MUMBAI SUBURBAN RAIL STATIONS: BUSINESS OPPORTUNITIES

Introduction

Mumbai is a megacity housing 1.25 crore residents. The transportation networks in the city are well developed and majority of the population relies on public transportation for their daily commuting needs. That has resulted in the city having one of the highest rates of public transport share among cities worldwide. The suburban railway is the most used with the lines carrying more than 7.5 million passengers daily on its 390km long network with more than 2,300 train services[1]. The network has 150 stations spanning over 7 lines. However, the infrastructure and facilities at each of the stations are not uniform or equivalent to the no. of users. Through this study, an attempt will be made to understand station characteristics and explore business opportunities.

Business problem

Having consistent customer base is extremely crucial to a business to grow and sustain itself. The travel patterns in Mumbai bring large crowds to its transit stations and thus many businesses choose to base their business to leverage this customer footfall. The known interchange points are an obvious choice. However, seldom suitable real estate at an affordable cost is available in these stations. In such cases, the businesses which are just starting up would look at the following prime questions to choose their location.

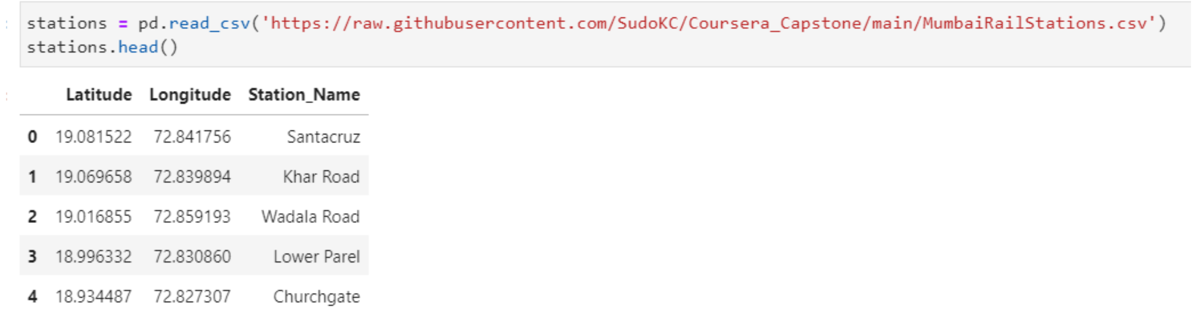
1. Which are the most happening suburban rail stations in Mumbai?
2. What are the prime categories of venues around the stations?
3. Which stations are worthy to host new passenger-oriented businesses?

Data description

Station location data in CSV

The analysis is based on locations of suburban rail transit stations in Mumbai. For the analysis, a selective list of 57 stations which lie within Mumbai district. The data is sourced from Google maps API to get latitudes and longitudes. The same are collated in a csv file to be read as a data frame. The dataset has 3 columns as follows

- Station Name
- Latitude
- Longitude



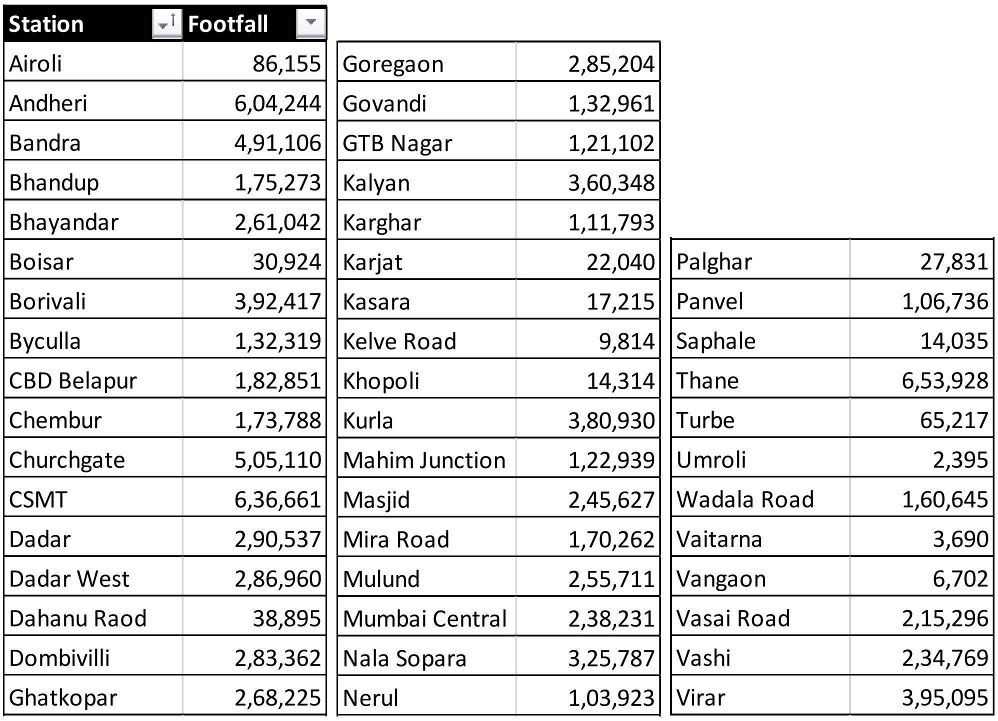
```
stations = pd.read_csv('https://raw.githubusercontent.com/SudoKC/Coursera_Capstone/main/MumbaiRailStations.csv')
stations.head()
```

