Neighborhoods in Chicago by Cuisine Type

November 4, 2019

1 Introduction

People of varying backgrounds, economic statuses, and tastes will shape the restaurant options located near them. So, it would seem likely that the different neighborhoods in a city would be composed of different sorts of food establishments and could therefore be detected by machine learning using restaurant data. This project investigates how closely established city neighborhoods align with clusters determined by restaurants and what sorts of restaurants are present in the various clusters. It also compares the restaurant compositions of different cities.

2 Data

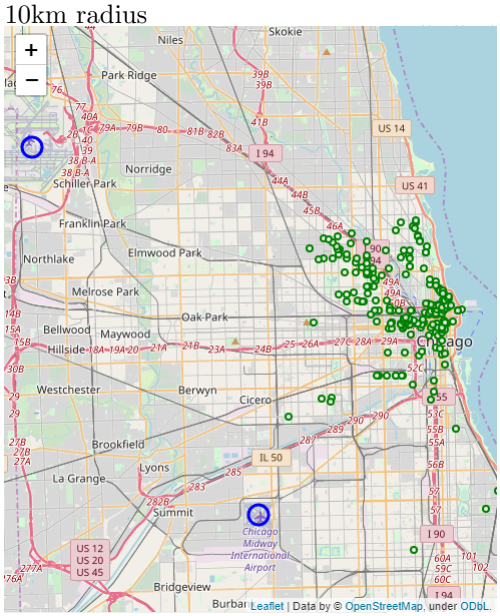
Foursquare's API has a search feature that allows the user to search for venues in a city which belong to certain categories. Among these categories are Food and Bars. Each of these categories is subdivided into smaller subcategories such as Afghan Restaurant, Burger Joint, etc. Given a city, we can consider its restaurants along with the corresponding list of categories each belongs to. Using these data, we can hope to understand the "culinary neighborhoods" of a city.

3 Methodology

3.1 Initial Data Collection

The initial data were collected using Foursquare API's explore feature with "Chicago" as the target city and "Food" and "Bars" as the target categories to search for. The radius was set to 10km. By paging through the results of multiple queries, a list of 248 unique restaurants and bars in Chicago were compiled, along with the latitude, longitude, and category of each. The distribution of these restaurants is depicted in Figure 1. Now compare an established map of Chicago neighborhoods in Figure 2 to our plot of restaurants. In each figure, the O'Hare and Midway airports have been marked with blue circles to make

Figure 1: All restaurants in Chicago obtained from venues-explore queries. 10km radius



comparison of the two maps simpler. The restaurant data lacks samples from many areas of interest, so we require a method of obtaining additional samples.

The Foursquare API's limitations presents issues for obtaining large quantities of samples for two reasons:

- The venues-explore feature was used to generate the initial list, so obviously does not yield enough results, even when the radius is extremely large. See Figure 3.
- The venues-search feature has a hard limit of 50 returned results with no ability to page through to obtain more, as venues-explore.

3.2 Expanding Dataset with DBScan

In order to collect more venues, we will use the two API calls venues-search and venues-explore on smaller regions that have disproportionately small numbers of venues. These regions must be automatically identified somehow. The most obvious approach is to grid out the city and query within gridboxes with few venues. However, there is an alternative approach that is both easy to code and can populate the dataset without any grid-like artifacts: DBScan.

DBScan is a clustering algorithm that classifies points as one of: core, boundary, and outlier. Core points are near many other points, where "many" and "near" are hyperparameters. Boundary points are near core points, and outliers are not near any other points. It then uses proximity of core and boundary

