IBM Applied D.S Capstone Project

Week 5

# Content in Notebook

. Import libraries

. Scrap data from webpage into a DataFrame

1. Data Preprocessing
2. Output as Prediction file ( .csv)

. Define Foursquare Credentials and Version

. Top 225 venues that are within a radius of 600 meters for each post office

1. Data Preprocessing
2. Output as Prediction file ( .csv)

. Analyze Each Postal Office For Venue Category

. List and display the top 5 existing facilities for each Pin Code

. Exploratory Visualization 1

. Feature Engineering for Business Problem

1. Simplification
2. Feature Selection
3. Handling Categorical Data (One Hot Encoding)

. Potential area for the development of different infrastructure

. Best place to stay with vital infrastructure facilities nearby

. Clustering And Exploratory Visualization 2

. Examine Clusters . Observations: . Acknowledgments

# 1. Import libraries

|  |  |  |
| --- | --- | --- |
|  | | |
|  | **import** numpy **as** np *# library to handle data in a vectorized manner* **import** pandas **as** pd |  |

In [1]:

|  |  |  |
| --- | --- | --- |
|  | **import** folium  **!**conda install -c conda-forge geopy --yes  **from** geopy.geocoders **import** Nominatim *# module to convert an address into*  **!**conda install -c conda-forge lxml --yes **import** requests **import** lxml.html **as** lh **from** sklearn.cluster **import** KMeans print("Libraries imported.") | *l* |
|  |

Collecting package metadata (current\_repodata.json): done

Solving environment: done ## Package Plan ## environment location: /home/jupyterlab/conda/envs/python

added / updated specs:

- geopy

The following packages will be downloaded:

package | build

---------------------------|-----------------

geographiclib-1.50 | py\_0 34 KB conda-forg e

geopy-1.21.0 | py\_0 58 KB conda-forg e

------------------------------------------------------------

Total: 92 KB The following NEW packages will be INSTALLED:

geographiclib conda-forge/noarch::geographiclib-1.50-py\_0 geopy conda-forge/noarch::geopy-1.21.0-py\_0

Downloading and Extracting Packages

geopy-1.21.0 | 58 KB | ##################################### | 1

00%

geographiclib-1.50 | 34 KB | ##################################### | 1

00%

Preparing transaction: done

Verifying transaction: done

Executing transaction: done

Collecting package metadata (current\_repodata.json): done

Solving environment: done ## Package Plan ## environment location: /home/jupyterlab/conda/envs/python

added / updated specs:

- lxml

The following packages will be downloaded:

package | build ---------------------------|-----------------

libxslt-1.1.33 | h7d1a2b0\_0 426 KB lxml-3.8.0 | py36\_0 3.8 MB conda-forg e

------------------------------------------------------------

Total: 4.2 MB The following NEW packages will be INSTALLED:

libxslt pkgs/main/linux-64::libxslt-1.1.33-h7d1a2b0\_0 lxml conda-forge/linux-64::lxml-3.8.0-py36\_0

Downloading and Extracting Packages

lxml-3.8.0 | 3.8 MB | ##################################### | 1

00%

libxslt-1.1.33 | 426 KB | ##################################### | 1

00%

Preparing transaction: done

Verifying transaction: done

Executing transaction: done Libraries imported.

2. Checking & Reading the 'mumbailatlong' file directly to avoid data scraping.

|  |
| --- |
| clean\_df **=** pd**.**read\_csv('mumbailatlong.csv',index\_col**=**'Unnamed: 0') clean\_df**.**head() |

In [2]:

Out[2]: City Post Office Pin Code Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | August Kranti Marg | 400036 | 18.963549 | 72.809989 |
| 1 | Mumbai | Aarey Milk Colony | 400065 | 19.156129 | 72.870722 |
| 2 | Mumbai | Andheri (East) | 400069 | 19.115883 | 72.854202 |
| 3 | Mumbai | Andheri (West) | 400058 | 19.117249 | 72.833968 |
| 4 | Mumbai | Antop Hill | 400037 | 19.020761 | 72.865256 |

# OR Scrap data from webpage into a DataFrame

|  |  |  |
| --- | --- | --- |
|  | | |
|  | url**=**'https://mumbai7.com/postal-codes-in-mumbai/'  *#Create a handle, page, to handle the contents of the website* page **=** requests**.**get(url)  *#Store the contents of the website under doc* doc **=** lh**.**fromstring(page**.**content)  *#Parse data that are stored between <tr>..</tr> of HTML* tr\_elements **=** doc**.**xpath('//tr') r **=** requests**.**get(url)    *#Create empty list* col**=**[] i**=**0  *#For each row, store each first element (header) and an empty list* **for** t **in** tr\_elements[0]: i**+=**1  name**=**t**.**text\_content() print ('%d:"%s"'**%**(i,name)) col**.**append((name,[]))    size\_of\_col **=** 3 **for** j **in** range(1,len(tr\_elements)): |  |

In [27]:

|  |  |  |
| --- | --- | --- |
|  | *#T is our j'th row*  T**=**tr\_elements[j]    *#If row is not of size 4, the //tr data is not from our table*  **if** len(T)**!=**size\_of\_col:  **break**    *#i is the index of our column* i**=**0    *#Iterate through each element of the row* **for** t **in** T**.**iterchildren(): data**=**t**.**text\_content() *#Check if row is empty* **if** i**>**0:  *#Convert any numerical value to integers* **try**:  data**=**int(data) **except**: **pass**  *#Append the data to the empty list of the i'th column* col[i][1]**.**append(data)  *#Increment i for the next column* i**+=**1    Dict**=**{title:column **for** (title,column) **in** col} df**=**pd**.**DataFrame(Dict) df |  |
|  |

1:"City"

2:"Post Office"

3:"Pin Code"

Out[27]: City Post Office Pin Code

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | Mumbai | August Kranti Marg | 400036 |
| 1 | Mumbai | Aarey Milk Colony | 400065 |
| 2 | Mumbai | Andheri (East) | 400069 |
| 3 | Mumbai | Andheri (West) | 400058 |
| 4 | Mumbai | Antop Hill | 400037 |
| ... | ... | ... | ... |
| 152 | Thane | Vasai East I/E | 401208 |
| 153 | Thane | Vasai Road East | 401210 |
| 154 | Thane | Vikramgad | 401605 |
| 155 | Thane | Virar | 401303 |
| 156 | Thane | Wagle Industrial Estate | 400604 |

157 rows × 3 columns

# 3. Finding the latitude and longitude of each zip code

|  |
| --- |
| df["Latitude"] **=** "" df["Longitude"] **=** "" df**.**shape |

In [28]:

Out[28]:

(157

,

5)

C

i

ty

P

ost

O

ffice

P

i

n

C

o

de

L

a

t

i

tu

de

L

on

gi

tu

de

0

M

um

bai

A

u

g

ust

K

r

a

nt

i

M

a

r

g

400036

df

**.**

head

()

In [29]:

Out[29]:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | Mumbai | Aarey Milk Colony | 400065 |
| 2 | Mumbai | Andheri (East) | 400069 |
| 3 | Mumbai | Andheri (West) | 400058 |
| 4 | Mumbai | Antop Hill | 400037 |

In [30]: df1 **=** df**.**loc[0:25] df2 **=** df**.**loc[26:50] df3 **=** df**.**loc[51:75] df4 **=** df**.**loc[76:100] df5 **=** df**.**loc[101:125] df6 **=** df**.**loc[126:150] df7 **=** df**.**loc[151:156]

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* to\_drop\_unknown **=** [] geolocator **=** Nominatim(user\_agent**=**"ny\_explorer") **for** index, row **in** df1**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df1**.**loc[index, 'Latitude'] **=** latitude df1**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [31]:

The geograpical coordinate of August Kranti Marg, Mumbai are 18.9635489, 72. 8099885.

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/pandas/core/i ndexing.py:966: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/ stable/user\_guide/indexing.html#returning-a-view-versus-a-copy self.obj[item] = s

The geograpical coordinate of Aarey Milk Colony, Mumbai are 19.1561292, 72.8 707223.

The geograpical coordinate of Andheri (East), Mumbai are 19.1158835, 72.8542 02.

The geograpical coordinate of Andheri (West), Mumbai are 19.1172495, 72.8339

68.

The geograpical coordinate of Antop Hill, Mumbai are 19.0207608, 72.8652556.

Cannot do: Anu Shakti Nagar, Mumbai, will drop index: 5

Cannot do: B A R C, Mumbai, will drop index: 6

The geograpical coordinate of Ballard Estate, Mumbai are 18.9366512, 72.8391

325.

The geograpical coordinate of Bandra (East), Mumbai are 19.0616565, 72.84981 11.

The geograpical coordinate of Bandra (West), Mumbai are 19.0583358, 72.83026 69.

The geograpical coordinate of Bangur Nagar, Mumbai are 19.1688142, 72.833677 7.

The geograpical coordinate of Barve Nagar, Mumbai are 19.0952831, 72.900177 6.

The geograpical coordinate of Bhandup, Mumbai are 19.1438684, 72.9384327.

The geograpical coordinate of Bhandup (East), Mumbai are 19.1485568, 72.9470

657.

Cannot do: Bhavani Shankar Road, Mumbai, will drop index: 14

The geograpical coordinate of Mumbai Central, Mumbai are 18.9695855, 72.8193

152.

Cannot do: Mumbai G P O, Mumbai, will drop index: 16

The geograpical coordinate of Borivli (East), Mumbai are 19.2251085, 72.8502

063.

The geograpical coordinate of Borivli (West), Mumbai are 19.2251085, 72.8502

063.

The geograpical coordinate of Borivli HO, Mumbai are 19.2314681, 72.8405735.

The geograpical coordinate of Chakala MIDC, Mumbai are 19.1152873, 72.861808 5.

The geograpical coordinate of Chembur, Mumbai are 19.0612128, 72.8975909.

The geograpical coordinate of Chinch Bunder, Mumbai are 18.9641312, 72.84163 78.

The geograpical coordinate of Colaba, Mumbai are 18.915091, 72.8259691.

Cannot do: Council Hall, Mumbai, will drop index: 24

The geograpical coordinate of Cumballa Hill, Mumbai are 18.9693073, 72.80653 75.

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer2") **for** index, row **in** df2**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df2**.**loc[index, 'Latitude'] **=** latitude df2**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [32]:

The geograpical coordinate of Dadar, Mumbai are 19.019282, 72.8428757.

The geograpical coordinate of Dahisar, Mumbai are 19.2494501, 72.8596206.

The geograpical coordinate of Delisle Road, Mumbai are 18.9932602, 72.831011

5.

The geograpical coordinate of Dharavi, Mumbai are 19.044463, 72.8586177.

The geograpical coordinate of F C I Mumbai, Mumbai are 18.9387711, 72.835335 5.

The geograpical coordinate of Ghatkopar (West), Mumbai are 19.0897194, 72.90

45972.

The geograpical coordinate of Girgaon, Mumbai are 18.9543165, 72.8179082.

The geograpical coordinate of Goregaon (East), Mumbai are 19.1692623, 72.855 2548.

The geograpical coordinate of Goregaon (West), Mumbai are 19.1633281, 72.841 1995.

The geograpical coordinate of Grant Road, Mumbai are 18.9644472, 72.8135727. The geograpical coordinate of Hutatma Chowk, Mumbai are 18.9341699, 72.8325 2.

The geograpical coordinate of I I T Mumbai, Mumbai are 18.9387711, 72.835335

5.

The geograpical coordinate of J B Nagar, Mumbai are 18.9387711, 72.8353355. The geograpical coordinate of Jacob Circle, Mumbai are 18.9810531, 72.826786

29995553.

The geograpical coordinate of Jogeshwari (East), Mumbai are 19.1369889, 72.8

649418.

The geograpical coordinate of Jogeshwari (West), Mumbai are 19.136394, 72.83

73817.

The geograpical coordinate of Juhu, Mumbai are 19.1070215, 72.8275275.

The geograpical coordinate of Kalbadevi, Mumbai are 18.9492575, 72.8279382. The geograpical coordinate of Kandivli (East), Mumbai are 19.2064788, 72.838 4505.

The geograpical coordinate of Kandivli (West), Mumbai are 19.2064788, 72.838 4505.

The geograpical coordinate of Khar, Mumbai are 19.0696584, 72.8398944.

The geograpical coordinate of Kharodi, Mumbai are 19.191684, 72.8140978.

The geograpical coordinate of Kurla, Mumbai are 19.0652797, 72.8793805.

The geograpical coordinate of Mahim, Mumbai are 19.0423145, 72.8398344. The geograpical coordinate of Malabar Hill, Mumbai are 18.9581616, 72.803366 5.

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer3") **for** index, row **in** df3**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df3**.**loc[index, 'Latitude'] **=** latitude df3**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [33]:

The geograpical coordinate of Malad (East), Mumbai are 19.1860219, 72.856318 1.

The geograpical coordinate of Malad (West), Mumbai are 19.1840129, 72.841215 5.

The geograpical coordinate of Mandpeshwar, Mumbai are 19.23584235, 72.852836

55957588.

The geograpical coordinate of Mandvi, Mumbai are 18.9550564, 72.8347919.

The geograpical coordinate of Mantralaya, Mumbai are 18.92766235, 72.8270393

0367538.

The geograpical coordinate of Marine Lines, Mumbai are 18.9456701, 72.82378 1.

The geograpical coordinate of Matunga, Mumbai are 19.0274356, 72.8501467. The geograpical coordinate of Mazgaon, Mumbai are 18.9680519, 72.84001187036

267.

The geograpical coordinate of Motilal Nagar, Mumbai are 19.1589951, 72.84019 707773174.

The geograpical coordinate of Mulund (East), Mumbai are 19.1704719, 72.96104 47.

The geograpical coordinate of Mulund (West), Mumbai are 19.1719717, 72.95119 56.