	Latitude	Longitude	Station_Name
0	19.081522	72.841756	Santacruz
1	19.069658	72.839894	Khar Road
2	19.016855	72.859193	Wadala Road
3	18.996332	72.830860	Lower Parel
4	18.934487	72.827307	Churchgate

[1] https://en.wikipedia.org/wiki/Mumbai_Suburban_Railway

Footfall at the stations

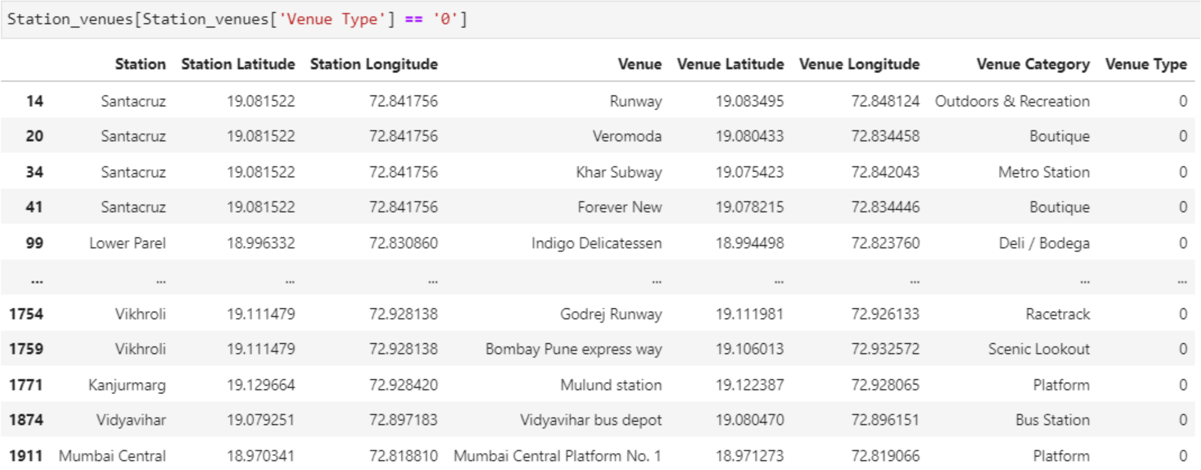
Project reports regarding the footfall at the railway stations will provide a valuable information regarding the footfall at each railway station. The report is available at https://mrvc.indianrailways.gov.in/view_section.jsp?lang=0&id=0,295,465. The data is collated and attached to the stations data to understand the potential of the station to serve as a great business location.



Station	Footfall
Airoli	86,155
Andheri	6,04,244
Bandra	4,91,106
Bhandup	1,75,273
Bhayandar	2,61,042
Boisar	30,924
Borivali	3,92,417
Byculla	1,32,319
CBD Belapur	1,82,851
Chembur	1,73,788
Churchgate	5,05,110
CSMT	6,36,661
Dadar	2,90,537
Dadar West	2,86,960
Dahanu Raod	38,895
Dombivilli	2,83,362
Ghatkopar	2,68,225
Goregaon	2,85,204
Govandi	1,32,961
GTB Nagar	1,21,102
Kalyan	3,60,348
Karghar	1,11,793
Karjat	22,040
Kasara	17,215
Kelve Road	9,814
Khopoli	14,314
Kurla	3,80,930
Mahim Junction	1,22,939
Masjid	2,45,627
Mira Road	1,70,262
Mulund	2,55,711
Mumbai Central	2,38,231
Nala Sopara	3,25,787
Nerul	1,03,923
Palghar	27,831
Panvel	1,06,736
Saphale	14,035
Thane	6,53,928
Turbe	65,217
Umroli	2,395
Wadala Road	1,60,645
Vaitarna	3,690
Vangaon	6,702
Vasai Road	2,15,296
Vashi	2,34,769
Virar	3,95,095

