TRAVEL HANDBOOK

ABSTRACT

The work presented here buckets top 50 most beautiful cities of the world into mutually exclusive clusters. The cluster labels are self-explanatory and depict the most prominent venue category in that particular cluster of cities.

Trehan, Rahul

Applied Data Science Capstone Project

Table of Contents

Capstone Project - The Battle of the Neighborhoods (Cities)

Applied Data Science Capstone by IBM/Coursera

Introduction

Background

Tourism is one of the biggest industries in the world with an approximate size of USD 10 trillion. With better means of travel, communication and ease of connectivity, the industry grew at an approximate CAGR of 3% between 2006 and 2017 as per the World Travel and Tourism Council (WTTC). Across the globe, the rise in middle class population and changing choices of the youth have made it nearly impossible for any country to avoid its tourism sector.

Nowadays, it has become very common for people to search for travel destinations online. There exist many online portals, websites, blogs etc. that cater to the needs of travel enthusiasts in helping them plan their vacations. Also, travel companies advise their clients basis their choices and preferences.

However, it is hard to find blogs that classify cities on an overall basis based on traits that can well explain the cities in terms of its most commonly visited venues & lifestyle.

This became the inspiration for carrying out this analysis as described in the “Business Problem” section below.

Problem Statement

In this project, we will investigate the Top 50 most beautiful cities in the world and would try to cluster them into mutually exclusive groups. Similar cities would be clustered basis the category of venues within a specified radius of the city.

We will use **Foursquare** location data to explore the cities' surroundings and cluster cities basis venues in their vicinity.

We will then find out the most commonly occurring trait that would certainly define any cluster based on the top 10 most commonly visited venues in that cluster of cities.

Lastly, we would represent the results in the form of a table and call it "***TRAVEL HANDBOOK***".

For reference, the list of cities can be accessed here : https://www.flightnetwork.com/blog/worlds-most-beautiful-cities/

Business Utility

This analysis would be of use to travel enthusiasts and/ or travel magazines/ online portals which frequently publish the list of most beautiful cities that one must visit at least once in a lifetime.

The added advantage here is that, one would be able to group cities into buckets and can make travel itineraries accordingly as per one's choice.

Moreover, this analysis can be carried out at a regular interval to have the latest set of cluster groups.

Data

I have taken the list of "THE WORLD'S 50 MOST BEAUTIFUL CITIES" from the website https://www.flightnetwork.com/blog/worlds-most-beautiful-cities/

The website contains a numbered list of the top 50 cities globally. As such, ***BeautifulSoup*** package has been used to scrape the website and get the top 50 cities in a tabular format.

The output after this step is as shown below:

```
In [23]: top_50_df
```

Out[23]:

	City_name
0	PARIS
1	NEW-YORK
2	LONDON
3	VENICE
4	VANCOUVER
5	BARCELONA
6	CAPE-TOWN
7	SAN-FRANCISCO
8	SYDNEY
9	ROME
10	SINGAPORE

In order to find out the geographical coordinates of these cities, the ***geopy*** package was used.

Next, ***Foursquare*** was used to explore the vicinity of these top 50 cities.

A **radius** of 20 km from the city center was chosen and **limit** of 1000 results per city was applied.

Upon completing the above steps, the data thus obtained looked like:

In [36]:
```
nearby_venues
```

Out[36]:

