IBM Capstone Project for Data Sciences

Exploring the desert: Analysis of venues density in Tucson - AZ

Maria Cecília Rodrigues do Prado

1. Introduction

Figure 1 - View of Downtown Tucson. Source: pixabay.com

The city of Tucson is located in the south of the state of Arizona in the United States. It is home of the 33rd largest population in the country, 2nd in its state, spread over 624 km^2. Although it stands among the most populated cities, its population density of 888 people/km² is low, meaning that the city buildings are rather scattered over the area than condensed. The city has not grown as much vertically as horizontally, so, tall buildings are not so common. Moreover, the city stands in the Sonoran Desert, so temperatures around 40ºC are common during 5 months, every year.

Moving around the city can be difficult, given the aforementioned conditions. The aim of this project is to provide meaningful information about the density of of venues in different Tucson neighborhoods to provide support for several choices, such as suitable areas to live, depending on one's need of public transportation, a potential good location for a business, priority areas for an enhancement in public transportation and other matters.

2. Data

2.1. Neighborhood names

The website *city-data.com* provides a range of information for US neighborhoods, including race and of inhabitants, household income, house values, education, means of transportation and many other information. From this source, we obtained the names of Tucson's neighborhoods.

2.2. The Foursquare API

The Foursquare API provides information about millions of venues and users from all over the world. With a developer's account, it is possible to search venues based on its location, category, name and many other features. In this study, we are interested in venues of all categories around specific geographic coordinates - the coordinates of our neighborhoods.

3. Methodology

3.1. Geographical coordinates

We performed a webscrap on the *city-data.com* website to get the neighborhood names. From neighborhood names, we used the geographic coordinates obtained by the Geopy geocoder. For simplicity's sake, the neighborhoods that were not found by the geocoder were not used in this study and we obtained a set of 21 neighborhoods. We then plotted the coordinates in a map, using the Folium library (fig. 2).

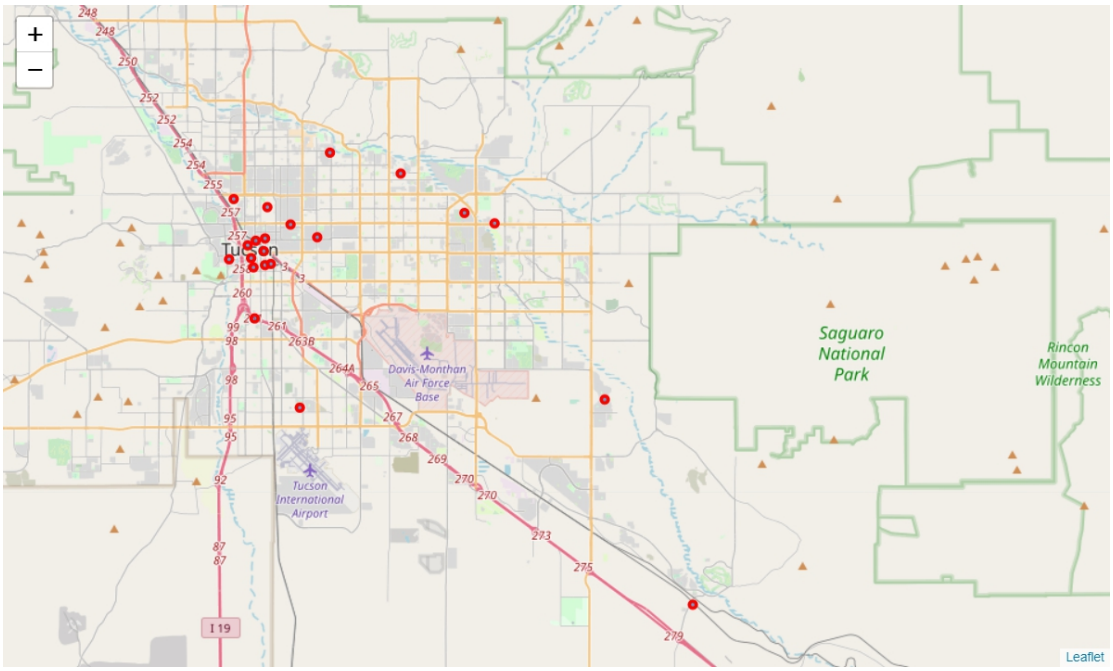

Figure 2 - Neighborhoods throughout Tucson represented by red dots.

