2019

The right spot for a bakery shop

FINDING THE RIGHT NEIGHBORHOOD

ALFONSO DE LOS REYES

Business requirement

1. Our client, a local chef would like to appoint a data science agency (the "Contractor") with the necessary expertise to design and conduct analysis to find the perfect location for a new **bakery shop** to be opened in **Singapore** in 6 months.

2. The bakery shop shouldn't be a typical bakery like the currently existent chains in the country, the place should offer a new experience in more quiet areas in which consumers appreciate bakery and coffee and therefore ambience is important. The client is still undecided about the positioning of the place in terms of a bakery shop or a café instead.

3. Context: Singapore is dense in terms of food offering; few cities can offer the amount of variety of restaurants packed in so small spaces. According to the local authorities, there are around 8000 food service outlets in the country and therefore having a clear idea of the marketing mix is key to be successful.

- **Objective**: **Understand which neighborhoods or areas are more appropriate to open a shop.**
- **Area**: **The entire Singapore.**

Restrictions / options	**Measure**
1. The place could be café or a bakery shop	*Frequency of bakery shops and cafes*
2. It should be accessible	*Near a metro station*
3. It should be a good area	*parks / gardens as measure*
4. Avoid high rental	*Avoid "business" areas like downtown*

Data sources:

1. Postal codes from Singapore http://download.geonames.org/export/zip/: The entire number of postal codes that exist in Singapore, beyond the district area code. The file contains 121154 total postal codes with longitude and latitude information.
2. Foursquares data of venues.

General methodology to be used:

1. Take a sample from total postal codes database (320 instances).
2. Plot the sample in Singapore.
3. Use the sample data to extract the venues information from the Foursquare database.
4. Process the venues information in terms of categories and group by area.
5. Use the processed information to form clusters using K-means.
6. Plot clusters
7. Understand the clusters.
8. See if any clusters are suitable to provide a solution to the client.

Step by step approach (Second week)

Install packages and import modules

```
〉 !conda install -c conda-forge geopy --yes;
〉 !conda install -c conda-forge geocoder --yes;
〉 !conda install -c conda-forge folium=0.5.0 --yes;
...
〉 from zipfile import ZipFile
〉 from sklearn.cluster import KMeans # import k-means from clustering stage
〉 import folium
〉 import numpy as np  # useful for many scientific computing in Python
〉 import pandas as pd # primary data structure library
〉 pd.set_option('display.max_columns', None)
〉 pd.set_option('display.max_rows', None)
〉 import requests
〉 import io
〉 import matplotlib.cm as cm
〉 import matplotlib.colors as colors
〉 import json
```

Downloading zip file and extracting it

```
〉 !wget -q -O 'zipsing' http://download.geonames.org/export/zip/SG.zip
...
〉 with ZipFile('zipsing', 'r') as zip:
〉     zip.extract('SG.txt')
```

Observations: File stored in the local environment as SG.txt

Parsing the txt file and renaming columns

```
〉 data = pd.read_csv('SG.txt',sep=' ', header=None)
〉 data.head(3)
```

output ┐

	0	1	2	3	4	5	6	7	8	9	10	11
0	SG	18906	Straits Boulevard	NaN	NaN	NaN	NaN	NaN	NaN	1.2758	103.8496	NaN
1	SG	18907	Straits Boulevard	NaN	NaN	NaN	NaN	NaN	NaN	1.2749	103.8517	NaN
2	SG	18910	Marina Gardens Drive	NaN	NaN	NaN	NaN	NaN	NaN	1.2796	103.8690	NaN

Observations: We need to drop and rename some columns

Dropping and renaming columns

```
〉 data.drop([0,3,4,5,6,7,8,11], axis = 1, inplace=True)
〉 print(data.shape)
```

output ┐

```
(121154, 4)
```

Observations: The file is huge, we need to take a random sample

Taking a sample of the original file, cleaning and sorting

```
〉 pcsubset = data.sample(n=320) #Postal code subset of the large database
〉 pcsub = pcsubset[:]
...
〉 pcsub.to_csv('sampledata.csv') #Saving the dataframe to the local environment
```

Observations: The kernel died in different occasions and as the sample is random, I had issues with the output later