Venues and popularity data

Foursquare location data is leveraged to identify venues and their popularity. This data has helped in identifying types of businesses in the vicinity of the stations as well as their popularity. The data will form a basis for understanding the characteristic of the suburban station.



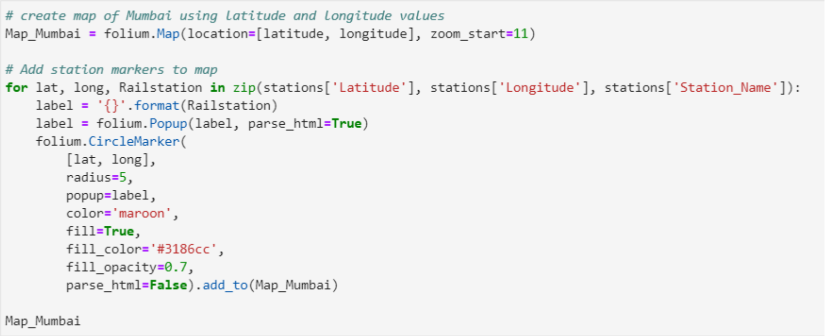
```
Station_venues[Station_venues['Venue Type'] == '0']
```

	Station	Station Latitude	Station Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Venue Type
14	Santacruz	19.081522	72.841756	Runway	19.083495	72.848124	Outdoors & Recreation	0
20	Santacruz	19.081522	72.841756	Veromoda	19.080433	72.834458	Boutique	0
34	Santacruz	19.081522	72.841756	Khar Subway	19.075423	72.842043	Metro Station	0
41	Santacruz	19.081522	72.841756	Forever New	19.078215	72.834446	Boutique	0
99	Lower Parel	18.996332	72.830860	Indigo Delicatessen	18.994498	72.823760	Deli / Bodega	0
...	...	...	...	...	...	...	...	...
1754	Vikhroli	19.111479	72.928138	Godrej Runway	19.111981	72.926133	Racetrack	0
1759	Vikhroli	19.111479	72.928138	Bombay Pune express way	19.106013	72.932572	Scenic Lookout	0
1771	Kanjurmarg	19.129664	72.928420	Mulund station	19.122387	72.928065	Platform	0
1874	Vidyavihar	19.079251	72.897183	Vidyavihar bus depot	19.080470	72.896151	Bus Station	0
1911	Mumbai Central	18.970341	72.818810	Mumbai Central Platform No. 1	18.971273	72.819066	Platform	0

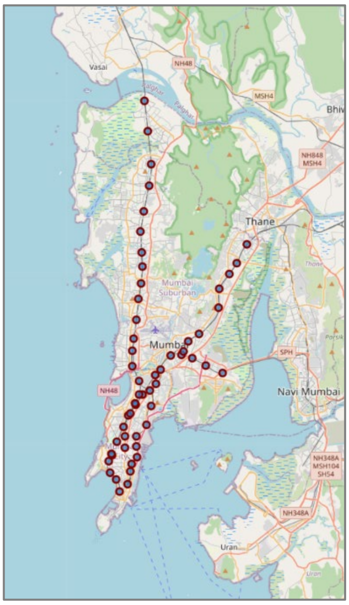
Methodology

Folium

Folium python library allows for visualizing spatial data manipulated in python easily on leaflet maps. All the station cluster visualizations are prepared using the folium library.

