Capstone Project – The Battle of Neighborhoods

INTRODUCTION

Every year, lots of people travel to different places for their vacations or get-togethers. People have different preferences and choices and they would want to select their destination vacation city according to that. This report proposes a business venture to create a mobile and web application to help people decide on finding a vacation city which matches their likings.

BUSINESS PROBLEM

It has always been a challenge for people to find a vacation destination as per their criteria. People just randomly search for most popular destinations on internet and take decisions. However a popular vacation city in the world does not mean that everyone will like that city for tourism. E.g. people with small kids may like Florida, but people with a taste of history may like Athens. Also some people wants family vacations whereas others may want to have bachelor theme vacation. Some people may like vacation in a quiet place whereas other people may need vacations with lots of activities.

People can search on internet for most popular vacation spots but they don't get an option to select one or more criteria based on which data can be presented for different cities. As a user, I would like to evaluate different cities in world based on my criteria and preference.

To provide a solution to this challenging problem, a new business venture in the shape of a mobile or web application is proposed. Using this app, people will get an option to assess different cities based on various categories like food, arts, entertainment, recreation etc. Users can select one or more cities, select one or more categories and then the application will provide comparison of these cities based on available data for likes, cost, activities, trends etc. As an output, users will see a table where they can see the selected cities rated based on different criteria. This will help them make an effective decision.

CityName	Venue Likes	Weighted Like	Low Cost
Athens, Greece	10441.2	2682.94	1377.69
Madrid, Spain	9797.8	2314.99	903.45
Paris, France	5552.8	1265.99	1185.60
Prague, Czech Republic	11354.0	2262.36	1434.04
Rome, Italy	6638.6	1168.34	1092.97
Vienna, Austria	6629.0	1551.86	935.91

DATA

User will have an option to select or enter name of one or more cities. Application will then use Nominatim geocoding API to retrieve the latitude and longitude of this location. This data will be further used to generate more data for analysis.

https://nominatim.openstreetmap.org/search?

For the purpose of this project and analysis, we will use the following six cities:

1. 'Athens, Greece'
2. 'Paris, France'
3. 'Prague, Czech Republic'
4. 'Vienna, Austria'
5. 'Madrid, Spain'
6. 'Rome, Italy'

Users will also have an option to select one or more categories and define a weightage to each of the category. If no weightage is defined, then each category will be given equal weightage. For the purpose of this project and analysis, we will use the following five categories:

1. 'Museum'
2. 'Shops and Service'
3. 'Italian Restaurant'
4. 'Bars'
5. 'Outdoors and Recreation'

For the actual analysis, we will fetch further data using Foursquare API's. Using these API's, we will get information of the venues which falls under the selected city and categories selected. API's like search, likes, and explore will be used to get information about the selected cities. This data will help us get the following data for each selected city and category.

1. How many users liked venues in each city
2. How much expensive are the venues in each city

Based on above data, aggregation and other calculations will be done to calculate the mean of likes, mean of weighted means and cost/price for each city. Users with use this analysis and information to make a decision on whether they would like to select this city as their destination for vacation.

Following FourSquare APIs will be called to retrieve data:

https://api.foursquare.com/v2/venues/explore?

https://api.foursquare.com/v2/venues/search?

https://api.foursquare.com/v2/venues/{}/likes?

METHODOLOGY

1. User will select one or more cities which they have short listed as their vacation destination and would like to use this application for making a final decision. For the purpose of this project, we have selected seven cities in Europe. The application can be enhanced later to have user select a list of cities.
2. The application gets the latitudes and longitudes of these cities by using Nominatim API. A function is written to which a list of city names is sent. The functions gets the latitudes and

longitudes for a city. We get two lists - one list has latitudes for all seven cities and second list has longitudes of all 7 cities.

