Capstone Project - The Battle of the Neighbourhoods
Project: "Dress to Impress"

APPLIED DATA SCIENCE CAPSTONE BY IBM

Silvia Ioana Carbunarea Prisecaru

silvia.prisecaru@gmail.com

1.EXECUTIVE SUMMARY

Bucharest has a burgeoning Cafe culture and offers residents an array of venues catering to every budget and desire. Andreea is a fashion vlogger moving to Bucharest, Romania, to follow her dream of opening her own coffee shop. The factors that will influence our decision are: diversity of neighbourhood amenities, closeness of similar neighbourhoods, number of existing coffee shops in the neighbourhood.

For this purpose, multiple sources were used to gather as much information about the neighbourhoods in Bucharest: Wikipedia, Foursquare, Geocoding, wall-street.ro. The methods used included: data visualization, one-hot-encoding, k-means clustering. Price/sqm was added as a later factor, to helped narrow down the list to one Neighbourhood: Drumul Taberei.

TABLE OF CONTENTS

2. INTRODUCTION

Bucharest is the capital of Romania. It is the nation's largest city in terms of area and population. The metropolitan area of Bucharest has an estimated population of 2.27 million people. The city is European Union's sixth largest in terms of population within city limits.

Bucharest has a growing cultural scene, in fields including the visual arts, performing arts, and nightlife. Unlike other parts of Romania, such as the Black Sea coast or Transylvania, Bucharest's cultural scene has no defined style, and instead incorporates elements of Romanian and international culture.

Sometime called "Little Paris", the today Bucharest city is a different town. Some buildings has been restored to keep the connection with the past, but the city is now a blend of old and modern.

Downtown area - include the most visited objectives of Bucharest: Parliament Palace, the Bucharest University, almost all museums from city, Calea Victoriei and the one of the oldest public garden from Bucharest: Cismigiu. Also, here you will find the Lipscani area, well known for the numerous bars, pubs and night life.

Bucharest has a burgeoning Cafe culture and offers residents an array of venues catering to every budget and desire. Thus, in recent years, it has become the main place the new generation from Romania move to try and make a living, while living the life of their dreams. This has also triggered an accelerated increase in traffic stalling, thus making it vital to live and work as close as possible, preferably in the same area.

3. BUSINESS PROBLEM

Andreea is a fashion vlogger moving to Bucharest, Romania, to follow her dream of opening her own coffee shop. She currently makes her living out of her Youtube and Instagram accounts, making movies about her whereabouts. She is looking to find out which neighbourhood from Bucharest is the most suitable to fit her highly active lifestyle. The neighbourhood should lack too many coffee shops, so that it could embrace her own business. Doing so, she hopes to maximize her chances of success.

The target audience for this project should also be other self-employed people looking for fame and cash-flow generated by their presence in certain places, in the city of Bucharest. Also, people looking to invest in consumer-oriented business should find this study helpful in decided where and why to invest in certain places.

4. DATA

Based on definition of our problem, factors that will influence our decision are:

- diversity of neighbourhood amenities

- closeness of similar neighbourhoods

- number of existing coffee shops in the neighbourhood

The data used for this report are sourced using multiple sources, in order to ensure accuracy and comparability.

4.1 LIST OF THE NEIGHBOURHOODS FROM BUCHAREST

This report used the Wikipedia page to identify all Neighbourhoods in Bucharest, Romania: https://en.wikipedia.org/wiki/Category:Districts_of_Bucharest. To do so, the page was initialized as a beautifulsoup object. Then, a csv file was created to write the row ;Neighbourhood', followed by the scrapping of the scrapping of the internet page to extract the list of the Neighbourhoods. Further, the dataframe df was created using panda.

