Coursera Capstone Project: Applied Data Science

Sandeep Kunwar

Pashchimanchal Campus, TU

Pokhara, Nepal

Contents

1 Introduction

1.1 Background of Study

Kathmandu city is the capital of Nepal. Kathmandu is and has been for many years the center of Nepal's history, art, culture, and economy. The main objective is to find where are the perfect spots in the valley where fast food retail chains can be put up.

1.2 Problem

To open a business such as hotels, restaurants, etc. that requires some research about the physical geography and status of population in their area. Data might contribute investors to find better place to open business and other people can get good food. The main objective to find the perfect spots in the valley where fast food retail cabins can be put up, aiming at the above demographic and maximize profits out of them.

2 Data

2.1 Neighborhoods

The data of the neighborhoods in Kathmandu can be extracted out by web scraping using BeautifulSoup library for Python. The neighborhood data is scraped from a Wikipedia webpage "https://en.wikipedia.org/wiki/List_of_neighborhoods_of_Kathmandu".

2.2 Venue Data

Venue Data from the location data obtained after Web Scraping, the venue data is found out by passing in the required parameters to the FourSquare API, and creating another DataFrame to contain all the venue details along with the respective neighborhoods.

```
explore_df_list = []

for i, nbd_name in enumerate(df['Neighbourhood']):

    try :
        ### Getting the data of neighbourhood
        nbd_name = df.loc[i, 'Neighbourhood']
        nbd_lat = df.loc[i, 'Latitude']
        nbd_lng = df.loc[i, 'Longitude']

        radius = 1000 # Setting the radius as 1000 metres
        LIMIT = 30 # Getting the top 30 venues

        url = 'https://api.foursquare.com/v2/venues/explore?client_id={} \
        &client_secret={}&ll={},{}&v={}&radius={}&limit={}'\
        .format(CLIENT_ID, CLIENT_SECRET, nbd_lat, nbd_lng, VERSION, radius, LIMIT)
        #url = 'https://api.foursquare.com/v2/venues/explore?client_id={} &client_secret={}&ll={},{}&oauth_token={}&v={}&quer


        results = json.loads(requests.get(url).text)
        results = results['response']['groups'][0]['items']

        nearby = json_normalize(results) # Flattens JSON

        # Filtering the columns
        filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
        nearby = nearby.loc[:, filtered_columns]

        # Renaming the columns
        columns = ['Name', 'Category', 'Latitude', 'Longitude']
        nearby.columns = columns

        # Gets the categories
        nearby['Category'] = nearby.apply(get_category_type, axis=1)

        # Gets the data required
        for i, name in enumerate(nearby['Name']):
            s_list = nearby.loc[i, :].values.tolist()  # Converts the numpy array to a python list
            f_list = [nbd_name, nbd_lat, nbd_lng] + s_list
            explore_df_list.append(f_list)

    except Exception as e:
        pass
```