The figure reveals a clear imbalance of neighborhoods scattering. Our goal is to perform an analysis as spatially uniform as the data allows, so we applied the K-Means clusterization method.

3.2. Clustering the neighborhoods

We applied the K-Means method to cluster our neighborhoods, to minimize information overlapping. The map shows 13 points scattered around the city and 8 points close to each other in the central area. Therefore, we set the value of K to 14. As we already knew the number of clusters we wanted, it was not necessary to perform metrics evaluation such as elbow or silhouette plots. The centroid of each cluster is presented in blue on figure 3.

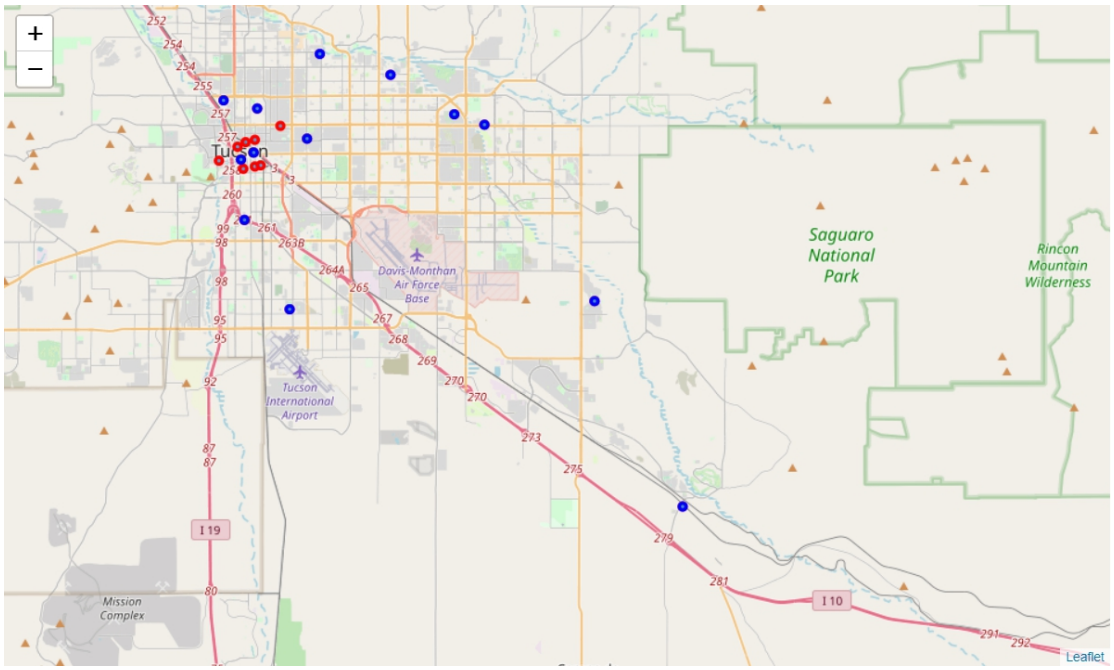

Figure 3 - Original neighborhoods in red and cluster centroids in blue.

The centroids positions correspond to the geographic coordinates of our interest. These coordinates were used in the search for venues in the Foursquare API.

3.3. Getting venues information from Foursquare API

The Foursquare API allows the users to set a maximum radius for the search and returns up to 50 closest and most relevant venues. If an area has a high density of venues, they will all be listed in a small radius search, but if an area has a low density of venues, a larger radius is needed to encompass the maximum number of venues. For this reason, we used radius values of radius of 3000, 1000, 500 and 100m in our searches for venues in each cluster centroid. Because the Foursquare free developers account allows a limited

number of diary calls, the searches results were stored into CSV files, so it was not necessary to further calling the API.