Importing Neighbourhoods in Bucharest from Wikipedia using BeautifulSoup

```
source = requests.get('https://en.wikipedia.org/wiki/Category:Districts_of_Bucharest').text
soup = BeautifulSoup(source, 'lxml')
```

```
csv_file = open('bucharest.csv', 'w')
csv_writer = csv.writer(csv_file)
csv_writer.writerow(['Neighbourhood'])
```

```
[7]: 15
```

```
mwcg = soup.find_all(class_ = "mw-category-group")
length = len(mwcg)
for i in range(1, length):
    lists = mwcg [i].find_all('a')
    for list in lists:
        nbd = list.get('title')
        csv_writer.writerow([nbd])
```

```
csv_file.close()
```

```
df = pd.read_csv('bucharest.csv')
```

The list consisted of 39 neighbourhoods, some of which had “, Bucharest” in their title- which needed to be cleaned up.

In [11]:
```
df.shape
```
Out[11]: (39, 1)

In [12]:
```
df.head()
```
Out[12]:

	Neighbourhood
0	Băneasa, Bucharest
1	Berceni, Bucharest
2	Bucureștii Noi
3	Centrul Civic
4	Colentina, Bucharest

In [13]:
```
df['Neighbourhood'] = df.Neighbourhood.str.replace(', Bucharest,?' , '')
```

In [14]:
```
df.head()
```
Out[14]:

	Neighbourhood
0	Băneasa
1	Berceni
2	Bucureștii Noi
3	Centrul Civic
4	Colentina

4.2 LATITUDE AND LONGITUDE OF NEIGHBOURHOODS

To locate the latitude and longitude of the neighbourhoods in Bucharest, this analysis used Google Maps API Geocoding. Using a function, the coordinates of each neighbourhood was extracted and appended to the lists lat & lng, which were added to the dataframe df.

```
latitudes = []
longitudes = []
distance = []

for nbd in df["Neighbourhood"] :
    address = nbd + ", Bucharest,Romania"
    url = 'https://maps.googleapis.com/maps/api/geocode/json?address={}&key={}'.format(address, google_api_key)
    obj = json.loads(requests.get(url).text)

    results = obj['results']
    lat = results[0]['geometry']['location']['lat']
    lng = results[0]['geometry']['location']['lng']

    latitudes.append(lat)
    longitudes.append(lng)
```

```
: df['Latitude'] = latitudes
  df['Longitude'] = longitudes
```

```
: df.head()
```

18]:

	Neighbourhood	Latitude	Longitude
0	Băneasa	44.493726	26.076048
1	Berceni	44.389221	26.118203
2	Bucureștii Noi	44.493619	26.031081
3	Centrul Civic	44.427285	26.092441
4	Colentina	44.465766	26.148647

4.3 LIST OF VENUES IN BUCHAREST'S NEIGHBOURHOODS

This report used Foursquare API to find amenities and their type and location in every neighbourhood from Bucharest A function was created to get venues within a radius of 100, taking into account the coordinates from the dataframe df. 73 venues were identified and 47 unique categories,

```python
def getNearbyVenues(names, latitudes, longitudes, radius=100):

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

        results = requests.get(url).json()["response"]['groups'][0]['items']
         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 = ['Neighbourhood',
                  'Neighbourhood Latitude',
                  'Neighbourhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