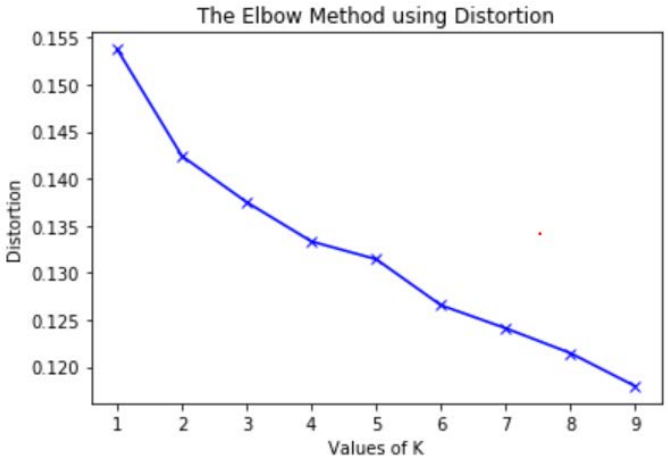
	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	PARIS	48.856697	2.351462	Cathédrale Notre-Dame de Paris	48.853124	2.349561	Church
1	PARIS	48.856697	2.351462	Fleux'	48.858763	2.354161	Furniture / Home Store
2	PARIS	48.856697	2.351462	Place de l'Hôtel de Ville – Esplanade de la Li...	48.856925	2.351412	Plaza
3	PARIS	48.856697	2.351462	Shakespeare & Company	48.852568	2.347096	Bookstore
4	PARIS	48.856697	2.351462	Miznon	48.857201	2.358957	Israeli Restaurant
5	PARIS	48.856697	2.351462	La Maison d'Isabelle	48.850007	2.348443	Bakery
6	PARIS	48.856697	2.351462	Centre Pompidou – Musée National d'Art Moderne	48.860730	2.351660	Art Museum
7	PARIS	48.856697	2.351462	Comme à Lisbonne	48.856767	2.356462	Café

This marks the end of the data gathering stage. Now we will proceed with describing the overall methodology employed in carrying out the analysis.

Methodology

This project aims to cluster top 50 most beautiful cities of the world which have similar category of venues/ features based on their vicinity data which has been obtained from Foursquare. The following steps were undertaken in a sequential manner to perform the analysis:

1. In first step we **scraped** the website to obtain the list of top 50 most beautiful cities globally.
2. Second step aims at obtaining the **geographical coordinates** of the above-mentioned cities.
3. In the third step, we will obtain the vicinity data from **Foursquare**.
4. In the fourth step, we will employ **KMeans** clustering technique to cluster the cities basis the data obtained from Foursquare. We will ***choose the best number of clusters*** by iterating through a range of cluster numbers (***Elbow Method of best k determination***). This is as shown below:



5. Upon performing clustering with k=6, the clusters are plotted on the map to visualize the clusters.
6. In the sixth step, we will obtain the ***top 10 most common venues*** for every city as shown below:

Out[110]:

	Cluster	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	10 C
0	3	AMSTERDAM	Hotel	Coffee Shop	Cocktail Bar	Breakfast Spot	Bookstore	Plaza	Canal	Café	Bar	Bak
1	1	ATHENS	Café	Coffee Shop	Bar	Historic Site	Meze Restaurant	Theater	Greek Restaurant	Cocktail Bar	Boutique	Boo
2	2	BANGKOK	Hotel	Coffee Shop	Shopping Mall	Park	Thai Restaurant	Noodle House	Spa	Bookstore	Clothing Store	Asi Res
3	4	BARCELONA	Hotel	Coffee Shop	Plaza	Tapas Restaurant	Burger Joint	Pizza Place	Breakfast Spot	Ice Cream Shop	Argentinian Restaurant	Par
4	2	BEIJING	Historic Site	Hotel	Park	Café	Shopping Mall	Brewery	Chinese Restaurant	Dumpling Restaurant	Pizza Place	Yur Res
5	1	BERGEN	Hotel	Bar	Coffee Shop	Café	Mountain	Restaurant	Bakery	Shopping Mall	Furniture / Home Store	Sea Res
6	0	BERLIN	Coffee Shop	Bookstore	Park	Ice Cream Shop	Gourmet Shop	Café	Hotel	Monument / Landmark	Bakery	Wir
7	1	BRUGES	Belgian Restaurant	Bar	Bakery	French Restaurant	Hotel	Bistro	Clothing Store	Sports Club	Pub	Tap Res

7. Next, for each cluster we will find the top 3 most visited venues for the cluster. In case of a ***tie***, we will go for the next top venue in the cluster.

