CSE 6040 Final Project
Final Report
Lauren Crabtree and Siddharth Shah

Introduction and previous research:

The topic we chose to do our project on is **"Potential Locations for New Restaurants"**. This topic is interesting because location plays an important role in a restaurant being successful. If a restaurant is opened in a location where there are fewer restaurants of the same category, then customers are more likely to visit this restaurant and make it successful.

To our knowledge, similar research has been done to find potential new locations for individual businesses - in both cases, coffee shops[1][2]. In both studies, additional factors like customer demographics and customer spending power are taken into consideration, while we are only looking at spatial data - locations of restaurants. Our study is slightly different in that we are finding potential locations for restaurants for multiple categories, while the existing studies only did the analysis for one specific category.

Problem definition:

The problem we are trying to address is: "How can we use data provided in the Yelp Challenge Dataset to evaluate where new restaurants for each type of cuisine can be opened?" Using attributes like the "names", "city", "categories" and "latitude/longitude" from the dataset, we will find existing locations of restaurants in Phoenix, and then evaluate optimal locations to open new restaurants. It is true that several other factors like customer demographics and customer spending power also affect the success of a restaurant, but for the purpose of this study, we are focusing only on the location for a new restaurant.

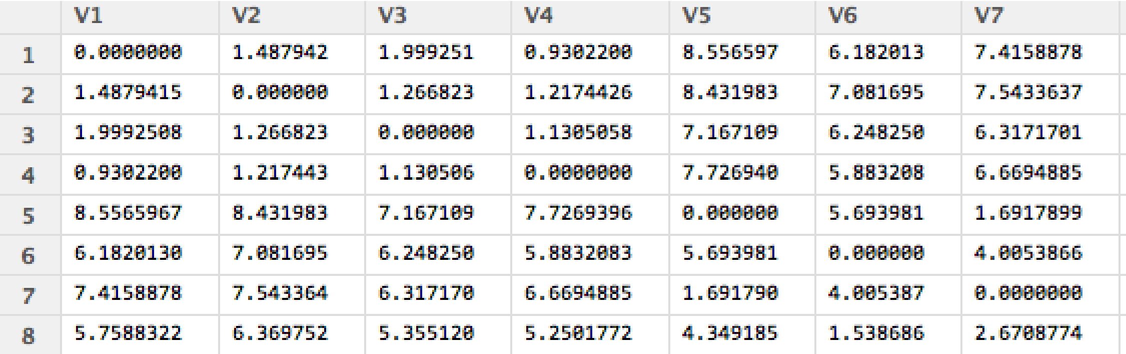
Methods:

- Description of the algorithm:

We used two different algorithms for this problem. They are as follows:

- Intuitive Algorithm:
  - We first attempted to use an intuitive algorithm to solve this problem. We created a matrix of the distances (in kilometers) between all the restaurants in one category - Greek. From this matrix, we then found for each column the minimum (nonzero) value and created an array of all these values. Then we found the maximum value from this array of all the minimum values. This would give us the largest minimum distance between two restaurants. Here is a snapshot of the distance matrix

    Here is a snapshot of the distance matrix:



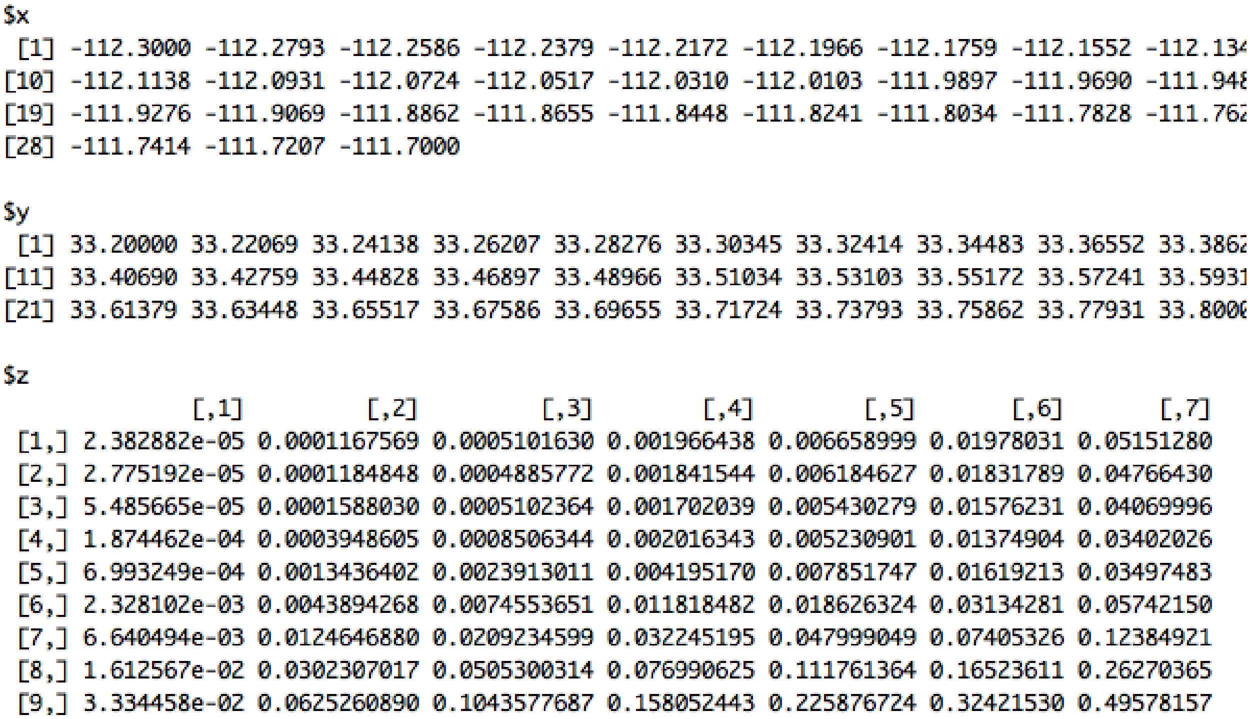
	V1	V2	V3	V4	V5	V6	V7
1	0.0000000	1.487942	1.999251	0.9302200	8.556597	6.182013	7.4158878
2	1.4879415	0.000000	1.266823	1.2174426	8.431983	7.081695	7.5433637
3	1.9992508	1.266823	0.000000	1.1305058	7.167109	6.248250	6.3171701
4	0.9302200	1.217443	1.130506	0.0000000	7.726940	5.883208	6.6694885
5	8.5565967	8.431983	7.167109	7.7269396	0.000000	5.693981	1.6917899
6	6.1820130	7.081695	6.248250	5.8832083	5.693981	0.000000	4.0053866
7	7.4158878	7.543364	6.317170	6.6694885	1.691790	4.005387	0.0000000
8	5.7588322	6.369752	5.355120	5.2501772	4.349185	1.538686	2.6708774

Figure 1: Distance matrix of Greek restaurants. Each entry Xij represents distance between restaurant i and restaurant j

- On further inspection we found two main issues with the algorithm. The first issue was that the data only includes restaurants with "official" Phoenix addresses. So restaurants in places like Tempe which is a suburb of Phoenix will not be considered. This was solved by including surrounding areas and creating a new distance matrix.

- Formal Method - Kernel Density Estimation (KDE)
  - Kernel Density Estimation is a way to estimate the probability density function of a random variable.
  - We used kernel density estimation and a filled contour graph of the kernel density estimation to graphical show the pattern of data points in each category as well as estimate the actual relative density values of data points within the geographical coordinate parameters. [Figure 5]
  - A density matrix was created to represent this estimation in a tabular format. From this matrix, we could find coordinates with low densities and use those areas to open new restaurants.

    Here is a snapshot of the density matrix for Greek restaurants.

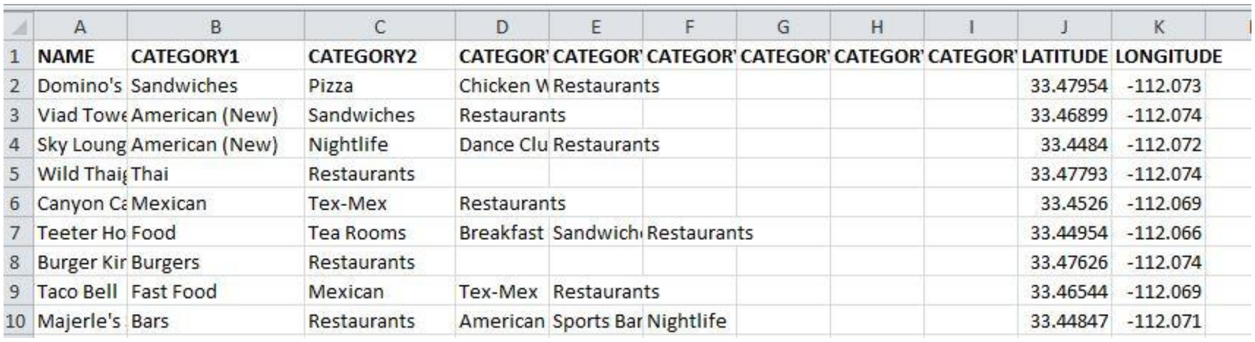


```
$x
 [1] -112.3000 -112.2793 -112.2586 -112.2379 -112.2172 -112.1966 -112.1759 -112.1552 -112.134
[10] -112.1138 -112.0931 -112.0724 -112.0517 -112.0310 -112.0103 -111.9897 -111.9690 -111.948
[19] -111.9276 -111.9069 -111.8862 -111.8655 -111.8448 -111.8241 -111.8034 -111.7828 -111.762
[28] -111.7414 -111.7207 -111.7000

$y
 [1] 33.20000 33.22069 33.24138 33.26207 33.28276 33.30345 33.32414 33.34483 33.36552 33.3862
[11] 33.40690 33.42759 33.44828 33.46897 33.48966 33.51034 33.53103 33.55172 33.57241 33.5931
[21] 33.61379 33.63448 33.65517 33.67586 33.69655 33.71724 33.73793 33.75862 33.77931 33.8000

$z
              [,1]         [,2]         [,3]        [,4]        [,5]       [,6]       [,7]
 [1,] 2.382882e-05 0.0001167569 0.0005101630 0.001966438 0.006658999 0.01978031 0.05151280
 [2,] 2.775192e-05 0.0001184848 0.0004885772 0.001841544 0.006184627 0.01831789 0.04766430
 [3,] 5.485665e-05 0.0001588030 0.0005102364 0.001702039 0.005430279 0.01576231 0.04069996
 [4,] 1.874462e-04 0.0003948605 0.0008506344 0.002016343 0.005230901 0.01374904 0.03402026
 [5,] 6.993249e-04 0.0013436402 0.0023913011 0.004195170 0.007851747 0.01619213 0.03497483
 [6,] 2.328102e-03 0.0043894268 0.0074553651 0.011818482 0.018626324 0.03134281 0.05742150
 [7,] 6.640494e-03 0.0124646880 0.0209234599 0.032245195 0.047999049 0.07405326 0.12384921
 [8,] 1.612567e-02 0.0302307017 0.0505300314 0.076990625 0.111761364 0.16523611 0.26270365
 [9,] 3.334458e-02 0.0625260890 0.1043577687 0.158052443 0.225876724 0.32421530 0.49578157
```

Figure 2: Density Matrix from KDE for Greek restaurants. Each entry Xij represents the density at latitude i and longitude j

– Software Packages Used and Implementation of the Algorithm:

- For extracting the data from the Yelp Challenge dataset, we used Python. The data was imported in Python using the "JSON" package. The extracted data was exported to a file using the "CSV" package. Here is a snapshot of the extracted data:



	A	B	C	D	E	F	G	H	I	J	K
1	NAME	CATEGORY1	CATEGORY2	CATEGOR'	CATEGOR'	CATEGOR'	CATEGOR'	CATEGOR'	CATEGOR'	LATITUDE	LONGITUDE
2	Domino's	Sandwiches	Pizza	Chicken W	Restaurants					33.47954	-112.073
3	Viad Towe	American (New)	Sandwiches	Restaurants						33.46899	-112.074
4	Sky Loung	American (New)	Nightlife	Dance Clu	Restaurants					33.4484	-112.072
5	Wild Thai	Thai	Restaurants							33.47793	-112.074
6	Canyon Ca	Mexican	Tex-Mex	Restaurants						33.4526	-112.069
7	Teeter Ho	Food	Tea Rooms	Breakfast	Sandwich	Restaurants				33.44954	-112.066
8	Burger Kir	Burgers	Restaurants							33.47626	-112.074
9	Taco Bell	Fast Food	Mexican	Tex-Mex	Restaurants					33.46544	-112.069
10	Majerle's	Bars	Restaurants	American	Sports Bar	Nightlife				33.44847	-112.071

Figure 3: Extracted Data

- For scraping data for neighboring cities of Phoenix, we used python to scrape directly from the Yelp website[3]. Using the "BeautifulSoup" package, we scraped the name, categories and addresses. Since the addresses on the Yelp website were physical and not the coordinates, we used an external website to convert[4].