3 Methodology

3.1 Folium

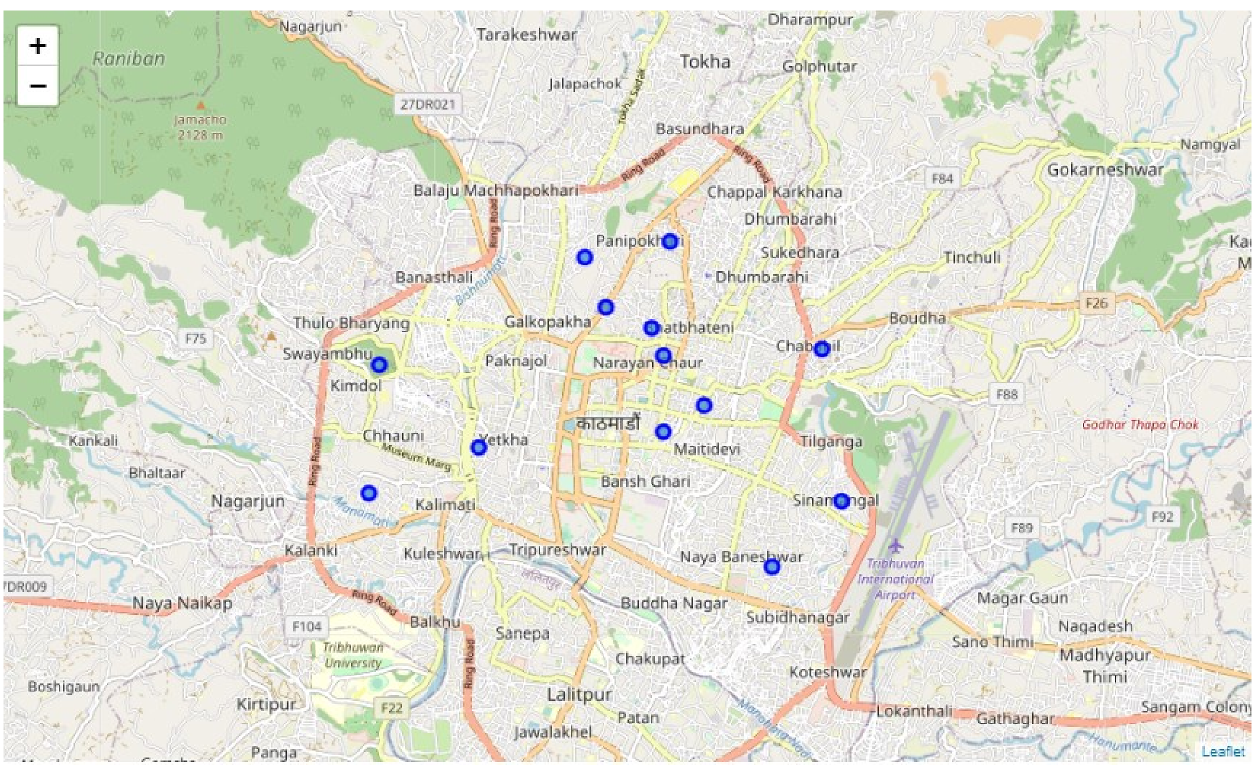
Folium builds on the data wrangling strengths of the Python ecosystem and the mapping strengths of the leaflet.js library. All cluster visualizations are done with help of Folium which in turn generates a Leaflet map made using OpenStreetMap technology.

```
# kathmandu latitude and longitude using Google search
ktm_lat = 27.7172
ktm_lng = 85.3240

# Creates map of Kathmandu using latitude and longitude values
map_ktm = folium.Map(location=[ktm_lat, ktm_lng], zoom_start=12)

# Add markers to map
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_ktm)

map_ktm
```



3.2 One Hot Encoding

One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. For the K-means Clustering Algorithm, all unique items under Venue Category are one-hot encoded.

```
# One hot encoding
ktm_onehot = pd.get_dummies(explore_df[['Venue Category']], prefix="", prefix_sep="")

# Add neighborhood column back to dataframe
ktm_onehot['Neighbourhood'] = explore_df['Neighbourhood']

# Move neighborhood column to the first column
fixed_columns = [ktm_onehot.columns[-1]] + ktm_onehot.columns[:-1].values.tolist()
ktm_onehot = ktm_onehot[fixed_columns]

ktm_onehot.head()
```

3.3 TOP 10 most common venues

Due to high variety in the venues, only the top 10 common venues are selected and a new DataFrame is made, which is used to train the K-means Clustering Algorithm.

	Neighbourhood	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	Baluwatar	Asian Restaurant	Café	Restaurant	French Restaurant	Diner	Coffee Shop	Breakfast Spot	Indian Restaurant	Bakery	Vietnamese Restaurant
2	Chabahil	Shopping Mall	Italian Restaurant	Historic Site	Multiplex	Spa	Restaurant	French Restaurant	Food	Fast Food Restaurant	Electronics Store
3	Dilli	Lake	Multiplex	Café	Outdoors & Recreation	Coffee Shop	Bus Station	Department Store	Dumpling Restaurant	Restaurant	Hotel
4	Gairidhara	Café	Hotel	Asian Restaurant	Restaurant	Vietnamese Restaurant	Himalayan Restaurant	Food	Hotel Bar	Ice Cream Shop	Jazz Club
5	Gyaneshwar	Asian Restaurant	American Restaurant	Historic Site	Multiplex	Fried Chicken Joint	Dumpling Restaurant	Lake	Café	Department Store	Outdoors & Recreation
6	Kalimati	Hotel	Tea Room	Indian Restaurant	Italian Restaurant	Bus Station	Vietnamese Restaurant	Department Store	Fried Chicken Joint	French Restaurant	Food
7	Lazimpat	Hotel	Restaurant	Asian Restaurant	Hostel	Café	Vietnamese Restaurant	Multiplex	French Restaurant	Garden	Himalayan Restaurant
8	Maru	Restaurant	Café	Asian Restaurant	Historic Site	Coffee Shop	Hotel	Electronics Store	Eastern European Restaurant	Plaza	Himalayan Restaurant
9	Naxal	Café	Hotel	Asian Restaurant	Restaurant	Multiplex	American Restaurant	Coffee Shop	Outdoors & Recreation	Himalayan Restaurant	Ice Cream Shop
10	Samakhushi	Hotel	Asian Restaurant	Restaurant	Vietnamese Restaurant	Pizza Place	Hostel	Hotel Bar	Indian Restaurant	Japanese Restaurant	Jazz Club
11	Sinamangal	Hotel	Airport Lounge	Airport Terminal	Airport	Indian Restaurant	Coffee Shop	Playground	Athletics & Sports	Airport Service	Diner

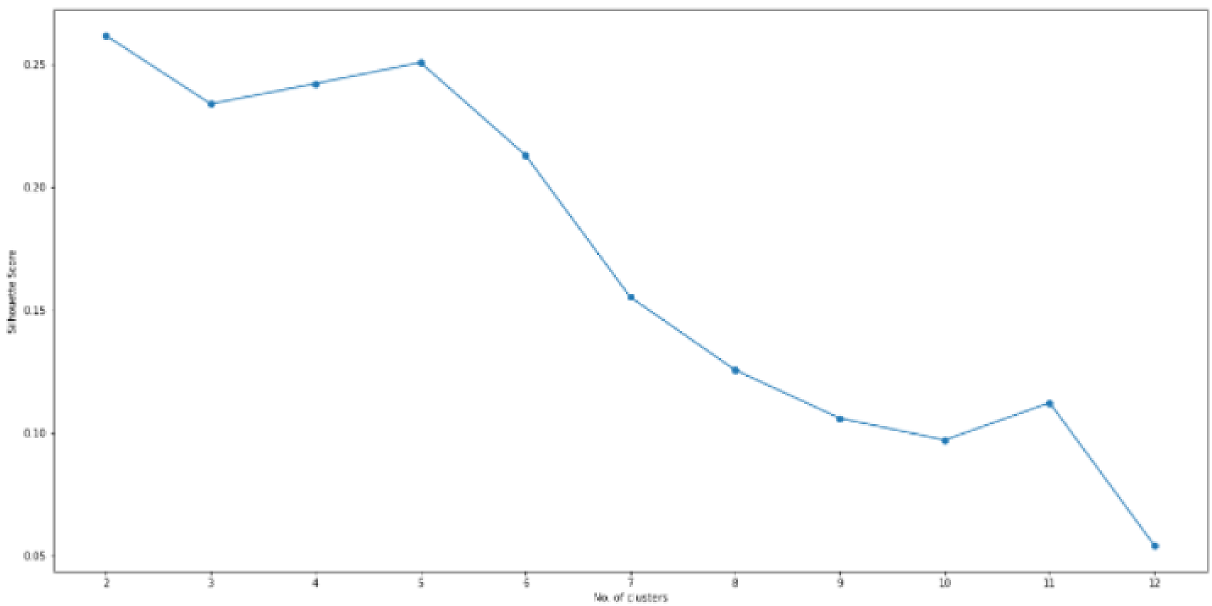
3.4 Optimal Number of clusters

Optimal number of clusters Silhouette Score is a measure of how similar an object is to its own cluster compared to other clusters (separation). The silhouette ranges from -1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