The geograpical coordinate of Mulund Colony, Mumbai are 19.1777778, 72.93835 11.

Cannot do: N I T I E, Mumbai, will drop index: 63

The geograpical coordinate of Nariman Point, Mumbai are 18.9259514, 72.82320

83.

The geograpical coordinate of Nehru Nagar, Mumbai are 19.0010485, 72.819426 5.

The geograpical coordinate of Pant Nagar, Mumbai are 19.0868209, 72.9156026.

The geograpical coordinate of Parel, Mumbai are 19.0094817, 72.8376614.

Cannot do: Poonam Ngr Jogeshwari (E), Mumbai, will drop index: 68 The geograpical coordinate of Prabhadevi, Mumbai are 19.0148811, 72.8279556.

The geograpical coordinate of Rajawadi, Mumbai are 19.07853645, 72.900194638

36587.

Cannot do: Rajbhavan, Mumbai, will drop index: 71

The geograpical coordinate of SEEPZ, Mumbai are 19.1253556, 72.875170171970 7.

The geograpical coordinate of Sahar, Mumbai are 19.0995955, 72.867284.

The geograpical coordinate of Saki Naka, Mumbai are 19.1082208, 72.8835824. The geograpical coordinate of Santacruz (East), Mumbai are 19.0815223, 72.84 17565.

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer4") **for** index, row **in** df4**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df4**.**loc[index, 'Latitude'] **=** latitude df4**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [34]:

The geograpical coordinate of Santacruz (West), Mumbai are 19.084877, 72.834 9727.

The geograpical coordinate of Santacruz P&T Colony, Mumbai are 19.0923132, 7

2.8446354.

The geograpical coordinate of Sewri, Mumbai are 18.9987796, 72.8544218.

The geograpical coordinate of Shivaji Nagar (Kurla), Mumbai are 19.1141005, 72.8622899.

The geograpical coordinate of Sion, Mumbai are 19.0465213, 72.8632834.

The geograpical coordinate of Tagore Nagar, Mumbai are 19.1176276, 72.932200 6.

The geograpical coordinate of Tank Road, Mumbai are 19.02816595, 72.86832536 911001.

The geograpical coordinate of Telecom Factory Deonar, Mumbai are 19.04690045

0000003, 72.91846239254781.

The geograpical coordinate of Tilak Nagar, Mumbai are 19.0692377, 72.897846

4.

The geograpical coordinate of Tulsiwadi, Mumbai are 18.9387711, 72.8353355.

Cannot do: Veer Jijamata Bhosle Udyan, Mumbai, will drop index: 86

Cannot do: Vesava (Versova), Mumbai, will drop index: 87

Cannot do: Vidyanagari, Mumbai, will drop index: 88 The geograpical coordinate of Vikhroli, Mumbai are 19.1114795, 72.928021.

The geograpical coordinate of Vile Parle (East), Mumbai are 19.0962884, 72.8

483799.

The geograpical coordinate of Vile Parle (West), Mumbai are 19.1038725, 72.8

402903.

The geograpical coordinate of Wadala, Mumbai are 19.0269192, 72.8759337.

The geograpical coordinate of Worli, Mumbai are 19.0116962, 72.8180702.

Cannot do: Airoli Mode, Mumbai, will drop index: 94

The geograpical coordinate of Belapur, Mumbai are 19.01898695, 73.0390947240 8442.

The geograpical coordinate of Ghansoli, Mumbai are 19.1193307, 72.9995096.

Cannot do: JNPT Town Ship, Mumbai, will drop index: 97

Cannot do: Konkan Bhawan, Mumbai, will drop index: 98

Cannot do: Krishi Utpanna Bazar, Mumbai, will drop index: 99

The geograpical coordinate of Mumbra, Mumbai are 19.2064735, 73.0179849.

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer5") **for** index, row **in** df5**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude print('The geograpical coordinate of {} are {}, {}.'**.**format(address df5**.**loc[index, 'Latitude'] **=** latitude df5**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [35]:

Cannot do: N A D Karanja, Mumbai, will drop index: 101

Cannot do: Nerul Mode, Mumbai, will drop index: 102

The geograpical coordinate of Turbhe, Mumbai are 19.0761648, 73.017661543696 02.

The geograpical coordinate of Uran, Mumbai are 18.8808, 72.9386.

The geograpical coordinate of Vashi, Mumbai are 19.075713, 73.0003541.

Cannot do: Agashi, Mumbai, will drop index: 106

Cannot do: Arnala, Mumbai, will drop index: 107

Cannot do: Balcum, Mumbai, will drop index: 108

Cannot do: Bassien, Mumbai, will drop index: 109

Cannot do: Bassien Road, Mumbai, will drop index: 110

The geograpical coordinate of Bhayandar, Mumbai are 19.1971521, 72.8113662. The geograpical coordinate of Bhayander (East), Mumbai are 19.1160161, 72.82

00221.

The geograpical coordinate of Boisar, Mumbai are 19.2091438, 72.8591832.

Cannot do: Bordi, Mumbai, will drop index: 114

Cannot do: Chinchani, Mumbai, will drop index: 115

Cannot do: Dahanu, Mumbai, will drop index: 116

Cannot do: Dahanu Road, Mumbai, will drop index: 117

Cannot do: Dapcheri, Mumbai, will drop index: 118

Cannot do: Ganeshpuri, Mumbai, will drop index: 119

Cannot do: Gholvad, Mumbai, will drop index: 120

Cannot do: Jakegram, Mumbai, will drop index: 121

Cannot do: Jawhar, Mumbai, will drop index: 122

Cannot do: Kalwa, Mumbai, will drop index: 123 Cannot do: Kasa, Mumbai, will drop index: 124

Cannot do: Kelwa, Mumbai, will drop index: 125

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer6") **for** index, row **in** df6**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df6**.**loc[index, 'Latitude'] **=** latitude df6**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [36]:

Cannot do: Kelwa Mahim, Mumbai, will drop index: 126

Cannot do: Kopri Colony, Mumbai, will drop index: 127

Cannot do: Kosbad Hill, Mumbai, will drop index: 128 The geograpical coordinate of Manor, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of Mira, Mumbai are 19.2820571, 72.8741437.

The geograpical coordinate of Mira Road, Mumbai are 19.1878962, 72.8365955.

Cannot do: Mokhada, Mumbai, will drop index: 132

Cannot do: Nalasopara (East), Mumbai, will drop index: 133

The geograpical coordinate of Naupada, Mumbai are 19.083579, 72.8886519.

The geograpical coordinate of Nirmal, Mumbai are 18.92732875, 72.82219717183

933.

Cannot do: Palghar H O, Mumbai, will drop index: 136 Cannot do: Papdi, Mumbai, will drop index: 137

Cannot do: Sandoz Baug, Mumbai, will drop index: 138

Cannot do: Satpati, Mumbai, will drop index: 139

Cannot do: Sopara, Mumbai, will drop index: 140

Cannot do: Suryanagar, Mumbai, will drop index: 141

Cannot do: Talasari, Mumbai, will drop index: 142

Cannot do: Tarapur, Mumbai, will drop index: 143

Cannot do: Tarapur App, Mumbai, will drop index: 144

Cannot do: Tarapur J/A, Mumbai, will drop index: 145

The geograpical coordinate of Thane (East), Mumbai are 19.0308262, 73.019853 7.

The geograpical coordinate of Thane (H Q), Mumbai are 19.609814, 73.4033791.

Cannot do: Umbarpada, Mumbai, will drop index: 148

The geograpical coordinate of Uttan, Mumbai are 19.2533188, 72.7905148.

Cannot do: Vajreshwari, Mumbai, will drop index: 150

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer7") **for** index, row **in** df7**.**iterrows():  address **=** row['Post Office'] **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address df7**.**loc[index, 'Latitude'] **=** latitude df7**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown**.**append(index) |

In [37]:

Cannot do: Vangam, Mumbai, will drop index: 151

Cannot do: Vasai East I/E, Mumbai, will drop index: 152

Cannot do: Vasai Road East, Mumbai, will drop index: 153

Cannot do: Vikramgad, Mumbai, will drop index: 154

The geograpical coordinate of Virar, Mumbai are 19.4531561, 72.8839655.

Cannot do: Wagle Industrial Estate, Mumbai, will drop index: 156

In [38]:

print

(

to\_drop\_unknown

)

[5, 6, 14, 16, 24, 63, 68, 71, 86, 87, 88, 94, 97, 98, 99, 101, 102, 106, 10

7, 108, 109, 110, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 12

5, 126, 127, 128, 132, 133, 136, 137, 138, 139, 140, 141, 142, 143, 144, 14 5, 148, 150, 151, 152, 153, 154, 156]

In [39]:

dfa **=** df1**.**append(df2, ignore\_index **=** **True**) dfa **=** dfa**.**append(df3, ignore\_index **=** **True**) dfa **=** dfa**.**append(df4, ignore\_index **=** **True**) dfa **=** dfa**.**append(df5, ignore\_index **=** **True**) dfa **=** dfa**.**append(df6, ignore\_index **=** **True**) dfa **=** dfa**.**append(df7, ignore\_index **=** **True**) cleandf **=** dfa**.**drop(to\_drop\_unknown)

In [40]:

cleandf

**.**

shape

(101, 5) Out[40]:

|  |
| --- |
| tosearch\_df **=** df**.**loc[to\_drop\_unknown] tosearch\_df**.**head()  tosearch\_df**.**reset\_index(drop**=True**) |

In [41]:

Out[41]: City Post Office Pin Code Latitude Longitude

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | Mumbai | Anu Shakti Nagar | 400094 |
| 1 | Mumbai | B A R C | 400085 |
| 2 | Mumbai | Bhavani Shankar Road | 400028 |
| 3 | Mumbai | Mumbai G P O | 400001 |
| 4 | Mumbai | Council Hall | 400039 |
| 5 | Mumbai | N I T I E | 400087 |
| 6 | Mumbai | Poonam Ngr Jogeshwari (E) | 400041 |
| 7 | Mumbai | Rajbhavan | 400035 |
| 8 | Mumbai | Veer Jijamata Bhosle Udyan | 400027 |
| 9 | Mumbai | Vesava (Versova) | 400061 |
| 10 | Mumbai | Vidyanagari | 400098 |
| 11 | Navi Mumbai | Airoli Mode | 400708 |
| 12 | Navi Mumbai | JNPT Town Ship | 400707 |
| 13 | Navi Mumbai | Konkan Bhawan | 400614 |
| 14 | Navi Mumbai | Krishi Utpanna Bazar | 400705 |
| 15 | Navi Mumbai | N A D Karanja | 400704 |
| 16 | Navi Mumbai | Nerul Mode | 400706 |
| 17 | Thane | Agashi | 401301 |
| 18 | Thane | Arnala | 401302 |
| 19 | Thane | Balcum | 400608 |
| 20 | Thane | Bassien | 401201 |
| 21 | Thane | Bassien Road | 401202 |
| 22 | Thane | Bordi | 401701 |
| 23 | Thane | Chinchani | 401503 |
| 24 | Thane | Dahanu | 401601 |
| 25 | Thane | Dahanu Road | 401602 |
| 26 | Thane | Dapcheri | 401610 |
| 27 | Thane | Ganeshpuri | 401206 |
| 28 | Thane | Gholvad | 401702 |
| 29 | Thane | Jakegram | 400606 |
| 30 | Thane | Jawhar | 401603 |

City Post Office Pin Code Latitude Longitude

31 Thane Kalwa 400605

|  |  |  |
| --- | --- | --- |
| 32 Thane | Kasa | 401607 |
| 33 Thane | Kelwa | 401401 |
| 34 Thane | Kelwa Mahim | 401402 |
| 35 Thane | Kopri Colony | 400603 |
| 36 Thane | Kosbad Hill | 401703 |
| 37 Thane | Mokhada | 401604 |
| 38 Thane | Nalasopara (East) | 401209 |
| 39 Thane | Palghar H O | 401404 |
| 40 Thane | Papdi | 401207 |
| 41 Thane | Sandoz Baug | 400607 |
| 42 Thane | Satpati | 401405 |
| 43 Thane | Sopara | 401203 |
| 44 Thane | Suryanagar | 401609 |
| 45 Thane | Talasari | 401606 |
| 46 Thane | Tarapur | 401502 |
| 47 Thane | Tarapur App | 401504 |
| 48 Thane | Tarapur J/A | 401506 |
| 49 Thane | Umbarpada | 401102 |
| 50 Thane | Vajreshwari | 401204 |
| 51 Thane | Vangam | 401103 |
| 52 Thane | Vasai East I/E | 401208 |
| 53 Thane | Vasai Road East | 401210 |
| 54 Thane | Vikramgad | 401605 |
| 55 Thane Wagle Industrial Estate | | 400604 |
| tosearch\_df  **.**  shape | |  |

In [42]:

|  |  |
| --- | --- |
| Out[42]: | (56, 5) |

In [43]: tosearch\_df1 **=** tosearch\_df**.**iloc[0:25] tosearch\_df2 **=** tosearch\_df**.**iloc[26:50] tosearch\_df3 **=** tosearch\_df**.**iloc[51:55]

|  |  |  |
| --- | --- | --- |
|  | | |
|  | *# Need to drop those Neighborhood that the geocode does not find* to\_drop\_unknown2 **=** []  geolocator **=** Nominatim(user\_agent**=**"ny\_explorer8") **for** index, row **in** tosearch\_df1**.**iterrows():  address **=** str(row['Pin Code']) **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude |  |

In [44]:

|  |  |  |
| --- | --- | --- |
|  | longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(addres tosearch\_df1**.**loc[index, 'Latitude'] **=** latitude tosearch\_df1**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown2**.**append(index) | s |
|  |

The geograpical coordinate of 400094, Mumbai are 19.037527767647283, 72.9281

4570708681.