- In order to subset the data and map the restaurant locations, we used R. For the map creation we used the “Google Maps” package in R using the longitude and latitude for all of the restaurants in the data set to create the map boundaries. After subsetting the data and plotting these categories, we created a spatial data frame using the ‘SP’ package and calculated the distance matrix.
- We used the “MASS” package in R to create a kernel density matrix of bivariate data with the “kde2d” estimator. This represented Phoenix and its neighboring cities in a 25x25 grid to give density estimations at each coordinate grid. We then created a filled contour map to give a visual representation of the density of restaurants in the area. As a result we were able to compare the actual density values and the filled contour plot with other categories.

- Charting and/or visualization that help make decisions in your analysis:

For both our methods, data visualization was an important component in making decisions.

- For the intuitive method, our first step was to plot all Greek restaurants in Phoenix on a map. [Figure 4]. This helped us get a measure of the density of the Greek restaurants and how they are spread in Phoenix. This plot also helped us make the decision of expanding to neighboring cities.
- For the KDE approach, we were able to create multiple contour maps of the kernel density estimates for several categories to observe the different hubs for each type of restaurant within our coordinate values.

Results:

– Quantitative evaluation of the method:

We ran our algorithm for Greek restaurants and found the optimal location to open a new Greek restaurant. Below is a plot of all Greek restaurants. The two restaurants that had the largest minimum distance according to our algorithm are represented in red and the optimal location would be in between those two restaurants.

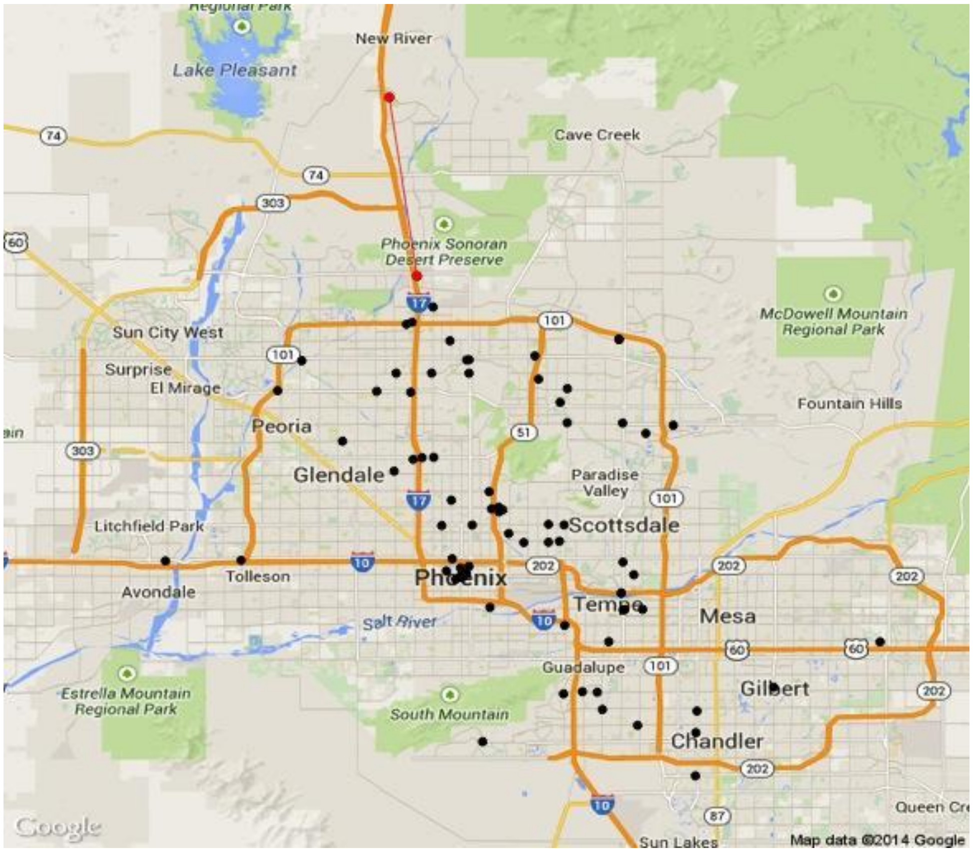


Figure 4: Plot of all Greek restaurants in Phoenix and neighboring cities. The red restaurants represent the solution and the optimal location is between those two restaurants.

