# Finding Similar Neighborhoods in different cities

Capstone project for Coursera

**Himank Kavathekar**

June, 2021

# Introduction

Relocating to a new city is a stressful task. People are used to a certain level of comfort that come from living in a known city, neighborhood. They know nearby places; like shops, restaurants or parks, etc. So when it comes to leaving that comfort and move to a completely unknown city, it is better to have some help to explore the new city. This project demonstrates how one can leverage the data available for the city to find a similar neighborhood to their current place of residence.

For the demonstration purposes, some specific locations are chosen in this project. Target audience is people that are looking to move from and to those specific locations. These people can directly use the findings of this project to their advantage, without any changes. But one can easily use the same methodology to analyse any other locations they see fit. The purpose of this project is to demonstrate the method in which one can use data to their advantage and make the process of moving to a new city a little easier.

In this project, we will look for a neighborhood similar to 'Baner, Pune, IN' in Banglore and Delhi.

# Data

We make use of the following data in this project:

1. List of neighborhoods in Bangalore, IN
2. List of Neighborhoods in Delhi, IN
3. Geological coordinates of all the neighborhoods
4. Nearby venues like restaurants, cafes, etc. in the above mentioned neighborhoods and in Baner, Pune.

We can get the neighborhoods data in Bangalore and Delhi from wikipedia. Using web-scraping and cleaning the data, we get fairly exhaustive lists of the neighborhoods in both these cities. Using python's geocoder package, we obtain the coordinates of these places. For nearby venues, we make use of the *Foursquare API.*

The foursquare API consists of venues data collected from over 100k sources and arranged according to the venue category. For any given place, one can explore upto 100 nearby venues in desired radius from the location. E.g. for Baner, Pune, we use foursquare api to get a list of nearby venues in within a distance of 750 meters. This list includes restaurants, cafes, fast food chains, etc. One can also explore ratings of venue given by Foursquare's city guide users.

# Methodology

The goal for this project is to compare different neighborhoods with each other. We will do this by comparing the nearby venues in the neighborhoods obtained from the Foursquare API. We are going to compare the neighborhoods in Bangalore and Delhi with a particular neighborhood- Baner- in Pune, IN.

The cities of Bangalore and Delhi are divided into suburbs according to the geography of the area, within which several neighborhoods are present. We will include these suburbs in our analysis, so that one can find a similar neighborhood to Baner in desired part of the town.

We start off by getting a list of neighborhoods in Bangalore and Delhi. This information is easily available on wikipedia. Web scraping these pages with python's requests and BeautifulSoup packages gives us these lists. To use foursquare API, we need to pass the geographical co-ordinates of the neighborhoods as argument to get information about nearby venues. Python's geopy package is an excellent tool to get these co-ordinates. Using this package, we obtain all the geographical information we need about the neighborhoods and combine all of it in a dataframe along with suburb and city, for future use.

Next, we will use the Foursquare API to get nearby venues in each of these neighborhoods. Foursquare API returns the list of nearby venues in json format. We will extract necessary information from these json data and populate a python pandas dataframe with it. This will include the venue category.

As we want to compare these neighborhoods using clustering, we will first combine all the neighborhoods data in one single dataframe. Here the city column will be useful to distinguish between neighborhoods and venues in different cities. We will be using the venue category for clustering the neighborhoods. As clustering algorithm is only applicable to numerical values, we will create dummy variables for each venue category in consideration. Then we will group this dataframe by neighborhoods and take the mean of frequency of occurence of each venue category in that neighborhood. This gives us the final data for clustering the neighborhoods.

Finally we use k-means clustering to create clusters of similar neighborhoods. K-means identifies k centroids and assigns each data point to one centroid based on euclidean distance. The centroids are then updated to minimize total distances in clusters, and then repeats the process. We will segregate our data into 5 clusters. We will be looking for neighborhoods in the same cluster as Baner, IN. These will be the most similar neighborhoods to Baner based on the nearby venues.

# Results

After dividing the neighborhoods in 5 clusters, we see that 39 neighborhoods in Bangalore and 35 neighborhoods in Delhi are similar to Bnaer in terms of local venues.

We further notice that Central Bangalore has the most number of neighborhoods similar to Baner with 10 neighborhoods in the same cluster. Whereas in case of Delhi, South Delhi has the most number of neighborhoods in the same cluster as Baner. South Delhi also has 10 neighborhoods similar to Baner.

**Neighborhoods in Bangalore similar to Baner are:**

Arekere, BTM Layout, Banashankari, Banaswadi, Bangalore Cantonment, Basavanagudi, Bengaluru Market, Bommanahalli, Bommasandra, C. V. Raman Nagar, Domlur, Electronic City, Gottigere, HSR Layout, Hebbal, Hoodi, Horamavu, Indiranagar, J. P. Nagar, Jalahalli, Jayanagar, Jeevan Bhima Nagar, Kammanahalli, Kengeri, Koramangala, Kothnur, Kumaraswamy Layout, Madiwala, Mahalakshmi Layout, Malleswaram, Marathahalli, Nagarbhavi, Padmanabhanagar, Rajajinagar, Sadashivanagar, Seshadripuram, Shivajinagar, Vasanth Nagar, Vijayanagar

**Neighborhoods in Delhi similar to Baner are:**

Adarsh Nagar, Barakhamba Road, Chanakyapuri, Chandni Chowk, Chattarpur, Chittaranjan Park, Connaught Place, Daryaganj, Defence Colony, East Vinod Nagar, Fateh Nagar, Gole Market, Green Park, Gulabi Bagh, Hauz Khas, Khanpur, Mayur Vihar, Moti Bagh, Munirka, Naraina Vihar, Nizamuddin East, Nizamuddin West, Okhla, Palam, Paschim Vihar, Patel Nagar, Pitam Pura, Safdarjung Enclave, Sainik Farm, Saket, Sarai Rohilla, Sarojini Nagar, South Extension, Sriniwaspuri, Vikaspuri

# Discussion

At this point, it is important to mention that the results obtained in this project are limited by the fact that Foursquare API has limited number of venues listed for Indian cities. We try to minimize this limitation by only including venue categories that appear more than 10 times in all the neighborhoods considered combined. How the results change with change in this threshold number is worth studying.

Another feature of this analysis is that it only tells which neighborhoods are similar to a given neighborhood in terms of local venues like shops, restaurants etc. One might want to look at more features while moving to a new place like prices of houses or crime stats. In that case, one can build up on the current methods to first filter similar neighborhoods and then perform further analysis on these neighborhoods to get the desired results.

# Conclusion

In this project we have gone through the process of identifying a problem, specifying the sources of data required and collecting it to solve said problem and performing clustering to find similar neighborhoods. Lastly we display our methodology and results to find neighborhoods similar to Baner, Pune, IN in Bangalore, IN and Delhi, IN. The findings of this project will help anyone who is looking to move from and to said locations. The methodology used can be helpful to anyone who wants to carry out similar analysis.