```python
bucharest_venues = getNearbyVenues(names=df['Neighbourhood'],
                                   latitudes=df['Latitude'],
                                   longitudes=df['Longitude']
                                  )
```

-

5. METHODOLOGY & RESULTS

Python offers a wide range of libraries and functions to support a robust and detailed data analysis.

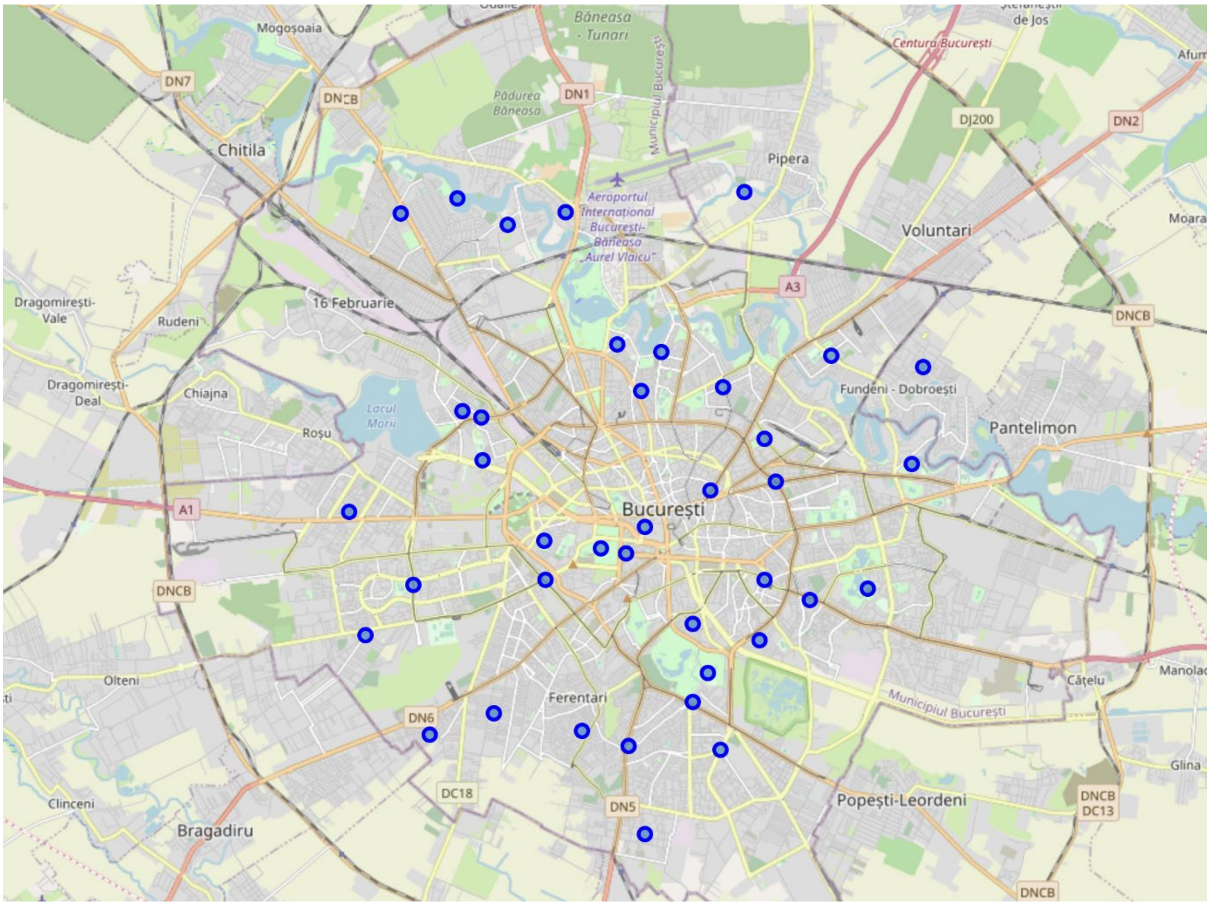
5.1 MAP VISUALIZATION

The analysis began by analysing the map of Bucharest and its 39 Neighbourhoods. It is visible that their position needs further information in order to be able to instruct Andreea which neighbourhood to choose.

Map of Bucharest's Neighbourhoods

```
from geopy.geocoders import Nominatim
address = 'Bucharest, Romania'

geolocator = Nominatim(user_agent="to_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
```

```
map_bucharest = folium.Map(location=[latitude, longitude], zoom_start=12)

for lat, lng, neighbourhood in zip(df['Latitude'], df['Longitude'], df['Neighbourhood']):
    label = '{}'.format(neighbourhood)
    label = folium.Popup(label, 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(map_bucharest)

map_bucharest
```

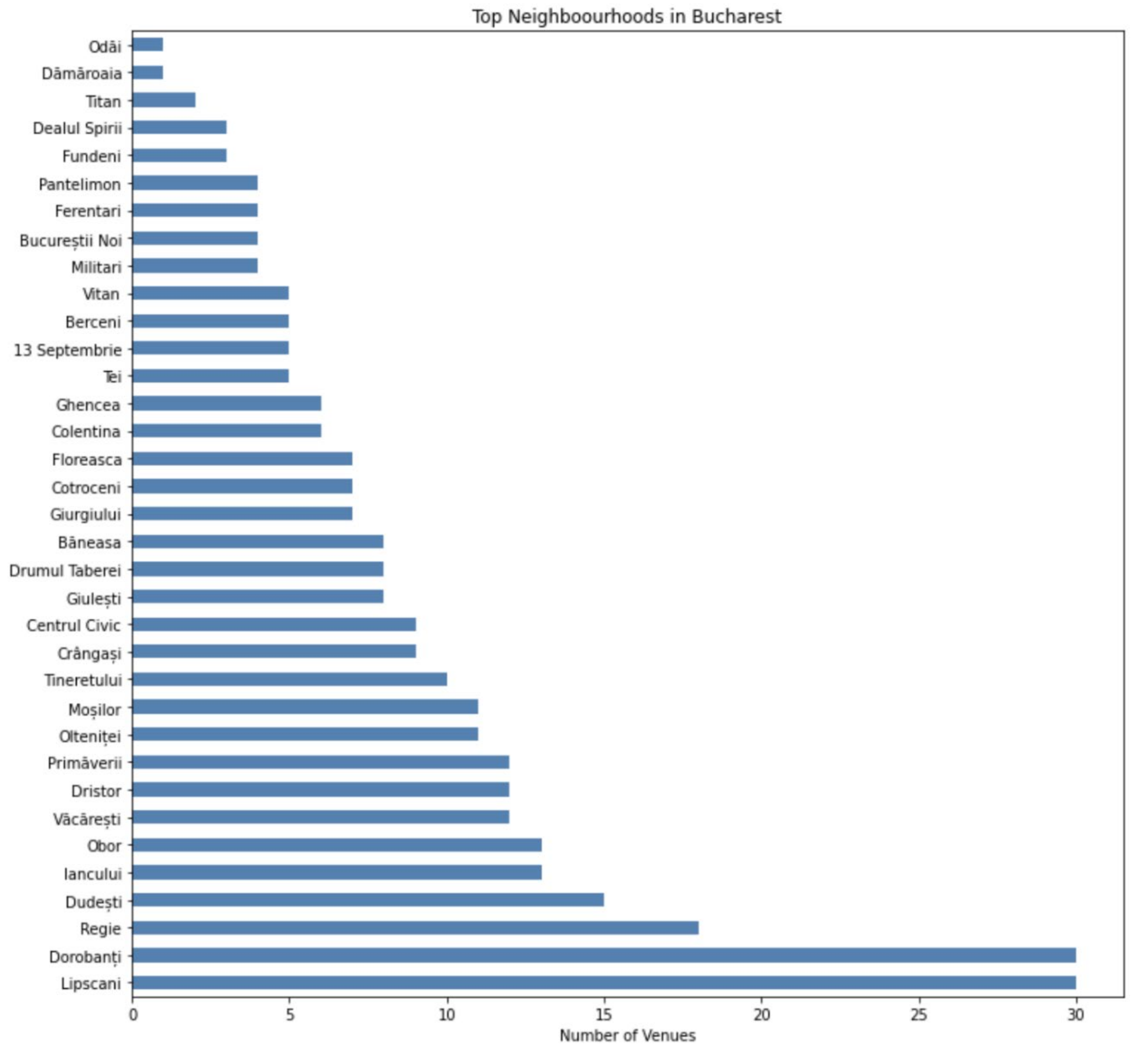


5.2 DATA VISUALIZATION

Matplotlib is another library that can be called by Python to visualise data. While plotting into a horizontal bar chart, there are 547 different venues, 110 unique categories from Bucharest within 300 meters radius, revied on Foursquare. It becomes highly apparent the most popular 3 places in terms of amenities are: Lipscani, Dorobanti and Iancului. However, these may not be the neighbourhoods that would fit the lifestyle and business ambitions of our young fashion vlogger.

```
]: import matplotlib as mpl
   import matplotlib.pyplot as plt
```

```
]: bc_venues.plot(kind='barh', figsize=(12, 12), color='steelblue')
   plt.xlabel('Number of Venues')
   plt.title('Top Neighboourhoods in Bucharest')
   plt.show()
```



5.4 ONE HOT ENCODING

Panda's method "get_dumies" was used to convert categorical values into binary. The one hot encoding process was built to prepare data to make predictions and clustering, taking into account the venue category. Furthermore, the resulted information was grouped by Neighbourhood, so that the frequency of each venue type would appear in the dataframe.

Analyzing each Neighbourhood

In [345]:
```
bucharest_onehot = pd.concat([bucharest_venues['Neighbourhood'],pd.get_dummies(bucharest_venues['Venue Category'])], axis=1)
print(bucharest_onehot.shape)
bucharest_onehot.head()
```

```
(308, 111)
```

Out[345]:

	Neighbourhood	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Auto Dealership	BBQ Joint	Bagel Shop	Bakery	Bar	...	Taco Place
0	Băneasa	0	0	0	0	0	0	0	0	0	...	0
1	Băneasa	0	0	0	0	0	0	0	0	0	...	0
2	Băneasa	0	0	0	0	0	0	0	0	0	...	0
3	Băneasa	0	0	0	0	0	0	0	0	0	...	0
4	Băneasa	0	0	0	0	0	0	0	0	0	...	0

5 rows × 111 columns

In [346]:
```
bucharest_grouped = bucharest_onehot.groupby('Neighbourhood').mean().reset_index()
bucharest_grouped
```

Out[346]:

	Neighbourhood	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Auto Dealership	BBQ Joint	Bagel Shop	Bakery	Bar	...	T Pl
0	13 Septembrie	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	...	0.000
1	Berceni	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.200000	0.000000	...	0.000
2	Bucureștii Noi	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.250000	0.000000	...	0.000
3	Băneasa	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.125000	0.000000	...	0.000

5.5 THE 5 MOST COMMON VENUES

We've built a function that calculates the most common venues per neighbourhood, in order to be better equipped when visualizing them.

Putting into Pandas framework

:48]:
```
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]
```

:49]:
```
num_top_venues = 5

indicators = ['st', 'nd', 'rd']

columns = ['Neighbourhood']
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))

neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighbourhood'] = bucharest_grouped['Neighbourhood']

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

neighborhoods_venues_sorted.head()
```

5.6 USING K-MEANS CLUSTERING

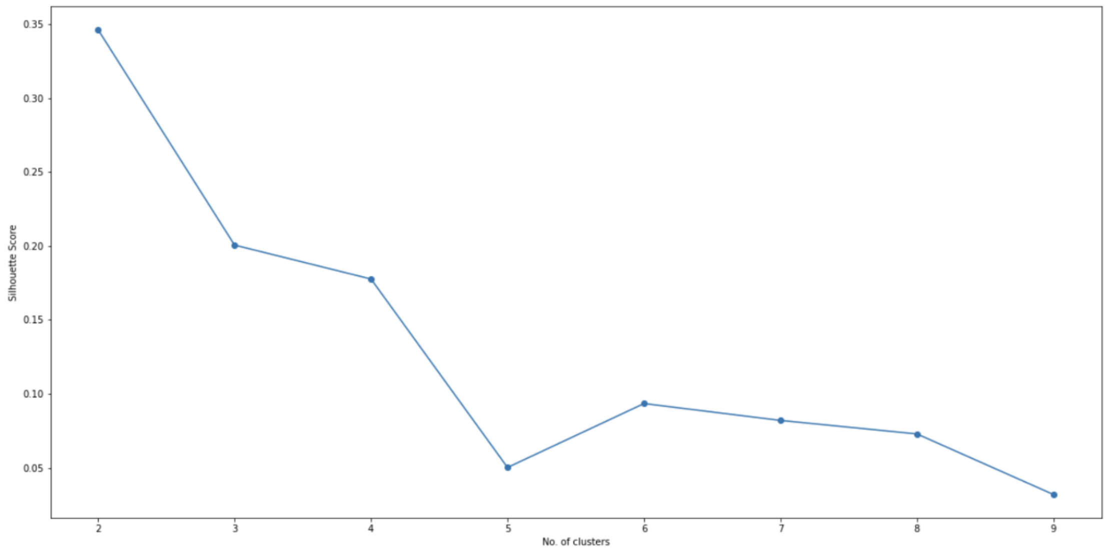
The encoded data needs to be trained using K-means Clustering algorithm to be able to group the neighbourhoods, based on their venues' types. First, we need to use Silhouette

Score to test de dissimilarity between clusters, and then data is fitted and neighbourhoods are clustered, using the optimal number of clusters.

```
### silhouette analysis  seeks to define the dissimilarity of clusters which means its a measure of how close each point in one cluster is to points in the neighboring clusters.
from sklearn.metrics import silhouette_samples, silhouette_score

indices = []
scores = []

for kclusters in range(2, max_range) :

    bgc = bucharest_grouped_clustering
    kmeans = KMeans(n_clusters = kclusters, init = 'k-means++', random_state = 0).fit_predict(bgc)

    score = silhouette_score(bgc, kmeans)

    indices.append(kclusters)
    scores.append(score)
```

```
plot(max_range, scores, "No. of clusters", "Silhouette Score")
```

```
]: plot(max_range, scores, "No. of clusters", "Silhouette Score")
```




```
]: opt = np.argmax(scores) + 2
   print(opt)
```

```
2
```

K-means for optimal number of clustering

```
kclusters = opt
kmeans = KMeans(n_clusters = kclusters, init = 'k-means++', random_state = 0).fit(bucharest_grouped_clustering)
```

```
neighborhoods_venues_sorted.drop(['Cluster Labels'], axis=1, inplace=True)
```

```
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
```

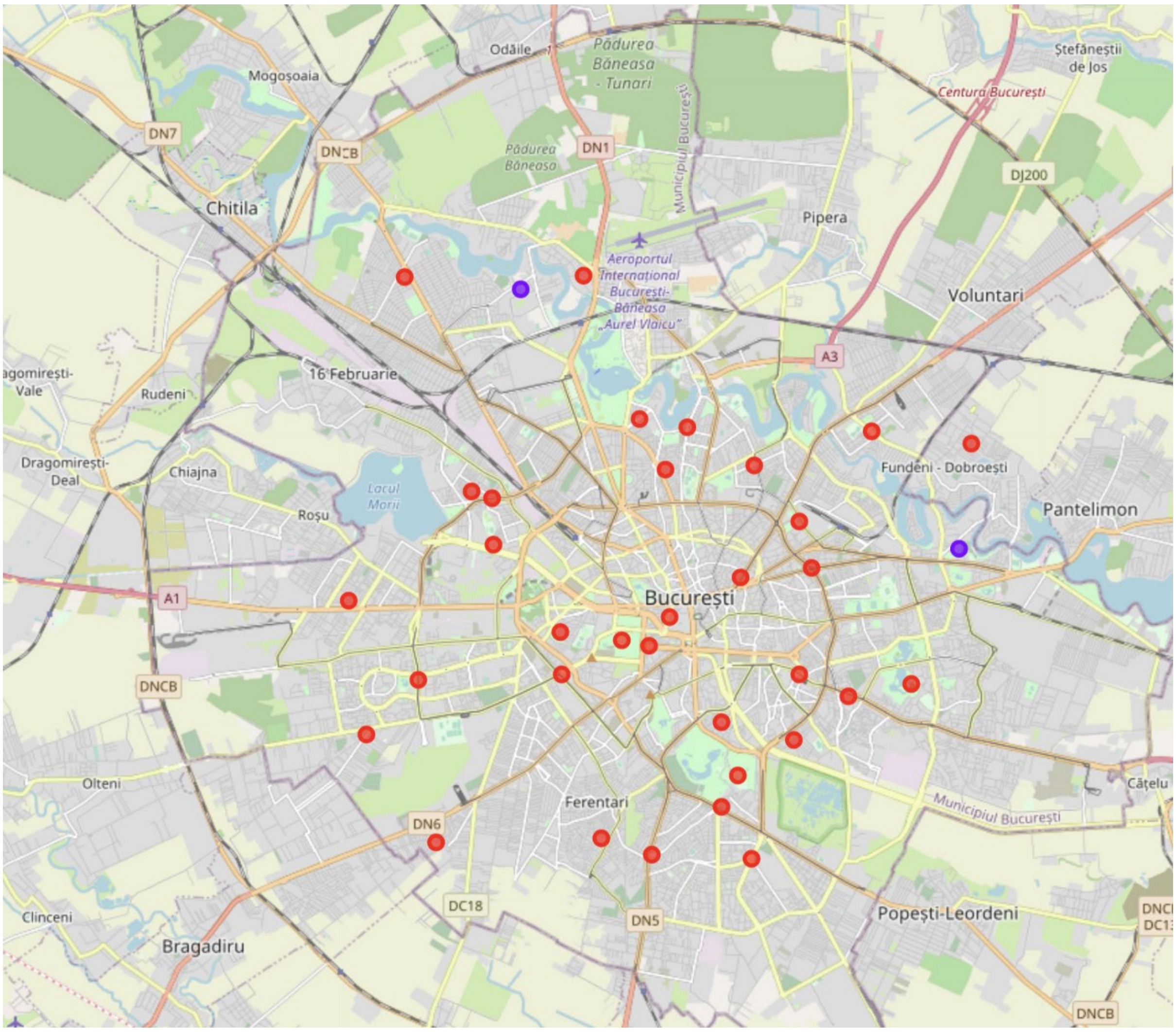
```
neighborhoods_venues_sorted.head()
```

]:

	Cluster Labels	Neighbourhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	0	13 Septembrie	Romanian Restaurant	Indian Restaurant	Plaza	Pizza Place	Department Store
1	0	Berceni	Pub	Lebanese Restaurant	Bakery	Cheese Shop	Fountain
2	0	Bucureștii Noi	Dessert Shop	Gym	Bakery	Supermarket	Fried Chicken Joint
3	0	Băneasa	Café	Restaurant	Tunnel	Theme Restaurant	Pizza Place
4	0	Centrul Civic	Restaurant	Theater	Romanian Restaurant	Clothing Store	Chocolate Shop

5.8 VIZUALIZATION OF CLUSTERS

Using Folium, we can visualize the clusters on the map, Most of the Neighbourhoods are grouped in Cluster 0, while 2 neighbourhoods situated at the extremes are in Cluster 1.



5.9 FURTHER DATA FILTERING

One request Andreea had was to move into a neighbourhood that has fewer Cafes, so that there will be a place to open her own Cafe business. Therefore, All neighbourhoods from Cluster 0 with Cafe listed in their top5 venues are filtered out. Moreover, Andreea is a highly sociable videoblogger. She needs many restaurants close by. So we should only choose neighbourhoods with restaurants listed as their most frequent venue.

```
In [376]: venues0a = venues0a[venues0a['4th Most Common Venue'] != 'Café']
```

```
In [377]: venues0a = venues0a[venues0a['5th Most Common Venue'] != 'Café']
```

```
In [378]: venues0a.shape
Out[378]: (27, 9)
```

```
In [379]: venues0a.head()
```

...

```
In [380]: venues0a['1st Most Common Venue'].value_counts()
```

...

Choosing Neighbourhood that has Restaurant as Most Common Venue

```
In [381]: venues=venues0a.loc[venues0a['1st Most Common Venue'] == 'Restaurant'].reset_index().drop(['index'],axis=1)
          venues
```

Out[381]:

	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Centrul Civic	44.427285	26.092441	0.0	Restaurant	Theater	Romanian Restaurant	Clothing Store	Chocolate Shop
1	Drumul Taberei	44.421340	26.034485	0.0	Restaurant	Grocery Store	Farmers Market	Park	Skating Rink
2	Fundeni	44.463675	26.173474	0.0	Restaurant	Bar	Wine Bar	Fried Chicken Joint	Dessert Shop
3	Tei	44.459806	26.118913	0.0	Restaurant	Doner Restaurant	Italian Restaurant	Electronics Store	Bar
4	Titan	44.420545	26.158415	0.0	Restaurant	Park	IT Services	Dessert Shop	Doner Restaurant

5.10 FURTHER INFORMATION

Andreea needs some pricing information, in order to decide whether she has the money to buy space for her own business. Data from an article was converted into a Dataframe and intersected with the filtered information.

```
: dict={'Neighbourhood':['Kiseleff','Aviatorilor','Herăstrau','Nordului','Dorobanți','Floreasca','Aviației','Unirii','Drumul Taberei','Giurgiului','Giulesti','Rahova','Ghe
]}
```

```
: df_price=pd.DataFrame(dict)
  df_price.head()
```

87]:

	Neighbourhood	Price/sqm
0	Kiseleff	2580
1	Aviatorilor	2580
2	Herăstrau	2410
3	Nordului	2410
4	Dorobanți	1990

```
: df_price = df_price.join(bucharest_merged.set_index('Neighbourhood'), on='Neighbourhood')
```

```
: df_price.head(20)
```

89]:

	Neighbourhood	Price/sqm	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Kiseleff	2580	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Aviatorilor	2580	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	Herăstrau	2410	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	Nordului	2410	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	Dorobanți	1990	44.459076	26.096738	0.0	Sushi Restaurant	Café	Bakery	Restaurant	Vegetarian / Vegan Restaurant
5	Floreasca	1990	44.466539	26.102152	0.0	Pool	Hotel	French Restaurant	Eastern European Restaurant	Lounge
6	Aviației	1870	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7	Unirii	1720	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
8	Drumul Taberei	1050	44.421340	26.034485	0.0	Restaurant	Grocery Store	Farmers Market	Park	Skating Rink
9	Giurgiului	1040	44.389770	26.093142	0.0	Playground	Pizza Place	Sandwich Place	Electronics Store	Supermarket
10	Giulesti	950	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
11	Rahova	940	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
12	Ghencea	940	44.411369	26.021560	0.0	Gym	Athletics & Sports	Pub	Supermarket	Bus Station

It turns out that Andreea should move and invest in Drumul Taberei as:

- it belongs to the most popular Cluster (Cluster 0)

- It has no Cafes listed in its top5 list

- it has Restaurants listed as its most frequent venue

- its price/sqm is at the lower end of the range.

7. DISCUSSION

In this report, it was attempted to use a combination of APIs in order to generated similar clusters of neighbourhoods from Bucharest. Based on the frequency of the venues located in these neighbourhoods, but also on the lack of presence of Cafes among the top 5 of the frequencies, a neighbourhood was to be selected that matches also Andreea's budget.

Further possible research could be done, were we to get a hold of further information about other dissimilarity criteria, such as: average income, average spending in Cafes, average time spent in cafes, and closeness of office building from cafes. These could be factored into the clustering scheme, for more accurate results.