IBM – Coursera

Data Science Specialization

Capstone Project – Project Report

CAPSTONE PROJECT FOR FINDING OPTIMIZED VENUE AT FRANCE

Rajavelu Angalan

2019

Table of Content:

1.)Problem Description :

This report is for the final course of the Data Science Specialization. A 9-courses series created by IBM, hosted on Coursera platform. The problem and the analysis approach are left for the learner to decide, with a requirement of leveraging the Foursquare location data to explore or compare neighborhoods or cities of your choice or to come up with a problem that you can use the Foursquare location data to solve.

In this project, the problem is to find the optimal location or finding the cityof cluster which has user preferred venue eg. BAR,PLAZA and GYM in France. To achieve this task, an analytical approach will be used, based on advance machine learning techniques and data analysis,concretely clustering and perhaps some data visualization techniques.

So can the city surrounding has user preferred venues ?
If so, what types of venues cluster has the most affect, both positively and negatively?

The Target Audience for this project is for who prefer to stay in hotel based on on their preferred venues(eg.Tourists).

2.)Data Preparation:

France cities were chosen as the observation target due to the following reasons:

With more than 10 million tourists a year, the French Riviera (French: Côte d'Azur), in southeastern France, is the second leading tourist destination in the country, after the Parisian region.

The availability of geo data which can be used to visualize the dataset onto a map.

France City Coordinates have been taken from below website as a csv file https://simplemaps.com/data/fr-cities

- FourSquare API which provides the surrounding venues of a given coordinates.

The process of collecting data as follows:

- Load the city co ordinates into data frame and clean the data by removing unnecessary filed's.
- Pass the obtained co rdinates to the foursquare API.The “explore” endpoint will return a list of surrounding venues in a pre-defined radius.
- Count the occurrence of each venue type in a neighborhood. Then apply one hot encoding to turn each venue type into a column with their occurrence as the value.

	A	B	C	D	E	F	G
1	city	lat	lng	country	iso2	capital	population
2	Paris	48.866667	2.333333	France	FR	primary	9904000
3	Lyon	45.748457	4.846711	France	FR	admin	1423000
4	Marseille	43.285413	5.37606	France	FR	admin	1400000
5	Lille	50.632971	3.058585	France	FR	admin	1044000
6	Nice	43.713644	7.25952	France	FR	927000	338620
7	Toulouse	43.599516	1.433188	France	FR	admin	847000
8	Bordeaux	44.840439	-0.5805	France	FR	admin	803000
9	Rouen	49.433333	1.083333	France	FR	admin	532559
10	Strasbourg	48.600381	7.787355	France	FR	admin	439972
11	Nantes	47.216509	-1.552379	France	FR	admin	438537
12	Metz	49.115461	6.175875	France	FR	minor	409186
13	Grenoble	45.171546	5.722387	France	FR	388574	158552
14	Toulon	43.117705	5.941712	France	FR	357693	168701
15	Montpellier	43.61092	3.87723	France	FR	minor	327254

3.)Methodology:

The methodology used to approach this problem includes some statistical exploration of the data an some visualization mail machine learning technique involved in the development of this project is clustering , in concrete the KMeans algorithm was used and implemented with python.

To Solve theses kind of optimal business location problems , we data from foursquare API.