Sorting values, resetting index and visualizing the table

```
〉 pcsub.sort_values(['area'], ascending=True, axis=0, inplace=True)
...
〉 pcsub = pcsub.reset_index(drop=True)
...
〉 pcsub.head()
```

output ┐

	postalcode	area	latitude	longitude
0	677851	Almond Street	1.3725	103.7724
1	69878	Amoy Street	1.2801	103.8468
2	544616	Anchorvale Crescent	1.3985	103.8921
3	567920	Ang Mo Kio Avenue 2	1.3774	103.8350
4	561203	Ang Mo Kio Avenue 3	1.3676	103.8446

Observations: Everything seems to be in order to plot the data

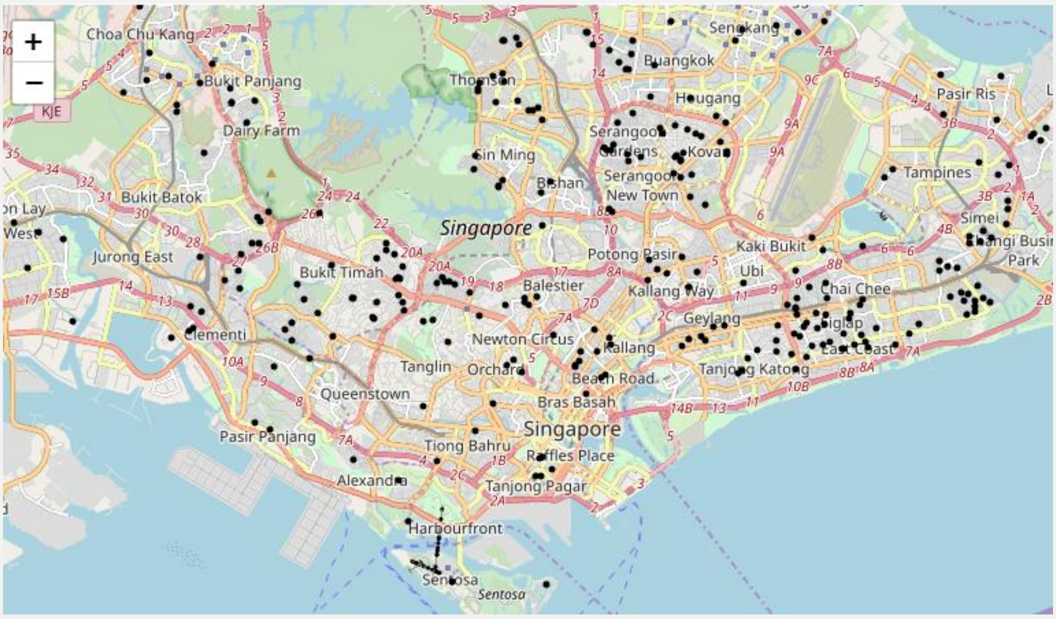
Define map coordinates and loop to add markers from random postal codes

```
〉 # Singapore latitude and longitude values
〉 latitude = 1.290270
〉 longitude = 103.8198
〉 singmap = folium.Map(location=[latitude, longitude], zoom_start=12)
〉 fgroup = folium.map.FeatureGroup() # instantiate a feature group for the incidents in the
  dataframe
```

Plot the map

```
〉 for lat, lng, in zip(pcsub['latitude'], pcsub['longitude']):
〉     fgroup.add_child(
〉         folium.features.CircleMarker(
〉             [lat, lng],
〉             radius=1, # define how big you want the circle markers to be
〉             color="black",
〉             fill=True,
〉             fill_color="black",
〉             fill_opacity=0.5
〉         )
〉     )
〉
〉 # add incidents to map
〉 singmap.add_child(fgroup)
```

output ┐



Observations: The black dots represents the 320 postal codes plotted. Why I choose to plot 320 points instead of 28 districts, because 28 districts is simply too small to cluster the data.