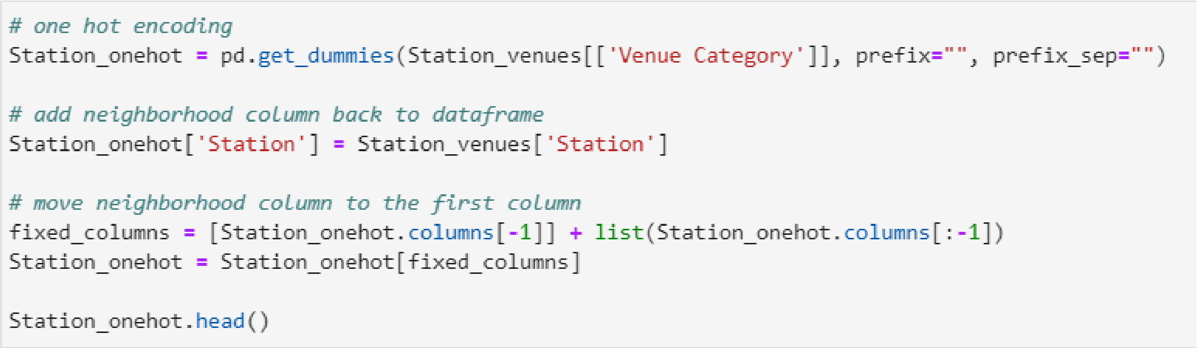


```
# create map of Mumbai using latitude and longitude values
Map_Mumbai = folium.Map(location=[latitude, longitude], zoom_start=11)

# Add station markers to map
for lat, long, Railstation in zip(stations['Latitude'], stations['Longitude'], stations['Station_Name']):
    label = '{}'.format(Railstation)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, long],
        radius=5,
        popup=label,
        color='maroon',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(Map_Mumbai)

Map_Mumbai
```



One hot encoding

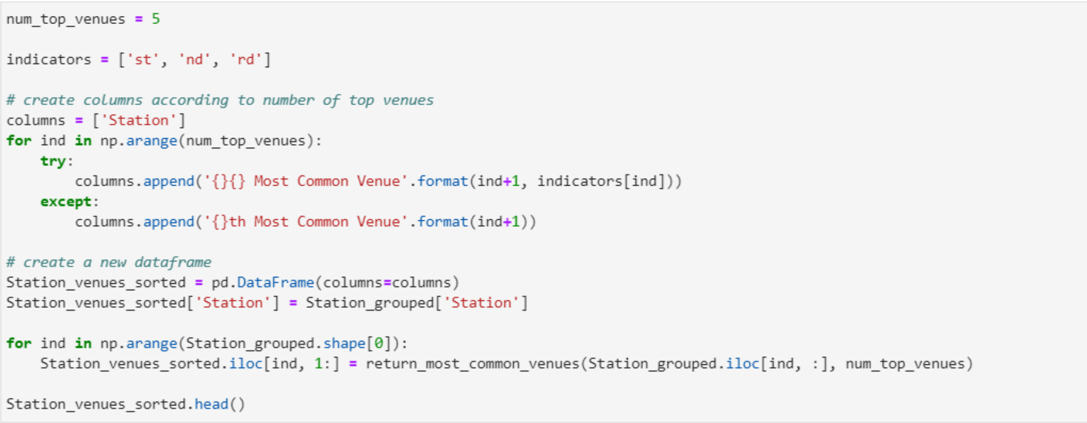
One hot encoding prepares categorical variables in the data suitable to be provided to ML algorithms and improves prediction. In the case of k-means clustering algorithm, unique venue categories in foursquare API data are on-hot encoded.



```
# one hot encoding
Station_onehot = pd.get_dummies(Station_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
Station_onehot['Station'] = Station_venues['Station']

# move neighborhood column to the first column
fixed_columns = [Station_onehot.columns[-1]] + list(Station_onehot.columns[:-1])
Station_onehot = Station_onehot[fixed_columns]

Station_onehot.head()
```