3. Using FourSquare APIs we retrieve data for these cities and categories. We loop thru the list of city and category and for each combination, we perform following steps:

    a. We get the top 50 venues for each city and category combinations using venue search API. E.g. we will get the top 50 bars in Paris. Also we will get top 50 museums in Paris and so on.
    b. For each of these venues, we use the 'like' API to get the counts of likes from users.
    c. For each of these venues, we use the 'explore' API to get the venue which costs less than $10 or more than $10. The data is listed under two columns.

    All the data is then stored in a dataframe as shown below.

	CityName	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Venue Likes	Parent Category Name	Trending Price Low	Trending Price High
0	Athens, Greece	37.9841493	23.7279843	Hellenic Motor Museum (Ελληνικό Μουσείο Αυτοκι...	37.991397	23.730054	Museum	119	Museum	57	47
1	Athens, Greece	37.9841493	23.7279843	Museum of illusions	37.976844	23.722807	Museum	19	Museum	58	96
2	Athens, Greece	37.9841493	23.7279843	National Archaeological Museum (Εθνικό Αρχαιολ...	37.989026	23.732529	History Museum	796	Museum	67	61
3	Athens, Greece	37.9841493	23.7279843	Museum of Islamic Art (Μουσείο Ισλαμικής Τέχνης)	37.979228	23.720305	Art Museum	55	Museum	51	103
4	Athens, Greece	37.9841493	23.7279843	Museum of the City of Athens (Μουσείο της Πόλε...	37.979077	23.731573	History Museum	19	Museum	49	103
5	Athens, Greece	37.9841493	23.7279843	Numismatic Museum (Νομισματικό Μουσείο)	37.977851	23.735339	Museum	119	Museum	49	105

4. Once we get the dataframe, we do some exploratory analysis
    - We use head, shape, and describe functions to understand the data. The dataframe contains 1205 rows which we will use for analysis. The minimum venue likes are 0 and maximum is 6831.
    - We got a count of venues returned for each city using value_counts. This tells the number of venues analyzed for each city. Prague has highest value of 227 and Rome has lowest value of 184.

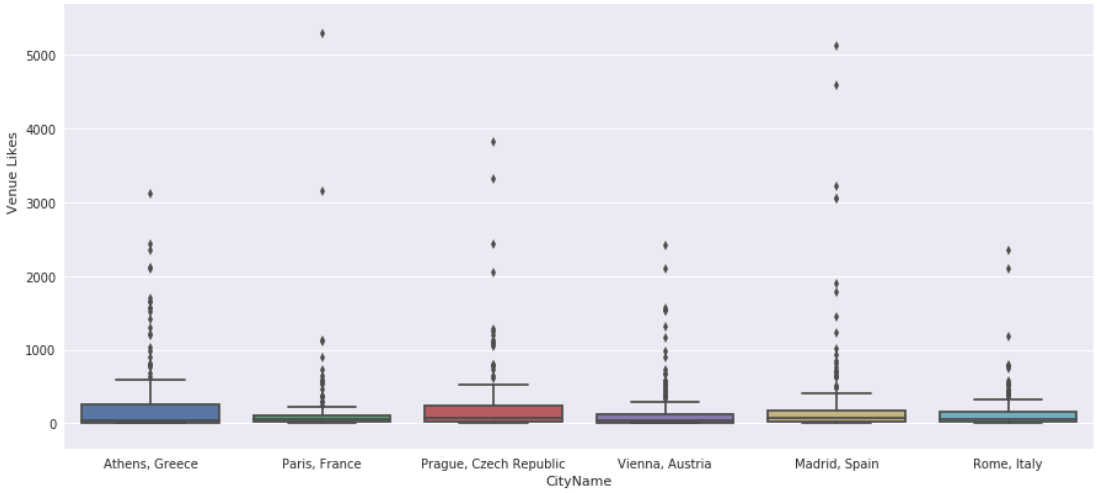
```
Prague, Czech Republic    227
Vienna, Austria           214
Athens, Greece            201
Paris, France             194
Madrid, Spain             185
Rome, Italy               184
```

    - We add a new column 'Price Low %'. We are only interested in understanding how many venues have low cost venues. We calculate the % using the following formula