Defining ID / secret / Version to use in Foursquare

```
CLIENT_ID = 'HJLPKNFYSMPOPHISTM0LKEJHZC1MK2ENDRP1RWT5NAWSYMCN' # your Foursquare ID
CLIENT_SECRET = 'G1I1EM0WQWTCR0JH0H334RF5FJMZMTJGXSLICV01B2P2MMMU' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version
```

Create function to loop through postal codes and get nearby venues information

```
LIMIT = 100 # limit of number of venues returned by Foursquare API

def getNearbyVenues(names, latitudes, longitudes, radius=1000):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            LIMIT)
```

```
        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']

        # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

    nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in
venue_list])
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

Apply function using the sample data postal codes

```
snv = getNearbyVenues(names=pcsubset['area'], latitudes=pcsubset['latitude'],
longitudes=pcsubset['longitude']) #Singapore nearby venues
```

Observations: This took a while and some errors were encountered with certain venues in other cases

Shape of venues data, backup and looking the table

```
print(snv.shape)
bnv = snv[:]
bnv.head(3)
```

output ┐

(17525, 7)

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Almond Street	1.3725	103.7724	正味 卤鸭面 Braised Duck	1.376867	103.773162	Noodle House
1	Almond Street	1.3725	103.7724	Chestnut Nature Park	1.371641	103.779792	Nature Preserve
2	Almond Street	1.3725	103.7724	Dairy Farm Nature Park	1.364448	103.776364	Nature Preserve

Observations: Another huge table of 17525 rows

Saving into the local environment the Foursquare data

To not repeat the process in case something happens

```
bnv.to_csv('foursquare.csv') #Saving the dataframe as CSV in case the kernel dies
```

One hot encoding / Preparing data to use in the K-means clustering

```
〉 ctgrtable = pd.get_dummies(bnv[['Venue Category']], prefix="", prefix_sep="")
〉 ctgrtable.shape
```

output ┐

```
(17525, 344)
```

Observations: 344 categories

Quick look into the table

```
〉 ctgrtable.head(5)
```

output ┐

	Accessories Store	American Restaurant	Aquarium	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Arts & Entertainment	Asian Restaurant
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0

Observations: Many empty areas in the borders of the country

Grouping by Neighborhood and checking the new shape

```
〉 groupt = temptb.groupby('Neighborhood').sum().reset_index()
〉 groupt.head()
〉 print(groupt.shape)
```

output ┐

```
(291, 344)
```

Observations: A dataframe of 291 rows (Areas) and 344 columns (Categories).

The groupt dataframe will be used in a paralel way for generating the top venues list but also to create the final dataframe for the clustering

1.- Preparing the dataframe for the k-mean clustering

```
〉 forcluster = groupt[:] # As a backup but also is the dataframe to be used for clustering
  purposes
```

Bringing the original latitude and longitude columns

```
〉 forcluster['latitude'] =
  forcluster.Neighborhood.map(pcsub.set_index('area')['latitude'].to_dict()) # Bringing the
  original latitude column
〉 forcluster['longitude'] =
  forcluster.Neighborhood.map(pcsub.set_index('area')['longitude'].to_dict()) # Bringing
  the original longitude column
```

```
〉    print(forcluster.shape) # This dataframe includes also the latitude and longitude
〉    forcluster.head(2)
```

output ┐

(291, 344)

	Neighborhood	Accessories Store	American Restaurant	Aquarium	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store
0	Almond Street	0	0	0	0	0	0	0	0
1	Amoy Street	0	0	0	0	1	1	0	0

Observations: This dataframe is ready to be used in the K-means process

2.- Top venues per neighborhood

```
〉    num_top_venues = 10
〉
〉    for hood in groupt['Neighborhood']:
〉        print("----"+hood+"----")
〉        temp = groupt[groupt['Neighborhood'] == hood].T.reset_index()
〉        temp.columns = ['venue','freq']
〉        temp = temp.iloc[1:]
〉        temp['freq'] = temp['freq'].astype(float)
〉        temp = temp.round({'freq': 2})
〉        print(temp.sort_values('freq',
     ascending=False).reset_index(drop=True).head(num_top_venues))
〉        print('\n')
```

output ┐

```
----Almond Street----
                venue  freq
0      Nature Preserve   3.0
1                 Park   3.0
2           Food Court   3.0
3         Noodle House   2.0
4               Market   1.0
5          Coffee Shop   1.0
6   Seafood Restaurant   1.0
7          Flea Market   1.0
8              Parking   1.0
9            Cafeteria   1.0
.
.
291 additional cases
```

Function definition and application to get a dataframe with the top 10 most common venues

```
〉    def return_most_common_venues(row, num_top_venues):
〉        row_categories = row.iloc[1:]
〉        row_categories_sorted = row_categories.sort_values(ascending=False)
〉
〉        return row_categories_sorted.index.values[0:num_top_venues]
```

```
〉    num_top_venues = 10
〉
〉    indicators = ['st', 'nd', 'rd']
〉
〉    # create columns according to number of top venues
〉    columns = ['Neighborhood']
```

```
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
topv = pd.DataFrame(columns=columns)
topv['Neighborhood'] = groupt['Neighborhood']

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

topv.shape
```

Check new dataframe

```
topv.head()
```

output ┐

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
0	Almond Street	Park	Nature Preserve	Food Court	Noodle House	Seafood Restaurant	Basketball Court	Market
1	Amoy Street	Hotel	Japanese Restaurant	Coffee Shop	Korean Restaurant	Café	Food Court	Bar
2	Anchorvale Crescent	Fast Food Restaurant	Food Court	Cosmetics Shop	Convenience Store	Supermarket	Coffee Shop	Bistro
3	Ang Mo Kio Avenue 2	Chinese Restaurant	Food Court	Asian Restaurant	Pet Store	Japanese Restaurant	Fast Food Restaurant	Thai Restaurant
4	Ang Mo Kio Avenue 3	Food Court	Chinese Restaurant	Coffee Shop	Fast Food Restaurant	Seafood Restaurant	Supermarket	Vegetarian / Vegan Restaurant

Observations: Everything seems to be in order

Backup, rename and check shape

```
topa = topv[:] # Backup
topa.rename(columns={'Neighborhood':'area'}, inplace =True)
print(topa.shape)
```

output ┐

```
(291, 11)
```

Running K-Means (Clusters are set to 10)

```
# set number of clusters
kclusters = 10

tgc = forcluster.drop('Neighborhood', 1)

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

```

```
〉 # check cluster labels generated for each row in the dataframe
〉 kmeans.labels_[0:10]
```
output ┐

```
(array([6, 3, 6, 9, 4, 2, 6, 2, 6, 0], dtype=int32)
```

Observations: Everything seems to be in order

Merging dataframes

Let's merge the dataframes that includes the cluster as well as the top 10 venues for each neighborhood.

```
〉 # add clustering labels
〉 topa.insert(0, 'Cluster Labels', kmeans.labels_)
〉
〉 tablemerged= pcsub[:]
〉
〉 # merge toronto_grouped with toronto_data to add latitude/longitude for each neighborhood
〉 tablemerged = tablemerged.join(topa.set_index('area'), on='area')
...
〉 tablemerged = tablemerged.drop('postalcode',1)
...
〉 tablemerged.head(5)
```
output ┐

	area	latitude	longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue
0	Almond Street	1.3725	103.7724	6	Park	Nature Preserve	Food Court	Noodle House
1	Amoy Street	1.2801	103.8468	3	Hotel	Japanese Restaurant	Coffee Shop	Korean Restaurant
2	Anchorvale Crescent	1.3985	103.8921	6	Fast Food Restaurant	Food Court	Cosmetics Shop	Convenience Store
3	Ang Mo Kio Avenue 2	1.3774	103.8350	9	Chinese Restaurant	Food Court	Asian Restaurant	Pet Store
4	Ang Mo Kio Avenue 3	1.3676	103.8446	4	Food Court	Chinese Restaurant	Coffee Shop	Fast Food Restaurant

Observations: This table is ready, now we can proceed to plot the data

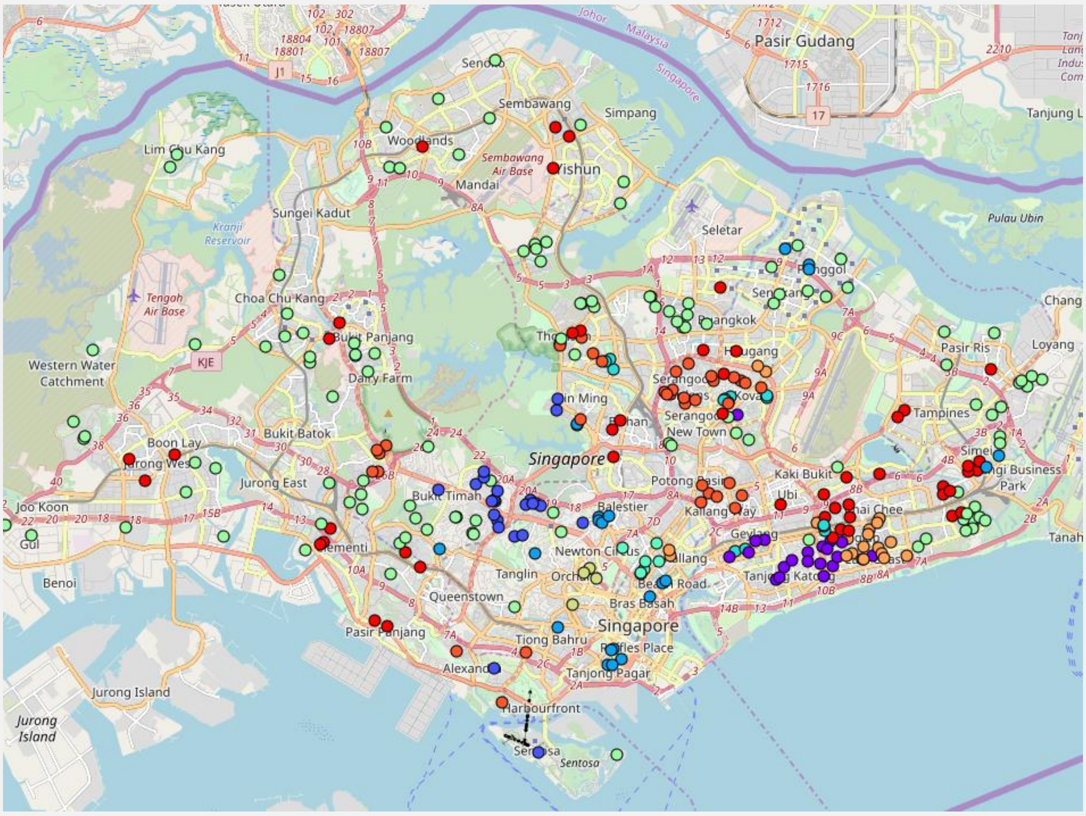
Plot the final map and clusters

```
〉 # create map
〉 map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)
〉
〉 # set color scheme for the clusters
〉 x = np.arange(kclusters)
〉 ys = [i + x + (i*x)**2 for i in range(kclusters)]
〉 colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
〉 rainbow = [colors.rgb2hex(i) for i in colors_array]
〉
```

```
# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(tablemerged['latitude'], tablemerged['longitude'],
tablemerged['area'], tablemerged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color= 'black',
        weight=1,
        fill=True,
        fill_color= rainbow[cluster -1],
        fill_opacity=1).add_to(map_clusters)

map_clusters
```

output



Cluster 0 Mixed Asian and coffee shops
Cluster 1 Traditional Asian restaurants
Cluster 2 Bakeries and cafes
Cluster 3 Downtown and business
Cluster 4 Popular venues
Cluster 5 Indian venues
Cluster 6 Entertainment, leisure, parks, utilities and food
Cluster 7 Upscale
Cluster 8 Coffee shops and Asian restaurants in traditional areas
Cluster 9 Food around Serangoon

Cluster analysis and final observations

We managed to divide postal codes in 10 clusters very distinct clusters that can be described as follows.

Cluster 0: Mixed restaurants and coffee shops in different areas

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 0, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
11	Bedok Reservoir Road	Food Court	Coffee Shop	Asian Restaurant	Fast Food Restaurant	Supermarket
12	Bedok Reservoir Road	Food Court	Coffee Shop	Asian Restaurant	Fast Food Restaurant	Supermarket
20	Boon Lay Drive	Asian Restaurant	Fast Food Restaurant	Japanese Restaurant	Chinese Restaurant	Food Court
26	Canberra Drive	Coffee Shop	Fast Food Restaurant	Chinese Restaurant	Asian Restaurant	Thai Restaurant
34	Changi Road	Chinese Restaurant	Coffee Shop	Seafood Restaurant	Indian Restaurant	Asian Restaurant
43	Clover Way	Coffee Shop	Chinese Restaurant	Food Court	Seafood Restaurant	Japanese Restaurant
69	Enterprise Road	Japanese Restaurant	Asian Restaurant	Coffee Shop	Fast Food Restaurant	Chinese Restaurant
71	Faber Green	Food Court	Coffee Shop	Chinese Restaurant	Japanese Restaurant	Shopping Mall
74	Fernvale Road	Coffee Shop	Food Court	Supermarket	Dessert Shop	Light Rail Station

Cluster 0, 🔴 is located in different areas of the city and is mostly composed of mixed Asian restaurants and coffee shops.

Cluster 1: Traditional asian restaurants

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 1, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
35	Charlton Road	Noodle House	Café	Chinese Restaurant	Coffee Shop	Food Court
60	Dunman Lane	Chinese Restaurant	Noodle House	Asian Restaurant	Café	Indian Restaurant
63	East Coast Road	Chinese Restaurant	Coffee Shop	Indian Restaurant	Noodle House	Asian Restaurant
64	East Coast Road	Chinese Restaurant	Coffee Shop	Indian Restaurant	Noodle House	Asian Restaurant
70	Everitt Road	Asian Restaurant	Chinese Restaurant	Noodle House	Indian Restaurant	Coffee Shop
86	Goodman Road	Chinese Restaurant	Noodle House	Asian Restaurant	Italian Restaurant	Café
87	Goodman Road	Chinese Restaurant	Noodle House	Asian Restaurant	Italian Restaurant	Café
95	Haigsville Drive	Chinese Restaurant	Asian Restaurant	Indian Restaurant	Noodle House	Food Court
109	Jalan Baiduri	Chinese Restaurant	Noodle House	Coffee Shop	Asian Restaurant	Food Court
147	Joo Chiat Avenue	Chinese Restaurant	Noodle House	Coffee Shop	Asian Restaurant	Food Court

Cluster 1,🟣 is located in traditional areas of the city and is mostly composed of Chinese restaurants and Noodle houses.

Cluster 2: Bakeries and cafes

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 2, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
7	Ash Grove	Bakery	Italian Restaurant	Food Court	Pizza Place	Seafood Restaurant
9	Barker Road	Café	Japanese Restaurant	Chinese Restaurant	Bus Station	Italian Restaurant
44	Cluny Park	Café	Garden	Bakery	Park	French Restaurant
45	Cluny Road	Café	Garden	Bakery	Park	Bus Stop
50	Coronation Road	Café	Bakery	Shopping Mall	Noodle House	Thai Restaurant
53	Cypress Avenue	Bus Station	Food Court	Seafood Restaurant	Indian Restaurant	Bakery
73	Farrer Road	Café	Garden	Bakery	French Restaurant	Shopping Mall
81	Gardenia Road	Café	Thai Restaurant	Ice Cream Shop	Chinese Restaurant	Spa
91	Greenwood Avenue	Italian Restaurant	Bakery	Pool	Gym	Chinese Restaurant
92	Greenwood Avenue	Italian Restaurant	Bakery	Pool	Gym	Chinese Restaurant

Cluster 2, 🔵 Concentrated in the Bukit Timah area, a residential area composed of cafes, bakery shops, gardens, metro station and shopping mall. There is minimal public housing in this area.

Cluster 3: Business amenities

```
tablemerged.loc[tablemerged['Cluster Labels'] == 3, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
1	Amoy Street	Hotel	Japanese Restaurant	Coffee Shop	Korean Restaurant	Café
32	Chancery Hill Walk	Café	Chinese Restaurant	Hotel	Bakery	Coffee Shop
33	Chancery Lane	Chinese Restaurant	Café	Coffee Shop	Hotel	Bakery
62	Duxton Hill	Japanese Restaurant	Coffee Shop	Bakery	Cocktail Bar	Hotel
82	Gentle Drive	Chinese Restaurant	Hotel	Café	Bakery	Coffee Shop
97	Havelock Road	Chinese Restaurant	Japanese Restaurant	Café	Noodle House	Coffee Shop
104	Holland Rise	Bakery	Japanese Restaurant	Bar	Chinese Restaurant	Ice Cream Shop
128	Jalan Novena	Chinese Restaurant	Hotel	Café	Bakery	Food Court
143	Jalan Tiga Ratus	Café	Coffee Shop	Fast Food Restaurant	Chinese Restaurant	Japanese Restaurant
210	Nassim Road	Hotel	Café	Park	Chinese Restaurant	Japanese Restaurant

Cluster 3, Concentrated in the financial district area, composed by hotels, mixed restaurants and cafes.