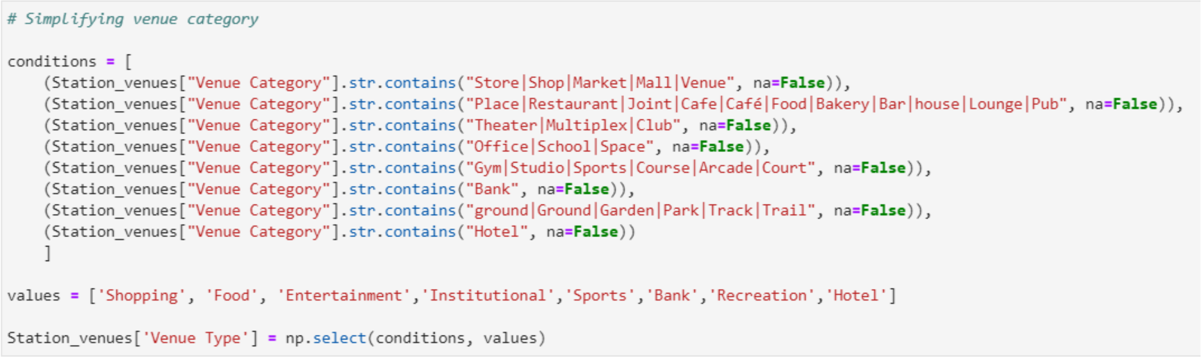
Top 5 most common venues

The foursquare API provided data with 166 unique categories. To reduce the variation among data, only top 5 venues are filters and fed to k-means clustering algorithm.



```
num_top_venues = 5

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Station']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
Station_venues_sorted = pd.DataFrame(columns=columns)
Station_venues_sorted['Station'] = Station_grouped['Station']

for ind in np.arange(Station_grouped.shape[0]):
    Station_venues_sorted.iloc[ind, 1:] = return_most_common_venues(Station_grouped.iloc[ind, :], num_top_venues)

Station_venues_sorted.head()
```

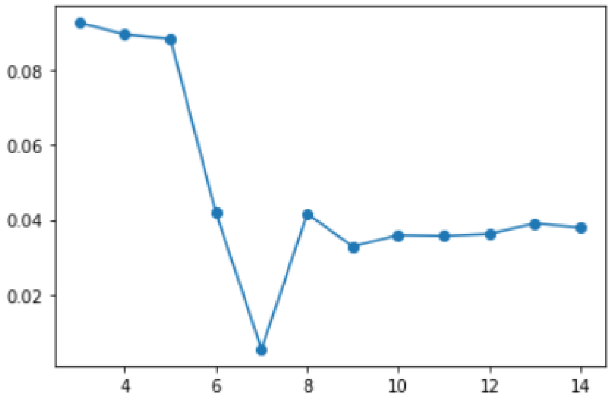
The dimensionality of the venues data was reduced for simplification. This was achieved by grouping similar venue types into venue category column.



```
# Simplifying venue category

conditions = [
    (Station_venues["Venue Category"].str.contains("Store|Shop|Market|Mall|Venue", na=False)),
    (Station_venues["Venue Category"].str.contains("Place|Restaurant|Joint|Cafe|Café|Food|Bakery|Bar|house|Lounge|Pub", na=False)),
    (Station_venues["Venue Category"].str.contains("Theater|Multiplex|Club", na=False)),
    (Station_venues["Venue Category"].str.contains("Office|School|Space", na=False)),
    (Station_venues["Venue Category"].str.contains("Gym|Studio|Sports|Course|Arcade|Court", na=False)),
    (Station_venues["Venue Category"].str.contains("Bank", na=False)),
    (Station_venues["Venue Category"].str.contains("ground|Ground|Garden|Park|Track|Trail", na=False)),
    (Station_venues["Venue Category"].str.contains("Hotel", na=False))
    ]

values = ['Shopping', 'Food', 'Entertainment','Institutional','Sports','Bank','Recreation','Hotel']

Station_venues['Venue Type'] = np.select(conditions, values)
```