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/pandas/core/i ndexing.py:966: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/ stable/user\_guide/indexing.html#returning-a-view-versus-a-copy self.obj[item] = s

Cannot do: 400085, Mumbai, will drop index: 6

The geograpical coordinate of 400028, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400001, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400039, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400087, Mumbai are 18.9387711, 72.8353355. Cannot do: 400041, Mumbai, will drop index: 68

The geograpical coordinate of 400035, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400027, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400061, Mumbai are 19.13373646053233, 72.81487

711581413.

The geograpical coordinate of 400098, Mumbai are 19.1090916, 72.8607705.

The geograpical coordinate of 400708, Mumbai are 19.17297875, 73.0035322.

The geograpical coordinate of 400707, Mumbai are 18.9387711, 72.8353355. The geograpical coordinate of 400614, Mumbai are 18.99038025, 73.06534926069

733.

The geograpical coordinate of 400705, Mumbai are 19.080266, 73.021173.

Cannot do: 400704, Mumbai, will drop index: 101

The geograpical coordinate of 400706, Mumbai are 19.047375, 73.0193.

The geograpical coordinate of 401301, Mumbai are 19.3849292, 72.897546.

The geograpical coordinate of 401302, Mumbai are 19.2023218, 73.0025371.

The geograpical coordinate of 400608, Mumbai are 19.2023218, 73.0025371. The geograpical coordinate of 401201, Mumbai are 19.3849292, 72.897546.

The geograpical coordinate of 401202, Mumbai are 19.3566852, 72.892828.

Cannot do: 401701, Mumbai, will drop index: 114

Cannot do: 401503, Mumbai, will drop index: 115

Cannot do: 401601, Mumbai, will drop index: 116

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer9") **for** index, row **in** tosearch\_df2**.**iterrows():  address **=** str(row['Pin Code']) **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address tosearch\_df2**.**loc[index, 'Latitude'] **=** latitude tosearch\_df2**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown2**.**append(index) |

In [45]:

Cannot do: 401610, Mumbai, will drop index: 118

The geograpical coordinate of 401206, Mumbai are 18.9387711, 72.8353355.

Cannot do: 401702, Mumbai, will drop index: 120

The geograpical coordinate of 400606, Mumbai are 19.13261205, 72.93201331422

414.

Cannot do: 401603, Mumbai, will drop index: 122 The geograpical coordinate of 400605, Mumbai are 19.2023218, 73.0025371.

Cannot do: 401607, Mumbai, will drop index: 124

Cannot do: 401401, Mumbai, will drop index: 125

Cannot do: 401402, Mumbai, will drop index: 126

The geograpical coordinate of 400603, Mumbai are 19.13261205, 72.93201331422

414.

Cannot do: 401703, Mumbai, will drop index: 128 Cannot do: 401604, Mumbai, will drop index: 132

Cannot do: 401209, Mumbai, will drop index: 133

Cannot do: 401404, Mumbai, will drop index: 136 The geograpical coordinate of 401207, Mumbai are 18.9387711, 72.8353355.

The geograpical coordinate of 400607, Mumbai are 19.2023218, 73.0025371.

Cannot do: 401405, Mumbai, will drop index: 139

The geograpical coordinate of 401203, Mumbai are 19.3849292, 72.897546.

Cannot do: 401609, Mumbai, will drop index: 141 Cannot do: 401606, Mumbai, will drop index: 142

Cannot do: 401502, Mumbai, will drop index: 143

Cannot do: 401504, Mumbai, will drop index: 144

Cannot do: 401506, Mumbai, will drop index: 145

The geograpical coordinate of 401102, Mumbai are 19.3849292, 72.897546.

|  |
| --- |
| *# Need to drop those Neighborhood that the geocode does not find* geolocator **=** Nominatim(user\_agent**=**"ny\_explorer10") **for** index, row **in** tosearch\_df3**.**iterrows():  address **=** str(row['Pin Code']) **+** ', Mumbai' **try**:  location **=** geolocator**.**geocode(address) latitude **=** location**.**latitude longitude **=** location**.**longitude  print('The geograpical coordinate of {} are {}, {}.'**.**format(address tosearch\_df3**.**loc[index, 'Latitude'] **=** latitude tosearch\_df3**.**loc[index, 'Longitude'] **=** longitude **except** AttributeError:  print('Cannot do: {}, will drop index: {}'**.**format(address, index)) to\_drop\_unknown2**.**append(index) |

In [46]:

Cannot do: 401103, Mumbai, will drop index: 151

The geograpical coordinate of 401208, Mumbai are 19.3849292, 72.897546.

The geograpical coordinate of 401210, Mumbai are 18.9387711, 72.8353355.

Cannot do: 401605, Mumbai, will drop index: 154

In [47]:

print

(

to\_drop\_unknown2

)

[6, 68, 101, 114, 115, 116, 118, 120, 122, 124, 125, 126, 128, 132, 133, 13

6, 139, 141, 142, 143, 144, 145, 151, 154]

In [48]: dfb **=** tosearch\_df1**.**append(tosearch\_df2, ignore\_index **=** **True**) dfb **=** dfb**.**append(tosearch\_df3, ignore\_index **=** **True**) dfb

Out[48]: City Post Office Pin Code Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | Anu Shakti Nagar | 400094 | 19.0375 | 72.9281 |
| 1 | Mumbai | B A R C | 400085 |  |  |
| 2 | Mumbai | Bhavani Shankar Road | 400028 | 18.9388 | 72.8353 |
| 3 | Mumbai | Mumbai G P O | 400001 | 18.9388 | 72.8353 |

Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4 | Mumbai | Council Hall | 400039 | 18.9388 | 72.8353 |
| 5 | Mumbai | N I T I E | 400087 | 18.9388 | 72.8353 |
| 6 | Mumbai | Poonam Ngr Jogeshwari (E) | 400041 |  |  |
| 7 | Mumbai | Rajbhavan | 400035 | 18.9388 | 72.8353 |
| 8 | Mumbai | Veer Jijamata Bhosle Udyan | 400027 | 18.9388 | 72.8353 |
| 9 | Mumbai | Vesava (Versova) | 400061 | 19.1337 | 72.8149 |
| 10 | Mumbai | Vidyanagari | 400098 | 19.1091 | 72.8608 |
| 11 | Navi Mumbai | Airoli Mode | 400708 | 19.173 | 73.0035 |
| 12 | Navi Mumbai | JNPT Town Ship | 400707 | 18.9388 | 72.8353 |
| 13 | Navi Mumbai | Konkan Bhawan | 400614 | 18.9904 | 73.0653 |
| 14 | Navi Mumbai | Krishi Utpanna Bazar | 400705 | 19.0803 | 73.0212 |
| 15 | Navi Mumbai | N A D Karanja | 400704 |  |  |
| 16 | Navi Mumbai | Nerul Mode | 400706 | 19.0474 | 73.0193 |
| 17 | Thane | Agashi | 401301 | 19.3849 | 72.8975 |
| 18 | Thane | Arnala | 401302 | 19.2023 | 73.0025 |
| 19 | Thane | Balcum | 400608 | 19.2023 | 73.0025 |
| 20 | Thane | Bassien | 401201 | 19.3849 | 72.8975 |
| 21 | Thane | Bassien Road | 401202 | 19.3567 | 72.8928 |
| 22 | Thane | Bordi | 401701 |  |  |
| 23 | Thane | Chinchani | 401503 |  |  |
| 24 | Thane | Dahanu | 401601 |  |  |
| 25 | Thane | Dapcheri | 401610 |  |  |
| 26 | Thane | Ganeshpuri | 401206 | 18.9388 | 72.8353 |
| 27 | Thane | Gholvad | 401702 |  |  |
| 28 | Thane | Jakegram | 400606 | 19.1326 | 72.932 |
| 29 | Thane | Jawhar | 401603 |  |  |
| 30 | Thane | Kalwa | 400605 | 19.2023 | 73.0025 |
| 31 | Thane | Kasa | 401607 |  |  |
| 32 | Thane | Kelwa | 401401 |  |  |
| 33 | Thane | Kelwa Mahim | 401402 |  |  |
| 34 | Thane | Kopri Colony | 400603 | 19.1326 | 72.932 |
| 35 | Thane | Kosbad Hill | 401703 |  |  |
| 36 | Thane | Mokhada | 401604 |  |  |
| 37 | Thane | Nalasopara (East) | 401209 |  |  |
| 38 | Thane | Palghar H O | 401404 |  |  |
| 39 | Thane | Papdi | 401207 | 18.9388 | 72.8353 |
| 40 | Thane | Sandoz Baug | 400607 | 19.2023 | 73.0025 |

Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 41 | Thane | Satpati | 401405 |  |  |
| 42 | Thane | Sopara | 401203 | 19.3849 | 72.8975 |
| 43 | Thane | Suryanagar | 401609 |  |  |
| 44 | Thane | Talasari | 401606 |  |  |
| 45 | Thane | Tarapur | 401502 |  |  |
| 46 | Thane | Tarapur App | 401504 |  |  |
| 47 | Thane | Tarapur J/A | 401506 |  |  |
| 48 | Thane | Umbarpada | 401102 | 19.3849 | 72.8975 |
| 49 | Thane | Vangam | 401103 |  |  |
| 50 | Thane | Vasai East I/E | 401208 | 19.3849 | 72.8975 |
| 51 | Thane | Vasai Road East | 401210 | 18.9388 | 72.8353 |
| 52 | Thane | Vikramgad | 401605 |  |  |

In [49]: dfb['Latitude']**.**replace('', np**.**nan, inplace**=True**) dfb**.**dropna(subset**=**['Latitude'], inplace**=True**) dfb

Out[49]: City Post Office Pin Code Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | Anu Shakti Nagar | 400094 | 19.037528 | 72.9281 |
| 2 | Mumbai | Bhavani Shankar Road | 400028 | 18.938771 | 72.8353 |
| 3 | Mumbai | Mumbai G P O | 400001 | 18.938771 | 72.8353 |
| 4 | Mumbai | Council Hall | 400039 | 18.938771 | 72.8353 |
| 5 | Mumbai | N I T I E | 400087 | 18.938771 | 72.8353 |
| 7 | Mumbai | Rajbhavan | 400035 | 18.938771 | 72.8353 |
| 8 | Mumbai | Veer Jijamata Bhosle Udyan | 400027 | 18.938771 | 72.8353 |
| 9 | Mumbai | Vesava (Versova) | 400061 | 19.133736 | 72.8149 |
| 10 | Mumbai | Vidyanagari | 400098 | 19.109092 | 72.8608 |
| 11 | Navi Mumbai | Airoli Mode | 400708 | 19.172979 | 73.0035 |
| 12 | Navi Mumbai | JNPT Town Ship | 400707 | 18.938771 | 72.8353 |
| 13 | Navi Mumbai | Konkan Bhawan | 400614 | 18.990380 | 73.0653 |
| 14 | Navi Mumbai | Krishi Utpanna Bazar | 400705 | 19.080266 | 73.0212 |
| 16 | Navi Mumbai | Nerul Mode | 400706 | 19.047375 | 73.0193 |
| 17 | Thane | Agashi | 401301 | 19.384929 | 72.8975 |
| 18 | Thane | Arnala | 401302 | 19.202322 | 73.0025 |
| 19 | Thane | Balcum | 400608 | 19.202322 | 73.0025 |
| 20 | Thane | Bassien | 401201 | 19.384929 | 72.8975 |
| 21 | Thane | Bassien Road | 401202 | 19.356685 | 72.8928 |
| 26 | Thane | Ganeshpuri | 401206 | 18.938771 | 72.8353 |

Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 28 | Thane | Jakegram | 400606 | 19.132612 | 72.932 |
| 30 | Thane | Kalwa | 400605 | 19.202322 | 73.0025 |
| 34 | Thane | Kopri Colony | 400603 | 19.132612 | 72.932 |
| 39 | Thane | Papdi | 401207 | 18.938771 | 72.8353 |
| 40 | Thane | Sandoz Baug | 400607 | 19.202322 | 73.0025 |
| 42 | Thane | Sopara | 401203 | 19.384929 | 72.8975 |
| 48 | Thane | Umbarpada | 401102 | 19.384929 | 72.8975 |
| 50 | Thane | Vasai East I/E | 401208 | 19.384929 | 72.8975 |
| 51 | Thane | Vasai Road East | 401210 | 18.938771 | 72.8353 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | August Kranti Marg | 400036 | 18.9635 | 72.81 |
| 1 | Mumbai | Aarey Milk Colony | 400065 | 19.1561 | 72.8707 |
| 2 | Mumbai | Andheri (East) | 400069 | 19.1159 | 72.8542 |
| 3 | Mumbai | Andheri (West) | 400058 | 19.1172 | 72.834 |
| 4 | Mumbai | Antop Hill | 400037 | 19.0208 | 72.8653 |
| ... | ... | ... | ... | ... | ... |
| 129 | Thane | Vasai Road East | 401210 | 18.9388 | 72.8353 |
| 130 | Mumbai | B A R C | 400085 | 19.0167 | 72.85 |
| 131 | Thane | Talasari | 401606 | 19.9167 | 73.2333 |
| 132 | Navi Mumbai | NAD Karanja | 400704 | 19.0778 | 72.8944 |
| 133 | Thane | Jawhar | 401603 | 19.9167 | 73.2333 |

In [50]: lst **=** [['Mumbai','B A R C',400085,19.0167,72.85 ],['Thane','Talasari',40160 dfc **=** pd**.**DataFrame(list(lst), columns **=**['City','Post Office','Pin Code','La dfc

Out[50]: City Post Office Pin Code Latitude Longitude

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | B A R C | 400085 | 19.0167 | 72.8500 |
| 1 | Thane | Talasari | 401606 | 19.9167 | 73.2333 |
| 2 | Navi Mumbai | NAD Karanja | 400704 | 19.0778 | 72.8944 |
| 3 | Thane | Jawhar | 401603 | 19.9167 | 73.2333 |