We implemented KDE for the same Greek restaurants. The filled contour plot created for Greek restaurants is below:

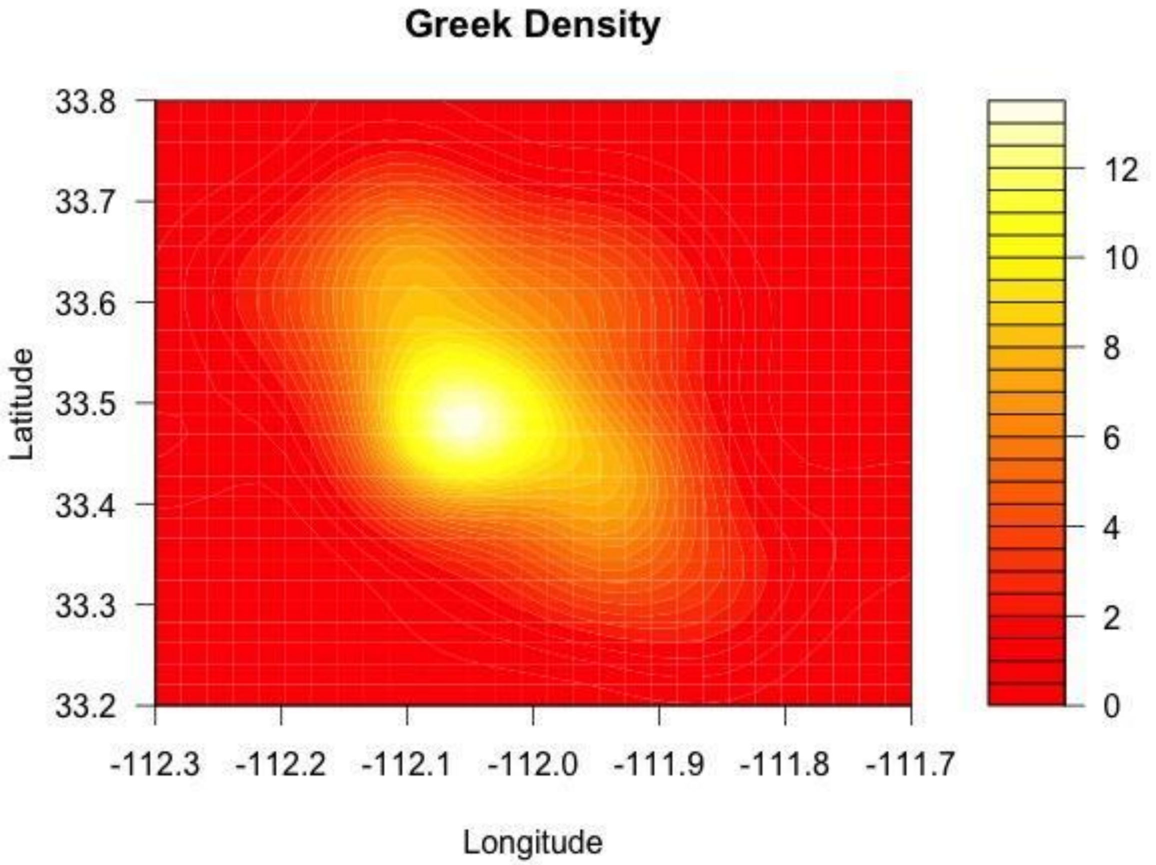


Figure 5: KDE contour plot. The latitudes are on the X axis and the longitudes on the Y axis. The legend is number of restaurants.

The kernel density contour plot more clearly articulates the actual ratio of restaurants within this specific location. Each density graph for every category has a unique pattern and slightly different central point. The highest density is reached around the point [33.5, -112.5]. It may be more valuable to open new restaurants in areas with low densities, which is consistent with the results we obtained from the intuitive method.

