Base station locations for bike share start-ups in Bengaluru

# Introduction

## 1.1 Background

Public transport has come a long way in India. From the early days of buses and trains, to the current age of metro mass transit systems and on-demand taxis like *Ola*, *Uber*, the transport options have expanded significantly and improved connectivity like never before. However, last mile connectivity transport is still a pain point due to relatively higher prices of on demand taxis and non-availability of other forms of transport. Navigating traffic in the much congested Indian metro cities is also a huge pain point for the modern commuter. To address such problems, a slew of new start-ups like *Bounce* and *Vogo* have come up in India, starting with the Indian start-up hub of Bengaluru. They provide IoT enabled bikes or scooters at various locations throughout a city. The user just has to download an app and use it to ‘unlock’ the bike and start their ride. Upon reaching their destination, they use the app to ‘lock’ the bike again. The need addressed by these start-ups is evident from the fact that *Bounce* became one of the fastest growing start-ups in India, clocking up to 60,000 rides a day.

## 1.2 The Problem

The problem at hand for these start-ups is deciding on the locations for the base stations at which to place their bikes. While *Bounce* operates on a dockless model i.e. the bikes can be parked anywhere as they have a GPS tracker and are IoT enabled to show their location in real time to prospective users, they still have base stations at which the bikes can be picked up. Foursquare location data can be very useful in identifying spots for where to have the base stations in a given locality. This can be done by looking at important mass transit points like metro stations and bus stations from where most users either take these bikes or arrive with these bikes. I will perform this exercise for the Indian metro city of Bengaluru.

# Data

## 2.2 Data Sources

Data on the localities and their pin codes for the city of Bengaluru were obtained from this link [here](https://www.mapsofindia.com/pincode/india/karnataka/bangalore/). The co-ordinates for these localities and their boundaries were acquired using Google Maps API for geocoding. A dataframe was made containing the pincode, corresponding localities under that pincode, and geographical co-ordinates for that pincode.

Using the co-ordinates of each pincode from the dataframe, the area was explored for major transit points like bus stations and metro stations. The co-ordinates of these transit points were then used in a K-Means clustering algorithm with the number of clusters set to 10, indicating 10 base stations per pincode. Using the co-ordinates of the centres of these clusters, the addresses of the locations which can be used as base stations were obtained by reverse geocoding using Google Maps API for each pincode.This was done for the all pincodes containing at least 20 transit points.

## 2.2 Data Cleaning

Data was downloaded in the form of a table available on the website. The data contained in the table was *Location, Pincode, State* and *District*. The data I needed was areas corresponding to each pincode, and the geographical co-ordinates for each pincode and its boundaries.

First, I grouped the table by *Pincode*, dropping out the *State* and *District* columns. The resulting table now only had pincodes and combined list of areas belonging to each pincode.

Next, using the pincodes in Google Maps API, the geographical co-ordinates for the pincodes and their boundaries or bounds were obtained. Bounds are the geographical co-ordinates of North-East and South-West points of an area, enclosing the area in a rectangle. The bounds are needed so that they can be used in the Foursquare API to restrict searches to a particular geographical area.



Figure - The dataframe *df* with the columns of Pincode, Location, Geographical Co-ordinates and their bounds

The dataframe obtained after performing the above two steps was ready for further analysis.

# Methodology

## 3.1 General Exploration

The dataframe *df* obtained after data cleaning was taken for further exploration. First, I mapped all the pincodes to see their distribution on the map.

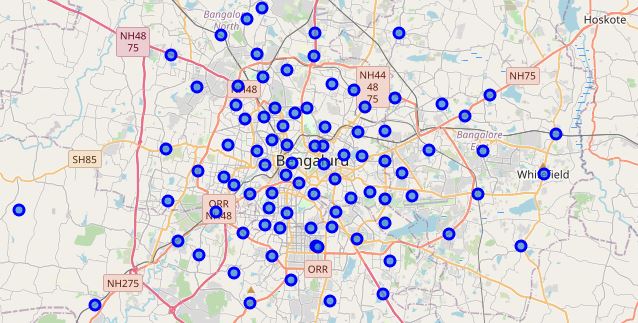


Figure - Map of Bengaluru with blue dots representing pincodes

I also created three functions-

1. *get\_dist*: A function to get the distance between each point in the dataframe(in Kilometre) to every other point once and return it in a list.
2. *show\_hist*: A function to plot the histogram of the points obtained in the last function.
3. *get\_stats*: A function to calculate the average, minimum and maximum distance between the points.

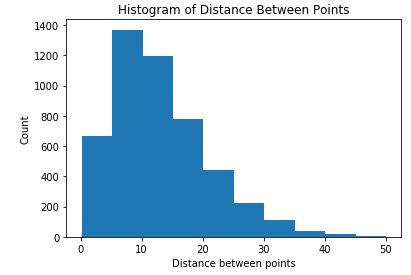
I applied *get\_dist* to *df* and used the output in *show\_hist* and *get\_stats:* 

Figure - Histogram showing the distribution of distances between the pincode areas in Bengaluru



Figure - The average, minimum and maximum distances between the pincode areas in Bengaluru

The first pincode from the dataframe was used, taking the geographical co-ordinates and bounds and feeding it to the Foursquare API. Here I chose the **use case** of people dropping off or taking the bikes from transit points(metro stations and bus stations), from where they would be proceeding on their onward journey. This was for last mile connectivity. Therefore, I explored the pincode to get the maximum possible number of transit points.