In [51]: clean\_df2 **=** dfb**.**append(dfc, ignore\_index **=** **True**) clean\_df2**.**shape

(33, 5) Out[51]:

In [52]: clean\_df **=** cleandf**.**append(clean\_df2, ignore\_index **=** **True**) clean\_df**.**shape

Out[52]: City Post Office Pin Code Latitude Longitude

134 rows × 5 columns

In [53]:

clean\_df

**.**

to\_csv

(

'mumbailatlong.csv'

)

# 3. Define Foursquare Credentials and Version

|  |
| --- |
| *# define Foursquare Credentials and Version*  CLIENT\_ID **=** 'NT2C1SLVDXCSGCZPWP3D0JM41SNOZMB5F4IYCL5CR2V20ZIP' *# your Fours*  CLIENT\_SECRET **=** 'PS1QM5EW1EVJPXYJK4JF45GJHMHZJBCDZAD4CXQJUMIDC4NM' *# your F*  VERSION **=** '20180605' *# Foursquare API version*    print('Your credentails:') print('CLIENT\_ID: ' **+** CLIENT\_ID) print('CLIENT\_SECRET:' **+** CLIENT\_SECRET) clean\_df\_new **=** clean\_df**.**copy() |

In [3]:

Your credentails:

CLIENT\_ID: NT2C1SLVDXCSGCZPWP3D0JM41SNOZMB5F4IYCL5CR2V20ZIP CLIENT\_SECRET:PS1QM5EW1EVJPXYJK4JF45GJHMHZJBCDZAD4CXQJUMIDC4NM

4. Checking & Reading the 'mumbaiexplore' file directly to avoid api calling.

|  |  |  |  |
| --- | --- | --- | --- |
| venues\_df **=** pd**.**read\_csv('mumbaiexplor venues\_df**.**head() | | e.csv',index\_col | **=**'Unnamed: 0') |
| Post Pin Latitude Office Code | Longitude | City VenueName | VenueLatitude VenueLong |

In [4]:

Out[4]:i

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Doolally Taproom | 18.963809 | 72.807 |
| 1 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | symphony | 18.963347 | 72.81 |
| 2 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Crossword | 18.963474 | 72.80 |
| 3 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Swati Snacks | 18.966442 | 72.81 |
| 4 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Gustoso | 18.964198 | 72.807 |



# OR Top 225 venues that are within a radius of 600 meters for each post office

|  |  |  |
| --- | --- | --- |
|  | | |
|  | radius **=** 600 LIMIT **=** 225 venues **=** [] |  |

In [7]:

|  |  |  |
| --- | --- | --- |
|  | **for** lat, long, pin, post, city **in** zip(clean\_df\_new['Latitude'], clean\_df\_new url **=** "https://api.foursquare.com/v2/venues/explore?client\_id={}&client\_ CLIENT\_ID, CLIENT\_SECRET, VERSION, lat,    results **=** requests**.**get(url)**.**json()["response"]['groups'][0]['items']  **for** venue **in** results: venues**.**append((  post, pin, lat, long, venue['venue']['name'], venue['venue']['location']['lat'], venue['venue']['location']['lng'], venue['venue']['categories'][0]['name'])) venues\_df **=** pd**.**DataFrame(venues) venues\_df**.**head() |  |
|  |

|  |
| --- |
| *# define the column names*  venues\_df**.**columns **=** ['Post Office', 'Pin Code', 'Latitude', 'Longitude', 'C    print(venues\_df**.**shape) venues\_df**.**head() |

In [11]:

(2340, 9)

Out[11]: Post Pin Latitude Longitude City VenueName VenueLatitude VenueLongi

Office Code

August Doolally

1. Kranti 400036 18.963549 72.809989 Mumbai Taproom 18.963809 72.807

Marg

August

1. Kranti 400036 18.963549 72.809989 Mumbai symphony 18.963347 72.81

Marg

August

1. Kranti 400036 18.963549 72.809989 Mumbai Crossword 18.963474 72.80

Marg

August Swati

1. Kranti 400036 18.963549 72.809989 Mumbai Snacks 18.966442 72.81

Marg

August

1. Kranti 400036 18.963549 72.809989 Mumbai Gustoso 18.964198 72.807

Marg

In [12]:



venues\_df

**.**

to\_csv

(

'mumbaiexplore.csv'

)

In [5]:

|  |  |  |
| --- | --- | --- |
| venues\_df**.**groupby(['Post Office', 'Pin Code', 'City'])**.**count()**.**head() | | |
|  |  | Latitude Longitude VenueName VenueLatitude VenueLongitu |
| Post Office | Pin Code | City |

Out[5]:d

Aarey

Milk 400065 Mumbai 2 2 2 2

Colony

Agashi 401301 Thane 3 3 3 3

Latitude Longitude VenueName VenueLatitude VenueLongitud

Post Pin City

Office Code

Airoli 400708 Navi 3 3 3 3

Mode Mumbai

Andheri 400069 Mumbai 14 14 14 14 1

(East)

Andheri 400058 Mumbai 22 22 22 22 2

(West)



|  |
| --- |
| print('There are {} uniques categories.'**.**format(len(venues\_df['VenueCatego |

In [6]: ry

There are 191 uniques categories.

In [7]:

venues\_df

[

'VenueCategory'

]

**.**

unique

()[:

20

]

array(['Brewery', 'Restaurant', 'Bookstore', 'Indian Restaurant', Out[7]:

'Pizza Place', 'Dessert Shop', 'History Museum',

'Salon / Barbershop', 'Hotel', 'Bakery', 'Café', 'Coffee Shop',

'Bar', 'Concert Hall', 'Italian Restaurant', 'Park', 'Lounge',

'Deli / Bodega', 'Sandwich Place', 'Clothing Store'], dtype=object)

# 5. Analyze Each Postal Office For Venue Category

|  |
| --- |
| *# one hot encoding*  mumbai\_onehot **=** pd**.**get\_dummies(venues\_df[['VenueCategory']], prefix**=**"", pre    *# add postal, borough and neighborhood column back to dataframe* mumbai\_onehot['Post Office'] **=** venues\_df['Post Office'] mumbai\_onehot['Pin Code'] **=** venues\_df['Pin Code'] mumbai\_onehot['City'] **=** venues\_df['City']    *# move postal, borough and neighborhood column to the first column* fixed\_columns **=** list(mumbai\_onehot**.**columns[**-**3:]) **+** list(mumbai\_onehot**.**column mumbai\_onehot **=** mumbai\_onehot[fixed\_columns]    print(mumbai\_onehot**.**shape) mumbai\_onehot**.**head() |

In [8]:

(2340, 194)

Out[8]: Post Pin Accessories Airport American

Office Code City Store Terminal Restaurant Amphitheater Aquarium A

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | August  Kranti  Marg | 400036 | Mumbai | 0 | 0 | 0 | 0 | 0 |
| 1 | August  Kranti  Marg | 400036 | Mumbai | 0 | 0 | 0 | 0 | 0 |
| 2 | August  Kranti  Marg | 400036 | Mumbai | 0 | 0 | 0 | 0 | 0 |

Post Pin Accessories Airport American

Office Code City Store Terminal Restaurant Amphitheater Aquarium A

August

1. Kranti 400036 Mumbai 0 0 0 0 0

Marg

August

1. Kranti 400036 Mumbai 0 0 0 0 0

Marg

1. rows × 194 columns



|  |
| --- |
| mumbai\_grouped **=** mumbai\_onehot**.**groupby(["Post Office", "Pin Code", "City"])    print(mumbai\_grouped**.**shape) mumbai\_grouped**.**head() |

In [9]:

(132, 194)

Out[9]: Post Pin Accessories Airport American

Office Code City Store Terminal Restaurant Amphitheater Aquarium A

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Aarey  Milk Colony | 400065 | Mumbai | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 1 | Agashi | 401301 | Thane | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 2 | Airoli  Mode | 400708 | Navi Mumbai | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3 | Andheri (East) | 400069 | Mumbai | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 4 | Andheri | 400058 | Mumbai | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

(West)

5 rows × 194 columns



# 6. List and display the top 5 existing facilities for each Pin Code

|  |  |  |
| --- | --- | --- |
|  | | |
|  | num\_top\_venues **=** 5    indicators **=** ['st', 'nd', 'rd']    *# create columns according to number of top venues* areaColumns **=** ["Post Office", "Pin Code", "City"] freqColumns **=** [] **for** ind **in** np**.**arange(num\_top\_venues): **try**:  freqColumns**.**append('{}{} Most Common Venue'**.**format(ind**+**1, indicato **except**:  freqColumns**.**append('{}th Most Common Venue'**.**format(ind**+**1)) columns **=** areaColumns**+**freqColumns    *# create a new dataframe*  neighborhoods\_venues\_sorted **=** pd**.**DataFrame(columns**=**columns) | r |

In [10]:

|  |  |  |
| --- | --- | --- |
|  | neighborhoods\_venues\_sorted['Post Office'] **=** mumbai\_grouped['Post Office'] neighborhoods\_venues\_sorted['Pin Code'] **=** mumbai\_grouped['Pin Code'] neighborhoods\_venues\_sorted['City'] **=** mumbai\_grouped['City']  **for** ind **in** np**.**arange(mumbai\_grouped**.**shape[0]):  row\_categories **=** mumbai\_grouped**.**iloc[ind, :]**.**iloc[3:] row\_categories\_sorted **=** row\_categories**.**sort\_values(ascending**=False**) neighborhoods\_venues\_sorted**.**iloc[ind, 3:] **=** row\_categories\_sorted**.**inde  *# neighborhoods\_venues\_sorted.sort\_values(freqColumns, inplace=True)* print(neighborhoods\_venues\_sorted**.**shape) neighborhoods\_venues\_sorted**.**head() | x |
|  |

(132, 8)

Out[10]: Post Pin 1st Most 2nd Most 3rd Most 4th Most 5th Most Office Code City Common Common Common Common Common

Venue Venue Venue Venue Venue

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Aarey  Milk Colony | 400065 | Mumbai | Fast Food  Restaurant | Lake | Yoga Studio | Flea Market | Farmers  Market |
| 1 | Agashi | 401301 | Thane | Cheese  Shop | Scenic  Lookout | Restaurant | Yoga Studio | Dhaba |
| 2 | Airoli  Mode | 400708 | Navi Mumbai | Cocktail Bar | Restaurant | Garden | Yoga Studio | Dhaba |
| 3 | Andheri (East) | 400069 | Mumbai | Hotel | Indian Restaurant | Chinese Restaurant | Fast Food  Restaurant | Shopping Mall |
| 4 | Andheri  (West) | 400058 | Mumbai | Indian Restaurant | Pub | Coffee Shop | Café | Gym / Fitness |

Center

|  |
| --- |
| *# Set manually to get proper fit in the map* address **=** 'Mumbai' latitude **=** 19.0760 longitude **=** 72.8777  print('The geograpical coordinate of {} are {}, {}.'**.**format(address, latitud |

In [11]:

The geograpical coordinate of Mumbai are 19.076, 72.8777.

|  |
| --- |
| mumbai\_merged **=** clean\_df**.**copy()  mumbai\_merged **=** mumbai\_merged**.**join(neighborhoods\_venues\_sorted[["Pin Code", print(mumbai\_merged**.**shape) mumbai\_merged**.**head() |

In [12]:

(136, 6)

Out[12]: City Post Office Pin Code Latitude Longitude 1st Most Common Venue

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | Mumbai | August Kranti Marg | 400036 | 18.963549 | 72.809989 | Coffee Shop |
| 1 | Mumbai | Aarey Milk Colony | 400065 | 19.156129 | 72.870722 | Fast Food Restaurant |
| 2 | Mumbai | Andheri (East) | 400069 | 19.115883 | 72.854202 | Hotel |
| 3 | Mumbai | Andheri (West) | 400058 | 19.117249 | 72.833968 | Indian Restaurant |
| 4 | Mumbai | Antop Hill | 400037 | 19.020761 | 72.865256 | Fast Food Restaurant |

# 7. Exploratory Visualization

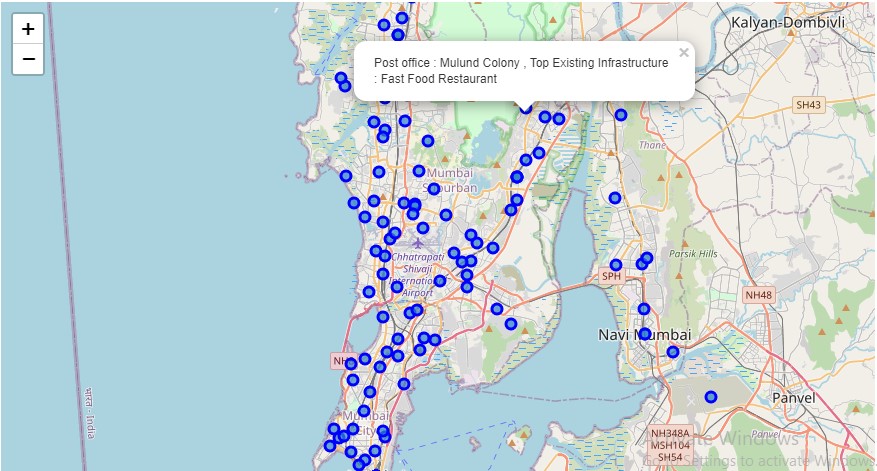
|  |  |  |
| --- | --- | --- |
|  | **import** folium  my\_map **=** folium**.**Map(location**=**[latitude, longitude], zoom\_start**=**11)  *# add markers to map*  **for** lat, lng, label1,common **in** zip(mumbai\_merged['Latitude'], mumbai\_merge labelnew **=** 'Post office : {} , Top Existing Infrastructure : {}'**.**for label **=** folium**.**Popup( labelnew, parse\_html**=True**) folium**.**CircleMarker( [lat, lng], radius**=**5, popup**=**label, color**=**'blue', fill**=True**,  fill\_color**=**'#3186cc', fill\_opacity**=**0.7, parse\_html**=False**)**.**add\_to(my\_map) my\_map | d m |
|  |