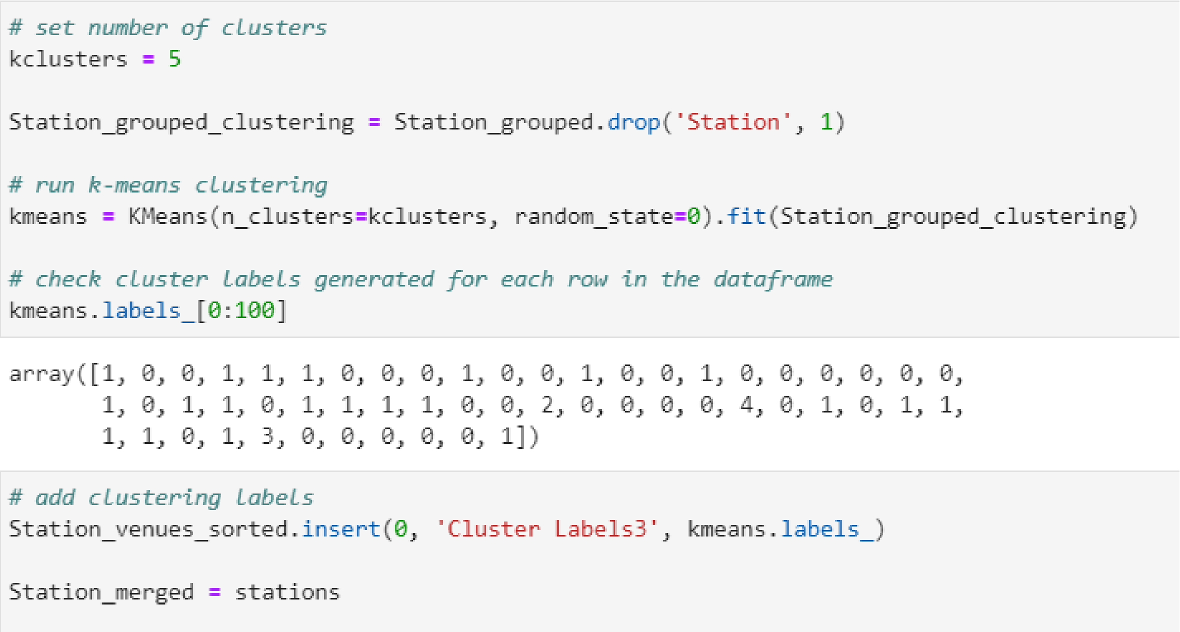
Optimal no. of clusters

Silhouette score is a measure of cohesiveness of the clusters based on similarity among the object of the cluster and the values range between +1 to -1. Higher value indicates that the objects in cluster and cohesive and different to the objects of other clusters. The silhouette score is used to identify optimal cluster size ranging from 2 to 20. The value to 5 clusters was chosen at the score showed a sharp drop for subsequent values.



K-means clustering

The simplified venues dataset was clustered using k-means algorithm. Due to the no. of categories being large in no. k-means algorithm was preferred owing to its efficient handling of these tasks.



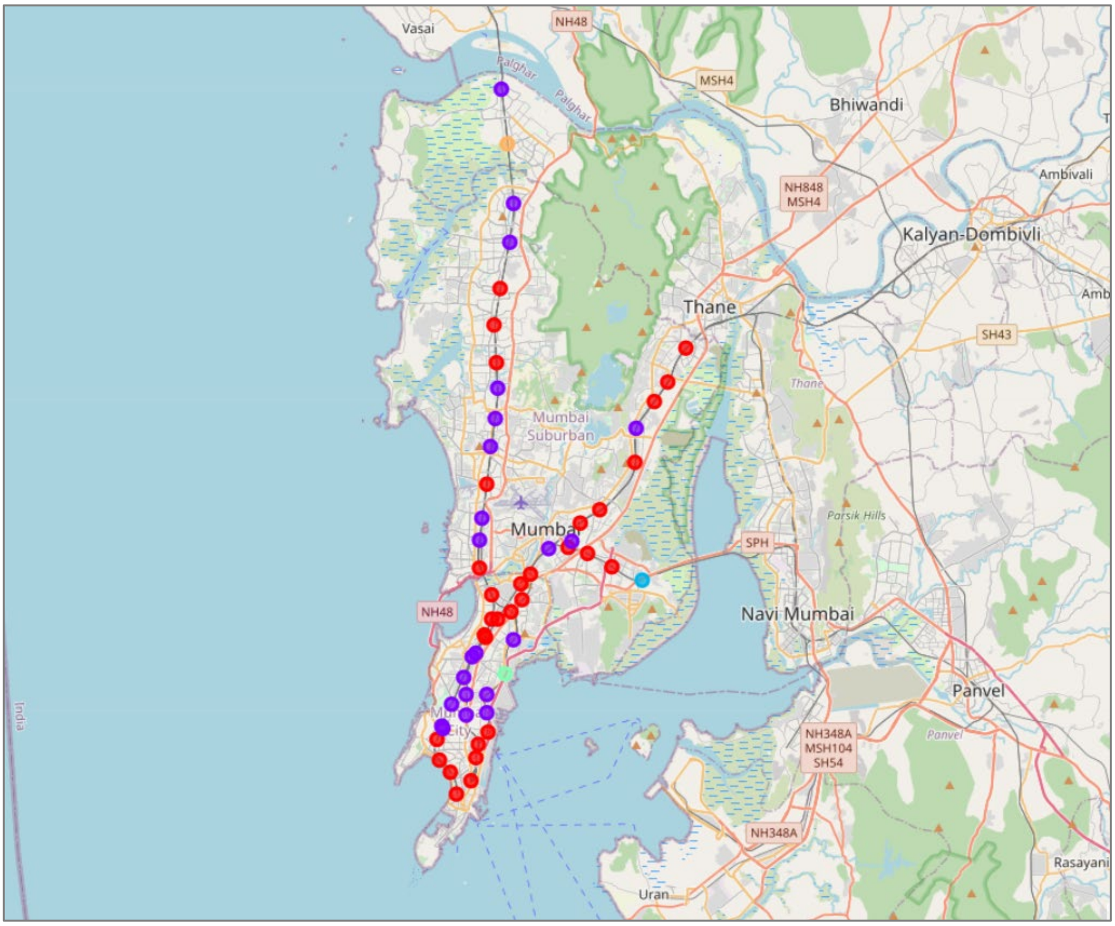
```
# set number of clusters
kclusters = 5

Station_grouped_clustering = Station_grouped.drop('Station', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(Station_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:100]
```

```
array([1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0,
       1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 2, 0, 0, 0, 0, 4, 0, 1, 0, 1, 1,
       1, 1, 0, 1, 3, 0, 0, 0, 0, 0, 1])
```

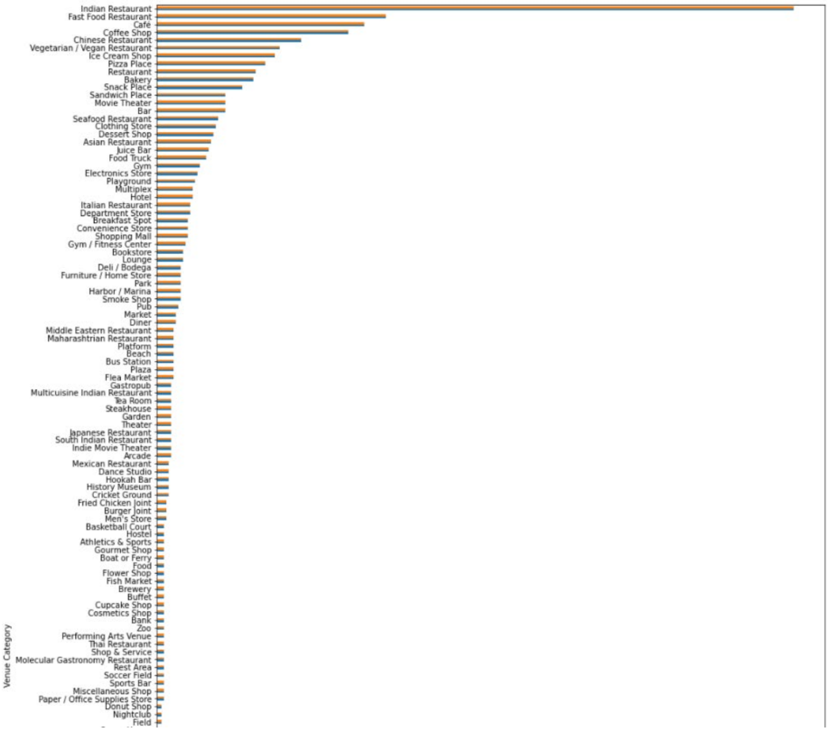
```
# add clustering labels
Station_venues_sorted.insert(0, 'Cluster Labels3', kmeans.labels_)

Station_merged = stations
```