Based on the Silhouette Score of various clusters below 13, the optimal cluster size is determined.

```
import matplotlib.pyplot as plt
%matplotlib inline

def plot(x, y, xlabel, ylabel):
    plt.figure(figsize=(20,10))
    plt.plot(np.arange(2, x), y, 'o-')
    plt.xlabel(xlabel)
    plt.ylabel(ylabel)
    plt.xticks(np.arange(2, x))
    plt.show()
```




```
from sklearn.metrics import silhouette_samples, silhouette_score

indices = []
scores = []

for kclusters in range(2, max_range) :

    # Run k-means clustering
    kgc = ktm_grouped_clustering
    kmeans = KMeans(n_clusters = kclusters, init = 'k-means++', random_state = 0).fit_predict(kgc)

    # Gets the score for the clustering operation performed
    score = silhouette_score(kgc, kmeans)

    # Appending the index and score to the respective lists
    indices.append(kclusters)
    scores.append(score)
```

3.5 K-means Clustering

The venue data is then trained using K-means Clustering Algorithm to get the desired clusters to base the analysis on. K-means was chosen as the variables (Venue Categories) are huge, and in such situations K-means will be computationally faster than other clustering algorithms.

```
kclusters = opt

# Run k-means clustering
kgc = ktm_grouped_clustering
kmeans = KMeans(n_clusters = kclusters, init = 'k-means++', random_state = 0).fit(kgc)
```

4 Results

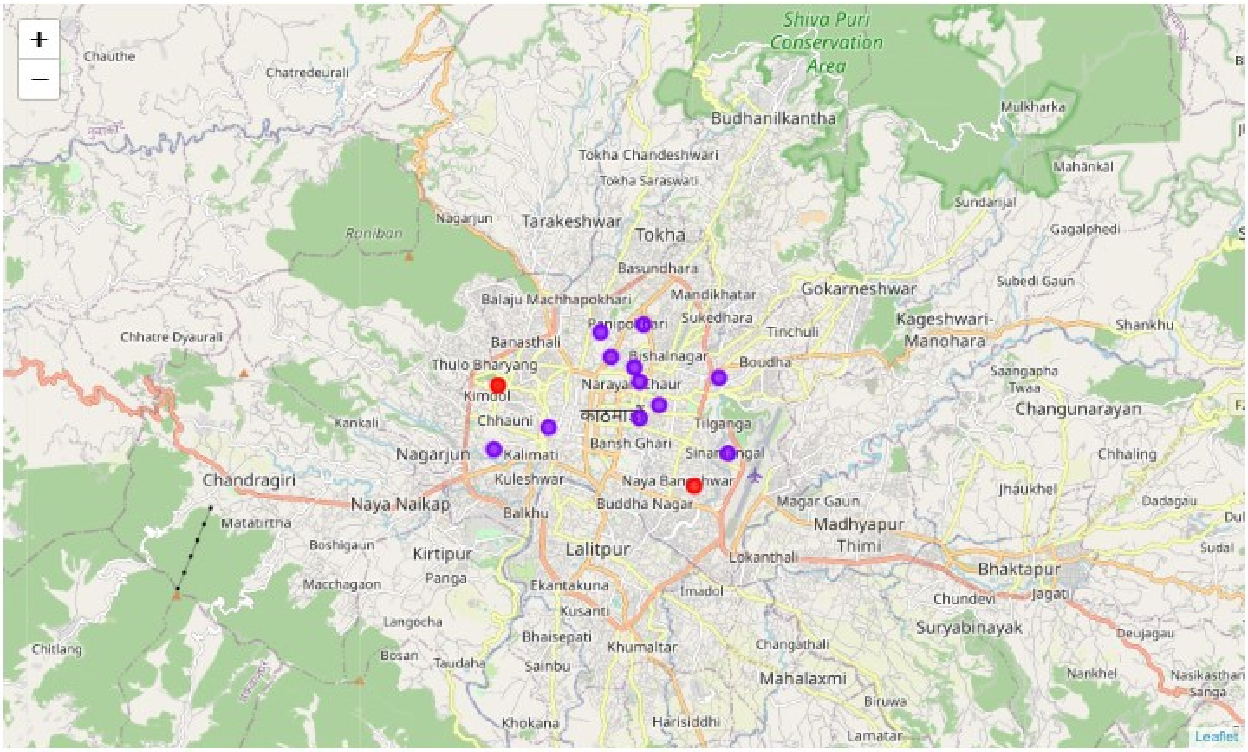
The neighborhoods are divided into n clusters where n is the number of clusters found using the optimal approach. The clustered neighborhoods are visualized using different colors so as to make them distinguishable.

```
# Create map
map_clusters = folium.Map(location=[ktm_lat, ktm_lng], zoom_start=12)

# Set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
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(ktm_merged['Latitude'], ktm_merged['Longitude'], ktm_merged['Neighbourhood'],
                                  ktm_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' (Cluster ' + str(cluster + 1) + ')', parse_html=True)
    map_clusters.add_child(
        folium.features.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7))

map_clusters
```



