Recommendation of a Manhattan Neighborhood for a Restaurant Location

Eric Luis Barroso Cavalcante

1. Introduction

The business problem that is addressed in this project is the choosing of a Manhattan neighborhood to install a restaurant of a chosen cuisine or type. Some variables influence the location of a restaurant in an area, such as: population density, median household income, median rent and number of restaurants already established. It can be argued that as the numbers of competitor restaurants increases it becomes more and more advisable to choose another area to locate a restaurant. Thus, it is the only mentioned variable that its increase reduces the chance of success of a new restaurant. As the other variables (population density, median household income, median rent) for a specific neighborhood increase the chance of success of a new restaurant in this area also increases.

The specific cuisine or type of restaurant is chosen by the stakeholder. Based on the neighborhoods' population density, household income, median rent and rate of neighborhood area by the number of restaurants of the type specified, the analysis provides a list of the neighborhoods from the most to the least promising one to locate the type of restaurant picked by the stakeholder.

2. Data

It has been utilized in this project three sources of data:

1) For the necessary latitude and longitude of the Manhattan neighborhoods it has been used the file newyork_data.json provided in the Lab "Segmenting and Clustering Neighborhoods in New York City" (week 3). This file contains the geographical coordinates for each neighborhood of the five boroughs of New York City.

2) The data of population, area, median household income and median rent for each Manhattan neighborhoodIt has been obtained from the address http://www.city-data.com/indexes/neighborhoods/NY/1/ by means of web scraping.

3) The names and categories of restaurants have been acquired using the "explore" endpoint of the Foursquare API. It is known that the Foursquare API provides the types and names of venues by means of defining a latitude/longitude and a radius of search.

The radius of a neighborhood, which is considered as the radius of search for the Foursquare API, has been obtained by means of its area using the circle area formula. The latitude/longitude given to the Foursquare API for the retrievel of the types and names of venues in a neighborhood is the one found in the mentioned file newyork_data.json.

The steps followed to acquire and structure the data are:

1) Create a file handler for the file newyork_data.json, download the data of the file which consists of a dictionary object and assign the value of the key "features" to the variable "neighborhoods_data". This value is a list object;

2) Create the DataFrame object "neighborhoods" which has "Borough", "Neighborhood", "Latitude", "Longitude" as columns;

3) Create a new Dataframe object picking only the Manhattan borough;

4) Discard sub-neighborhoods of the given Manhattan neighborhoods;

5) Rename properly some of the Manhattan neighborhoods to fit the webpages of the respective neighborhoods contained in the city-data webpage;

6) Create a list object with the column "Neighborhood" of the DataFrame object "manhattan_data";

7) Copy the previous list and rename some of the Manhattan neighborhoods to fit the content of the html document of the respective neighborhoods contained in the city-data webpage;

8) Create a list of urls with the names of the neighborhoods present in the list object "list_manhattan_neigh";

9) With the possession of the list object "url_list" and the list object "list_manhattan_neigh_beta", create list objects for the area, population, median household income, median rent of the Manhattan neighborhoods making use of python regular expressions to web scrape the html documents of each neighborhood found in the city-data webpage;

10) Add the columns "Approx_Radius_Meters", "Population/Km2", "Median_Income_Dollars" and "Median_Rent_Dollars" to the existing DataFrame object "manhattan_data";

11) Define a function (named “getOverVenues”) that creates a DataFrame object containing the venues names, geographical coordinates and category type of all neighborhoods making use of the Foursquare API;

12) Use the function “getOverVenues” to create the DataFrame object "manhattan_venues" that aggregates the venues' names, geographical coordinates and category type of all neighborhoods making use of the Foursquare API.

3. Methodology

This project provides the stakeholder the possibility of choosing a specific type of restaurant (for instance: French, BBQ Joint, Steakhouse, Vietnamese) and with the possession of the quantity of the chosen type of restaurant, the population density, the median household income and the median rent for each Manhattan neighborhood the system provides a list ranking the neighborhoods from the most to the least promising one to install the specified type of restaurant.

In the previous item we have collected the required data and gathered them in two DataFrame objects:

1) "manhattan_data": contains as columns the neighborhoods, their latitudes, their longitudes, their approximate radii, their population density, their median household income and their median rent (shape = (29, 7));

2) "manhattan_venues": contains as columns the neighborhoods, the name of the venues, their latitudes, their longitudes and their category (shape = (2502, 5)).

It is visualized the Manhattan map and the markers of the centroids of each Manhattan neighborhood with the use of the folium library.

