation in the market as many business’ have made their mark in the night life industry with multiple chains around the city like SOCIAL, House of Commons, Truffles, Sherlock’s etc.

For someone who wants to start a restaurant or other night life venues, choice of location is very critical as in can make or break them. This project can help with this problem as it will cluster different neighborhoods of the city based on the number of venues already available.

1.1 Business Problem:

The objective here is to use the KMeans algorithm to cluster the different areas based on the number of night life venues present in them. The algorithm will help to differentiate areas with High, Medium and Low density of such venues.

It can help a businessman or anyone who wants to build such venues take a small step forward by atleast showing them the areas where they can get the lion’s share of profit instead of stepping into a place where it is already crowded with old established brands

1.2 Target Audience:

This project is particularly useful to property developers and investors looking to open or invest in new night life projects in the pub capital of India. It may also be helpful to anyone who considers night life venues as a considerable factor to where they stay or where they hang out.

2. Data:

For this project, we needed:

- List of all of the areas of Bangalore
    - Sources:
        - https://pincode.net.in/
        - https://www.indiatvnews.com/pincode/karnataka/bangalore/bangalore-city
- The area's respective Latitude and Longitude
    - Using the Geopy Library
- List of all the venues within a certain radius of the coordinates
    - Using the FourSquare API

2.1 Data Collection:

It wasn't a direct process to get the areas along with their coordinates as it isn't available on the internet, so we first get a list of all the areas in Bangalore and then by the geocoder API we try to get their respective coordinates. There might be certain minor flaws as the exact coordinates might be difficult to get via the API and in certain examples, I had to manually change the coordinates to their correct ones.

1. First, we make a csv file with all the areas of Bangalore we could find from the websites mentioned above, I did it simply by copying the

	A	B	C
1	Area		
2	A F Station Yelahanka		
3	Agram		
4	Amruthahalli		
5	Anandnagar Bangalore		
6	Arabic College		
7	Attur		
8	Austin Town		
9	Banaswadi		
10	Bangalore International Airport		
11	Bangalore Sub Foreign Post		
12	Bellandur		
13	Benson Town		
14	Bhattarahalli		
15	C.V.Raman Nagar		
16	CMM Court Complex		
17	Devanagundi		
18	Devasandra		
19	Doddagubbi		

2. We then read this file in Jupyter using Pandas

```
In [9]: 1 blr_df.head()
```

Out[9]:

	Area
0	A F Station Yelahanka
1	Agram
2	Amruthahalli
3	Anandnagar Bangalore
4	Arabic College

3. We then use the geocoder API to get the coordinates of all the respective areas.
4. We then convert the coordinates into a dataframe and copy the columns of this dataframe into the main dataframe.

```
In [19]: 1 blr_df.head()
```

Out[19]:

	Area	Latitude	Longitude
0	A F Station Yelahanka	13.12682	77.610660
1	Agram	12.99840	77.571690
2	Amruthahalli	13.06684	77.595100
3	Anandnagar Bangalore	12.96348	77.702020
4	Arabic College	13.03344	77.619345

5. We then store this to a csv file and read that csv as the main csv for the project.
6. After getting this lot of data another data that had to be collected were the venues around the areas of Bangalore, that was done using the Foursquare API.
7. For the foursquare API we have to enter our CLIENT_ID and CLIENT_SECRET.
8. We then have to send a GET request to their server and they'll return a json file and the entries have been appended to a list called 'venues'.

Out[8]:

	Area	Latitude	Longitude	VenueName	VenueLatitude	VenueLongitude	VenueCategory
0	A F Station Yelahanka	13.12682	77.61066	Cafe Potenza	13.121925	77.623036	Café
1	A F Station Yelahanka	13.12682	77.61066	cafe coffee day	13.145366	77.617906	Coffee Shop
2	A F Station Yelahanka	13.12682	77.61066	Cafe Coffee Day	13.094997	77.597301	Café
3	A F Station Yelahanka	13.12682	77.61066	Cafe Coffee Day	13.099390	77.588282	Café
4	A F Station Yelahanka	13.12682	77.61066	A2B restaurant	13.152166	77.620648	Indian Restaurant

```
In [9]: 1 venues_df.shape
Out[9]: (6911, 7)
```