In [13]:

Out[13]: Make this Notebook Trusted to load map: File -> Trust Notebook

Showcasing map if not displayed in Github

|  |
| --- |
| **from** IPython.display **import** Image **from** IPython.core.display **import** HTML  Image(url**=** "https://raw.githubusercontent.com/RohitLearner/IBM-Applied-D.S-C |

In [14]: Out[14]: 

# 8. Feature Engineering for Business Problem

In [15]:

venues\_df

[

'VenueCategory'

]

**.**

unique

()

array(['Brewery', 'Restaurant', 'Bookstore', 'Indian Restaurant', Out[15]:

'Pizza Place', 'Dessert Shop', 'History Museum',

'Salon / Barbershop', 'Hotel', 'Bakery', 'Café', 'Coffee Shop',

'Bar', 'Concert Hall', 'Italian Restaurant', 'Park', 'Lounge', 'Deli / Bodega', 'Sandwich Place', 'Clothing Store',

'Chinese Restaurant', 'Farmers Market', 'Diner', 'Food Court',

'Cosmetics Shop', 'Food Truck', 'Fast Food Restaurant', 'Lake', 'Shopping Mall', 'Camera Store', 'Smoke Shop', 'Bus Station',

'Vegetarian / Vegan Restaurant', 'Pub', 'Gym / Fitness Center',

'Bagel Shop', 'Nightclub', 'Snack Place', 'Ice Cream Shop',

'Pharmacy', 'Grocery Store', 'Trail', 'Multiplex',

'Parsi Restaurant', 'Irani Cafe', 'Plaza', 'Seafood Restaurant',

'Hostel', 'Train Station', 'Outdoors & Recreation', 'Sports Club',

'General Entertainment', 'Gym', 'College Auditorium',

'French Restaurant', 'Gourmet Shop', 'Salad Place',

'German Restaurant', 'Event Space', 'Arcade', 'Asian Restaurant', 'Spanish Restaurant', 'BBQ Joint', "Women's Store",

'Indie Movie Theater', 'Comedy Club', 'Mediterranean Restaurant',

'Burger Joint', "Men's Store", 'Beer Garden', 'Steakhouse',

'Fried Chicken Joint', 'Electronics Store', 'Donut Shop',

'Department Store', 'New American Restaurant',

'Middle Eastern Restaurant', 'Factory', 'Light Rail Station',

'Track Stadium', 'Tea Room', 'Bengali Restaurant', 'Movie Theater',

'Platform', 'Convenience Store', 'Boutique', 'Sporting Goods Shop',

'Shoe Store', 'Historic Site', 'Food', 'Cocktail Bar',

'Performing Arts Venue', 'Garden', 'Market', 'Whisky Bar',

'Gift Shop', 'Thai Restaurant', 'Spa', 'Furniture / Home Store',

'Theater', 'Stadium', 'Other Great Outdoors', 'Racetrack',

'Japanese Restaurant', 'Yoga Studio', 'Frozen Yogurt Shop',

'Flower Shop', 'Maharashtrian Restaurant', 'Breakfast Spot',

'Juice Bar', 'Music Venue', 'Athletics & Sports', 'Flea Market', 'Harbor / Marina', 'Beach', 'Mexican Restaurant', 'Gastropub',

'Opera House', 'Multicuisine Indian Restaurant', 'Road', 'Farm',

'Jewelry Store', 'Bank', 'Monument / Landmark',

'Mughlai Restaurant', 'Noodle House', 'Campground',

'American Restaurant', 'North Indian Restaurant', 'Hookah Bar',

'Business Service', 'Hotel Bar', 'Supermarket', 'Video Store',

'Kids Store', 'Cheese Shop', 'Bike Rental / Bike Share',

'Beer Bar', 'Amphitheater', 'Szechuan Restaurant',

'Food & Drink Shop', 'Arts & Crafts Store', 'Intersection',

'Art Gallery', 'Scenic Lookout', 'Chaat Place', 'Field',

'Cricket Ground', 'Tennis Court', 'Bistro', 'Bridal Shop',

'Cupcake Shop', 'Aquarium', 'South Indian Restaurant',

'Gaming Cafe', 'Soup Place', 'Playground', 'Mountain', 'Wine Bar', 'Mobile Phone Shop', 'Government Building', 'Airport Terminal',

'Indian Chinese Restaurant', 'Dance Studio', 'Miscellaneous Shop',

'Accessories Store', 'Moving Target', 'Cafeteria', 'Gym Pool',

'Baseball Field', 'Paper / Office Supplies Store', 'Sports Bar',

'Surf Spot', 'Health & Beauty Service', 'Bed & Breakfast', 'Resort', 'Falafel Restaurant', 'Burrito Place',

'Dim Sum Restaurant', 'Toy / Game Store', 'Bowling Alley',

'Theme Park', 'Hawaiian Restaurant', 'Punjabi Restaurant',

'Wine Shop', 'Boat or Ferry', 'Dhaba', 'Recreation Center',

'Creperie', 'Auto Garage', 'Automotive Shop'], dtype=object)

In [16]:

|  |
| --- |
| *# Quality Infrastructure*  search\_query**=** ['Restaurant', 'Hotel', 'Farmers Market', 'Shopping Mall', 'Electronics Store', 'Indie Movie Theater', 'Light  'Theater','ATM', 'Office', 'Bus Station', 'Bank', 'Resort', 'Hospital', 'Police Station', 'School',  'Convention Center', 'College Auditorium', 'Government Build  ] print(len(search\_query)) |

'Gy

33

In [17]: quality\_dataframe **=** []

quality\_dataframe**=** venues\_df**.**loc[venues\_df['VenueCategory']**.**isin(search\_quer quality\_dataframe**.**shape

(473, 9) Out[17]:

In [18]:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | quality\_dataframe |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Post Pin  Office Code | Latitude | Longitude | City | VenueName | VenueLatitude | Venue | |

Out[18]:L

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | symphony | 18.963347 | 7 |
| 8 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Krishna  Palace  Residency Hotel | 18.962266 | 7 |
| 11 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | Moshe's | 18.963438 | 7 |
| 16 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | August  Kranti Maidan | 18.963433 | 7 |
| 18 | August  Kranti  Marg | 400036 | 18.963549 | 72.809989 | Mumbai | di bella | 18.965556 | 7 |
| ... | ... | ... | ... | ... | ... | ... | ... |  |
| 2327 | B A R C | 400085 | 19.016700 | 72.850000 | Mumbai | Ramee  Guestline Hotel | 19.017085 | 7 |
| 2330 | B A R C | 400085 | 19.016700 | 72.850000 | Mumbai | wadala bus depot | 19.014879 | 7 |
| 2332 | B A R C | 400085 | 19.016700 | 72.850000 | Mumbai | Agyari Gardens | 19.018896 | 7 |
| 2334 | Talasari | 401606 | 19.916700 | 73.233300 | Thane | Jawhar Bus Station | 19.919062 | 7 |
| 2339 | Jawhar | 401603 | 19.916700 | 73.233300 | Thane | Jawhar Bus Station | 19.919062 | 7 |

473 rows × 9 columns



|  |
| --- |
| *# one hot encoding*  qualitymumbai\_onehot **=** pd**.**get\_dummies(quality\_dataframe[['VenueCategory']],  *# add postal, borough and neighborhood column back to dataframe* qualitymumbai\_onehot['Post Office'] **=** quality\_dataframe['Post Office'] qualitymumbai\_onehot['Pin Code'] **=** quality\_dataframe['Pin Code'] qualitymumbai\_onehot['City'] **=** quality\_dataframe['City']  *# move postal, borough and neighborhood column to the first column* fixed\_columns **=** list(qualitymumbai\_onehot**.**columns[**-**3:]) **+** list(qualitymumba qualitymumbai\_onehot **=** qualitymumbai\_onehot[fixed\_columns]    print(qualitymumbai\_onehot**.**shape) qualitymumbai\_onehot**.**head()  print(qualitymumbai\_onehot**.**columns**.**values) |

In [19]:

(473, 27)

['Post Office' 'Pin Code' 'City' 'Airport Terminal' 'Bank' 'Bus Station'

'Business Service' 'Café' 'College Auditorium' 'Electronics Store' 'Farmers Market' 'Garden' 'Government Building' 'Gym / Fitness Center'

'Hotel' 'Indie Movie Theater' 'Light Rail Station' 'Market'

'Monument / Landmark' 'Park' 'Pharmacy' 'Playground' 'Resort'

'Restaurant' 'Shopping Mall' 'Theater' 'Train Station']

In [20]:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| qualitymumbai\_grouped **=** qualitymumbai\_onehot print(qualitymumbai\_grouped**.**shape) qualitymumbai\_grouped**.**head() | | | | **.** | groupby([ |  | "Post Office", | "Pin Co |
| (111, 27)  Post Office | Pin Code | Airport City Terminal | Bank | Bus Station | Business Service | Café | College  Auditorium | Electron S |

Out[20]: to

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Agashi | 401301 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | Airoli  Mode | 400708 | Navi Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | Andheri (East) | 400069 | Mumbai | 0 | 0 | 1 | 0 | 0 | 0 |
| 3 | Andheri  (West) | 400058 | Mumbai | 0 | 0 | 0 | 0 | 2 | 0 |
| 4 | Anu  Shakti Nagar | 400094 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |

5 rows × 27 columns



(111

,

28)

In [21]:

qualitymumbai\_grouped

[

'Total infrastructure'

]

**=**

qualitymumbai\_grouped

[

qua

l

In [22]:

qualitymumbai\_grouped

**.**

shape

Out[22]:

What are the best locations in Mumbai as per infrastructure?

|  |
| --- |
| qualitymumbai\_grouped[qualitymumbai\_grouped['Total infrastructure'] **==** qual |

In [23]:

Out[23]: 9

Post Office Bandra (West)

|  |  |
| --- | --- |
| Pin Code | 400050 |
| City | Mumbai |
| Airport Terminal | 0 |
| Bank | 0 |
| Bus Station | 0 |
| Business Service | 0 |
| Café | 10 |
| College Auditorium | 1 |
| Electronics Store | 1 |

9

|  |  |
| --- | --- |
| Farmers Market | 1 |
| Garden | 0 |
| Government Building | 0 |
| Gym / Fitness Center | 3 |
| Hotel | 1 |
| Indie Movie Theater | 1 |
| Light Rail Station | 0 |
| Market | 0 |
| Monument / Landmark | 0 |
| Park | 1 |
| Pharmacy | 0 |
| Playground | 0 |
| Resort | 0 |
| Restaurant | 1 |
| Shopping Mall | 1 |
| Theater | 0 |
| Train Station | 0 |
| Total infrastructure | 21 |

Which all areas lack the infrastructure facilities?

In [24]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| badquality **=** badquality | qualitymum | bai\_grouped[qualitymumbai\_ | grouped[ |  | 'Total infrastructu |
| Post Office | Pin Code | Airport Bus  City Terminal Bank Station | Business Service | Café | College E Auditorium |

Out[24]:le

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Agashi | 401301 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | Anu Shakti Nagar | 400094 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | Bassien | 401201 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 16 | Bhandup  (East) | 400042 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 19 | Bhayander (East) | 401105 | Thane | 0 | 0 | 0 | 0 | 1 | 0 |
| 20 | Boisar | 401501 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 35 | Ghansoli | 400701 | Navi Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 43 | Jacob  Circle | 400011 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 44 | Jakegram | 400606 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |

Post Pin Airport Bus Business College Ele

Office Code City Terminal Bank Station Service Café Auditorium

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 45 | Jawhar | 401603 | Thane | 0 | 0 | 1 | 0 | 0 | 0 |
| 50 | Kopri Colony | 400603 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 51 | Krishi  Utpanna Bazar | 400705 | Navi Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 52 | Mahim | 400016 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 73 | Nerul Mode | 400706 | Navi Mumbai | 0 | 0 | 0 | 0 | 1 | 0 |
| 86 | Santacruz  P&T Colony | 400029 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 90 | Sopara | 401203 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 91 | Tagore Nagar | 400083 | Mumbai | 0 | 0 | 0 | 0 | 1 | 0 |
| 92 | Talasari | 401606 | Thane | 0 | 0 | 1 | 0 | 0 | 0 |
| 97 | Umbarpada | 401102 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 98 | Uran | 400702 | Navi Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |
| 100 | Vasai East I/E | 401208 | Thane | 0 | 0 | 0 | 0 | 0 | 0 |
| 109 | Wadala | 400031 | Mumbai | 0 | 0 | 0 | 0 | 0 | 0 |

22 rows × 28 columns



# 9. Potential area for the development of infrastructure of different kinds

Looking for area in development of infrastructure in postal office.