The analytical steps required for producing results are:

1) Set the column "Neighborhood" of the DataFrame object "manhattan_data" as index and sort (in alphabetical order) the index of this new DataFrame object called "manhatan_data_ordered" (shape = (29, 6));

2) Create a Series object (named "total_venues_per_neighborhood") to get the total number of venues of each neighborhood using the DataFrame object "manhattan_venues" and applying to it the groupby and count methods (shape = (29, 1));

3) Take the DataFrame object "manhattan_venues" and create a new one (named "manhattan_onehot") containing as columns the neighborhoods and the category of the venues as dummy features (shape = (2502, 302));

4) Create a new DataFrame object (named "g_manhattan_onehot") by grouping the column "Neighborhood" of the DataFrame object "manhattan_onehot" and taking the sum of the other columns which represent the categories of the venues (shape = (29, 301));

5) Create a list object (named "food_columns") of only the categories of venues of the DataFrame object "g_manhattan_onehot" that represent restaurants (len = 84);

6) Create a new DataFrame object (named "g1_manhattan_onehot") with only the columns of the DataFrame object "g_manhattan_onehot" that represent restaurants (shape = (29, 84));

7) The stakeholder is required to type the kind of restaurant he or she desires to install (suppose the stakeholder typed "Italian Restaurant");

8) Create a Series object (named "food_places") to get the total number of "Italian Restaurant" of each neighborhood using the DataFrame object "g1_manhattan_onehot" (shape = (29, 1));

9) Create a copy of the DataFrame object "manhattan_data_ordered", drop the columns "Latitude" and "Longitude", add the column "Km2/Food_Places" using the area of the neighborhoods and the Series object "food_place" and name the resulting DataFrame object as "df_ordered" (shape = (29, 4));

10) Perform a scaling of the columns of the DataFrame object "df_ordered" which represents a ranking of the values of each column ("Population/Km2", "Median_Income_Dollars", "Median_Rent_Dollars" and "Km2/Food_Places");

11) Create a ndarray object ("sum_X") where each element represents the sum of the scaled values of "Population/Km2", "Median_Income_Dollars", "Median_Rent_Dollars" and "Km2/Food_Places" for a neighborhood, conveying, therefore, a score for each neighborhood.

The larger the element of "sum_X" the better the chance of success of the "type_restaurant" chosen by the stakeholder in that neighborhood.

12) Add the ndarray object "sum_X" as the column "Score" to the DataFrame object "df_ordered" and also add the list object ""foods_list" as the column "Food_Places";

13) Sort the DataFrame object "df_ordered" by the column "Score" from the largest to the smallest and take the resulting DataFrame object and assign it to the variable "df".

4. Results and Discussion

For the kind of restaurant chosen by the stakeholder ("Italian Restaurant"), our analysis shows that the best suitable neighborhood to install a restaurant of the cuisine is Wal Street and the worst is East Harlem. Of the six most adequate neighborhoods to place a Italian Restaurant four of them do not have a unique restaurant of this kind.

The fact that Wall Street is the first and Chinatown is the second best recommended neighborhood, despite the zero number of Italian restaurants in the latter, is related to the other three variables (population density, median household income and median rent), which are all higher in Wall Street. The median household income of Wal Street is 83% larger than Chinatown's and the median rent in Wall Street is 174% higher than Chinatown's.

The neighborhoods of Turtle Bay, West Village, Noho, Upper East Side and Upper West Side already have a considerable number of Italian restaurants, however it is still mmuch more recommended to locate a restaurant of the kind in these areas than in the East Village, Washington Heights and Inwood due to economic factors.

Another discussion that can be made with the result shown in the DataFrame object "df" is related to the Greenwitch Village and Little Italy neighborhoods. Greenwitch Village has only one less Italian restaurant than Little Italy, nevertheless it is much better positioned than Little Italy. This can be attributed to the median household income and to the median rent of the Greenwitch Village, which are far superior than those of Little Italy. On the other hand, in this case we detect a limitation of the present project since it does not consider as variable, in its current version, the number of tourists that visits a neighborhood. This limitation it is certainly a point to be addressed in a next version of the project.

5. Conclusion

The main purpose of this project has been to rank the Manhattan neighborhoods best suited to have a certain type of restaurant in its area. The variables taken into account to perform this ranking are the density of population, median household income, median

rent and the rate of neighborhood area by the number of restaurants of the type specified. The larger are these variables for a neighborhood the greater is its rank.

One interesting feature of the project is that allows the stakeholder to choose from a variety of 84 kinds of restaurants, i.e., the same number of possibilites contained in the Foursquare API.

One limitation detected by the current version of the project is that it does not take into account as a ranking variable the number of tourists that visit a neighborhood. We will address this limitation in the following version of the project.