9. There are 204 unique venue categories

In [10]:
```
print('There are {} uniques categories.'.format(venues_df['VenueCategory'].nunique()))
```

```
There are 204 uniques categories.
```

In [11]:
```
#List of all the Categories
venues_df['VenueCategory'].unique()[:-1]
```

Out[11]:
```
array(['Café', 'Coffee Shop', 'Indian Restaurant', 'Clothing Store',
       'American Restaurant', 'Train Station', 'Smoke Shop', 'Food Truck',
       'Vegetarian / Vegan Restaurant', 'Fast Food Restaurant',
       'South Indian Restaurant', 'Golf Course', 'Ice Cream Shop',
       'Department Store', 'Resort', 'Snack Place', 'Art Gallery', 'Gym',
       'Bakery', 'Hotel', 'Karnataka Restaurant', 'Shopping Mall',
       'Food & Drink Shop', 'Multiplex', 'Gym / Fitness Center',
       'Steakhouse', 'Bowling Alley', 'Racetrack', 'Tea Room', 'Lounge',
       'Movie Theater', 'Pub', 'Seafood Restaurant',
       'Monument / Landmark', 'Donut Shop', 'Park', 'Italian Restaurant',
       'Electronics Store', 'Burger Joint', 'Cricket Ground',
       'French Restaurant', 'Bar', 'Mexican Restaurant',
       'Japanese Restaurant', 'Motorcycle Shop', 'Asian Restaurant',
       'Cupcake Shop', 'Gas Station', 'Boutique', 'Bistro',
       'Chinese Restaurant', 'Thai Restaurant', 'Brewery',
       'Sandwich Place', 'Pizza Place', 'Bubble Tea Shop', 'Building',
       'Dessert Shop', 'Flea Market', 'Airport', 'Lake', 'Bus Station',
       'Badminton Court', 'Trail', 'Soccer Field', 'Office', 'BBQ Joint',
       'Andhra Restaurant', 'Creperie', 'Restaurant',
       'Sporting Goods Shop', 'Toy / Game Store', 'Recreation Center',
       'Kerala Restaurant', "Men's Store", 'History Museum',
       'Maharashtrian Restaurant', 'Salon / Barbershop',
       'Fried Chicken Joint', 'Breakfast Spot', 'Farmers Market',
       'Performing Arts Venue', 'Mediterranean Restaurant', 'Sports Bar',
       'Athletics & Sports', 'Furniture / Home Store',
```

10. Now we have all the data that is necessary, after this we’ll be cleaning and feature engineering as per the problem

3. Methodology:

1. For the first step we have to one-hot encode our dataframe to get categorical data into numerical.

```
(6911, 205)
```

Out[12]:

	Area	ATM	Accessories Store	Afghan Restaurant	Airport	Airport Service	Airport Terminal	American Restaurant	Andhra Restaurant	Arcade	...	Trail	Train Station	Udupi Restaurant	Vegetari / Vega Restaura
0	A F Station Yelahanka	0	0	0	0	0	0	0	0	0	...	0	0	0	
1	A F Station Yelahanka	0	0	0	0	0	0	0	0	0	...	0	0	0	

2. Then we group the dataframe on the 'Area' column and then take the mean of all the entries.

```
(101, 205)
```

Out[13]:

	Area	ATM	Accessories Store	Afghan Restaurant	Airport	Airport Service	Airport Terminal	American Restaurant	Andhra Restaurant	Arcade	...	Trail	Train Station	Udupi Restaurant	Vegetarian / Vegan Restaurant	V F
0	A F Station Yelahanka	0.0	0.0	0.0	0.000000	0.0	0.0	0.052632	0.00	0.0	...	0.000000	0.052632	0.0	0.052632	
1	Adugodi	0.0	0.0	0.0	0.000000	0.0	0.0	0.010000	0.01	0.0	...	0.000000	0.000000	0.0	0.010000	
2	Agara	0.0	0.0	0.0	0.000000	0.0	0.0	0.020000	0.00	0.0	...	0.010000	0.000000	0.0	0.000000	
3	Agram	0.0	0.0	0.0	0.000000	0.0	0.0	0.000000	0.00	0.0	...	0.000000	0.000000	0.0	0.010000	
4	Amruthahalli	0.0	0.0	0.0	0.018182	0.0	0.0	0.018182	0.00	0.0	...	0.018182	0.000000	0.0	0.000000	

