Battle of Neighborhoods

by Alberto Iriberri (Dec 2019)

The project (Jupyter Notebook)

- You can find the Jupyter Notebook that I used to ellaborate the data and reach the final conclussions here:
- Los Angeles New restaurant Project
- and also you may find interesting the Notebook I programmed to specifically extract Los Angeles data from the Wikipedia:
- Los Angeles Wikipedia Extractor

1.- Introduction (Business requirements)

- As we discussed in the orevious week project, the main goal of this project is to find a good opportunity in Los Angeles, CA for opening a new restaurant.
- We need to take into account that Los Angeles is multi-cultural city (probably one of the most diverse in the world) and opening a new restaurant may be a financial risk if you don't measure all the different variables.
- Our financial partners don't need to start a restaurant for an specific food type. They have the money to start a new business and they think a restaurant in LA, where they have their headquarters, could be a good choice.
- Based on that, we need to accomplish two requirements and use available free internet data:
- Select the **food type** for our restaurant
- Choose the **neighborhood** where we will place our restaurant

2.- Data Requirements

- We will retrieve our data from two different sources:
- **Wikipedia**: list of Los Angeles neighborhoods & districts
- **Foursquare**: list of restaurants for each of the neighborhoods
- *We will also use Nominatim API to get geographic coordinates for each place*

3.- Data understanding

- List of neighborhoods:
  - We need to clean or remove those districts that seem to have incorrect information in Wikipedia. After analyzing each of them, it is not relevant to remove part of them as they are only small zones. Note that even we remove a small district, their restaurants will be in the venues list because they are enough close to another main district or neighborhood.
- List of venues (restaurants):
  - Though we specified **Restaurant** as the query for Foursquare API it returns some venues that are not really restaurants. We have to detect these cases and axclude them from the final list.
- Coordinates (GEO):
  - Some of the places are not geolocated. As we did with incorrect neighborhoods, we remove these places.

4.- Data preparation

- **Nominatin API problems**
- As we mentioned above, we are using Nominating to get the GEO coords for all the places we will analyze. When we iterate the list of places and invoke Nominnating API, we get a **Service unavailable** error.
- We dealt with this problem in the next way:
- Set a delay of 1 second between each invocation
- Save the results we get (for every coordinates we get) so that we don't need to repeat the process for the same items
- Data filtering and transformation will be done always on the origina data, and not on data previously transformed or manipulated. This will allow us to perform as many tests as we need without the need of retrieving the original data again.
- The original data is contained in two different csv files:
  - Neighborhoods .csv file
  - Venues .csv file

5.- Methodology I

- o better understand the field and the problem we have to solve, we need to use visual tools that help us to make decisions.
- First we will classify the data we have to understand which are the LA people preferences. For instance, probably we will not find any Scottish restaurant (or very few) but a bunch of Mexican or Latin restaurants.
- Once we know the preferences we will choose a restaurant of this type based on these:
- **If we choose a type of food that LA people doesn't like now, we will have no competitors, but also we will have no customers**
- **If we choose a type of food that LA people like now but we place it where there are a lot of competitors with the same food type, we will not have a lot of customers**
- **We need to place a restaurant specialized in a food type that LA people likes in a neighborhood with few competitors. Does this exist?**

5.- Methodology II

- We need two tools for our purpose:
  - **Ranking tools**: Python Pandas and Numpy will be enough
  - **Clustering tools**: As we are clustering based on geographic coordinates it is recommendable to use DBSCAN algorithm
- So, the steps to reach our conclusions will be:
  - Rank all the food types in LA to know what customers like
  - Select our top n food types to make the clustering analysis with
  - Select all the existing restaurant for the selected categories and place it in the map
  - Use clustering tools to divide the map into different numbered clusters
  - Classify food types / clusters using a weighted matrix
  - Analyze with maps and the matrix which are our options
  - Make a decision/recommendation