Cluster 4: Popular food, hawker centers and Asian restaurants

```
tablemerged.loc[tablemerged['Cluster Labels'] == 4, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
4	Ang Mo Kio Avenue 3	Food Court	Chinese Restaurant	Coffee Shop	Fast Food Restaurant	Seafood Restaurant
5	Ang Mo Kio Avenue 3	Food Court	Chinese Restaurant	Coffee Shop	Fast Food Restaurant	Seafood Restaurant
6	Ang Mo Kio Avenue 3	Food Court	Chinese Restaurant	Coffee Shop	Fast Food Restaurant	Seafood Restaurant
83	Geylang Road	Chinese Restaurant	Noodle House	Food Court	Seafood Restaurant	Hotel
84	Geylang Road	Chinese Restaurant	Noodle House	Food Court	Seafood Restaurant	Hotel
99	Highland Road	Chinese Restaurant	Noodle House	Food Court	Café	Coffee Shop
100	Highland Road	Chinese Restaurant	Noodle House	Food Court	Café	Coffee Shop
117	Jalan Hock Chye	Chinese Restaurant	Noodle House	Coffee Shop	Food Court	Café

Cluster 4, Concentrated along the Geylang area is composed of food court and fast food mostly.

Cluster 5: Indian area and restaurants

```
tablemerged.loc[tablemerged['Cluster Labels'] == 5, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
25	Cambridge Road	Indian Restaurant	Chinese Restaurant	Café	Hotel	Coffee Shop
52	Cuff Road	Indian Restaurant	Café	Chinese Restaurant	Bakery	Hotel
59	Dunlop Street	Indian Restaurant	Café	Chinese Restaurant	Hotel	Vegetarian / Vegan Restaurant
111	Jalan Besar	Indian Restaurant	Chinese Restaurant	Café	Hostel	BBQ Joint
235	Race Course Road	Indian Restaurant	Chinese Restaurant	Café	Hotel	Dessert Shop
249	Serangoon Road	Indian Restaurant	Chinese Restaurant	Café	Hotel	Vegetarian / Vegan Restaurant
299	Upper Dickson Road	Indian Restaurant	Café	Chinese Restaurant	Hotel	Vegetarian / Vegan Restaurant

Cluster 5, Concentrated in little India mostly consists of Indian restaurants, cafes and hotels mostly.

Cluster 6: Entertainment, leisure, parks, utilities and food

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 6, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Almond Street	Park	Nature Preserve	Food Court	Noodle House	Seafood Restaurant
2	Anchorvale Crescent	Fast Food Restaurant	Food Court	Cosmetics Shop	Convenience Store	Supermarket
8	Ashwood Grove	Fast Food Restaurant	Food Court	Pizza Place	Grocery Store	Shopping Mall
10	Bedok Avenue	Beach	Campground	Resort	Food Court	Grocery Store
13	Bedok Ria Walk	Noodle House	Dessert Shop	Malay Restaurant	Coffee Shop	Café
14	Bedok Road	Asian Restaurant	Noodle House	Chinese Restaurant	BBQ Joint	Seafood Restaurant
15	Belgravia Drive	Café	Bus Stop	Thai Restaurant	Asian Restaurant	Bakery
16	Belgravia Drive	Café	Bus Stop	Thai Restaurant	Asian Restaurant	Bakery
17	Bilal Lane	Bus Line	Coffee Shop	Noodle House	Soccer Field	Flower Shop
21	Brookvale Drive	Café	Bakery	College Cafeteria	Grocery Store	Military Base

Cluster 6, All around the city, this cluster is composed by supermarkets, food, beaches, parks, gun ranges, fast food, bus stops, etc.