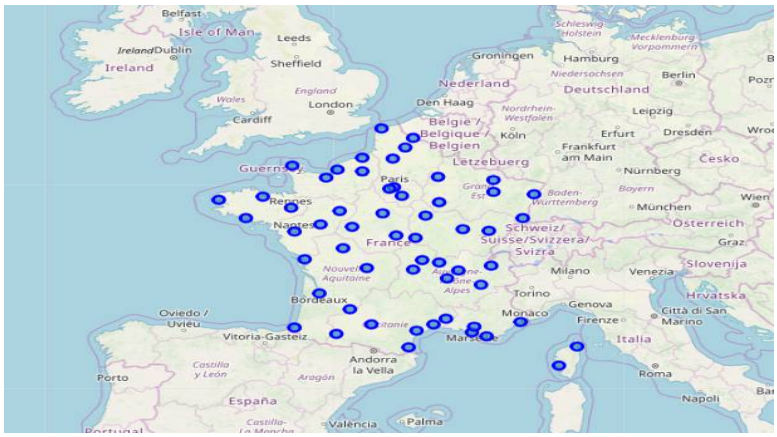
France has 37 sites inscribed in the UNESCO's World Heritage List and features cities or sites of high cultural interest (Paris being the foremost, but also Loire Valley, Toulouse, Strasbourg, Bordeaux, Lyon, and others), beaches and seaside resorts, ski resorts, and rural regions that many enjoy for their beauty and tranquillity (green tourism). Small and picturesque French villages of quality heritage (such as Collonges-la-Rouge, Locronan, or Montsoreau) are promoted through the association Les Plus Beaux Villages de France (literally "The Most Beautiful Villages of France"). The "Remarkable Gardens" label is a list of the over two hundred gardens classified by the French Ministry of Culture. This label is intended to protect and promote remarkable gardens and parks.

With all this being considered, it was decided that the mail gal to solve this problem.

To continue this line , the Foursquare API was used to obtain the needed data about the venues in each cities , but to use Foursquare API , it was first necessary to transform the raw data to something the Foursquare API was capable to handle. Basically, the coordinates of each cities were added.

	CITY	LATITUDE	LONGITUDE	COUNTRY	F
0	Paris	48.866667	2.333333	France	
1	Lyon	45.748457	4.846711	France	
2	Marseille	43.285413	5.376060	France	
3	Lille	50.632971	3.058585	France	
4	Nice	43.713644	7.259520	France	

The cities were plotted into map of Franc based on the co ordinates.



Then define the foursquare URL based on credentials of yours to get the venues.

Define Foursquare Credentials and Version

```
CLIENT_ID = 'ESYH340ZLYESMFLKUKCHDQ33YNUJINGWDUPRBZC21VVYTFMT' # your Foursquare ID
CLIENT_SECRET = 'EYRI0QRQTSMVWD5AWU1JGD4FXZBNCPOXM4NRO1TKBS3EVHOZ' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: ESYH340ZLYESMFLKUKCHDQ33YNUJINGWDUPRBZC21VVYTFMT
CLIENT_SECRET:EYRI0QRQTSMVWD5AWU1JGD4FXZBNCPOXM4NRO1TKBS3EVHOZ
```

```
results = requests.get(url).json()
results
```


```
{'meta': {'code': 200, 'requestId': '5c3861e46a6071298d257090'},
 'response': {'suggestedFilters': {'header': 'Tap to show:',
   'filters': [{'name': 'Open now', 'key': 'openNow'}]},
  'headerLocation': 'Place Vendôme',
  'headerFullLocation': 'Place Vendôme, Paris',
  'headerLocationGranularity': 'neighborhood',
  'totalResults': 244,
  'suggestedBounds': {'ne': {'lat': 48.8711670045, 'lng': 2.340161078526742},
   'sw': {'lat': 48.8621669955, 'lng': 2.326504921473258}},
  'groups': [{'type': 'Recommended Places',
    'name': 'recommended',
    'items': [{'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4cbdcb0b7148f04d510aefab',
       'name': 'Pierre Hermé',
       'location': {'address': "39 avenue de l'Opéra",
        'lat': 48.86822151447183,
        'lng': 2.333396617684349,
        'labeledLatLngs': [{'label': 'display',
```

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Paris	48.866667	2.333333	Pierre Hermé	48.868222	2.333397	Pastry Shop
1	Paris	48.866667	2.333333	Le Roch Hotel & Spa Paris	48.866200	2.332995	Hotel
2	Paris	48.866667	2.333333	Cantine California	48.867401	2.332017	Food Truck
3	Paris	48.866667	2.333333	Boulangerie Aki	48.866211	2.335458	Bakery
4	Paris	48.866667	2.333333	Brasserie Réjane	48.865486	2.334824	Restaurant

Once the venues of each cities obtained then calculate the frequency of occurrence of each venue.

```
----Agen----
                venue  freq
0         Supermarket  0.33
1        Dance Studio  0.33
2                Park  0.33
3  Accessories Store  0.00
4       Optical Shop  0.00
```

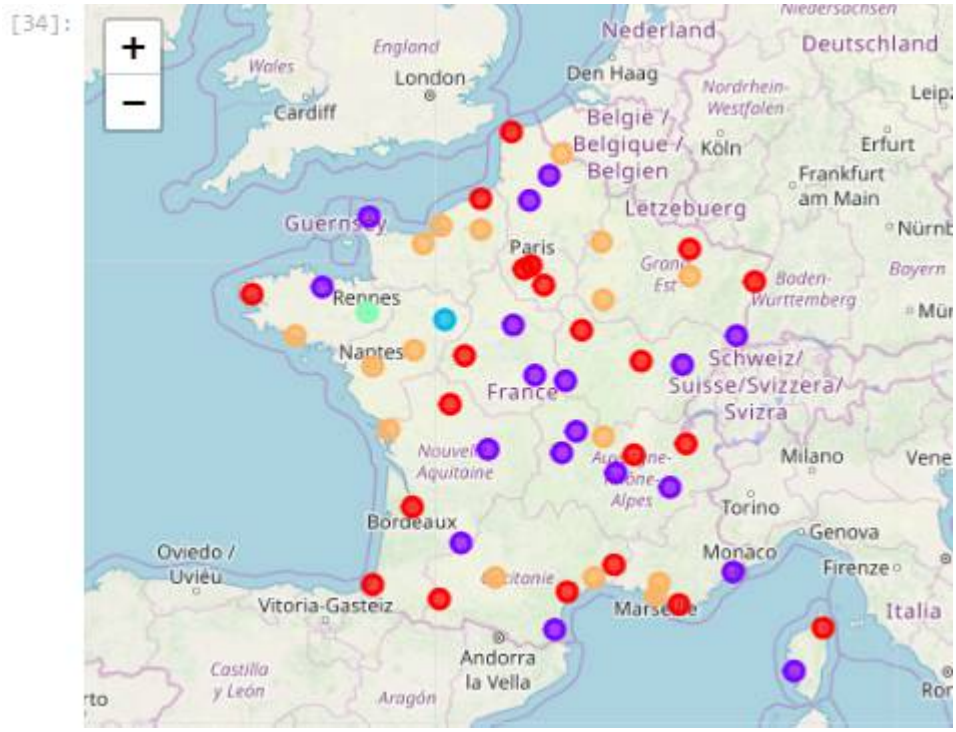
With this data-set we should know which venue is most common venues of each city.

	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Agen	Supermarket	Dance Studio	Park	Women's Store	Electronics Store	Food & Drink Shop	Food	Flower Shop	Fish & Chips Shop	Financial or Legal Service
1	Aix-en-Provence	French Restaurant	Plaza	Bar	Pedestrian Plaza	Pub	Bagel Shop	Burger Joint	Italian Restaurant	Asian Restaurant	Ice Cream Shop
2	Ajaccio	Hotel	Sushi Restaurant	French Restaurant	Restaurant	Café	Plaza	Art Museum	Arts & Crafts Store	Food Truck	Food & Drink Shop

Apply the logic of Kmeans algorithm and extract the clustering.

	CITY	LATITUDE	LONGITUDE	COUNTRY	POPULATION	Cluster Labels	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	8th Most Common Venue
0	Paris	48.866667	2.333333	France	9904000	0	Japanese Restaurant	Hotel	French Restaurant	Café	Ramen Restaurant	Jewelry Store	Pastry Shop	Bakery
1	Lyon	45.748457	4.846711	France	1423000	0	Restaurant	Diner	Bistro	Pizza Place	Plaza	Hobby Shop	Italian Restaurant	Sandwich Place
2	Marseille	43.285413	5.376060	France	1400000	4	Plaza	Bus Stop	Lounge	French Restaurant	Cupcake Shop	Church	Scenic Lookout	Hotel
3	Lille	50.632971	3.058585	France	1044000	4	French Restaurant	Bar	Japanese Restaurant	Pub	Cocktail Bar	Italian Restaurant	Coffee Shop	Plaza
4	Nice	43.713644	7.259520	France	338620	1	French Restaurant	Plaza	Seafood Restaurant	Mediterranean Restaurant	Gym	Farmers Market	Women's Store	Doner Restaurant

Once the clusters are obtained then plot the data in map.