5 rows × 205 columns

3. From the list of categories, we now select all the necessary ones, as our problem statement includes night life venues, we select all the night life related categories we could find and create a new dataframe.

Select particular columns based on scenario

In [14]:
```
blr_night = blr_grouped[["Area",'Brewery','Lounge',"Nightclub", "Pub", "Sports Bar", "Gastropub", "Bistro", "Beer Bar", "Coc
```

In [15]:
```
blr_night.head()
```

Out[15]:

	Area	Brewery	Lounge	Nightclub	Pub	Sports Bar	Gastropub	Bistro	Beer Bar	Cocktail Bar	Bar	Beer Garden
0	A F Station Yelahanka	0.000000	0.00	0.0	0.00	0.00	0.0	0.00	0.0	0.00	0.00	0.0
1	Adugodi	0.040000	0.06	0.0	0.03	0.00	0.0	0.01	0.0	0.01	0.01	0.0
2	Agara	0.010000	0.03	0.0	0.02	0.01	0.0	0.00	0.0	0.00	0.01	0.0
3	Agram	0.010000	0.05	0.0	0.01	0.00	0.0	0.01	0.0	0.00	0.02	0.0
4	Amruthahalli	0.018182	0.00	0.0	0.00	0.00	0.0	0.00	0.0	0.00	0.00	0.0

4. We now make use of K-Means algorithm to cluster the areas into 4 clusters. We selected 4 as we could find better results that were much better interpretable. We also add the cluster labels from the labels_ returned from kmeans.

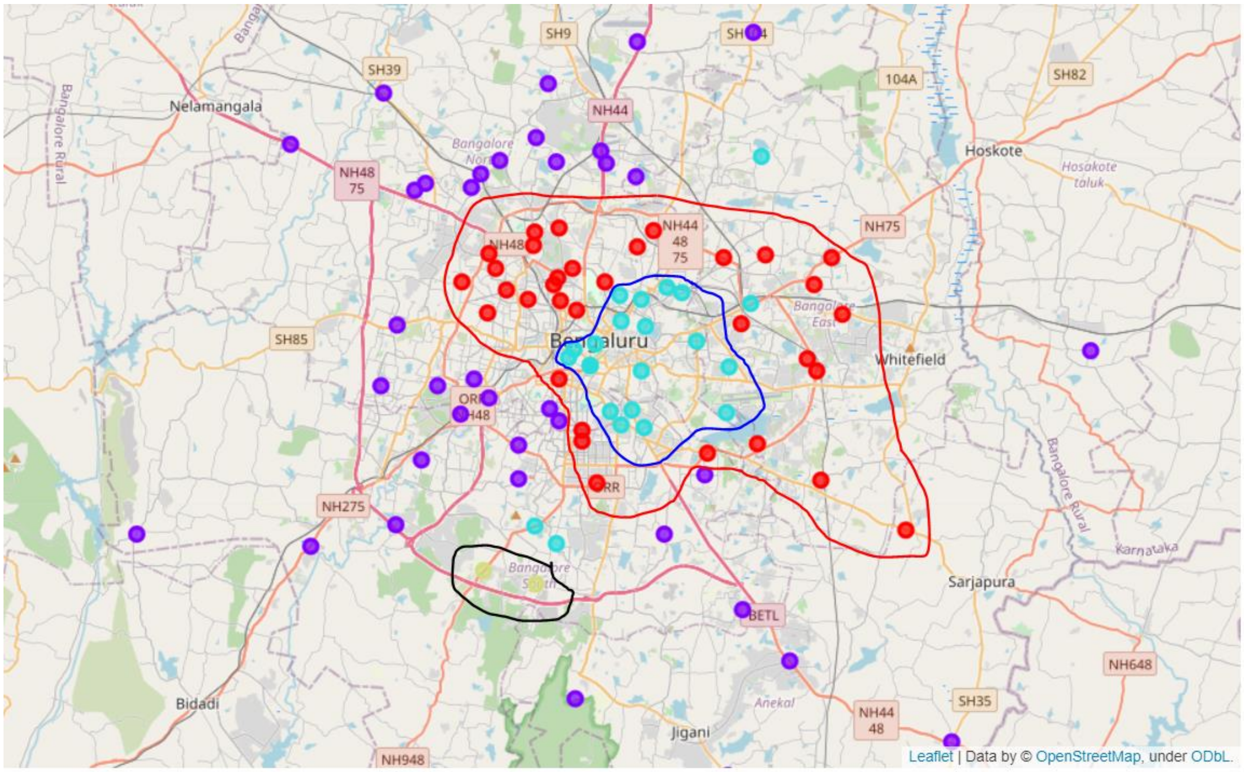
Out[18]:

	Area	Brewery	Lounge	Nightclub	Pub	Sports Bar	Gastropub	Bistro	Beer Bar	Cocktail Bar	Bar	Beer Garden	Cluster Labels
0	A F Station Yelahanka	0.000000	0.00	0.0	0.00	0.00	0.0	0.00	0.0	0.00	0.00	0.0	1
1	Adugodi	0.040000	0.06	0.0	0.03	0.00	0.0	0.01	0.0	0.01	0.01	0.0	2
2	Agara	0.010000	0.03	0.0	0.02	0.01	0.0	0.00	0.0	0.00	0.01	0.0	0
3	Agram	0.010000	0.05	0.0	0.01	0.00	0.0	0.01	0.0	0.00	0.02	0.0	0
4	Amruthahalli	0.018182	0.00	0.0	0.00	0.00	0.0	0.00	0.0	0.00	0.00	0.0	1

5. After this we merge it the foremost dataframe to get the coordinates of all the areas.

4. Result:

For the results we will plot the clusters on the Bangalore map using folium library.



As from this map we can clearly say that there is an interior cluster, a mid-outer cluster, an outer cluster and a small cluster towards the lower left of the map.

11. The inner cluster of cyan dots represent – Cluster 2
12. The mid-outer cluster of red dots represent – Cluster 0
13. The outer cluster of violet dots represent – Cluster 1
14. The small cluster towards lower left of map represent – Cluster 3

4.1 Ranking by score:

To make the analysis easier we make a 'score' by summing up half the value of the one-hot encoded columns for each cluster.

Observations:

In [44]:
```
print('The score of cluster_0 ranges from {} to {}'.format(min(cluster_0['Score']),max(cluster_0['Score'])))
print('The score of cluster_1 ranges from {} to {}'.format(min(cluster_1['Score']),max(cluster_1['Score'])))
print('The score of cluster_2 ranges from {} to {}'.format(min(cluster_2['Score']),max(cluster_2['Score'])))
print('The score of cluster_3 ranges from {} to {}'.format(min(cluster_3['Score']),max(cluster_3['Score'])))
```

```
The score of cluster_0 ranges from 0.060000000000000005 to 0.14
The score of cluster_1 ranges from 0.0 to 0.0625
The score of cluster_2 ranges from 0.07692307692307693 to 0.16000000000000003
The score of cluster_3 ranges from 0.1 to 0.1111111111111111
```

This shows that if we consider the maximum score of all clusters.