1. Write your choice of infrastructur for which postal area has highest potential

|  |
| --- |
| yourchoiceinfra **=** 'Restaurant' *# Select your choice of infrastructue from V* badqualitychoice **=** qualitymumbai\_grouped[qualitymumbai\_grouped[yourchoicein badqualitychoice['Post Office'] |

In [25]:

2 Andheri (East) Out[25]:

4 Anu Shakti Nagar 6 B A R C

7 Ballard Estate

11 Barve Nagar ... 102 Vashi

103 Veer Jijamata Bhosle Udyan

106 Vikhroli

1. Wadala
2. Worli

Name: Post Office, Length: 65, dtype: object

2. Write your choice of area for which one infrastructure has highest potential

|  |
| --- |
| yourchoicearea **=** 'Mantralaya' *# Change with the name of postal area where* infraqualitychoice **=** qualitymumbai\_grouped[qualitymumbai\_grouped['Post Offic infraqualitychoice **=** infraqualitychoice**.**reset\_index() |

|  |
| --- |
| *# making data frame from csv file*  print("These are infrastructures with highest potential in" , yourchoiceare **for** i **in** range(len(infraqualitychoice)) : **if** (infraqualitychoice**.**iloc[i, 1] **==** 0): print(infraqualitychoice**.**iloc[i, 0]) |

In [26]: In [27]:

These are infrastructures with highest potential in Mantralaya area :

Airport Terminal

Bank

Bus Station

Business Service

College Auditorium

Farmers Market

Garden

Government Building Indie Movie Theater

Light Rail Station

Market

Monument / Landmark

Park

Pharmacy

Playground

Resort

Train Station

Recheck your choice infrastructure at your choice postal office

(Need to be done as Limitation on Range and Limit of Foursquare)

|  |
| --- |
| **from** pandas.io.json **import** json\_normalize  search\_query **=** 'School' LIMIT **=** 5 radius **=** 500 latitude **=** 19.016700 *# Latitude of your choice postal office from*  longitude **=** 72.850000 *# Longitude of your choice postal office from*  VERSION **=** 20180604  url **=** 'https://api.foursquare.com/v2/venues/search?client\_id={}&client\_secre results **=** requests**.**get(url)**.**json() results |

In [28]:

{'meta': {'code': 200, 'requestId': '5eb668501a4b0a0028d31527'}, Out[28]:

'response': {'venues': [{'id': '4f426fd2e4b0085fef97baaa',

'name': 'Auxilium Convent School',

'location': {'address': 'Wadala (W)',

'crossStreet': 'Off Katrak Rd',

'lat': 19.01620168738538,

'lng': 72.85447591009165,

'labeledLatLngs': [{'label': 'display',

'lat': 19.01620168738538,

'lng': 72.85447591009165}],

'distance': 474,

'postalCode': '400031',

'cc': 'IN',

'city': 'Mumbai',

'state': 'Mahārāshtra',

'country': 'India',

'formattedAddress': ['Wadala (W) (Off Katrak Rd)',

'Mumbai 400031',

'Mahārāshtra',

'India']},

'categories': [{'id': '4bf58dd8d48988d1ab941735',

'name': 'Student Center',

'pluralName': 'Student Centers',

'shortName': 'Student Center',

'icon': {'prefix': 'https://ss3.4sqi.net/img/categories\_v2/education/s tudentcenter\_',

'suffix': '.png'},

'primary': True}],

'referralId': 'v-1589012416',

'hasPerk': False},

{'id': '4dbb90906e810768bf48c704',

'name': 'King George School',

'location': {'address': 'Hindu Colony',

'lat': 19.02166087460775,

'lng': 72.84838333178422,

'labeledLatLngs': [{'label': 'display',

'lat': 19.02166087460775,

'lng': 72.84838333178422}],

'distance': 577,

'cc': 'IN',

'city': 'Mumbai',

'state': 'Mahārāshtra',

'country': 'India',

'formattedAddress': ['Hindu Colony', 'Mumbai', 'Mahārāshtra', 'Indi a']},

'categories': [{'id': '4bf58dd8d48988d13d941735',

'name': 'High School',

'pluralName': 'High Schools',

'shortName': 'High School',

'icon': {'prefix': 'https://ss3.4sqi.net/img/categories\_v2/building/sc hool\_',

'suffix': '.png'},

'primary': True}],

'referralId': 'v-1589012416',

'hasPerk': False},

{'id': '4f19aa3be4b0a9e6db8d7446',

'name': 'School',

'location': {'lat': 19.013098120952943,

'lng': 72.84912760380683,

'labeledLatLngs': [{'label': 'display',

'lat': 19.013098120952943,

'lng': 72.84912760380683}],

'distance': 411,

'cc': 'IN',

'city': 'Mumbai',

'state': 'Mahārāshtra',

'country': 'India',

'formattedAddress': ['Mumbai', 'Mahārāshtra', 'India']},

'categories': [{'id': '4bf58dd8d48988d1a6941735',

'name': 'Law School',

'pluralName': 'Law Schools',

'shortName': 'Law School',

'icon': {'prefix': 'https://ss3.4sqi.net/img/categories\_v2/education/l awschool\_',

'suffix': '.png'},

'primary': True}],

'referralId': 'v-1589012416',

'hasPerk': False},

{'id': '51bfeeee498e9cf446697ed5',

'name': 'Good Luck Motor Training School', 'location': {'lat': 19.01681065035709,

'lng': 72.8468601633293,

'labeledLatLngs': [{'label': 'display',

'lat': 19.01681065035709,

'lng': 72.8468601633293}],

'distance': 330,

'cc': 'IN',

'country': 'India',

'formattedAddress': ['India']},

'categories': [{'id': '4bf58dd8d48988d124951735',

'name': 'Automotive Shop',

'pluralName': 'Automotive Shops',

'shortName': 'Automotive',

'icon': {'prefix': 'https://ss3.4sqi.net/img/categories\_v2/shops/autom otive\_',

'suffix': '.png'},

'primary': True}],

'referralId': 'v-1589012416',

'hasPerk': False},

{'id': '52f4ef6511d2b5a364b6975a',

'name': 'Dadar Parsee Youths Assembly High School',

'location': {'address': 'Dadar Parsi Colony, Dadar East',

'lat': 19.01784324645996,

'lng': 72.85298156738281,

'labeledLatLngs': [{'label': 'display',

'lat': 19.01784324645996,

'lng': 72.85298156738281}],

'distance': 338,

'postalCode': '400014',

'cc': 'IN',

'city': 'Mumbai',

'state': 'Mahārāshtra',

'country': 'India',

'formattedAddress': ['Dadar Parsi Colony, Dadar East',

'Mumbai 400014',

'Mahārāshtra',

'India']},

'categories': [{'id': '4bf58dd8d48988d198941735',

'name': 'College Academic Building',

'pluralName': 'College Academic Buildings',

'shortName': 'Academic Building',

'icon': {'prefix': 'https://ss3.4sqi.net/img/categories\_v2/education/a cademicbuilding\_',

'suffix': '.png'},

'primary': True}],

'referralId': 'v-1589012416',

'hasPerk': False}]}}

In [29]:

*# assign relevant part of JSON to venues* venues **=** results['response']['venues'] *# tranform venues into a dataframe* dataframe **=** json\_normalize(venues)

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel\_lau ncher.py:4: FutureWarning: pandas.io.json.json\_normalize is deprecated, use pandas.json\_normalize instead after removing the cwd from sys.path.

|  |  |  |
| --- | --- | --- |
|  | | |
|  | *# keep only columns that include venue name, and anything that is associat* clean\_columns **=** ['name', 'categories'] **+** [col **for** col **in** dataframe**.**columns | *e* |

In [30]:

|  |  |  |
| --- | --- | --- |
|  | clean\_dataframe **=** dataframe**.**loc[:,clean\_columns]    *# function that extracts the category of the venue* **def** get\_category\_type(row): **try**:  categories\_list **=** row['categories'] **except**:  categories\_list **=** row['venue.categories']  **if** len(categories\_list) **==** 0:  **return** **None** **else**:  **return** categories\_list[0]['name']    *# filter the category for each row*  clean\_dataframe['categories'] **=** clean\_dataframe**.**apply(get\_category\_type, a  *# clean column names by keeping only last term*  clean\_dataframe**.**columns **=** [column**.**split('.')[**-**1] **for** column **in** clean\_datafra    clean\_dataframe**.**head() | x |
|  |

Out[30]: name categories address crossStreet lat lng labeledLatLngs

Auxilium

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Convent  School | Student  Center | Wadala  (W) | Off Katrak Rd | 19.016202 | 72.854476 | [{'label': 'display', 'lat':  19.01620168738538... |
| 1 | King  George School | High  School | Hindu  Colony | NaN | 19.021661 | 72.848383 | [{'label': 'display', 'lat':  19.02166087460775... |
| 2 | School | Law School | NaN | NaN | 19.013098 | 72.849128 | [{'label': 'display', 'lat':  19.01309812095294... |
| 3 | Good  Luck  Motor  Training School | Automotive Shop | NaN | NaN | 19.016811 | 72.846860 | [{'label': 'display', 'lat':  19.01681065035709... |
| 4 | Dadar  Parsee  Youths  Assembly High | College  Academic Building | Dadar  Parsi  Colony,  Dadar East | NaN | 19.017843 | 72.852982 | [{'label': 'display', 'lat':  19.01784324645996... |

School



# 10. Best place to stay within a city for vital infrastructure facilities

In [34]:

|  |  |
| --- | --- |
| quality\_infra\_mumbai2 **=** pd**.**read\_csv('es quality\_infra\_mumbai2**.**head() | sentialinfra.csv',index\_col**=**'Unnamed |
| Post Latitude Longitude VenueName  Office | VenueLatitude VenueLongitude VenueCat |

Out[34]: eg

August Cumballa Emerge

0 Kranti 18.963549 72.809989 Hill Heart 18.963778 72.809091 Ro

Marg hospital

Post Latitude Longitude VenueName VenueLatitude VenueLongitude VenueCateg Office

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Andheri (East) | 19.115883 | 72.854202 | criticare hospital | 19.118263 | 72.850639 | Hosp |
| 2 | Andheri  (West) | 19.117249 | 72.833968 | Sujay Hospital | 19.115915 | 72.834270 | Hosp |
| 3 | Antop  Hill | 19.020761 | 72.865256 | sai hospital | 19.023059 | 72.862824 | Hosp |
| 4 | Ballard Estate | 18.936651 | 72.839132 | Seaman  Hospital | 18.936524 | 72.839380 | Medical Ce |



# OR FIND ALL THE SEARCH QUERY IN ALL POSTAL OFFICE

|  |
| --- |
| search\_query2**=** ['Hospital','Food', 'Hotel', 'Shopping Mall', 'Pharmacy', 'Metro Station', 'Train Station', 'ATM', 'Office', 'Police Station', 'School', 'College & University  ]  categoryId **=** ['4bf58dd8d48988d104941735','4d4b7105d754a06374d81259', '4bf58d '4bf58dd8d48988d1fd931735', '4bf58dd8d48988d129951735', '52f2ab '4bf58dd8d48988d10a951735', '50be8ee891d4fa8dcc7199a7','4bf58dd '4bf58dd8d48988d163941735'] |

In [32]:

|  |
| --- |
| **from** pandas.io.json **import** json\_normalize radius **=** 500 VERSION **=** 20180604  *# Quality Infrastructure*  search\_query2 **=** 'Park'  categoryId **=** '4bf58dd8d48988d163941735' LIMIT **=** 1 |

In [33]:

In [44]:

|  |
| --- |
| **def** getNearbyVenues(names, lat1, long1, radius):  venues\_list**=**[] **for** name, lat, lng **in** zip(names, lat1, long1):    *# create the API request URL*  url1 **=** 'https://api.foursquare.com/v2/venues/search?client\_id={}&cl  *# make the GET request*  results **=** requests**.**get(url1)**.**json()["response"]["venues"]  *# return only relevant information for each nearby venue* venues\_list**.**append([( name, lat, lng, v['name'],  v['location']['lat'], v['location']['lng'], v['categories'][0]['name']) **for** v **in** results])    nearby\_venues **=** pd**.**DataFrame([item **for** venue\_list **in** venues\_list    **return**(nearby\_venues) |

**for**

In [45]:

|  |  |  |
| --- | --- | --- |
|  | names**=**clean\_df['Post Office'] latitudes**=**clean\_df['Latitude'] longitudes**=**clean\_df['Longitude']  all\_venues **=** getNearbyVenues(names,latitudes, longitudes, radius ) |  |
|  |

In [46]:

|  |
| --- |
| *# convert the venues list into a new DataFrame*  *# define the column names*  all\_venues**.**columns **=** ['Post Office','Latitude', 'Longitude','VenueName',    print(all\_venues**.**shape) all\_venues**.**head() |

'Ve

(23, 7)

Out[46]: Post Latitude Longitude VenueName VenueLatitude VenueLongitude VenueCat

Office

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Bandra (West) | 19.058336 | 72.830267 | D'Monte  Park  Recreation  Centre | 19.059630 | 72.825995 | Event |
| 1 | Bangur  Nagar | 19.168814 | 72.833678 | Jogger's park | 19.164701 | 72.835153 |  |
| 2 | Colaba | 18.915091 | 72.825969 | Harish  Mahindra  Children's Park | 18.914867 | 72.823688 |  |
| 3 | Cumballa  Hill | 18.969307 | 72.806538 | Amarsons  Park, Breach Candy | 18.972317 | 72.806327 |  |
| 4 | Ghatkopar  (West) | 19.089719 | 72.904597 | Joggers park,L B S marg.  Ghatkopar west | 19.087582 | 72.901977 |  |



Done for Hospital, Food, Hotel, Shopping Mall, Pharmacy,Metro Station, Train Station,

ATM, Office, Bus Station, Bank,Market, Police Station, School, College, Park

In [47]:

quality\_infra\_mumbai

**=**

all\_venues

**.**

copy

()

In [48]:

*#quality\_infra\_mumbai2 = quality\_infra\_mumbai.copy()*

|  |  |  |
| --- | --- | --- |
| quality\_infra\_mumbai2 **=** quality\_inf quality\_infra\_mumbai2**.**shape | ra\_mumbai2**.** | append(quality\_infra\_mumbai, |
| (1019, 7) |  |  |
| quality\_infra\_mumbai2**.**tail(30) |  |  |
| Post Office Latitude Longitude | VenueName | VenueLatitude VenueLongitude V |

In [49]:

Out[49]: In [35]:

Out[35]:e

JNPT Town Government

989 Ship 18.938771 72.835335 Dental 18.939510 72.836700

College

Post Office Latitude Longitude VenueName VenueLatitude VenueLongitude Ve

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 990 | Nerul Mode | 19.047375 | 73.019300 | Dr. D Y Patil  Dental  College & Hospital | 19.045169 | 73.024025 | M |
| 991 | Ganeshpuri | 18.938771 | 72.835335 | Government  Dental College | 18.939510 | 72.836700 |  |
| 992 | Papdi | 18.938771 | 72.835335 | Government  Dental College | 18.939510 | 72.836700 |  |
| 993 | Vasai Road  East | 18.938771 | 72.835335 | Government  Dental College | 18.939510 | 72.836700 |  |
| 994 | B A R C | 19.016700 | 72.850000 | Akbar  Peerbhoy College | 19.020737 | 72.850264 | Co |
| 995 | NAD Karanja | 19.077800 | 72.894400 | S K Somaiya  College of  Arts,  Science And Commerce | 19.077490 | 72.899439 |  |
| 996 | Bandra (West) | 19.058336 | 72.830267 | D'Monte  Park  Recreation  Centre | 19.059630 | 72.825995 |  |
| 997 | Bangur Nagar | 19.168814 | 72.833678 | Jogger's park | 19.164701 | 72.835153 |  |
| 998 | Colaba | 18.915091 | 72.825969 | Harish  Mahindra  Children's Park | 18.914867 | 72.823688 |  |
| 999 | Cumballa Hill | 18.969307 | 72.806538 | Amarsons  Park, Breach Candy | 18.972317 | 72.806327 |  |
| 1000 | Ghatkopar  (West) | 19.089719 | 72.904597 | Joggers park,L B S marg.  Ghatkopar west | 19.087582 | 72.901977 |  |
| 1001 | Goregaon (West) | 19.163328 | 72.841200 | Guru Nanak Park | 19.159897 | 72.839256 |  |
| 1002 | Jogeshwari (East) | 19.136989 | 72.864942 | Hemant Karkare Park | 19.137349 | 72.868155 |  |
| 1003 | Juhu | 19.107021 | 72.827528 | Pushpa Narsee Park | 19.104546 | 72.830918 |  |
| 1004 | Kharodi | 19.191684 | 72.814098 | Kalpana  Park | 19.196491 | 72.814888 |  |
| 1005 | Malabar Hill | 18.958162 | 72.803367 | Priyadarshini Park | 18.957515 | 72.799614 |  |
| 1006 | Mandpeshwar | 19.235842 | 72.852837 | Joggers park | 19.237044 | 72.850103 |  |
| 1007 | Motilal Nagar | 19.158995 | 72.840197 | Guru Nanak Park | 19.159897 | 72.839256 |  |

Post Office Latitude Longitude VenueName VenueLatitude VenueLongitude Ve

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1008 | Pant Nagar | 19.086821 | 72.915603 | Arun Vaidya  Park | 19.085515 | 72.912799 |
| 1009 | Parel | 19.009482 | 72.837661 | Pramod  Mahajan Kala Park | 19.013319 | 72.837708 |
| 1010 | Rajawadi | 19.078536 | 72.900195 | joggers park | 19.077364 | 72.903292 |
| 1011 | Santacruz  (West) | 19.084877 | 72.834973 | Muktananda Park | 19.081794 | 72.832387 |
| 1012 | Ghansoli | 19.119331 | 72.999510 | CIDCO Park | 19.121827 | 73.003010 |
| 1013 | Turbhe | 19.076165 | 73.017662 | Cidco Nature Park | 19.076810 | 73.012673 |
| 1014 | Bhayandar | 19.197152 | 72.811366 | Kalpana  Park | 19.196491 | 72.814888 |
| 1015 | Boisar | 19.209144 | 72.859183 | Joggers  Park (Thakur  Zagadu  Singh  Charitable t... | 19.210427 | 72.863646 |
| 1016 | Mira | 19.282057 | 72.874144 | sai baba nagar park | 19.283890 | 72.870278 |
| 1017 | Thane (East) | 19.030826 | 73.019854 | mother teresa park nerul east | 19.028223 | 73.021927 |
| 1018 | Vesava | 19.133736 | 72.814877 | Park | 19.137666 | 72.815483 |

(Versova)

In [51]:



quality\_infra\_mumbai2

**.**

to\_csv

(

'essentialinfra.csv'

)

# CLUSTERING THE IMPORTED DATASET

In [36]:

quality\_infra\_mumbai2

[

'VenueCategory'

]

**.**

unique

()

array(['Emergency Room', 'Hospital', 'Medical Center', 'Eye Doctor', Out[36]:

'Veterinarian', "Doctor's Office", 'Bus Line', 'Sandwich Place',

'Fast Food Restaurant', 'Fried Chicken Joint', 'Bagel Shop',

'Snack Place', 'Seafood Restaurant', 'Indian Restaurant',

'Food Court', 'Chinese Restaurant', 'Food Truck', 'Cafeteria',

'Pizza Place', 'Café', 'Middle Eastern Restaurant',

'Vegetarian / Vegan Restaurant', 'Asian Restaurant', 'Restaurant',

'Deli / Bodega', 'Burger Joint', 'Breakfast Spot',

'Comfort Food Restaurant', 'Dessert Shop', 'Hotel',

'Bed & Breakfast', 'Motel', 'Boarding House', 'Resort', 'Hostel',

'Shopping Mall', 'Building', 'Pharmacy', 'Light Rail Station',

'Metro Station', 'Train Station', 'Platform', 'Train', 'ATM',

'Bank', 'Campaign Office', 'Office', 'Coworking Space', 'Tech Startup', 'Conference Room', 'Bus Station', 'Market',

'Flea Market', 'Police Station', 'High School',

'Elementary School', 'Driving School', 'School',

'Adult Education Center', 'Middle School', 'Student Center',

'College Science Building', 'College Academic Building',

'College Cafeteria', 'College Administrative Building',

'College & University', 'General College & University',

'College Classroom', 'Medical School',

'College Engineering Building', 'College Math Building',

'College Arts Building', 'College Auditorium', 'Community College',

'Law School', 'College Technology Building',

'College Communications Building', 'Event Space', 'Park', 'Garden'], dtype=object)

|  |
| --- |
| *# one hot encoding*  quality\_mumbai\_onehot **=** pd**.**get\_dummies(quality\_infra\_mumbai2[['VenueCategory  *# add postal, borough and neighborhood column back to dataframe*  quality\_mumbai\_onehot['Post Office'] **=** quality\_infra\_mumbai2['Post Office']  *# move postal, borough and neighborhood column to the first column* fixed\_columns **=** list(quality\_mumbai\_onehot**.**columns[**-**1:]) **+** list(quality\_mumb quality\_mumbai\_onehot **=** quality\_mumbai\_onehot[fixed\_columns]    print(quality\_mumbai\_onehot**.**shape) quality\_mumbai\_onehot**.**head() |

In [37]:

(1019, 81)

Out[37]: Post Adult Asian Bagel Bed & Boarding Breakfast

Office ATM Education Restaurant Shop Bank Breakfast House Spot Build Center

August

1. Kranti 0 0 0 0 0 0 0 0

Marg

1. Andheri 0 0 0 0 0 0 0 0

(East)

1. Andheri 0 0 0 0 0 0 0 0

(West)

1. Antop 0 0 0 0 0 0 0 0

Hill

1. Ballard 0 0 0 0 0 0 0 0

Estate

1. rows × 81 columns



|  |
| --- |
| qualitymumbai\_grouped **=** quality\_mumbai\_onehot**.**groupby(["Post Office"])**.**sum(    print(qualitymumbai\_grouped**.**shape) qualitymumbai\_grouped**.**head() |

In [38]:

(122, 81)

Out[38]: Post Adult Asian Bagel Bed & Boarding Breakfast

Office ATM Education Restaurant Shop Bank Breakfast House Spot Build Center

Aarey

1. Milk 0 0 0 0 0 0 0 0

Colony

1. Airoli 0 0 0 0 0 0 0 0

Mode

1. Andheri 1 0 0 0 1 0 0 0

(East)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Post Office | ATM | Adult  Education Center | Asian  Restaurant | Bagel Shop | Bank | Bed & Breakfast | Boarding House | Breakfast Spot | Build |

1. Andheri 1 0 0 0 1 0 0 0

(West)

1. Antop 1 0 0 1 1 0 0 0

Hill

1. rows × 81 columns



In [39]: qualitymumbai\_grouped['Total infrastructure'] **=** qualitymumbai\_grouped[qual

In [40]: qualitymumbai\_groupedmax **=** qualitymumbai\_grouped[qualitymumbai\_grouped['Tot print("Best place to stay within a city for vital infrastructure facilities qualitymumbai\_groupedmax[['Post Office', 'Total infrastructure']] print(qualitymumbai\_groupedmax**.**shape)

Best place to stay within a city for vital infrastructure facilities : (19, 82)

In [41]: mumbai\_merged2 **=** qualitymumbai\_grouped**.**copy() mumbai\_merged2 **=** mumbai\_merged2**.**join(clean\_df[["Pin Code",'Latitude', 'Long

|  |
| --- |
| fixed\_columns **=** list(mumbai\_merged2**.**columns[**-**3:]) **+** list(mumbai\_merged2**.**  mumbai\_merged2 **=** mumbai\_merged2[fixed\_columns]    print(mumbai\_merged2**.**shape) mumbai\_merged2**.**head() |

In [42]: colu

(122, 85)

Out[42]: Pin Post Adult Asian Bagel B

Code Latitude Longitude Office ATM Education Restaurant Shop Bank Break Center

Aarey

1. 400065 19.156129 72.870722 Milk 0 0 0 0 0

Colony

1. 400708 19.172979 73.003532 Airoli 0 0 0 0 0

Mode

1. 400069 19.115883 72.854202 Andheri 1 0 0 0 1

(East)

1. 400058 19.117249 72.833968 Andheri 1 0 0 0 1

(West)

1. 400037 19.020761 72.865256 Antop 1 0 0 1 1

Hill

1. rows × 85 columns



|  |  |  |
| --- | --- | --- |
|  | | |
|  | *# set number of clusters* kclusters **=** 3    mumbai\_2\_grouped\_clustering **=** mumbai\_merged2[["Total infrastructure"]] |  |

In [43]:

|  |  |  |
| --- | --- | --- |
|  | *# run k-means clustering*  kmeans **=** KMeans(n\_clusters**=**kclusters, random\_state**=**0)**.**fit(mumbai\_2\_grouped\_c  *# check cluster labels generated for each row in the dataframe* kmeans**.**labels\_[0:10] |  |
|  |

array([1, 1, 2, 2, 2, 1, 1, 2, 2, 1], dtype=int32) Out[43]:

|  |
| --- |
| *# create a new dataframe that includes the cluster as well as the top 10 ven* mumbai\_mergedfinal **=** mumbai\_merged2**.**copy()  *# add clustering labels*  mumbai\_mergedfinal["Cluster Labels"] **=** kmeans**.**labels\_ print(mumbai\_mergedfinal**.**shape) mumbai\_mergedfinal**.**head() *# check the last columns!* |

In [44]:

(122, 86)

Out[44]: Pin Post Adult Asian Bagel B

Code Latitude Longitude Office ATM Education Restaurant Shop Bank Break Center

Aarey

1. 400065 19.156129 72.870722 Milk 0 0 0 0 0

Colony

1. 400708 19.172979 73.003532 Airoli 0 0 0 0 0

Mode

1. 400069 19.115883 72.854202 Andheri 1 0 0 0 1

(East)

1. 400058 19.117249 72.833968 Andheri 1 0 0 0 1

(West)

1. 400037 19.020761 72.865256 Antop 1 0 0 1 1

Hill

1. rows × 86 columns



# 11. Exploratory Visualization 2

|  |
| --- |
| *# Set manually to get proper fit in the map* address **=** 'Mumbai' latitude **=** 19.0760 longitude **=** 72.8777  print('The geograpical coordinate of {} are {}, {}.'**.**format(address, latitud |

In [45]:

The geograpical coordinate of Mumbai are 19.076, 72.8777.

|  |  |  |
| --- | --- | --- |
|  | | |
|  | map\_clusters **=** folium**.**Map(location**=**[latitude, longitude], zoom\_start**=**11)  *# set color scheme for the clusters* x **=** np**.**arange(kclusters)  rainbow **=** [ 'red', 'blue', 'orange', 'darkgreen', 'darkblue  *# add markers to map* markers\_colors **=** []  **for** lat, lng, label1,common, cluster **in** zip(mumbai\_mergedfinal['Latitude'] labelnew **=** 'Post office : {} , Total infrastructure : {}'**.**format(labe label **=** folium**.**Popup( labelnew, parse\_html**=True**) folium**.**CircleMarker(  [lat, lng], | '  , l |

In [46]:

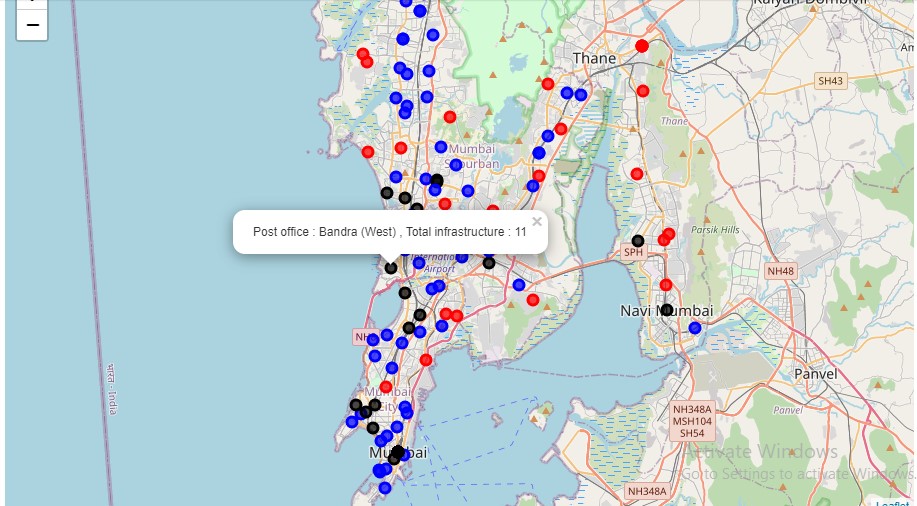
|  |  |  |
| --- | --- | --- |
|  | radius**=**5, popup**=**label, color**=**rainbow[cluster**-**1], fill**=True**,  fill\_color**=**rainbow[cluster**-**1], fill\_opacity**=**0.7, parse\_html**=False**)**.**add\_to(map\_clusters) map\_clusters |  |
|  |