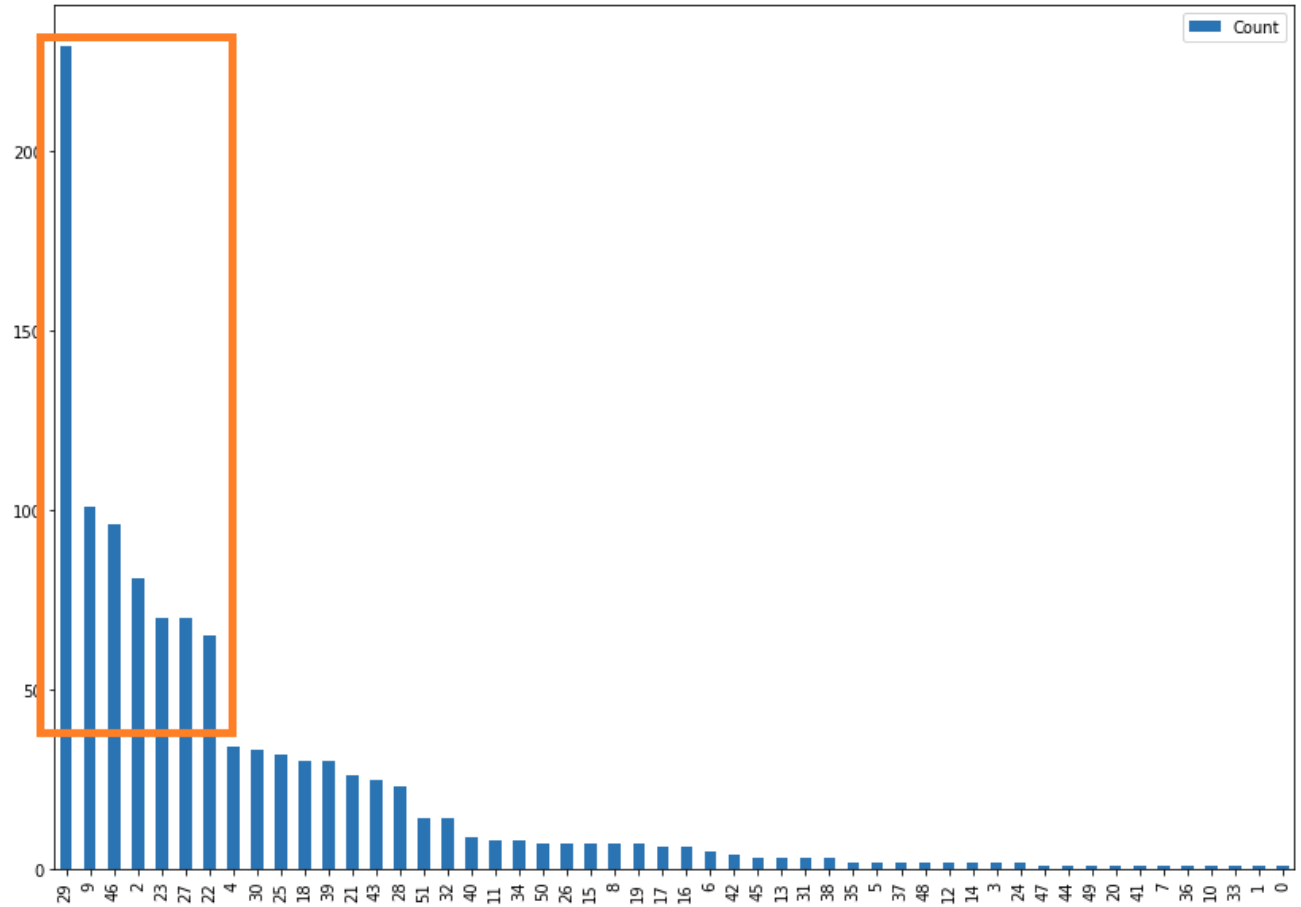
6.- Modeling

- **Important:** Take into account that the details of the data modeling/analysis are contained in the Notebook. Here I will only place the images to easily explain the process