The ranking will be sort of like this

Cluster_2 > Cluster_0 > Cluster_1 > Cluster_3

1. As we know the inner cluster is represented by cluster _2 which also has the highest max score, which inferences that the inner cluster has maximum number of night life venues.
2. The mid-outer cluster represented by cluster_0 has the second highest max score as it is not that far from the hotspots.
3. Cluster_1 is third as it is farther away from the city but more venues than cluster_3
4. Cluster_3 has the least with lowest number of venues

4.2 Cluster Wise Analyzing

Cluster_2:

In [45]:
```
#List of all areas in cluster_2
cluster_2['Area']
```

Out[45]:
```
73                   H.K.P. Road
74                   Austin Town
75             Dharmaram College
76                   Doddagubbi
77                     Chickpet
78             Science Institute
79                       Domlur
80          Sivan Chetty Gardens
81                  Fraser Town
82                      Adugodi
83             Doddakallasandra
84             Sampangiramnagar
85                   Hampinagar
86             Maruthi Sevanagar
87     Bangalore Dist Offices Bldg
88      Bangalore Sub Foreign Post
89               Lingarajapuram
90                         EPIP
91                  Benson Town
92                  Koramangala
93              Jeevanbhimanagar
94      Bnagalore Viswavidalaya
95                       Bolare
96                  Indiranagar
97                          NAL
98                Doorvaninagar
Name: Area, dtype: object
```

Cluster_2 has areas like Koramangala, Indiranagar which are hotspots of night life activity.

In [46]:
```python
#Picking up a random area from cluster 2
c2_koramangala = venues_df[(venues_df['Area']=='Koramangala') & ((venues_df['VenueCategory']=='Lounge')|(venues_df['VenueCat
```

In [47]:
```python
len(c2_koramangala)
```

Out[47]: 14

In [48]:
```python
#Picking up the area with highest score in cluster 2
c2_indiranagar = venues_df[(venues_df['Area']=='Indiranagar') & ((venues_df['VenueCategory']=='Lounge')|(venues_df['VenueCat
```

In [49]:
```python
len(c2_indiranagar)
```

Out[49]: 16

We can also check this out by the number of venues that comes under these 2 areas with Koramangala having 14 venues and Indiranagar having 16 venues which ranks among the most number of venues.

Cluster_0:

In [50]:
```python
#List of areas in cluster_0
cluster_0['Area']
```

Out[50]:
```
0                  Yeshwanthpur Bazar
1                  Basaveshwaranagar
2                            Laggere
3                        Dommasandra
4                      Bhattarahalli
5                  Jayangar III Block
6                          Jayanagar
7                      C.V.Raman Nagar
8                          Carmelram
9                         Chamrajpet
10                         J.C.Nagar
11                         J P Nagar
12          ISRO Anthariksha Bhavan
13                             Hoodi
14                        Devasandra
15                     H.A.L II Stage
16                     Gayathrinagar
17                    Doddanekkundi
18                         Bapagrama
19          Mahalakshmipuram Layout
20                         Bellandur
21    Bangalore International Airport
22                     Arabic College
23                        Anandnagar
24                       Malleswaram
25          P&T Col. Kavalbyrasandra
26                    Sadashivanagar
27                     Nandinilayout
28                             Agram
29                         Mathikere
30                         Banaswadi
31                 Malleswaram West
32                             Agara
```

These are some of the areas in cluster_0

In [52]:
```python
#Picking up area with highest score in cluster 0
c0_cvraman = venues_df[(venues_df['Area']=='C.V.Raman Nagar') & ((venues_df['VenueCategory']=='Lounge')|(venues_df['Venue
```

In [53]:
```python
len(c0_cvraman)
```

Out[53]: 14

In [54]:
```python
#Picking up random area in cluster 0
c0_malleswaram =  venues_df[(venues_df['Area']=='Malleswaram') & ((venues_df['VenueCategory']=='Lounge')|(venues_df['Venu
```

In [55]:
```python
len(c0_malleswaram)
```

Out[55]: 8

We can also verify the score of cluster_0 by taking the area with the highest score among all in cluster_0 and a random area from the same cluster and seeing that the number of venues is quiet high but not as high as cluster_2.

Cluster_1:

```
In [56]:  1  #List of areas in cluster_1
          2  cluster_1['Area']

Out[56]: 34           Gaviopuram Extension
         35                Ullalu Upanagar
         36                 Tarabanahalli
         37                    HSR Layout
         38                     Haragadde
         39                      G.K.V.K.
         40                 Jalahalli East
         41                    Rv Niketan
         42             Peenya Dasarahalli
         43                 Jalahalli West
         44                  Nayandahalli
         45                    Nagarbhavi
         46                   Kodigehalli
         47                   Kumbalagodu
         48                   Magadi Road
         49                     Jalahalli
         50               Electronics City
         51          A F Station Yelahanka
         52                 Vidyaranyapura
         53                  Amruthahalli
         54                        Anekal
         55                    Ashoknagar
         56                      Attibele
         57                         Attur
         58                    Bagalgunte
         59                       Bagalur
         60          Banashankari III Stage
         61                  Bannerghatta
         62                  Basavanagudi
         63                   Bettahalsur
```

These are some of the areas in cluster_1

In [58]:
```
#Picking up area with highest score in cluster 1
c1_nagarbhavi = venues_df[(venues_df['Area']=='Nagarbhavi') & ((venues_df['VenueCategory
```

In [59]:
```
len(c1_nagarbhavi)
```

Out[59]: 3

In [61]:
```
#Picking up random area in cluster 1
c1_begur = c0_cvraman = venues_df[(venues_df['Area']=='Begur') & ((venues_df['VenueCatego
```

In [62]:
```
len(c1_begur)
```

Out[62]: 4

As we see these areas have lesser number of venues than cluster_0 and cluster_2.

Cluster_3:

In [63]:
```
#List of areas in cluster_3
cluster_3['Area']
```

Out[63]:
```
99          Anjanapura
100    Thalaghattapura
Name: Area, dtype: object
```

Cluster_3 has only 2 areas within it and also significantly lesser number of venues.

In [65]:
```
c3_thalaghattapura = venues_df[(venues_df['Area']=='Thalaghattapura') & ((venues_df['VenueCategory']=='Loung
```

In [66]:
```
len(c3_thalaghattapura)
```

Out[66]: 1

In [67]:
```
c3_anjanapura = venues_df[(venues_df['Area']=='Anjanapura') & ((venues_df['VenueCategory']=='Lounge')|(venue
```

In [68]:
```
len(c3_anjanapura)
```

Out[68]: 1

4.3 Final Result:

As verified from the number of venues per cluster taking the highest scoring area and a random area from the cluster, we can say that the clustering is correct as we have predicted that according to number of venues the ranking is:

Cluster_2 > Cluster_0 > Cluster_1 > Cluster_3

Cluster	**Area**	**Number of Venues**
0	CV Raman Nagar	14
	Malleswaram	8
1	Nagarbhavi	3
	Begur	4
2	Koramangala	14
	Indiranagar	16
3	Thalaghattapura	1
	Anjanapura	1

5. Discussion

In this section, I would be discussing the observations I have noted and the recommendation that I can make based on the results.
This analysis is performed on limited data. There may be some discrepancies based on coordinate data. But if good amount of data is available there is scope to come up with better results.

- There is high competition in Cluster_2 so it is very risky to open business in these areas and is probably a saturated market with brands with strong hold on customer loyalty.
- Cluster_0 has potential as it is not as saturated as cluster_2 but not as far from the city heart as cluster_1.
- Cluster_1 is a tricky call to make as with the ever growing population of a city like Bangalore with major traffic issues, many people who are about the age of 35+ and well settled financially are moving away from the city to more peaceful areas, so it wouldn’t be a bad choice to set up a branch sub chain or a major hub where people can come out of the city as a weekend trip.
- Cluster_3 is not a very good call as far as I can tell but if there is a influx of people there than it would be a great place to set up a new business as it is almost untouched.

6. Conclusion

What we wanted to achieve from this project was met and we could cluster Bangalore into potential clusters for the target audience and we clearly showed which cluster is a saturated market and which is not and also showed clusters that had future potential.

For future branching of this project into bigger projects we can first of all try to get more accurate coordinate data and also use other data like demographic data that includes population, income, age-group etc. to find out even better clusters and could be a full-fledged application with a bit of hardwork and patience.