Now if the user wants to apply the venues to the hotel venue clusters.

```
map_clusters
```

4.)Results :

The result obtain are 5 clusters of very different venue distribution.the foloowing are distribution of clusters.

USER_CLUSTER 1:

There are 4 cities where the data venues displayed for user selection input.

USER_CLUSTER 1

Cluster based on user selection

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

	CITY	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
10	Metz	Bar	French Restaurant	Plaza	Italian Restaurant	Hotel	Sandwich Place	Pub	Coffee Shop	Department Store	Fast Food Restaurant
32	Brest	Hotel	Fast Food Restaurant	Sandwich Place	Shopping Mall	Pedestrian Plaza	Bookstore	Electronics Store	Café	Thai Restaurant	Bar
36	Annecy	Hotel	Department Store	Bar	Clothing Store	Pizza Place	Candy Store	Shopping Mall	Mobile Phone Shop	Sandwich Place	Café
55	Auxerre	Hotel	French Restaurant	Tourist Information Center	Historic Site	Harbor / Marina	Grocery Store	Pizza Place	Restaurant	Bar	Plaza

USER_CLUSTER 2:

There are 2 cities where the data venues displayed for user selection input.

USER_CLUSTER 2

Cluster based on user selection

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

	CITY	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
44	Bourges	Plaza	Pub	French Restaurant	Hotel	Bar	Tourist Information Center	Park	Department Store	Clothing Store	Snack Place
52	Nevers	French Restaurant	Historic Site	Supermarket	Diner	Dessert Shop	Park	Bar	Hotel	Dance Studio	Department Store

USER_CLUSTER 3:

There are no cities where the data venues displayed for user selection input.

USER_CLUSTER 3

Cluster based on user selection

```
tot_cluster.loc[tot_cluster['Cluster Labels'] == 2, tot_cluster.columns[[0] + list(range(6, france_merged.shape[1]))]]
```

CITY	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue

USER_CLUSTER 4:

There are no cities where the data venues displayed for user selection input.

USER_CLUSTER 4

Cluster based on user selection

```
tot_cluster.loc[tot_cluster['Cluster Labels'] == 3, tot_cluster.columns[[0] + list(range(6, france_merged.shape[1]))]]
```

CITY	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue

USER_CLUSTER 5:

There are 4 cities where the data venues displayed for user selection input.

USER_CLUSTER 5 ¶

Cluster based on user selection

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

	CITY	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	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
3	Lille	French Restaurant	Bar	Japanese Restaurant	Pub	Cocktail Bar	Italian Restaurant	Coffee Shop	Plaza	Burger Joint	Hotel
9	Nantes	Bar	French Restaurant	Plaza	Hotel	Coffee Shop	Restaurant	Burger Joint	Indian Restaurant	Tea Room	Bistro
14	Nancy	Bar	French Restaurant	Hotel	Italian Restaurant	Nightclub	Plaza	Coffee Shop	Cosmetics Shop	Historic Site	Pizza Place
25	Angers	Bar	French Restaurant	Pub	Lounge	Sandwich Place	Indian Restaurant	Italian Restaurant	Department Store	Japanese Restaurant	Hotel

5.)Discussion :

It is interesting how the venues and people from different cities varies to one another. The main differentiation is after the clusters filtered upon the user inputs but also we could see some common venues among the clusters.

As a recommendation, it must be said in study to make better predictions about the where to locate cluster city with user venue. for example if tourist want to locate the city with hotel clusters based on bar,plaza,gym etc..

6.)Conclusion :

As far as we can see with this data, some of the clusters are not populated because of user filter.

It is highly possible that user_cluster 1 & 5 has more cities which has the user preferences of hotel cluster. If the user input data should perform with more data and logic also framed in proper way then we can provide more accurate output .

7.)References

https://developer.foursquare.com/docs/api/venues/

https://simplemaps.com/data/fr-cities

https://www.coursera.org/