Using the melt function to have all the venues in one column

In [139]:
```
clus_0_melt=pd.melt(cluster_0, id_vars=['Cluster','City'], value_vars=cluster_0.columns[2:12], var_name='Common_venues', value_name='Venue_name')
```

In [157]:
```
clus_0_melt
```

Out[157]:

	Cluster	City	Common_venues	Venue_name
0	0	BERLIN	1st Most Common Venue	Coffee Shop
1	0	CHICAGO	1st Most Common Venue	Park
2	0	DUBLIN	1st Most Common Venue	Café
3	0	EDINBURGH	1st Most Common Venue	Café
4	0	LONDON	1st Most Common Venue	Park
5	0	NEW-YORK	1st Most Common Venue	Park
6	0	PRAGUE	1st Most Common Venue	Café
7	0	QUEBEC-CITY	1st Most Common Venue	Park
8	0	RIO-DE-JANEIRO	1st Most Common Venue	Coffee Shop
9	0	SAN-DIEGO	1st Most Common Venue	Park
10	0	SAN-FRANCISCO	1st Most Common Venue	Park
11	0	SYDNEY	1st Most Common Venue	Café
12	0	TORONTO	1st Most Common Venue	Park
13	0	VANCOUVER	1st Most Common Venue	Park
14	0	BERLIN	2nd Most Common Venue	Bookstore

8. Finally, each cluster is labelled as per the top venue.

Analysis

Although, the nature of data does not warrant much of Exploratory Data Analysis, the analysis phase primarily involves step by step implementation to draw conclusions from the raw data. They are:

1. Checking the number of venues obtained per city. The output of which is also shown:

```
In [41]: df_top_50_required.groupby('City').count()
```

Out[41]:

City	Latitude	Longitude	Venue Category
AMSTERDAM	200	200	200
ATHENS	100	100	100
BANGKOK	200	200	200
BARCELONA	200	200	200
BEIJING	100	100	100
BERGEN	100	100	100
BERLIN	200	200	200
BRUGES	200	200	200
BUDAPEST	200	200	200

We see that number of venues obtained for some cities is lower than others. Maximum venues for any city are 200. This might be due to lesser number of venues within the specified radius. We may also conclude that the city is not that big.

2. Creating dummy variables using one-hot encoding on the Venue Category column

```
In [44]: df_top_50_required_onehot.head()
```

Out[44]:

	Accessories Store	Adult Boutique	African Restaurant	Airport	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Amphitheater	Antique Shop	...	Wine Bar	Wine Shop	Winery	Women's Store	F
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	C
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	C
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	C
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	C
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	C

5 rows × 390 columns

We see from the above that there are 390 unique venue categories.

3. Getting mean score per city across all venue categories

```
In [62]: df_grouped.head()
```

Out[62]:

City	Accessories Store	Adult Boutique	African Restaurant	Airport	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Amphitheater	Antique Shop	...	Wine Bar	Wine Shop	Winery
AMSTERDAM	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0
ATHENS	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.02	0.0	0.0
BANGKOK	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0
BARCELONA	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.01	0.0	0.0
BEIJING	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0

5 rows × 390 columns

```
In [63]: df_grouped.shape
```

Out[63]: (50, 390)

4. Performing K-Means Clustering on the location data. The output of which looks like:

```
In [92]: df_with_labels.head()
```

Out[92]:

	Cluster	City	Accessories Store	Adult Boutique	African Restaurant	Airport	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	...	Wine Bar	Wine Shop	Winery	Women's Store	R
0	3	AMSTERDAM	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0	0.01	0.
1	1	ATHENS	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.02	0.0	0.0	0.00	0.
2	2	BANGKOK	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0	0.00	0.
3	4	BARCELONA	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.01	0.0	0.0	0.00	0.
4	2	BEIJING	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.00	0.0	0.0	0.00	0.

5 rows × 392 columns

5. Next is to get the top 10 most common venues for each city which is as shown below:

```
In [111]: cluster_features.head()
```

Out[111]:

	Cluster	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
0	3	AMSTERDAM	Hotel	Coffee Shop	Cocktail Bar	Breakfast Spot	Bookstore	Plaza	Canal	Café	Bar	Bakery
1	1	ATHENS	Café	Coffee Shop	Bar	Historic Site	Meze Restaurant	Theater	Greek Restaurant	Cocktail Bar	Boutique	Bookstore
2	2	BANGKOK	Hotel	Coffee Shop	Shopping Mall	Park	Thai Restaurant	Noodle House	Spa	Bookstore	Clothing Store	Asian Restaurant
3	4	BARCELONA	Hotel	Coffee Shop	Plaza	Tapas Restaurant	Burger Joint	Pizza Place	Breakfast Spot	Ice Cream Shop	Argentinian Restaurant	Park
4	2	BEIJING	Historic Site	Hotel	Park	Café	Shopping Mall	Brewery	Chinese Restaurant	Dumpling Restaurant	Pizza Place	Yunnan Restaurant

This ends the Analysis phase. Next we will filter for each cluster number and try to find out the labels for each cluster.

Results & Discussion

Results

By now, we have clustered the cities ensuring that the clustering exercise produces the most ***optimum number of clusters*** into which the cities will be grouped.

Next, we shall filter for each cluster number and try to find out the Top 3 venues that define the ***cluster's uniqueness***.

The outcome of the analysis has been presented in the table below:

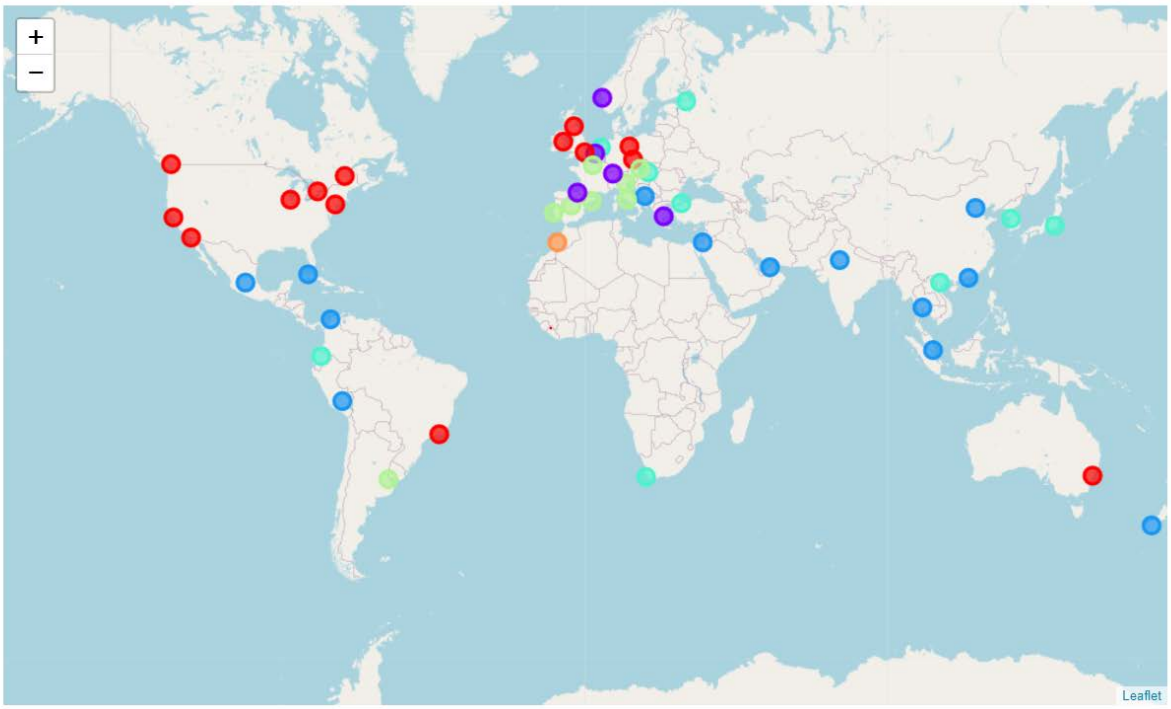
Cluster	Cities	Number Of cities	Top 3 Venues (Count in the cluster)	Cluster label
0	➢ Berlin ➢ Chicago ➢ Dublin ➢ Edinburgh ➢ London ➢ New York ➢ Prague ➢ Quebec-city ➢ Rio-de-Janerio ➢ San-Diego ➢ San-Francisco ➢ Sydney ➢ Toronto ➢ Vancouver	14	➢ Park (14) ➢ Coffee-shop (12) ➢ Ice cream shop (10)	**Parks**
1	➢ Athens ➢ Bergen ➢ Bruges ➢ San-Sebastian ➢ Zurich	5	➢ Bar (5) ➢ Hotel (4) ➢ Bakery (3)	**Bars**
2	➢ Bangkok ➢ Beijing	13	➢ Hotel (13) ➢ Cafe (8)	**Hotels**