We then grouped the information of the files and stored in a Pandas data frame containing only our data of interest: latitude, longitude and distance to centroid. At this stage, we also dealt with duplicate data, which were dropped. We could then, analyse the venue density for each neighborhood.

3.4 - Clustering Neighborhoods based on distance of venues

To further understand the general characteristics of venues distribution over each neighborhood cluster, we applied de K-Means method once again, using the average distance of venues to the centroids and their standard deviation. We then applied the Elbow (fig.4) and Silhouette (fig.5) methods to base the choice of cluster numbers.

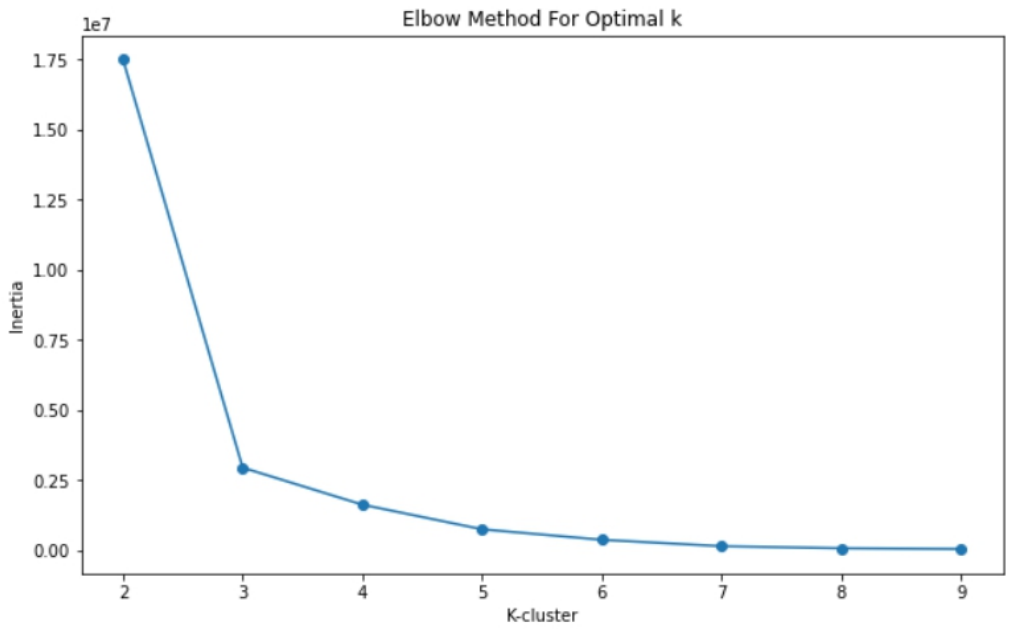

Figure 4 - Elbow method curve

In Figure 4, it is possible to detect that the angle of lines between clusters 2 and 3 and 3 and 4 is smaller than the angles of other segments. It implies that the best value for number of clusters is 3.

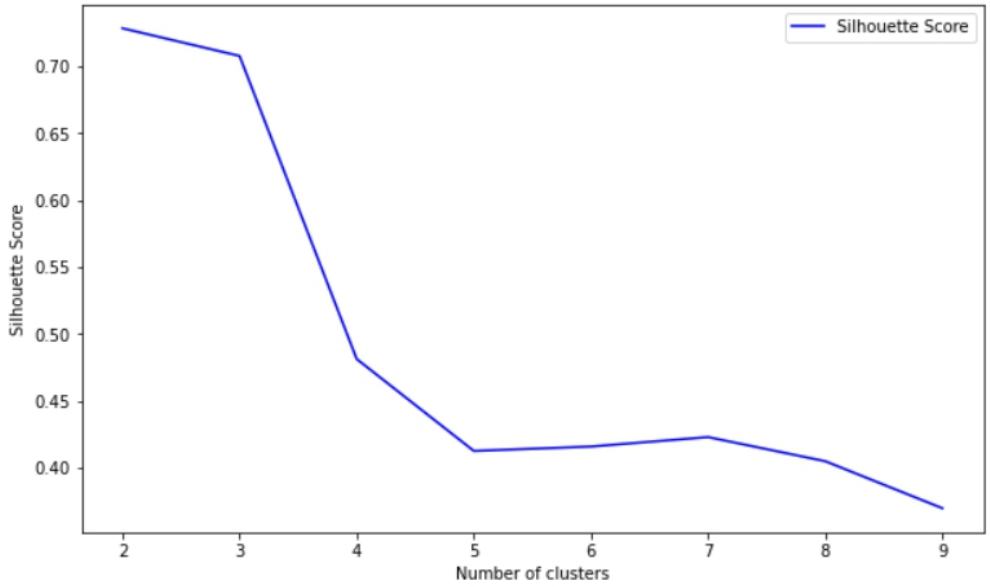

Figure 5 - Silhouette method

Using the Silhouette method, we verified that 3 is good choice for our number of clusters, due to its high value compared to other numbers higher than 2.

4. Results

Figure 6 represents all the venues obtained through the Foursquare API, around each cluster's centroid.

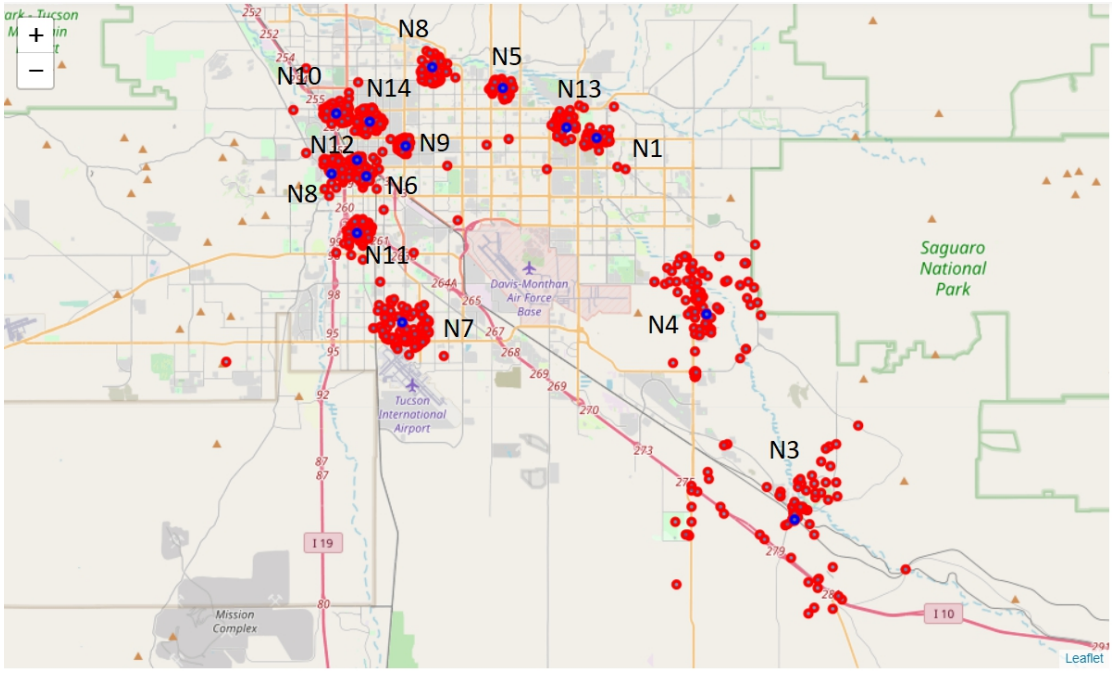

Figure 6 - Venues around each cluster's centroid.

The figure shows that neighborhoods located on south and east parts of the city tend to present lower density than the neighborhoods located on the north and central parts. Below is the average distance and standard deviation for each neighborhood cluster.