Out[46]: Make this Notebook Trusted to load map: File -> Trust Notebook

# CLUSTERING THE IMPORTED DATASET

|  |
| --- |
| **from** IPython.display **import** Image **from** IPython.core.display **import** HTML  Image(url**=** "https://raw.githubusercontent.com/RohitLearner/IBM-Applied-D.S-C |

In [47]:

Out[47]: 

# 12. Examine Clusters

Cluster 0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| mumbai\_mergedfinal**.**loc[mumbai\_mergedfinal['Cluster Labels'] **==** 0] | | | | |  |
| Pin  Code Latitude Longitude | Post  Office ATM | Adult  Education Center | Asian  Restaurant | Bagel Shop | Bank |

In [48]:

Out[48]:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 12 | 400050 | 19.058336 | 72.830267 | Bandra (West) | 1 | 0 | 0 | 0 | 1 |
| 18 | 400028 | 18.938771 | 72.835335 | Bhavani  Shankar Road | 1 | 0 | 0 | 0 | 1 |
| 24 | 400093 | 19.115287 | 72.861808 | Chakala MIDC | 1 | 1 | 0 | 0 | 1 |
| 25 | 400071 | 19.061213 | 72.897591 | Chembur | 1 | 0 | 0 | 0 | 1 |

1. 400039 18.938771 72.835335 Council 1 0 0 0 1

Hall

1. 400026 18.969307 72.806538 Cumballa 1 0 0 0 1

Hill

1. 400014 19.019282 72.842876 Dadar 1 0 0 0 1
2. 400074 18.938771 72.835335 F C I 1 0 0 0 1

Mumbai

1. 401206 18.938771 72.835335 Ganeshpuri 1 0 0 0 1
2. 400086 19.089719 72.904597 Ghatkopar 1 0 0 0 1

(West)

1. 400004 18.954317 72.817908 Girgaon 1 0 0 0 1
2. 400007 18.964447 72.813573 Grant Road 1 0 0 0 1
3. 400023 18.934170 72.832520 Hutatma 1 0 0 0 1

Chowk

1. 400076 18.938771 72.835335 I I T 1 0 0 0 1

Mumbai

1. 400059 18.938771 72.835335 J B Nagar 1 0 0 0 1
2. 400707 18.938771 72.835335 JNPT Town 1 0 0 0 1

Ship

51 400049 19.107021 72.827528 Juhu 1 0 0 0 1

61 400016 19.042314 72.839834 Mahim 1 0 0 0 1

67 401403 18.938771 72.835335 Manor 1 0 0 0 1

70 400019 19.027436 72.850147 Matunga 1 0 0 0 1

1. 400008 18.969586 72.819315 Mumbai 1 0 0 0 1

Central

1. 400001 18.938771 72.835335 Mumbai G 1 0 0 0 1

P O

1. 400087 18.938771 72.835335 N I T I E 1 0 0 0 1

83 400602 19.083579 72.888652 Naupada 1 0 0 0 1

88 401207 18.938771 72.835335 Papdi 1 0 0 0 1

92 400035 18.938771 72.835335 Rajbhavan 1 0 0 0 1

1. 400055 19.081522 72.841756 Santacruz 1 0 0 0 1

(East)

1. 400054 19.084877 72.834973 Santacruz 0 0 0 0 2

(West)

Shivaji

101 400043 19.114100 72.862290 Nagar 1 1 0 0 1

(Kurla)

107 400603 19.030826 73.019854 Thane 1 0 0 0 1

(East)

109 400034 18.938771 72.835335 Tulsiwadi 1 0 0 0 1

112 401210 18.938771 72.835335 Vasai Road 1 0 0 0 1

East

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 113 | 400703 | 19.075713 | 73.000354 | Vashi | 1 | 0 | 0 | 0 | 1 |
| 114 | 400027 | 18.938771 | 72.835335 | Veer  Jijamata  Bhosle Udyan | 1 | 0 | 0 | 0 | 1 |
| 118 | 400057 | 19.096288 | 72.848380 | Vile Parle (East) | 0 | 0 | 0 | 0 | 2 |
| 119 | 400056 | 19.103873 | 72.840290 | Vile Parle (West) | 0 | 0 | 0 | 0 | 1 |

36 rows × 86 columns



# Cluster 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| mumbai\_mergedfinal**.**loc[mumbai\_mergedfinal['Cluster Labels'] **==** 1] | | | | |  |
| Pin  Code Latitude Longitude | Post  Office ATM | Adult  Education Center | Asian  Restaurant | Bagel Shop | Bank |

In [49]: Out[49]:

1. 400065 19.156129 72.870722 Aarey Milk 0 0 0 0 0

Colony

1. 400708 19.172979 73.003532 Airoli Mode 0 0 0 0 0
2. 400094 19.037528 72.928146 Anu Shakti 0 0 0 0 0

Nagar

1. 401302 19.202322 73.002537 Arnala 1 0 0 0 1

9 400608 19.202322 73.002537 Balcum 1 0 0 0 1

14 400084 19.095283 72.900178 Barve 1 0 0 0 0

Nagar

17 400042 19.148557 72.947066 Bhandup 1 0 0 0 1

(East)

19 401101 19.197152 72.811366 Bhayandar 0 0 0 0 0 36 400701 19.119331 72.999510 Ghansoli 1 0 0 0 1

46 400011 18.981053 72.826786 Jacob 0 0 0 0 2

Circle

48 401603 19.916700 73.233300 Jawhar 0 0 0 0 0

50 400102 19.136394 72.837382 Jogeshwari 0 0 0 0 1

(West)

53 400605 19.202322 73.002537 Kalwa 1 0 0 0 1

57 400095 19.191684 72.814098 Kharodi 1 0 0 0 0

Krishi

59 400705 19.080266 73.021173 Utpanna 0 0 0 0 0

Bazar

77 400082 19.177778 72.938351 Mulund 0 0 0 0 1

Colony

85 400706 19.047375 73.019300 Nerul 1 0 0 0 1

Mode

94 400099 19.099595 72.867284 Sahar 1 0 0 0 1

96 400607 19.202322 73.002537 Sandoz 1 0 0 0 1

Baug

100 400015 18.998780 72.854422 Sewri 0 0 0 0 1

1. 400083 19.117628 72.932201 Tagore 1 0 0 0 1

Nagar

1. 401606 19.916700 73.233300 Talasari 0 0 0 0 0 105 400033 19.028166 72.868325 Tank Road 0 0 0 0 1 110 400613 19.076165 73.017662 Turbhe 0 0 0 0 0

111 400702 18.880800 72.938600 Uran 0 0 0 0 0

115 400061 19.133736 72.814877 Vesava 0 0 0 0 1

(Versova)

120 400031 19.026919 72.875934 Wadala 1 0 0 0 0

27 rows × 86 columns



# Cluster 2

In [65]:

Out[65]:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| mumbai\_mergedfinal**.**loc[mumbai\_mergedfinal['Cluster Labels'] **==** 2] | | |  |  |
| Pin  Code Latitude Longitude Post Office ATM | Adult  Education Center | Asian  Restaurant | Bagel Shop | Ba |

n

1. 400069 19.115883 72.854202 Andheri 1 0 0 0

(East)

1. 400058 19.117249 72.833968 Andheri 1 0 0 0

(West)

1. 400037 19.020761 72.865256 Antop Hill 1 0 0 1
2. 400036 18.963549 72.809989 August Kranti 1 0 0 0

Marg

1. 400085 19.016700 72.850000 B A R C 1 0 0 0
2. 400038 18.936651 72.839132 Ballard Estate 0 0 0 0
3. 400051 19.061657 72.849811 Bandra (East) 1 0 0 0

13 400090 19.168814 72.833678 Bangur Nagar 1 0 0 0

1. 400611 19.018987 73.039095 Belapur 0 0 0 0
2. 400078 19.143868 72.938433 Bhandup 1 0 0 0
3. 401501 19.209144 72.859183 Boisar 0 0 0 0
4. 400066 19.225108 72.850206 Borivli (East) 0 0 0 0
5. 400092 19.225108 72.850206 Borivli (West) 0 0 0 0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Post Office | ATM | Adult  Education Center | Asian  Restaurant | Bagel Shop | Ban |

1. 400091 19.231468 72.840574 Borivli HO 1 0 0 0
2. 400009 18.964131 72.841638 Chinch 0 0 0 0

Bunder

1. 400005 18.915091 72.825969 Colaba 1 0 0 0
2. 400068 19.249450 72.859621 Dahisar 1 0 0 0
3. 400013 18.993260 72.831012 Delisle Road 1 0 0 0
4. 400017 19.044463 72.858618 Dharavi 1 0 0 0
5. 400063 19.169262 72.855255 Goregaon 1 0 0 0

(East)

1. 400062 19.163328 72.841200 Goregaon 1 0 0 0

(West)

47 400606 19.132612 72.932013 Jakegram 1 0 0 0

49 400060 19.136989 72.864942 Jogeshwari 1 0 0 0

(East)

52 400002 18.949258 72.827938 Kalbadevi 1 0 0 0

1. 400101 19.206479 72.838451 Kandivli 1 0 0 0

(East)

1. 400067 19.206479 72.838451 Kandivli 1 0 0 0

(West)

1. 400052 19.069658 72.839894 Khar 1 0 0 0

58 400603 19.132612 72.932013 Kopri Colony 1 0 0 0

60 400070 19.065280 72.879381 Kurla 1 0 0 0

1. 400006 18.958162 72.803367 Malabar Hill 1 0 0 0
2. 400097 19.186022 72.856318 Malad (East) 1 0 0 0
3. 400064 19.184013 72.841216 Malad (West) 1 0 0 0
4. 400103 19.235842 72.852837 Mandpeshwar 1 0 0 0
5. 400003 18.955056 72.834792 Mandvi 0 0 0 0
6. 400032 18.927662 72.827039 Mantralaya 1 0 0 0
7. 400020 18.945670 72.823781 Marine Lines 1 0 1 0
8. 400010 18.968052 72.840012 Mazgaon 1 0 0 0
9. 401104 19.282057 72.874144 Mira 1 0 0 0
10. 401107 19.187896 72.836596 Mira Road 1 0 1 0
11. 400104 19.158995 72.840197 Motilal Nagar 1 0 0 0
12. 400081 19.170472 72.961045 Mulund (East) 0 0 0 0
13. 400080 19.171972 72.951196 Mulund 1 0 0 0

(West)

1. 400704 19.077800 72.894400 NAD Karanja 1 0 0 0
2. 400021 18.925951 72.823208 Nariman Point 1 0 0 0

84 400024 19.001048 72.819427 Nehru Nagar 1 0 0 0

Adult Asian Bagel

Post Office ATM Education Restaurant Shop Ban

Center

1. 401304 18.927329 72.822197 Nirmal 1 0 0 0
2. 400075 19.086821 72.915603 Pant Nagar 1 0 0 0
3. 400012 19.009482 72.837661 Parel 0 0 0 0
4. 400025 19.014881 72.827956 Prabhadevi 1 0 0 0
5. 400077 19.078536 72.900195 Rajawadi 1 0 0 0

93 400096 19.125356 72.875170 SEEPZ 1 0 0 0

95 400072 19.108221 72.883582 Saki Naka 1 0 0 0

99 400029 19.092313 72.844635 Santacruz 1 0 0 0

P&T Colony

102 400022 19.046521 72.863283 Sion 1 0 0 0

Telecom

106 400088 19.046900 72.918462 Factory 0 0 0 0

Deonar

108 400089 19.069238 72.897846 Tilak Nagar 1 0 0 0

1. 400098 19.109092 72.860771 Vidyanagari 1 0 0 0
2. 400079 19.111480 72.928021 Vikhroli 1 0 0 0

121 400018 19.011696 72.818070 Worli 1 0 0 0

59 rows × 86 columns



13. Observations:

Most of the infrastructures are concentrated in the Southern areas of Mumbai city, with the highest number in cluster 0 and moderate number in cluster 2. On the other hand, cluster 1 has a very low number of infrastructures in the neighborhoods. This represents a great opportunity and high potential areas to open new infrastructures as it is very little to no competition from existing varied infrastructures. Meanwhile, one can specifically check the infrastructure of choice against the postal office choice area.

A person who is planning to build infrastructure with unique selling propositions and lives prosperously to stand out from the competition can also open new infrastructures in neighborhoods in cluster 2 with moderate competition and supporting adequate no. of infrastructures. Lastly, people with planning to settle in the city are advised to start in cluster 0 which already has a high concentration of infrastructures.

# 14. Acknowledgement

Conclusion:

In this project, I have gone through the process of identifying the business problems, specifying the data required, extracting and preparing the data, visualizing the results, performing machine learning by clustering the data into 3 clusters based on their frequency similarities, tackling and reaching to a definitive solution to business problems (mentioned in results). Lastly, the project is providing recommendations to the relevant stakeholders i.e. business developers regarding the best locations to open a new infrastructure. The project also provides visitors and immigrants to the city regarding postal office areas for growth and living prosperously.

In [ ]: