Data science capstone project report

By Nupur Saboo

Introduction

Paris is the most popular city in France with an estimated population of about 2 million people as of 2020 in about 150 sq. kilometers. Paris is also one of the most desirable tourist destinations in the world. It is known as the city of love and is widely known for its fashion sense, pastries and cafes. The city accommodates a lot of creative people who keep moving from one place to another looking for inspirations and often spend hours working in one place, making cafes a very popular and regularly visited spot for many.

Corner shops and popular streets are ideal to set up a café, but to succeed with yet another café in a market with such high competition, one must provide affordable and delicious looking merchandise, services and have a unique ambiance which people will not find easily elsewhere.

Business problem

People working on projects need a place to sit where they can consume beverage such as tea, coffee etc. and eat fast food that does not require the use of both hands and is not very messy or time consuming. Thus, the aim will be to set up the café in a place which attracts a lot of people, thereby allowing owners of the outlets to earn maximum profit out of them.

Data

The data of this project comes from multiple sources.

- **Neighborhood**

The data for neighborhoods in Paris was extracted by web scraping using BeautifulSoup library for Python. The neighborhood data is scrapped form the Wikipedia page.

```
data = requests.get("https://en.wikipedia.org/wiki/Category:Suburbs_of_Paris").text

soup = BeautifulSoup(data,'html.parser')

neighborhoodList = []

for row in soup.find_all("div", class_="mw-category")[0].findAll("li"):
    neighborhoodList.append(row.text)

paris_df = pd.DataFrame({"Neighborhood": neighborhoodList})

paris_df.head()
```

- **Geocoding**

```
def get_latlng(neighborhood):
    lat_lng_coords = None
    while(lat_lng_coords is None):
        g = geocoder.arcgis('{}, Paris, France'.format(neighborhood))
        lat_lng_coords = g.latlng
    return lat_lng_coords

coords = [ get_latlng(neighborhood) for neighborhood in paris_df["Neighborhood"].tolist
```

- **Venue Data**

Venue data is found by passing the required parameters in the Foursquare API and creating a data frame to contain all the details.

```
for lat, long, neighborhood in zip(paris_df['Latitude'], paris_df['Longitude'], paris_d

  url = 'https://api.foursquare.com/v2/venues/explore'

  params = dict(
  client_id=CLIENT_ID,
  client_secret=CLIENT_SECRET,
  v='20180323',
  ll='40.7243,-74.0018',
  query='coffee',
  limit=1
  )
resp = requests.get(url=url, params=params)
data = json.loads(resp.text
results = requests.get(url).json()["response"]['groups'][0]['items']


    for venue in results:
        venues.append((
```

Methodology

An analysis of principals, methods and rules have been made to ensure that the interface works accurately.

- ## One hot encoding

It is a process by which categorical variables are converted into a form that could be turned into ML algorithm to improve predictions.

```
[39]: paris_oh = pd.get_dummies(venues_df[['VenueCategory']], prefix="", prefix_sep="")

      # add neighborhood column back to dataframe
      paris_oh['Neighborhoods'] = venues_df['Neighborhood']

      # move neighborhood column to the first column
      fixed_columns = [paris_oh.columns[-1]] + list(paris_oh.columns[:-1])
      paris_oh = paris_oh[fixed_columns]

      print(paris_oh.shape)
      paris_oh.head()
```

- ## Folium

All cluster visualization is done using folium which in turn generates a map.

```
[54]: 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(parisclusters)]
      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(paris_merged['Latitude'], paris_merged['Longitude'],
          label = folium.Popup(str(poi) + ' - Cluster ' + str(cluster), parse_html=True)
          folium.CircleMarker(
              [lat, lon],
              radius=5,
              popup=label,
              color=rainbow[cluster-1],
              fill=True,
              fill_color=rainbow[cluster-1],
```

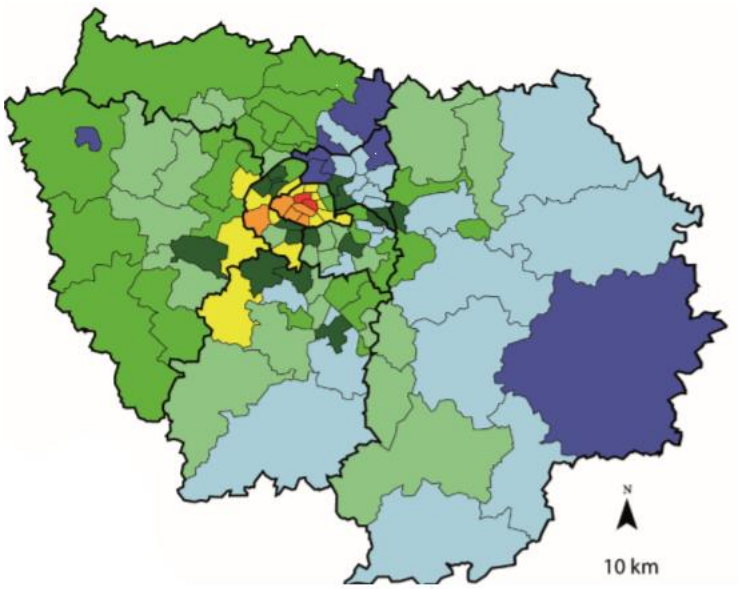

Fig-Population distribution in Paris.

Results

The neighborhoods are divided into n clusters. The clustered neighborhoods are seen using different colors to make them distinguishable.

Conclusion

Montmartre is the best location for cafes as it has a high population is close to some major tourist spot, but because of high competition in that region from the already existing cafes, Belleville will be a more suitable place to set up a new café.