5 Discussion

After analyzing the various clusters produced by the Machine learning algorithm, a prime fit cluster number is used to solve the problem of finding a cluster with common venue.

	Neighbourhood	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	Baluwatar	Asian Restaurant	Café	Restaurant	French Restaurant	Diner	Coffee Shop	Breakfast Spot	Indian Restaurant	Bakery	Vietnamese Restaurant
2	Chabahil	Shopping Mall	Italian Restaurant	Historic Site	Multiplex	Spa	Restaurant	French Restaurant	Food	Fast Food Restaurant	Electronics Store
3	Dilli	Lake	Multiplex	Café	Outdoors & Recreation	Coffee Shop	Bus Station	Department Store	Dumpling Restaurant	Restaurant	Hotel
4	Gairidhara	Café	Hotel	Asian Restaurant	Restaurant	Vietnamese Restaurant	Himalayan Restaurant	Food	Hotel Bar	Ice Cream Shop	Jazz Club
5	Gyaneshwar	Asian Restaurant	American Restaurant	Historic Site	Multiplex	Fried Chicken Joint	Dumpling Restaurant	Lake	Café	Department Store	Outdoors & Recreation
6	Kalimati	Hotel	Tea Room	Indian Restaurant	Italian Restaurant	Bus Station	Vietnamese Restaurant	Department Store	Fried Chicken Joint	French Restaurant	Food
7	Lazimpat	Hotel	Restaurant	Asian Restaurant	Hostel	Café	Vietnamese Restaurant	Multiplex	French Restaurant	Garden	Himalayan Restaurant
8	Maru	Restaurant	Café	Asian Restaurant	Historic Site	Coffee Shop	Hotel	Electronics Store	Eastern European Restaurant	Plaza	Himalayan Restaurant
9	Naxal	Café	Hotel	Asian Restaurant	Restaurant	Multiplex	American Restaurant	Coffee Shop	Outdoors & Recreation	Himalayan Restaurant	Ice Cream Shop
10	Samakhushi	Hotel	Asian Restaurant	Restaurant	Vietnamese Restaurant	Pizza Place	Hostel	Hotel Bar	Indian Restaurant	Japanese Restaurant	Jazz Club
11	Sinamangal	Hotel	Airport Lounge	Airport Terminal	Airport	Indian Restaurant	Coffee Shop	Playground	Athletics & Sports	Airport Service	Diner

6 Conclusion:

As the middle class will grow at a rapid rate in the next upcoming years, opening food outlets catered for that section of the society will see a massive increase in footfall, which would lead to a further increase in business.