Using the Foursquare API and the bounds of the first pincode in the dataset, I got 50 transit points. The output of the API call was then stored as a dataset containing the locations of the 50 transit points and their co-ordinates.

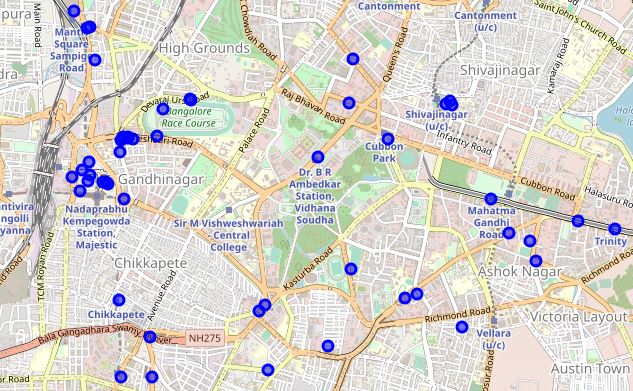


Figure - Map of Pincode 560001 showing all transit points as blue dots

The distribution of the points, as seen on the map, was such that there was higher density at some parts of the area and relatively less at the others. Now, I needed to find the best possible locations for base stations based on how the points were distributed in the area in terms of density of points. If sections of the map had higher number of points, the centres of those sections were to be chosen as locations for base stations. For such density based clustering, K means is generally the option used and I used it to cluster the points into 10 clusters i.e K=10. The no. of clusters here had to be chosen and not arrived at by inflection point method because the requirement here was to have a certain number of base stations in a locality. The cluster centres of these clusters were then taken as base station locations and plotted.

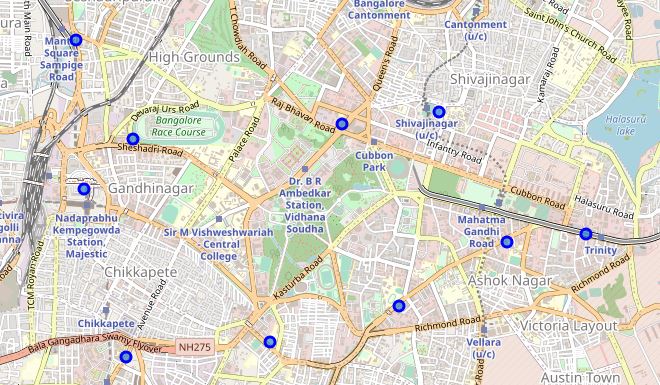


Figure - Map of cluster centres in pincode 560001(purported base station locations)

## 3.2 Solving the problem

Having addressed the problem for a single pincode in Bengaluru, scope was widened to cover the entire city. Using a while loop, I applied Foursquare API to find out no. of transit points for each pincode. I set the criteria that at least 20 transit points had to be present in that area for further analysis. I obtained 19 areas which had at least 20 transit points. I created a dataframe to house these data points.

Then, functions were created to solve the problem for all the pincodes. Three functions were created to be applied to the dataset obtained in the step above:

1. *Points*: The first function, which has dataframe and index as arguments, was used to get the co-ordinates corresponding to the respective pincode.
2. *Kmeans\_clustering*: The second function, which has dataframe as an argument, applied K means clustering to the data points in the dataframe and returns dataframe containing the cluster centres.
3. *Address:* The third function, which also has a dataframe as an argument, converts geographical coordinates to addresses using Google Maps API and returns dataframe containing both the coordinates and addresses as output .**The addresses are an approximation which matches the co-ordinates with the closest possible address**.

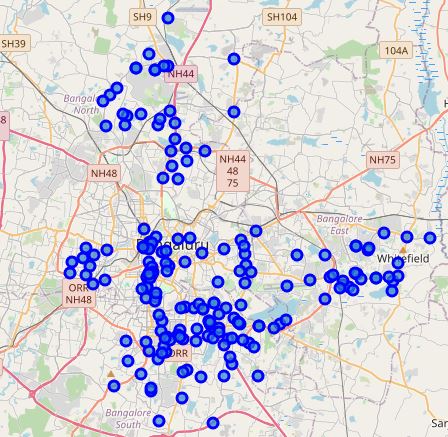
Next, the above functions were used in a While loop, looping over the index of the dataframe. The results obtained were appended to a dataframe. The dataframe,*df\_final1* which now contained geographical co-ordinates and addresses of all purported base stations, was then mapped to see the distribution of data points.

Figure - Map of Bengaluru with all the purported base stations mapped out ( *df\_final1*)

# Results

The dataframe obtained above, on mapping showed that some of the data points are overlapping and there are data points which are too close. On using the *get\_distance* and then *get\_stats* on *df\_final1*,I got the following results:



That the minimum distance was zero indicated that there were some duplications among the locations. Also, some of the locations were too close to each other. On using *dataframe.nunique* function, the duplicate locations were found.

Hence, to have an even distribution of base stations, I removed all duplicate locations using *dataframe.duplicated* method. The locations which are too close to each other were removed by looping over the entire dataframe comparing the distance between each point with every other point. Here I eliminated the points which were closer than 1 kilometre i.e, if two points were closer than 1 km, one of the points was dropped and the other was kept. They were put into a dataframe,*df\_final*

**

Figure 8- The first 5 rows of *df\_final* showing the address of base stations, their co-ordinates and pincodes

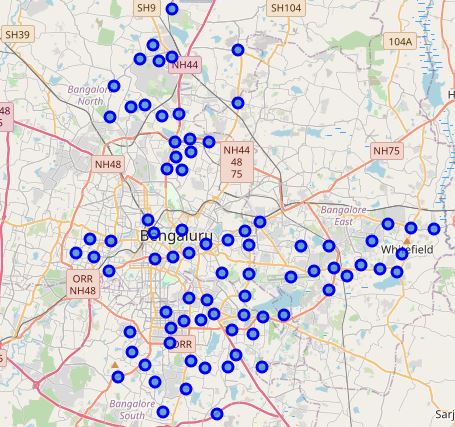


Figure - Map of Bengaluru with purported base stations (*df\_final*)

Now that the final set of base stations was obtained, I looked at the distribution of distances and stats for *df\_final*.

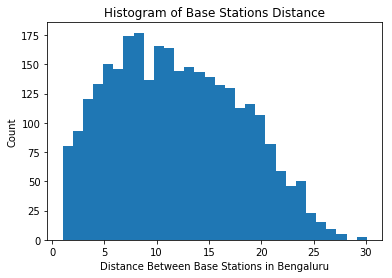


Figure 10-Histogram showing distance between base stations



As can be seen, the average distance in between the base stations is 11.49 km and the maximum distance is 30 km. Then, I created a boxplot using using the data obtained from *get\_dist:*

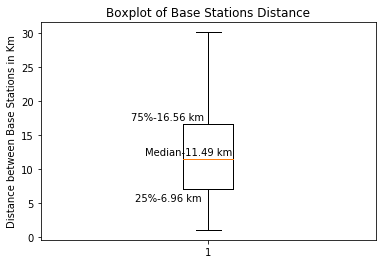


Figure 11- Boxplot of Base Stations Distance

From the boxplot, it could be seen that most of the distances (75%) were below 16.56 km. A quarter of the distances were below 6.96 km and the median distance was 11.49 km.

Then, I went on to look at the number of base stations per pincode. I grouped *df\_final*  by pincode and made a bar chart of the results:

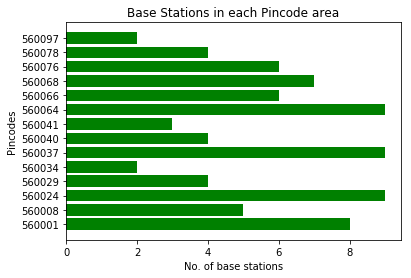


Figure 12- Barchart of No. of Base Stations Per Pincode

From the grouped data, it was seen that there were 3 pincodes which had the maximum number of base stations, 9, as can be seen in the bar chart above. The average number of base stations per pincode was found to be 5.56.

# Discussion

From the results obtained, it was seen that most of the base stations (75%) were within approximately 17 km of each other and 50% of them were within 11.5 km of each other. The maximum distance between the base stations was found to be 30km. This indicated that most of the base stations were within a 30 km area, while the whole city of Bengaluru spans a distance of 50 km. The span of the distance was limited due to the criteria chosen while conducting K-means clustering where only pincodes which had more than 20 transit points were chosen for clustering with K-means, leading to only getting 19 of the total 99 pin codes in the city. As I did this for only a single use case where a person rents a scooter available at a base station close to a transit point to go on their onward journey, this limited the scope of the problem. The scope could have been widened by using lower number of transit points for clustering or just by clustering separately for areas with lower transit points and adding it on in the final analysis. Another use case which could be explored is in areas where there are fewer transit points which might be underserved areas, which could be helped by having base stations at existing transit points or other prominent locations in the area, so that such areas could be served.

More data could be incorporated into doing the above analysis. While I used just locations of transit points and clustered it based on density, other data like population in the area, use of public transport vs. private transport, availability of public transport in an area, Traffic density in an area. Data like this can be used to decide on the minimum distance to be maintained between base stations and how to optimize the base station locations further. More populous areas could be assigned more base stations. Areas where there is higher usage of private transport can have more base stations to serve the last mile transportation needs or shorter distance transportation needs there better. Areas with higher traffic could favour usage of two wheelers due to improved travel times in traffic, where more base stations could be added.

# Conclusion

In the above exercise, I set out to find the locations of base stations to be set up in areas of Bengaluru city based on number of transit points in a particular area by pincode and applying K-means clustering to that area individually and then consolidating that data on a citywide level.This data could be used with modifications incorporating other data like population in an area, traffic density, use of public transport etc. to get better and more optimized results