– Charting and/or visualization for the end-user

The end-user would use a single script that asks them what kind of restaurant they want to open, and then proceeds to plot the current restaurants of that category on a map, evaluate the top recommendations for new locations for restaurants using both the intuitive method and KDE and plot those on the map as well. Currently we have created a prototype of this script for Greek restaurants.

– How your proposed approach has solved the question

Both the methods give the user valid recommendations for locations to open new restaurants in, which was one of the main targets of our project. Through both the methods, one of the insights we gained is that it may be more valuable to open a new restaurant in the outskirts of the city, where restaurant density is lower and distances between restaurants is larger. For a large city like Phoenix, the center of the city is already quite saturated with restaurants.

It is important to remember that only the location is being considered here, which is why locations towards the edge of the city were recommended. If other factors like customer spending power and customer demographics were taken into consideration, the results may be different.

Another insight we had is that certain categories have stronger "hubs" of restaurants - areas where that category has a high number of restaurants. We compared the density of a moderately popular category - Greek (83 restaurants) with a highly popular category - Chinese (238 restaurants). The maximum density found through KDE for Chinese restaurants was 18.9 restaurants and for Greek restaurants it was 12.6 restaurants. Below is a plot of Greek and Chinese restaurants on the same map which helps visualize this finding.

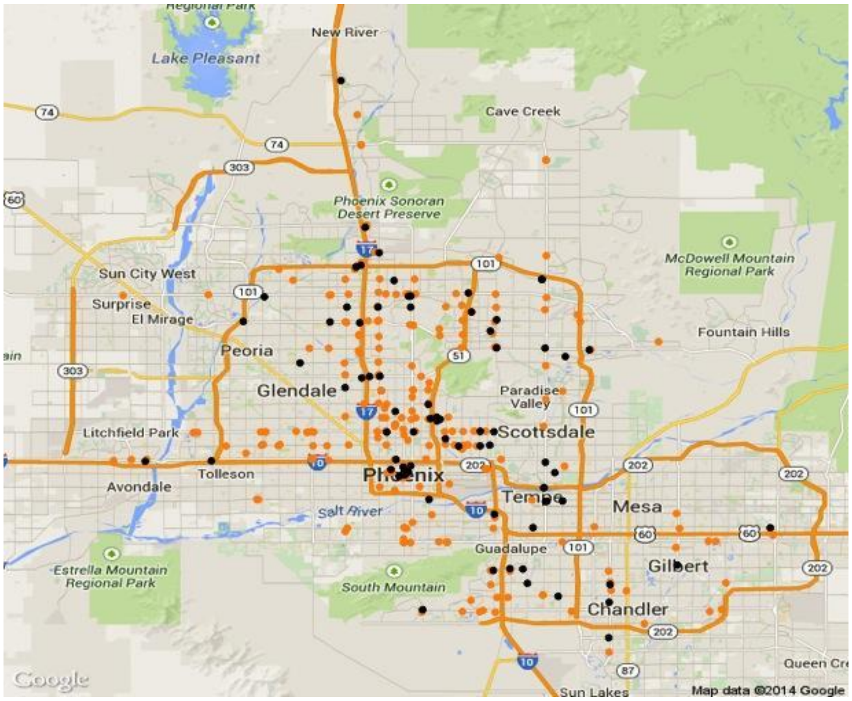


Figure 6: Greek (black) and Chinese (orange) restaurants on the same map.

Division of work:

- Siddharth was responsible for extracting required data from the Yelp dataset. Siddharth was also responsible for scraping the Yelp website to get data for neighboring cities. This required the additional step of cleaning the scraped data to match the format of the data from the dataset (converting physical addresses to coordinates). This was all done in Python

- Lauren was responsible for implementing both the intuitive method and KDE using the data obtained. This was done in R
- Both Siddharth and Lauren collaborated to come up with the approach that should be taken - including coming up with the intuitive method, the decision to expand outside Phoenix, and using KDE.

Appendix

[1] http://www.gis.smumn.edu/GradProjects/RingoL.pdf
[2] http://www4.ncsu.edu/~livia/M1.htm
[3]http://www.yelp.com/search?find_desc=restaurants&find_loc=Phoenix%2C+AZ#l=p:AZ:Phoenix::
[4] http://www.gpsvisualizer.com/geocoder/