6.1 The restaurants dataframe

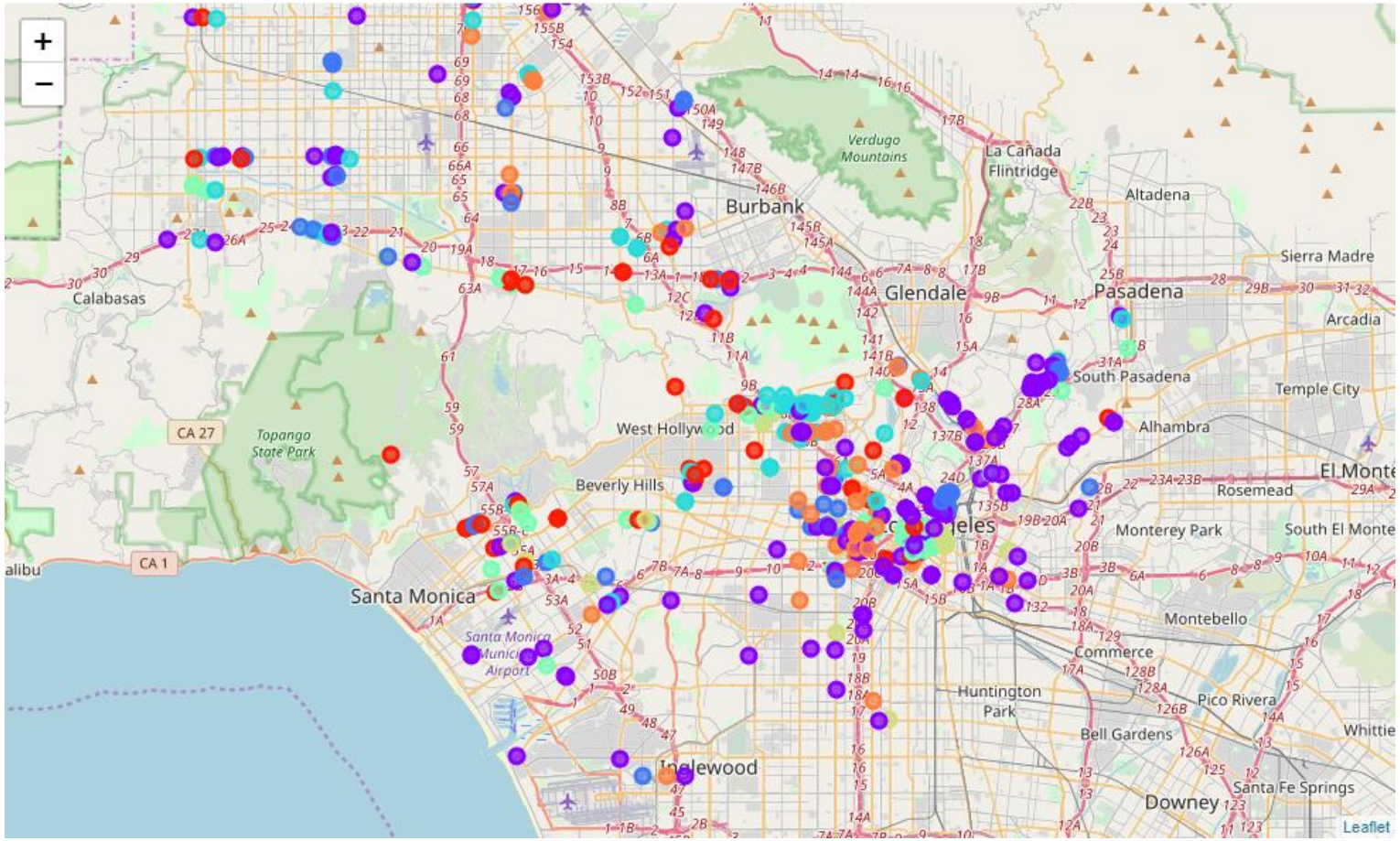
	Neighborhood	Lat	Lon	Type
0	Angelino Heights	34.078103	-118.261575	French Restaurant
1	Angelino Heights	34.066298	-118.253917	Sushi Restaurant
2	Angelino Heights	34.060190	-118.257793	Mexican Restaurant
3	Angelino Heights	34.078205	-118.261114	Mexican Restaurant
6	Angelino Heights	34.077936	-118.259632	Mexican Restaurant
...	...	...	...	...
1946	Woodland Hills	34.168249	-118.602936	Mediterranean Restaurant
1947	Woodland Hills	34.168300	-118.615780	Mexican Restaurant
1948	Woodland Hills	34.168103	-118.600244	Thai Restaurant
1949	Woodland Hills	34.168107	-118.600664	Sushi Restaurant
1950	Woodland Hills	34.166969	-118.592416	Mexican Restaurant