Figure 2: Recognized neighborhoods in 1993

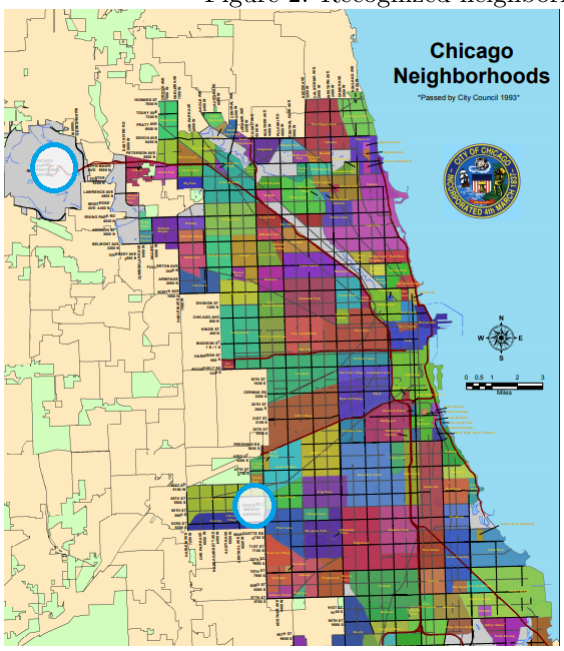


Figure 3: All restaurants in Chicago obtained from venues-explore queries. 50km radius

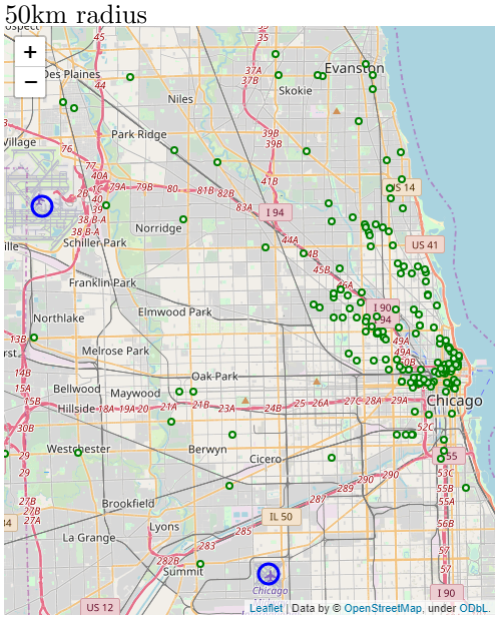
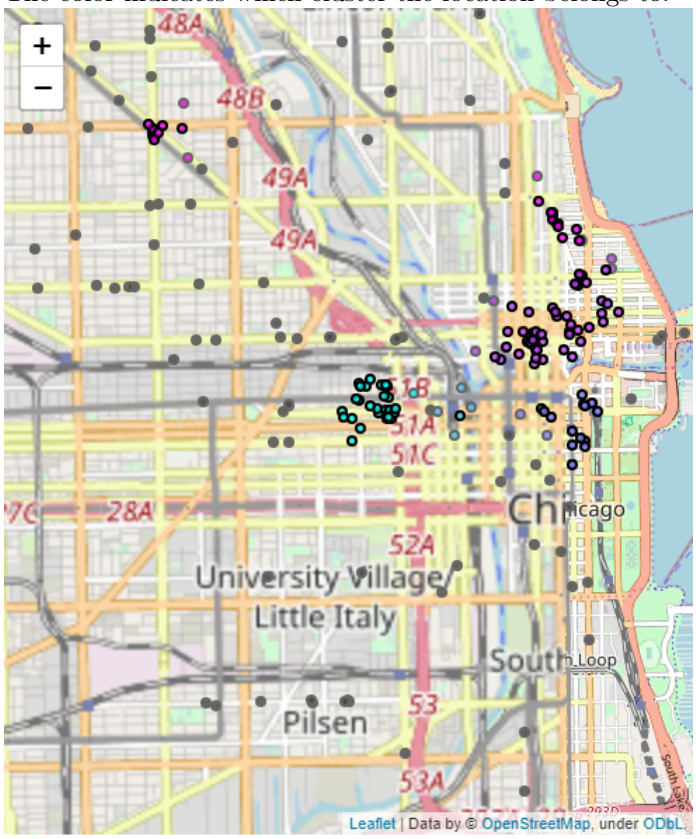


Figure 4: Results of DBScan. Outliers are gray and boundary points are lighter. The color indicates which cluster the location belongs to.



points to cluster the data. Of most interest for our purposes is the fact that the algorithm automatically identifies points which are located in regions with low data density relative to some radius, which are natural candidates for a new set of queries.

The initial dataset of only 248 locations was expanded to over 7000 locations in the Chicago area, and Figure 5 shows the distribution more accurately matches the coverage of Figure 2. The process by which this expansion took place is as follows: All outliers and boundary points of the DBScan algorithm are collected, and a search about each of them with a search radius r is conducted. Any new venues are added to the dataset, and then a new DBScan is conducted. This process was repeated until performance greatly slowed and the coverage of the dataset was acceptable. During each step of the procedure, the search radius r was increased if no new locations were found at a given spot, and the neighborhood radius of the DBScan algorithm was decreased if insufficiently many boundary and outliers were found.

Figure 5: The final dataset locations

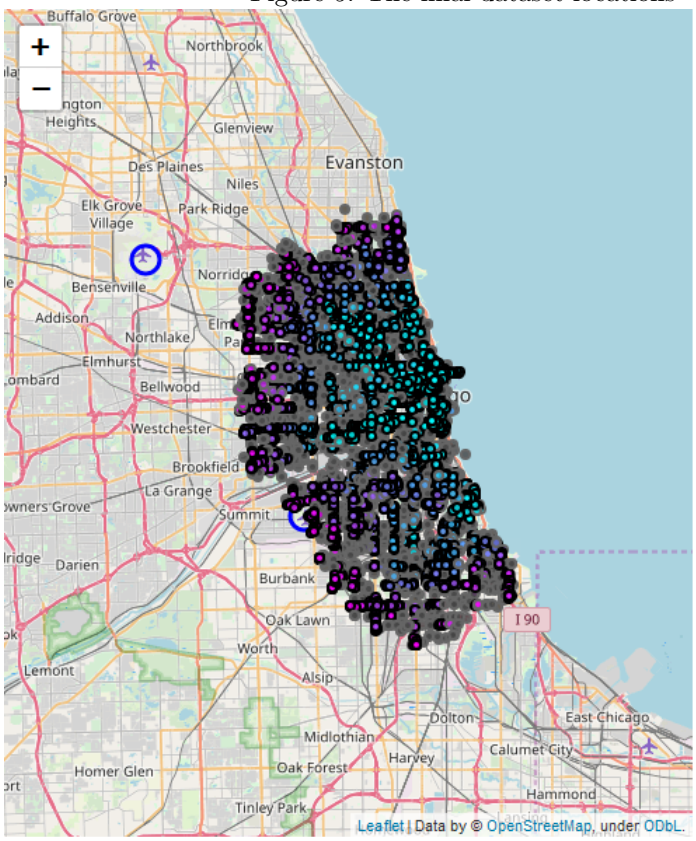
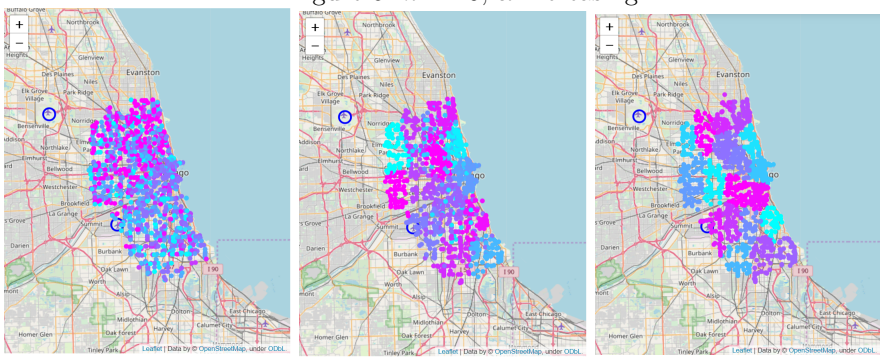


Figure 6: $k = 20$, α increasing



3.3 Preliminary Analysis

To group the venues by location and cuisine type, the categorical variable indicating the type of food or bar was onehot encoded and the latitude/longitude coordinates normalized. We now view the resulting row as a geographical data point $(lat, long)$ along with an associated "feature" data point $(iscat_1, iscat_2, \ldots, iscat_N)$. We weight the data as $\alpha(lag, long) + (1 - \alpha)(iscat_1, \ldots, iscat_N)$ for $\alpha \in [0, 1]$. This weighting allows flexbility between viewing the data as a set of locations and as a set of feature vectors. The latitude and longitude features a normalized prior to weighting, but the feature vectors' columns are not, due to lengthy processing time.

After the normalization and weighting, the scikit-learn k-means algorithm is applied for various choices of $k \in \{20, 40, 80\}$ and $\alpha \in \{0.2, 0.5, 0.8\}$. Figures 6, 7, and 8 depict the results, with only 20% of the points plotted for viewability.

In order to choose an appropriate value for α, we can compute two quantities: something representing the uniformity of restaurants types in a cluster and another representing how geographically dispersed a cluster is. The former can be measured by consider the proportions $p_1, \ldots, p_R$ of each of the classes of restaurants in the cluster and computing $\sum_r p_r^2$. This is the Herfindahl index from economics, and it measures concentrations of firms in a market but can be adapted here to qunatify concentration. It gives a number between 0 and 1 that is larger if there are a few large proportions. We can measure geographical dispersion by simply computing the variance of the square distance from the centroid of the cluster. Making these calculations for various values of α yields the left image in Figure 9.

In order to make a choice that balances having low geographical dispersion and high venue concentration, we can seek α that maximizes the ratio

$$\frac{\text{venue concentraion}}{\text{square distance variance}}$$

and we find a value of $\alpha = 0.4$ might be a good choice.

Figure 7: $k = 80$, α increasing

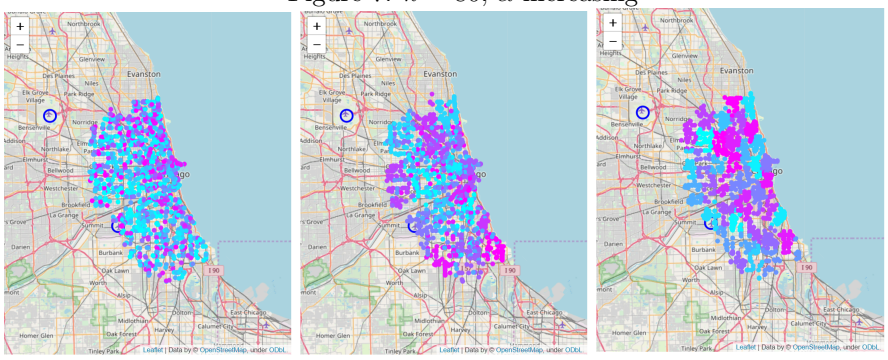


Figure 8: $k = 80$, α increasing

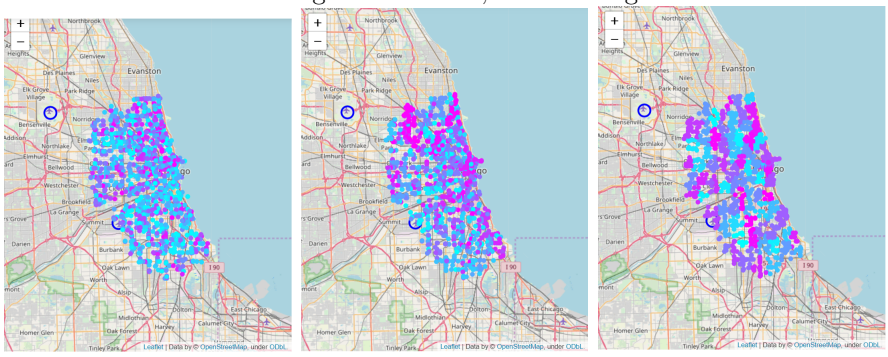


Figure 9: Graphs informing the choice of α

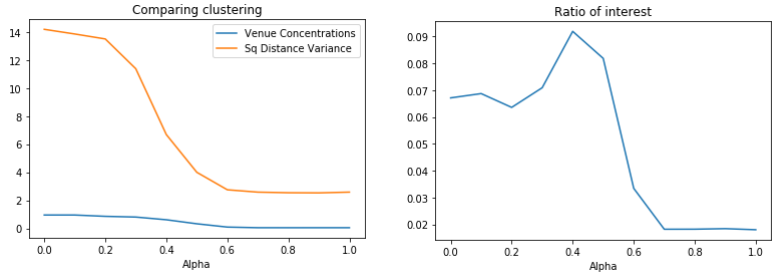
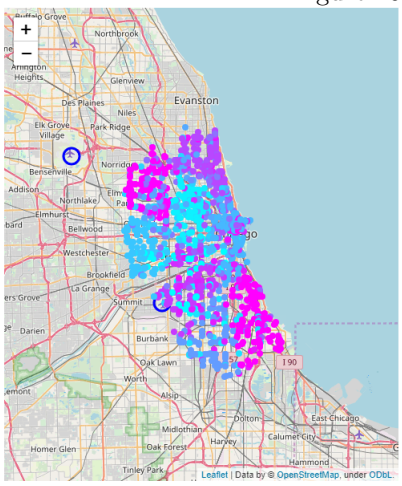


Figure 10: $\alpha = 0.4$ clustering



Group 0: Mexican Restaurant
Group 1: Sandwich Place, Café, Cocktail Bar
Group 2: Fast Food Restaurant
Group 3: Fast Food Restaurant
Group 4: Bar, Pizza Place
Group 5: American Restaurant, Italian Restaurant
Group 6: Mexican Restaurant
Group 7: American Restaurant, Fried Chicken Joint, Sandwich Place, BBQ Joint, Pizza Place
Group 8: Restaurant
Group 9: American Restaurant
Group 10: Bar
Group 11: Chinese Restaurant
Group 12: Coffee Shop
Group 13: Sandwich Place
Group 14: Food, Bakery, Ice Cream Shop, Seafood Restaurant, Taco Place, Sandwich Place
Group 15: Pizza Place
Group 16: Bar
Group 17: Donut Shop
Group 18: Pizza Place, Sandwich Place, Food, Bakery
Group 19: Food, American Restaurant, Fried Chicken Joint

3.4 Final Analysis

The clustering algorithm was run with $k = 20$ clusters and $\alpha = 0.4$ The resulting clusters are depicted on a map in Figure 10. To the right of the map is a list of each cluster and all restaurant categories in it which comprise more than 5% of the cluster. We observe there are several clusters which contain only one sort of restaurant.

To understand what the algorithm has done, we'll examine each of the groups by category. We'll start with the categories containing only one sort of restaurant in the list of Figure 10 and compare each two the results of k-means clustering the original dataset. We will start by examining the clusters that contain only one kind of venue category representing more than 5% of that cluster.

3.4.1 Mexican Restaurants

Figure 11 depicts plots of the two Mexican restaurant clusters, as well as the k-means clustering of all Mexican restaurants in the data set. Note the similarity between the two, but also that the clusters from our analysis have cut out some of the outliers in the south neighborhoods.

3.4.2 Fast Food Restaurants

Figure 12 depicts plot of the two fast food clusters, as well as the k-means clustering of all fast food restaurants in the data set. The two plots are nearly identical, so not much interesting information was gained from the clusters.

Figure 11: Two clusterings of Mexican restaurants

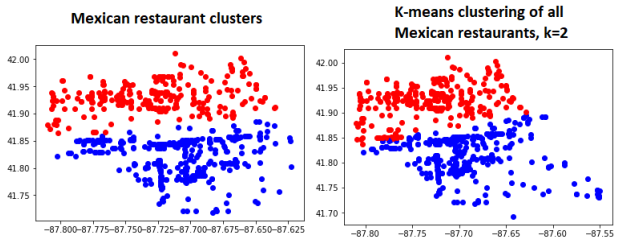
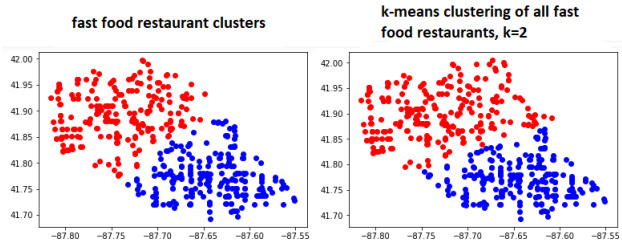


Figure 12: Two clusterings of fast food restaurants



3.4.3 Bars

Figure 13 depicts the bar clusters and the k-means clustering of bars in the original dataset with $k = 2$.

Observe that a number of bars in the western side of the red group is not present in the algorithm's clustering. They have been placed into a cluster 4 consisting of both bars and pizza joints, which we will examine momentarily.

3.4.4 Restaurants

Several venues are simply labelled "restaurant". The clustering algorithm has produced a single cluster for them, and we compare it to the "restaurants"

Figure 13: Two clusterings of bars

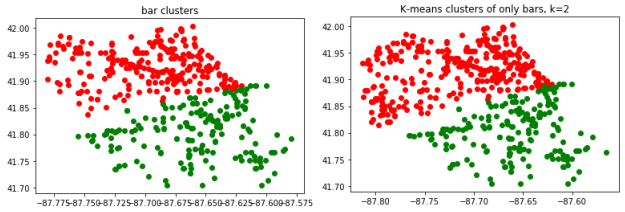


Figure 14: Venues labelled restaurant

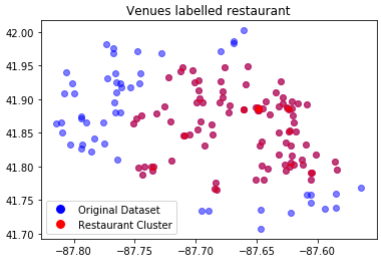
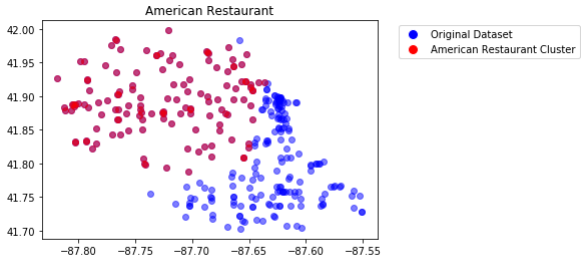


Figure 15: American restaurants



from the original dataset. Figure 14 plots both together on one graph. We see that the two share many points in common, but the restaurant cluster is, not surprisingly, picked out of the center area of the venues called "restaurant".

3.4.5 American Restaurants

The American restaurant cluster comprises American restaurants in the north and west parts of all American restaurants. See Figure 15. The south and east areas belong to clusters with multiple labels, shown below.

3.4.6 Chinese Restaurants

The Chinese restaurant cluster is located on the south and east portion of all Chinese restaurants. Figure 16 shows the Chinese restaurant cluster, as well as all Chinese restaurants.

3.4.7 Coffee and Donuts

Coffee and donut shops are popular enough to have gained their own categories. Figure 17 plots both of the clusters along with all venues labelled "coffee shop" and "donut shop". Both have picked out venues that are clustered near the

Figure 16: Chinese restaurants

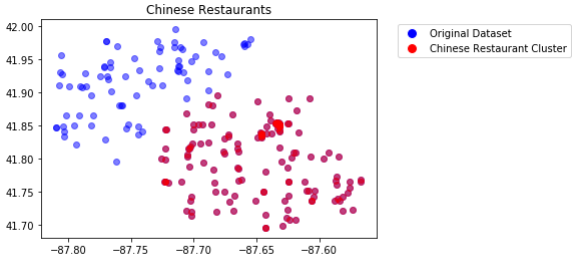
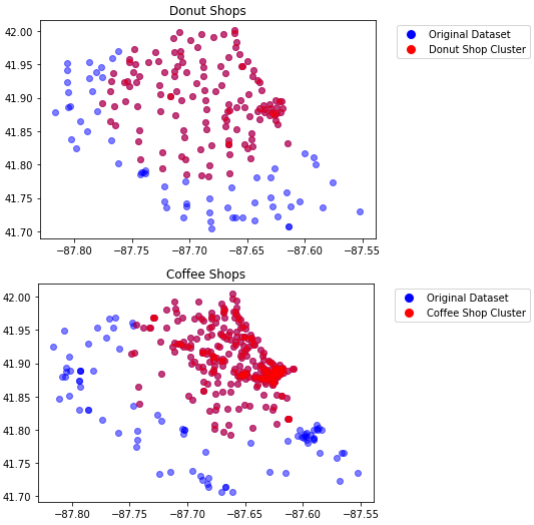


Figure 17: Coffee and Donuts

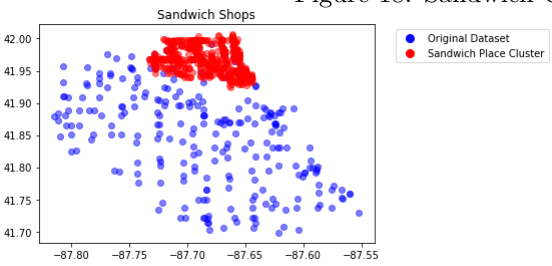


Loop area of Chicago. The coffee shop cluster has also picked up many non-coffee shop venues, labelled as bright red in the figures. Observe that neither of these categories would naturally arise from k-means clustering, since the dividing boundary is not linear.

3.4.8 Sandwich Place Cluster

The cluster containing a high concentration of sandwich places is interesting, as it is highly clustered and also contains many, many venues that are not sandwich places, as the amount of bright red in Figure 18 indicates. Examining the distributions of venue categories in that group, we find that only 5.03% of the venues are sandwhich shops, with several other venue categories in the 3 or 4% range.

Figure 18: Sandwich Cluster



3.4.9 Multicategory Clusters

Motivated by the anomolous sandwich cluster, we will reevaluate the sorts of venues in each cluster by lowering the threshold to 1%. This leaves clusters 0, 2, 3, 6, 8, 9, 10, 11, 12, 15, 16 and 17 with only one venue category, forcing only "the sandwich cluster" off the list. Clusters on this list have many, many kinds of restaurants above the 1% threshold. These clusters are geographically distinct, as seen in Figure 19. They are also distinct in the popular kinds of venues in them, as the following chart shows:

Group 1		Group 4	
Sandwich Place	5.79%	**Bar**	9.24%
Café	5.44%	Pizza Place	5.25%
Cocktail Bar	5.09%	Ice Cream Shop	4.71%
Bakery	4.56%	Sandwich Place	4.70%
Dive Bar	3.68%	Bakery	4.17%

Group 5		Group 7	
American Restaurant	6.5%	American Restaurant	10.63%
Italian Restaurant	5.3%	Fried Chicken Joint	8.38%
Sandwich Place	4.7%	Sandwich Place	7.98%
Café	4.40%	**BBQ Joint**	6.34%
New American Restaurant	3.70%	Pizza Place	6.13%

Group 13		Group 14	
Sandwich Place	5.03%	Food	16.53%
Bakery	4.56%	Bakery	9.39%
Indian Restaurant	4.40%	Ice Cream Shop	6.73%
Diner	3.30%	**Seafood Restaurant**	6.32%
Breakfast Spot	3.30%	**Taco Place**	5.92%

Group 18		Group 19	
Pizza Place	8.82%	Food	8.65%
Sandwich Place	6.08%	American Restaurant	7.28%
Food	5.29%	Fried Chicken Joint	6.87%
Bakery	5.10%	**Caribbean Restaurant**	4.95%
Chinese Restaurant	4.90%	Sandwich Place	4.67%

Categories that are unique to that group are highlited, and each cluster has at least one.

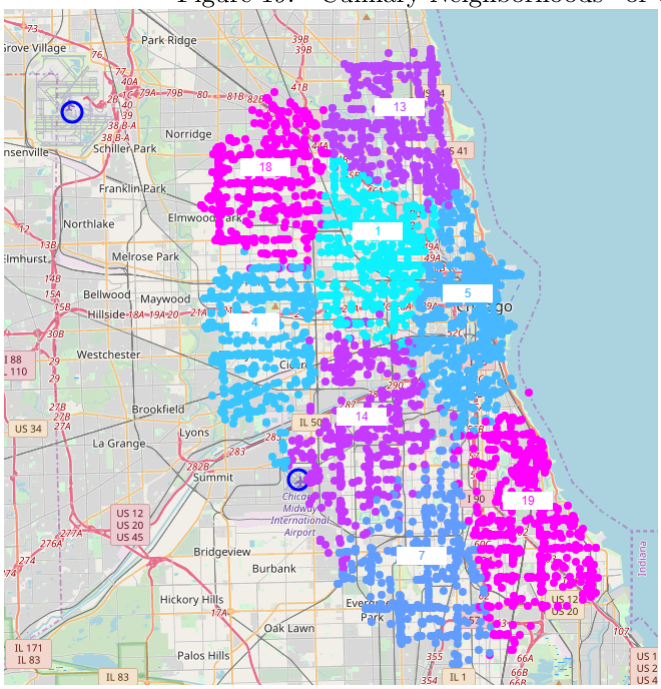
Note that it should not be assumed that many of the city's Chinese restaurants are located in group 18, as many Chinese restaurants have been placed into group 11. Geographically speaking, it need not even distinguish that group from the other seven present in the chart. It may be possible that the other groups' locations have more Chinese restaurants that have been placed into group 11. Categories which are already accounted for in a single-category cluster should be viewed with suspicion in this chart.

As one final piece of analysis, let's do a k-means clustering on the venues located in the multi-venue clusters to compare, where $k = 8$ to match the number of multi-venue clusters. The results in Figure 20 and look quite similar to Figure 16.

4 Conclusions and Further Directions

Reviewing the output of the clustering algorithm with $\alpha = 0.4$, it makes sense to conjecture that what has happened is that popular restaurants dispersed widely

Figure 19: "Culinary Neighborhoods" of Chicago



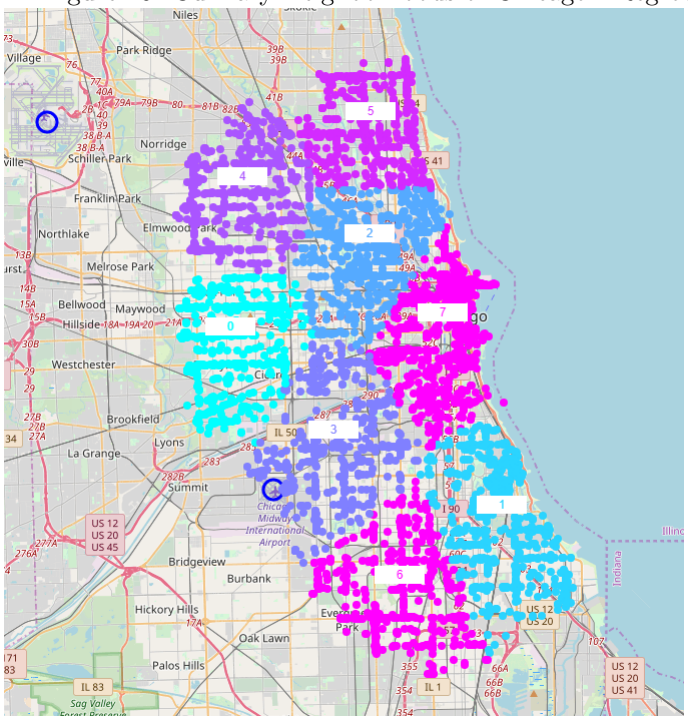
across the city such as Mexican and Chinese cuisine were largely filtered out from the data and placed in clusters on their own. Sufficiently large clusters of one cuisine are then further subdivided using k-means. Of the remaining restaurant venues that are not as popular in Chicago, another round of k-means clustering is performed. Thus, it seems that all this method of classifying regions of the city by cuisine type truly does is find popular restaurant types!

The fact that this method requires more care to get results out of is not surprising. If I use k-means on onehot encoded features, the dummy variables will spread out cuisines of different types. The most populated clusters of cuisine types will swallow up one or two of the centroids, leaving the other centroids to be divided among the remaining dummy variable columns. So it would seem that understanding the right mixture of α and k is the key to getting better information out of this method.

So, in order to produce more interesting maps of neighborhoods by cuisine, perhaps the following ideas will help:

- Try a new algorithm entirely!
- Try different values of k, α. The value $\alpha = 0.4$ was chosen via maximizing a ratio that was chosen arbitrarily, so perhaps a more meaningful value could be found. The value $k = 20$ was also chosen rather arbitrarily and is an essential hyperparameter.

Figure 20: Culinary Neighborhoods of Chicago - Regrouped by k-Means



- Filtering out popular cuisines dispersed across the whole city from the beginning might lead to more interesting results.