Results

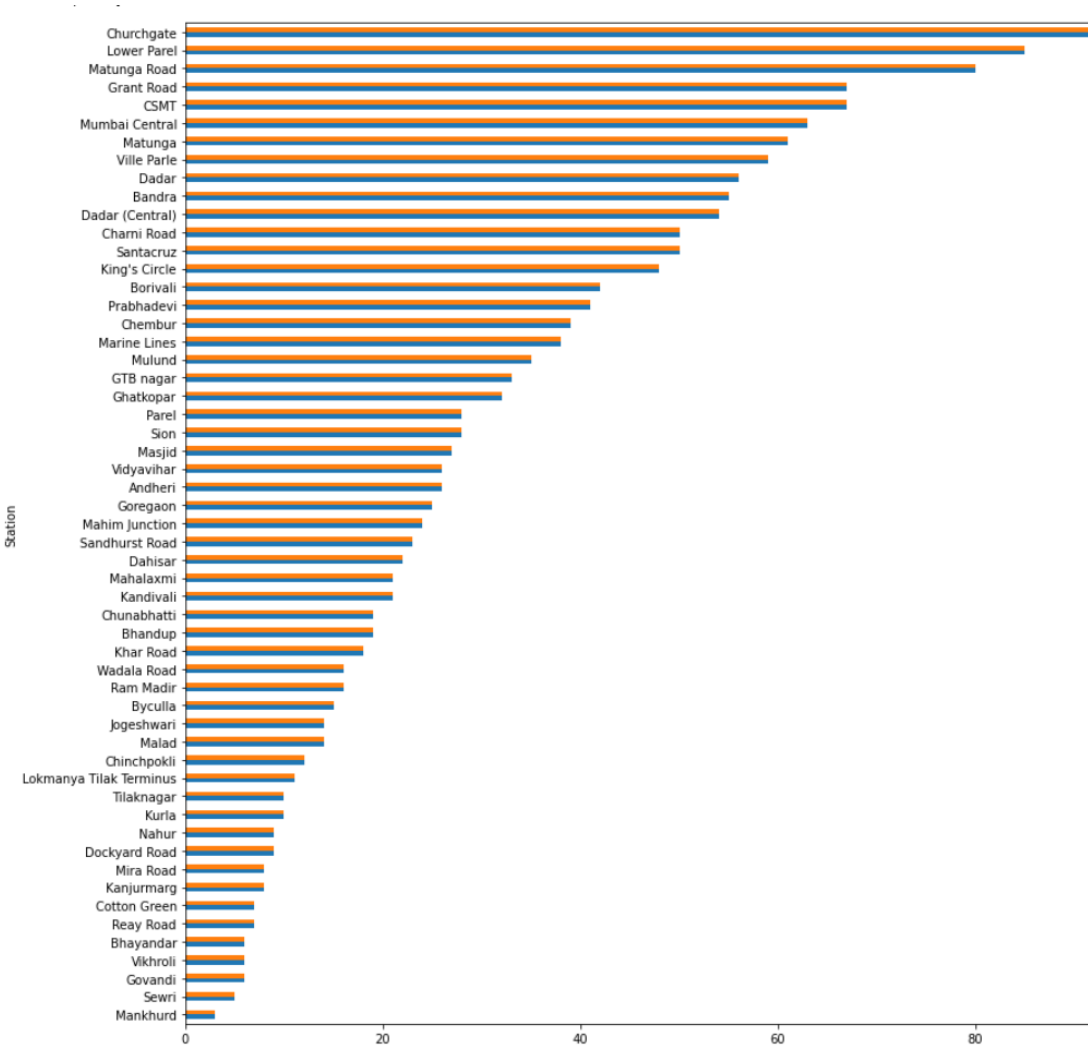
The 5 clusters formed by k-means clustering algorithm shown in the map give a good understanding of station characteristics.



The clusters formed based on Foursquare data provide deep insight into the characteristics of the stations. It is evident that food businesses are at forefront as the most popular businesses at the stations. Indian restaurants and Fast-food restaurants being the most chosen.



The employment hubs of Churchgate and Lower Parel along with Matunga Road came out as the most buzzing stations in the city.



Discussion

The rich dataset available from Foursquare API is a great way to identify development and business potential at the suburban transit stations. These methods are going to be crucial in the post-pandemic world where business decisions will have to be even more cautious. Owing to the limited popularity for Foursquare app in India, it is advisable to enhance the study by user perception survey at the stations.

Conclusion

- Mumbai with is growing suburbs and metro stations being made in the area, can look at stations with high footfall but low count of popular venues to target them as a business opportunity.
- As the food businesses are the most popular businesses, the stations which do not have them as a popular spot pose an opportunity for entrepreneurs to set up a food business at such stations.