6.2 Choosing the top n food types

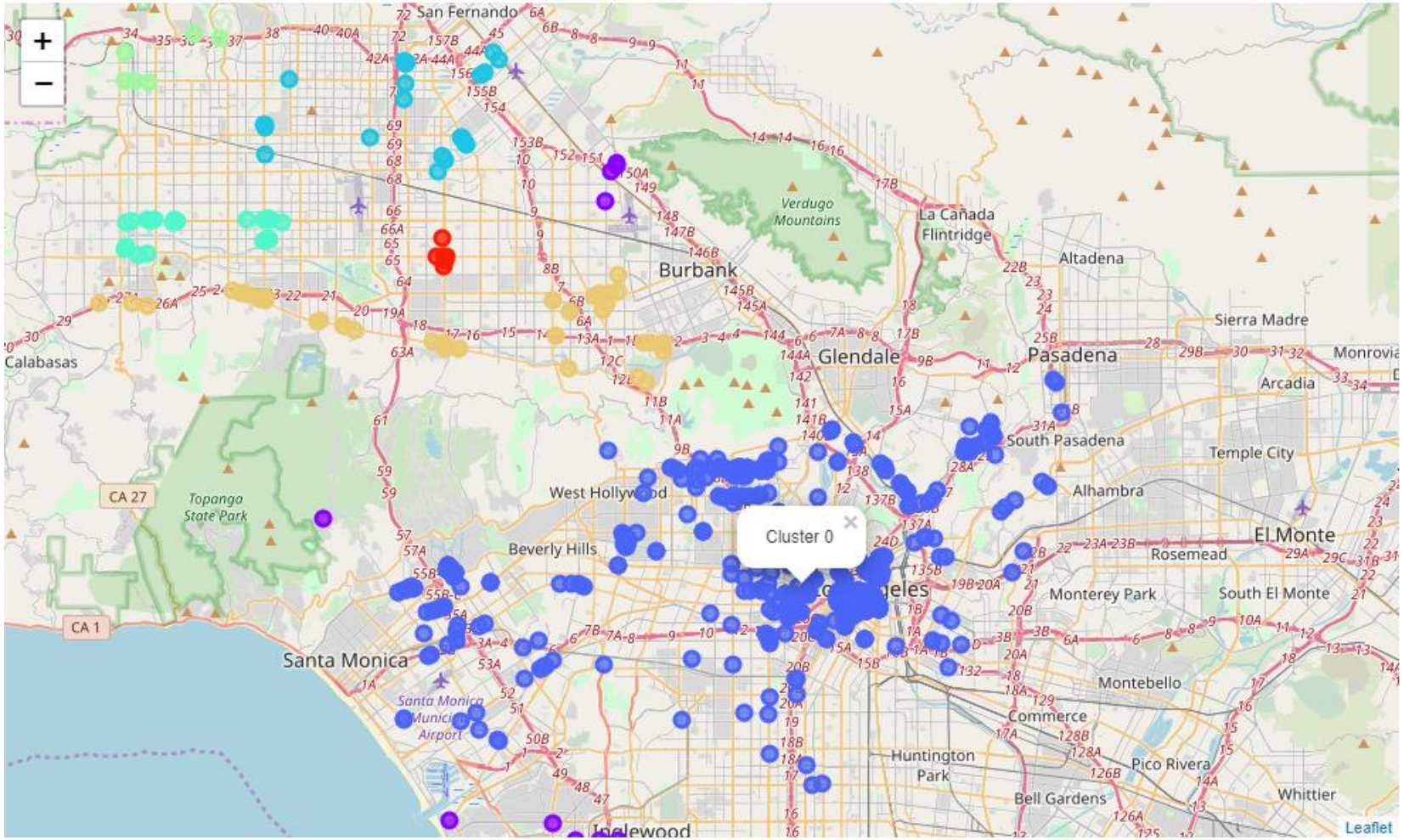


	Type
0	Mexican Restaurant
1	Chinese Restaurant
2	Thai Restaurant
3	American Restaurant
4	Japanese Restaurant
5	Latin American Restaurant
6	Italian Restaurant