	➢ Cartagena ➢ Cusco ➢ Dubai ➢ Dubrovnik ➢ Havana ➢ Hong Kong ➢ Jaipur ➢ Jerusalem ➢ Queenstown ➢ San-Miguel-de-Allende ➢ Singapore		➢ Restaurant (8)	
3	➢ Amsterdam ➢ Budapest ➢ Cape-Town ➢ Hanoi ➢ Istanbul ➢ Quito ➢ Seoul ➢ St-Petersburg ➢ Tokyo	9	➢ Coffee Shop (9) ➢ Hotel (9) ➢ Park (6)	**Coffee Shops**
4	➢ Barcelona ➢ Buenos-Aires ➢ Lisbon ➢ Madrid ➢ Paris ➢ Rome ➢ Venice ➢ Vienna	8	➢ Plaza (8) ➢ Ice Cream Shop (7) ➢ Park (6)	**Plazas**
5	➢ Marrakesh	1	➢ Bed & Breakfast (1) ➢ Lounge (1) ➢ Moroccan Restaurant (1)	**Bed & Breakfast**

The same can be represented on the Map as:

```
In [108]: map_clusters
```

Out[108]:



Discussion

We have obtained **six** major cluster of cities characterized by **Parks, Bars, Hotels, Coffee Shops, Plazas and Bed & Breakfast**. All clusters barring Bed & Breakfast have multiple cities in them.

The city of Marrakesh seems to be quite interesting as it did not share similarities with the remaining 49 cities. Something to really check out!!

Also, for cluster 3, Coffee Shops and Hotels share the same count, i.e. 3. We have however chosen **Coffee Shops** for the following reasons:

- Coffee Shops does not qualify as label for any other cluster
- Hotels has already been assigned to cluster 2

Conclusion

Purpose of this project was to identify similar cities among top 50 most beautiful cities globally in terms of venues in their vicinity. This was primarily done in order to understand the cities better in terms of their lifestyle and choices. The objective behind this project was to help travel enthusiasts in planning and make intelligent decisions while travelling/ recommending travel itineraries. This is equally useful for online portals/ travel magazines.

Note: While it is fully understandable that each city may have a different most common venue, we will try to label the clusters basis the top 3 venues for each cluster and for its underlying cities.

Also, cluster label/ Speciality_Hangouts, thus obtained after the analysis, clearly imply that the specialty will be present in **at least** the top 10 most visited place in any city of that particular cluster.

While this analysis tries to present a holistic view of the top 50 cities, final decision to consider one's travel plans will be strictly one's own decision and this may depend on several other factors and preferences.