Cluster 7: Upscale venues

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 7, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
24	Cairnhill Road	Hotel	Boutique	Japanese Restaurant	Chinese Restaurant	Bakery
67	Emerald Hill Road	Japanese Restaurant	Hotel	Boutique	Shopping Mall	Clothing Store
114	Jalan Elok	Boutique	Hotel	Japanese Restaurant	Sushi Restaurant	Bakery
164	Leonie Hill	Boutique	Shopping Mall	Japanese Restaurant	Bubble Tea Shop	Hotel

Cluster 7, Concentrated near downtown area, composed by hotels, boutiques and Japanese restaurants mostly.

Cluster 8: Coffee shops and traditional restaurants in residential areas

```
1 tablemerged.loc[tablemerged['Cluster Labels'] == 8, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
31	Cavan Road	Café	Chinese Restaurant	Coffee Shop	Indian Restaurant	Hostel
58	Dunbar Walk	Coffee Shop	Chinese Restaurant	Indian Restaurant	Noodle House	Asian Restaurant
75	Fidelio Street	Coffee Shop	Indian Restaurant	Chinese Restaurant	Supermarket	Dessert Shop
78	Fourth Street	Coffee Shop	Indian Restaurant	Chinese Restaurant	Café	Dessert Shop
79	Frankel Street	Chinese Restaurant	Coffee Shop	Indian Restaurant	Noodle House	Asian Restaurant
121	Jalan Keris	Coffee Shop	Noodle House	Café	Chinese Restaurant	Dessert Shop
122	Jalan Keris	Coffee Shop	Noodle House	Café	Chinese Restaurant	Dessert Shop
142	Jalan Terang Bulan	Coffee Shop	Chinese Restaurant	Café	Asian Restaurant	Supermarket
144	Jalan Yasin	Chinese Restaurant	Coffee Shop	Indian Restaurant	Food Court	Noodle House
160	Lavender Street	Chinese Restaurant	Café	Hotel	Indian Restaurant	Coffee Shop

Cluster 8, Located in a traditional area of Singapore, is composed of coffee shops, cafes and Asian restaurants.

Cluster 9: Food around Serangoon

```
tablemerged.loc[tablemerged['Cluster Labels'] == 9, tablemerged.columns[[0] + list(range(4, tablemerged.shape[1]))]]
```

	area	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
3	Ang Mo Kio Avenue 2	Chinese Restaurant	Food Court	Asian Restaurant	Pet Store	Japanese Restaurant
18	Blandford Drive	Chinese Restaurant	Food Court	Bus Station	Park	Bakery
19	Bodmin Drive	Chinese Restaurant	Food Court	Park	Bakery	Noodle House
22	Bukit Merah Central	Chinese Restaurant	Coffee Shop	Food Court	Bus Station	Gym
23	Burghley Drive	Chinese Restaurant	Food Court	Bus Station	Park	Bakery
27	Cardiff Grove	Chinese Restaurant	Japanese Restaurant	Coffee Shop	Bus Station	Asian Restaurant
29	Casuarina Road	Bus Station	Chinese Restaurant	Food Court	Restaurant	Japanese Restaurant
30	Casuarina Road	Bus Station	Chinese Restaurant	Food Court	Restaurant	Japanese Restaurant
36	Cheng Soon Crescent	Korean Restaurant	Chinese Restaurant	Café	Food Court	College Cafeteria
39	Chiselhurst Grove	Chinese Restaurant	Park	Coffee Shop	Pub	Bus Station

Cluster 9, Asian restaurants concentrated in the Serangoon area

Final conclusions

According to our objective **[understand which neighborhoods or areas are more appropriate to open a shop].** Now we have a very good idea of how the different venues are distributed in the city with 2 or 3 main areas concentrating a larger number of bakery shops, cafes and similar.

Opening a bakery/café in Cluster 2 will offer a higher scale experience with a more modern experience, with younger affluent customers willing to pay more, furthermore the Cluster 2 area / Bukit Timah has some of the densest clusters of luxury condominiums and therefore potential to continue growing and is close to MRT station and is also composed of parks and leisure areas. On the other hand, cluster 8 also has plenty of coffee shops, however it is located in a much more traditional area and opening there would be an error. This area is also limited in terms of disposable income.

It is our recommendation then to open in the Bukit Timah area as shown in the map below.