6.3 Map distribution of top 7 food types restaurants



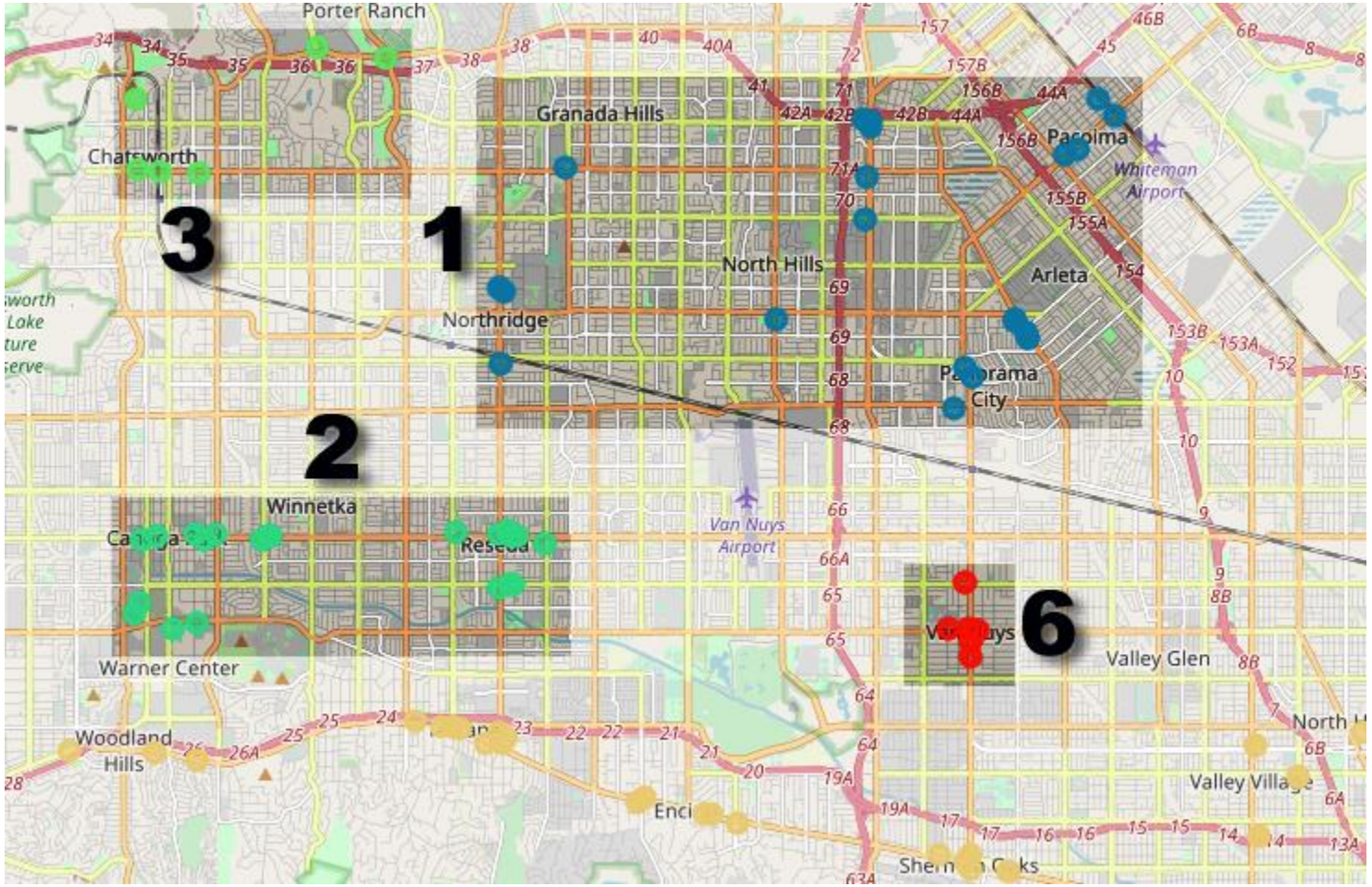
6.4 Clustering places with DBSCAN



7 Type of Food x Cluster Matrix

	Type	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6
0	American Restaurant	0.000000	0.987179	0.564103	0.528846	0.000000	0.000000
1	Chinese Restaurant	0.982906	0.825641	1.179487	1.474359	0.688034	0.589744
2	Italian Restaurant	2.991453	1.256410	0.717949	0.598291	0.000000	0.717949
3	Japanese Restaurant	0.000000	0.717949	0.000000	0.769231	0.478632	0.000000
4	Latin American Restaurant	0.356125	0.000000	0.000000	1.923077	0.000000	0.256410
5	Mexican Restaurant	0.878443	0.830128	0.948718	1.423077	0.632479	1.897436
6	Thai Restaurant	0.961538	1.076923	0.923077	0.865385	1.076923	0.000000

8 Final Analysis



	Cluster	Food type	Weight
0	Cluster 1	Italian Restaurant	2.991453
1	Cluster 2	Italian Restaurant	1.256410
2	Cluster 3	Chinese Restaurant	1.179487
3	Cluster 4	Latin American Restaurant	1.923077
4	Cluster 5	Thai Restaurant	1.076923
5	Cluster 6	Mexican Restaurant	1.897436

Conclusion

- **The weighted matrix shows that Cluster 1 and Cluster 2 are very good places to set an Italian Restaurant**

Restaurant

“Regression to Pasta”