        'Trending Price Low' / ('Trending Price Low' + 'Trending Price High')

	CityName	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Venue Likes	Parent Category Name	Trending Price Low	Trending Price High	Price Low %	Weighted_Likes
0	Athens, Greece	37.9841493	23.7279843	Hellenic Motor Museum (Ελληνικό Μουσείο Αυτοκι...	37.991397	23.730054	Museum	119	Museum	57	47	54.81	47.6
1	Athens, Greece	37.9841493	23.7279843	Museum of illusions	37.976844	23.722807	Museum	19	Museum	58	96	37.66	7.6
2	Athens, Greece	37.9841493	23.7279843	National Archaeological Museum (Εθνικό Αρχαιολ...	37.989026	23.732529	History Museum	796	Museum	67	61	52.34	318.4
3	Athens, Greece	37.9841493	23.7279843	Museum of Islamic Art (Μουσείο Ισλαμικής Τέχνης)	37.979228	23.720305	Art Museum	55	Museum	51	103	33.12	22.0
4	Athens, Greece	37.9841493	23.7279843	Museum of the City of Athens (Μουσείο της Πόλε...	37.979077	23.731573	History Museum	19	Museum	49	103	32.24	7.6

Plotted **Box plot** – which showed that data is very scattered. There are some outliers also which needs to be treated.



- Grouped the dataset by City Name and calculated the total of likes for venues in each city.

CityName	Venue Likes
Athens, Greece	10441.2
Madrid, Spain	9797.8
Paris, France	5552.8
Prague, Czech Republic	11354.0
Rome, Italy	6638.6
Vienna, Austria	6629.0

5. We slice the dataset to get limited data so that we can further analyze the data. The slicing is done to first understand the data from 'Venue Likes' point of view.

```
VenueDataLikes = VenueData[['CityName','Parent Category Name','Venue Likes']]
VenueDataLikes
```

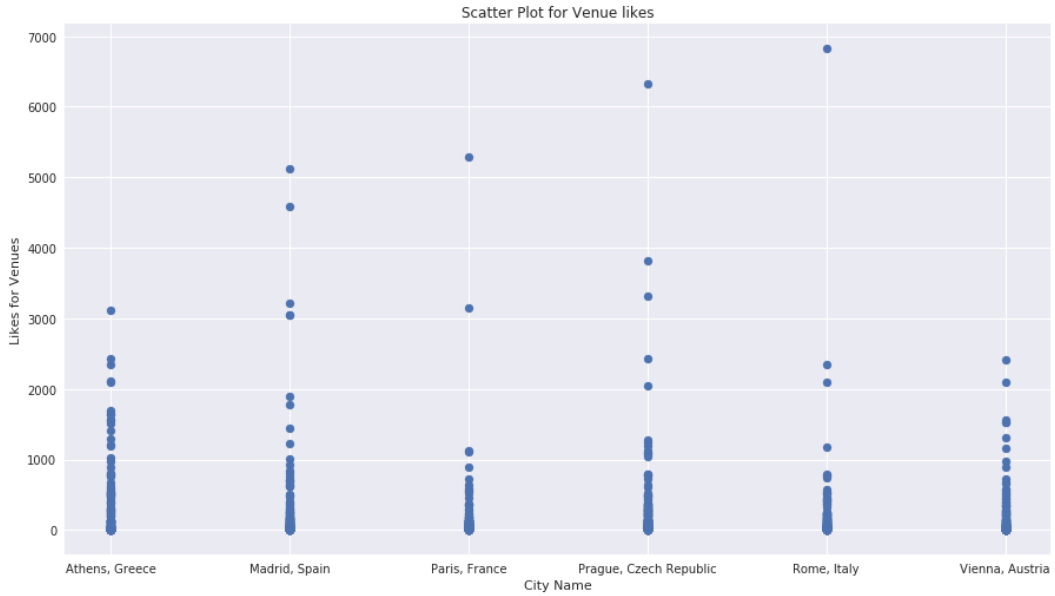
]:

	CityName	Parent Category Name	Venue Likes
0	Athens, Greece	Museum	119
1	Athens, Greece	Museum	19
2	Athens, Greece	Museum	796
3	Athens, Greece	Museum	55

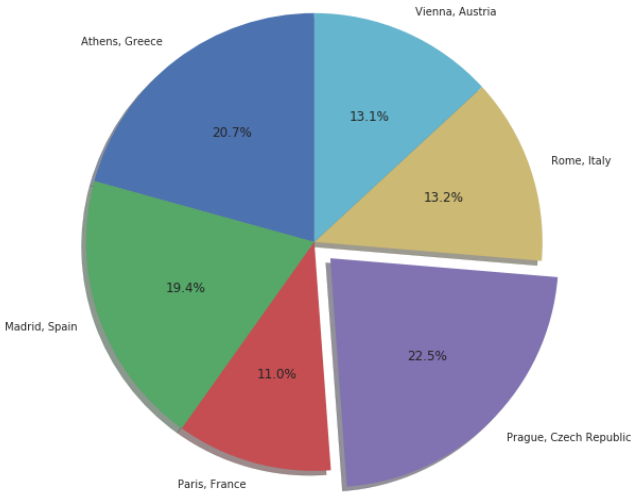
6. We group the dataset by CityName to get a sum of 'Venue Likes'

CityName	Venue Likes
Athens, Greece	10441.2
Madrid, Spain	9797.8
Paris, France	5552.8
Prague, Czech Republic	11354.0
Rome, Italy	6638.6
Vienna, Austria	6629.0

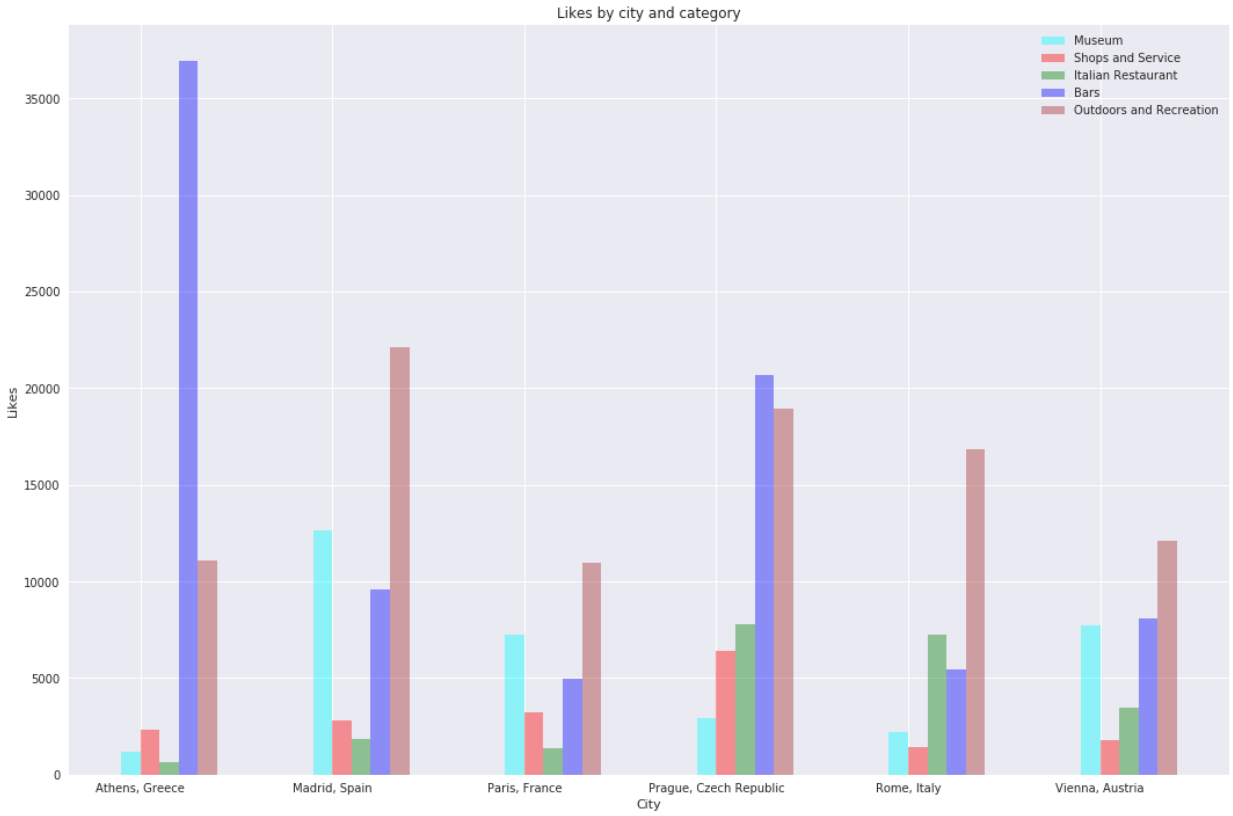
7. We draw a scatter plot for city and venue likes to understand the distribution of data.



