**Exploring Restaurants in Los Angeles**

1. **Introduction**

Restaurant business is one of the most sought-after businesses in Los Angeles (LA). LA, in particular, is an amazing place to dine out owing to its wide variety of international cuisines, the quality of food and the services offered. Given the high demand for quality, a restaurant needs to get several factors right in order to survive here. In this brief report, I tried to investigate some of these factors by exploring the restaurants based in the various neighborhoods of Los Angeles (LA). I have tried to keep the discussion general when it comes to the kinds of restaurant and their features and rather focus on the global features like density of restaurants in a region, their frequency and so on.

The *business* direction that I have in mind is actually two-fold. One is directed towards individuals looking for places to eat or even people looking for places to rent based on restaurant types or frequency. The other direction is for corporations/individuals looking to open a new restaurant.

* 1. **Specific Plan**

I plan to use 2 ideas for segregating/clustering the neighborhoods based on their restaurant venues. One idea is to find the density of restaurants in each neighborhood. This will help anyone trying to open a new restaurant by either avoiding overcrowded areas or alternately could help finding popular venues to avoid competition.

The next idea to explore is the kind of restaurant. Here, one can again use clustering but now based on the frequency of occurrence of each restaurant in a Neighborhood just like the one done in the New York data set. Once we find the relevant cluster here, we can then look for its intersection with the clusters found above to fine tune the relevant neighborhood where one wants to open his/her restaurant.

1. **Data collection, cleaning and wrangling**

I acquired the location data from this [website](https://usc.data.socrata.com/dataset/Los-Angeles-Neighborhood-Map/r8qd-yxsr). There is an API endpoint [link](https://usc.data.socrata.com/resource/9utn-waje.json) for the json file in this website as well.

**2.1 Data cleaning**

Data cleaning was pretty straightforward. Lot of redundant columns were dropped. Some columns like the ‘type’ or the geometry of the boundaries (‘the\_geom’) of a neighborhood were also removed. One major problem which I came to realize after plotting the folium map is that latitude and longitude values are swapped with each other. Once that is fixed along with renaming of some columns, the final table looked as shown below:

Table 1:

**A screenshot of a cell phone

Description automatically generated**

The data consisted of 272 unique neighborhoods. To make the analysis slightly easier, I used a distance function from the LA central coordinates to reduce the number of rows to 199. This is just so that I can reduce the number of calls to the foursquare location app. The final map with all the neighborhoods is shown below –

**A close up of a map

Description automatically generated**

The centroid coordinates of LA was found by using the geocoder library in the geopy module of python. Then to explore venues in a given neighborhood, we use Foursquare which was introduced earlier in the course.

**2.2** **Data Wrangling**

Now, we need to transform the data to bring it into a useful form for analysis. Nearby venues for each of the neighborhoods were collected using Foursquare and only the restaurant venues were stored for further analysis.

Table 2:

**A screenshot of a cell phone

Description automatically generated**

As mentioned earlier, we will be analyzing restaurant location to predict areas for new restaurant set ups. First part is basically exploring the restaurants and computing the densities for each neighborhood. Then we can form clusters with high, low and medium densities. This is a very good starting point for stakeholders as it gives them an idea of which regions are already saturated with restaurants and which are brimming with opportunity.

Then we again use the same clustering technique to cluster neighborhoods now based on their frequency in the neighborhood. This is similar to what was done in the New York dataset. Then one can use both these kinds of clusters to now find densities of particular kind of restaurants in a region.

A further exploration could also be looking at the distance of a particular restaurant from neighborhood center and cluster regions based on that. We don't explore this here. But one can easily do a follow up in this direction using my analysis.

1. **Analysis**

Similar to the New York dataset, we create one hot vectors for each restaurant venue. For the densities part, we group by neighborhood and find the number of restaurants in each neighborhood by using the sqmi column in Table 1 above. Then we evoke the KMeans() algorithm. First we use the elbow technique to find the optimal k for the KMeans(). This was found to be 5. Hence, we use n\_clusters = 5 for our analysis. The cluster distribution was obtained as follows :

A screenshot of a cell phone

Description automatically generated

We can clearly see the segregation into high, medium and low densities. To, simplify our analysis, we rename the labels for the clusters as follows :

{1 & 3 : high, 0 & 4 : medium, 2 : low}

Next, we find the most common restaurant venues for each neighborhood by computing their frequency of occurrence. This would give us an idea about the most sought-after cuisine in each neighborhood. Then we use this to run the KMeans() algorithm again. The elbow technique was not significantly helpful as it was throwing different values everytime we ran it. So we just chose an average of those different values which turned out to be close to 10.

The segmentation resulted in the following map : A picture containing text, map

Description automatically generated

The clusters are color coded above. Just as an example, I showed a popup in the above figure for a neighborhood in cluster 1 called Arcadia marked in purple. All the purple marked neighborhoods are in cluster 1.

1. **Visualization of Clusters**

Finally, let’s see how this cluster information can be combined to evoke some interesting results. First, we combine both the cluster labels into one table bu using the join function using the ‘Neighborhood’ column as the key.

The following shows a portion of the table -A screenshot of a cell phone

Description automatically generated