8. We draw a pie chart which displays the pie for each city based on the total likes for venues within that city. The pie which has maximum likes is expanded.



9. To further explore, we draw a bar chart, which displays the venue likes for each city and each category. This gives a good comparison of all likes for all cities.



10. After analyzing for venue likes, we would like to apply weights for each category to the dataset to understand what difference does this makes. We add a new column and apply the weights to the venue likes - ''Weighted_Likes'

```
VenueData['Weighted_Likes'] = VenueData.apply(GetWeighted, axis=1)
```

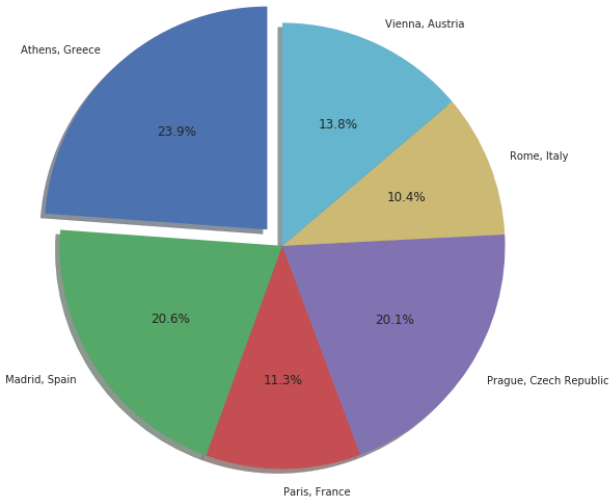
```
VenueData.head(5)
```

]:

	CityName	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Venue Likes	Parent Category Name	Trending Price Low	Trending Price High	Price Low %	Weighted_Likes
0	Athens, Greece	37.9841493	23.7279843	Hellenic Motor Museum (Ελληνικό Μουσείο Αυτοκι...	37.991397	23.730054	Museum	119	Museum	57	47	54.807692	47.6
1	Athens, Greece	37.9841493	23.7279843	Museum of Illusions	37.976844	23.722807	Museum	19	Museum	58	96	37.662338	7.6
2	Athens, Greece	37.9841493	23.7279843	National Archaeological Museum (Εθνικό Αρχαιολ...	37.989026	23.732529	History Museum	796	Museum	67	61	52.343750	318.4

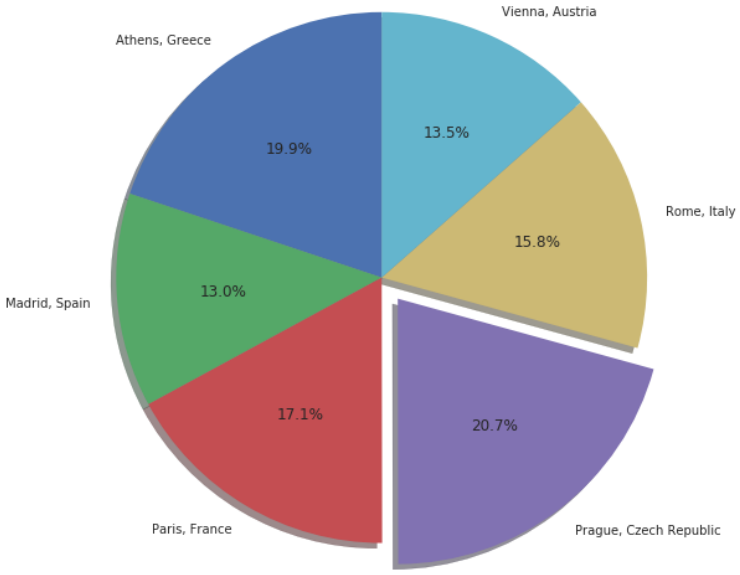
11. We further slice, group, and draw pie chart to see which city stands on top based on weighted likes. The pie which has maximum weighted likes is expanded

CityName	Weighted_Likes
Athens, Greece	2682.94
Madrid, Spain	2314.99
Paris, France	1265.99
Prague, Czech Republic	2262.36
Rome, Italy	1168.34
Vienna, Austria	1551.86



12. After analyzing for 'Venue likes' and 'Weighted venue likes', we will further slice, group, and draw pie chart to see which city stands on top based on Low cost %. The pie which has lowest cost % will be expanded.

CityName	Price Low %
Athens, Greece	1377.692
Madrid, Spain	903.448
Paris, France	1185.598
Prague, Czech Republic	1434.042
Rome, Italy	1092.974
Vienna, Austria	935.906



13. Finally, we put all the data in a single dataframe which will be used by a user to analyze the city which best suits his criteria.

CityName	Venue Likes	Weighted Like	Low Cost
Athens, Greece	10441.2	2682.94	1377.69
Madrid, Spain	9797.8	2314.99	903.45
Paris, France	5552.8	1265.99	1185.60
Prague, Czech Republic	11354.0	2262.36	1434.04
Rome, Italy	6638.6	1168.34	1092.97
Vienna, Austria	6629.0	1551.86	935.91

RESULTS

We finally get a dataframe where we have data for all three factors. Based on these factors, Users can analyze as below and take a decision on which city would they prefer to go for vacation. We evaluate each city by 3 factors:

- Likes for Top 50 venues in each city
- Likes for Top 50 venues in each city by applying different weights to each category
- Cost/Price of trending venues

Athens: Very Good venue likes, Highest Weighted likes, but cost/price is also second highest of all cities

Madrid: Both Venue and weighted likes are third in list. However the cost/price is highest.

Paris: Both Venue and weighted likes are lowest and the cost/price is also average in the list.

Prague: Highest Venue likes and lowest cost/price. So this seems the best in the list of cities.

Rome: Average venue and weighted likes, but cost is towards higher side.

Vienna: Average venue and weighted likes. However cost is towards lower side.

DISCUSSION

- We observes that the box plot displays a very wide range of data elements. We need to analyze the variation and wrangle the data to produce more effective results.
- Outliers should be treated based on scatter and box plots.
- There is a significant different in the value counts for each city. E.g. Prague has 227 rows whereas Rome has 184. We should investigate and enhance our dataset.
- For pricing, we have made an assumption that anything less than $10 will be considered as low price and anything above $10 will be considered as high. We can change the criteria and analyze what different does this makes.
- In the dataframe, we have venues with likes = 0. We should exclude these rows to make the analysis more effective.

CONCLUSION

This project allows users to assess different cities based on the likes for venues, weighted likes for venues, and low cost percentage and for certain categories selected by user. This will help users to select the vacation city as per their preferences. The application also displays data analysis in different graphic formats which provides a good data visualization to the users. Finally the relevant data for each selected city is displayed in a dataframe for user to make a comparative analysis. As we move forward, we can add more features and analysis factors to the application so that users can make more in depth study to make a decision.