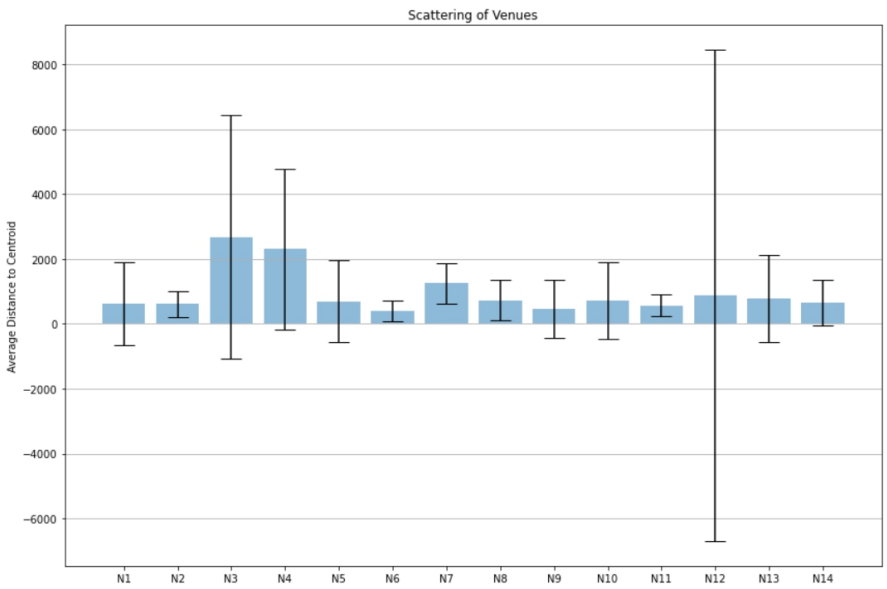

Figure 7 - Average distances of venues to clusters' centroids (blue bars) and their standard deviation.

Figure 8 presents the clusters calculated based on distance and standard deviation of venues. We called the red points "Cluster 1", the blue point "Cluster 2" and green points "Cluster 3".

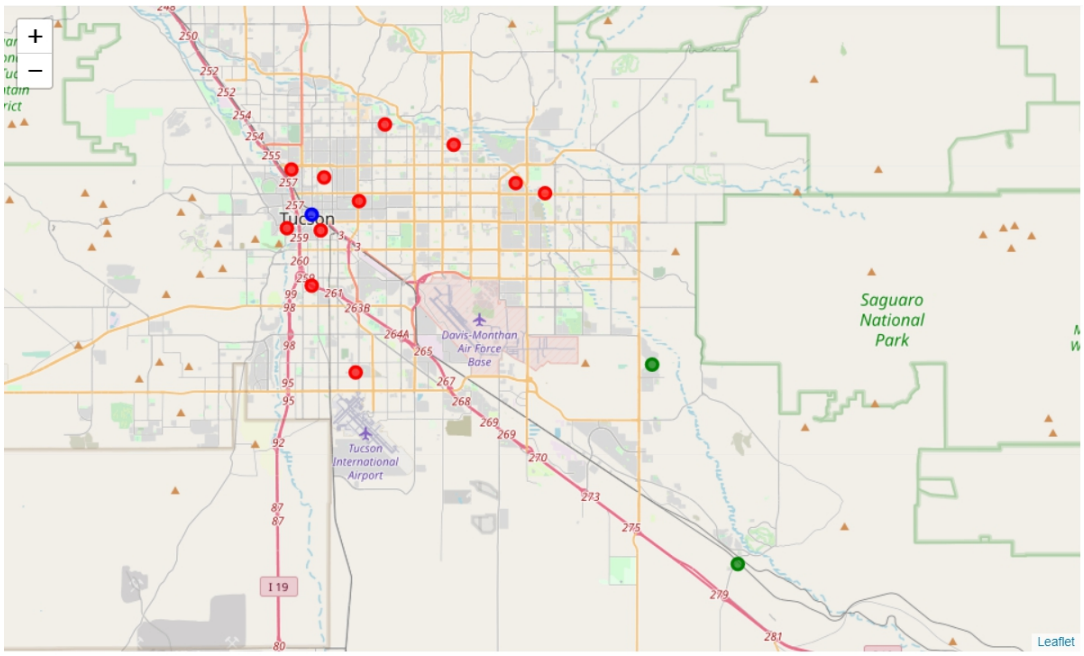

Figure 8 - Final clustering. Red: Cluster 1, Blue: Cluster 2, Green: Cluster 3.

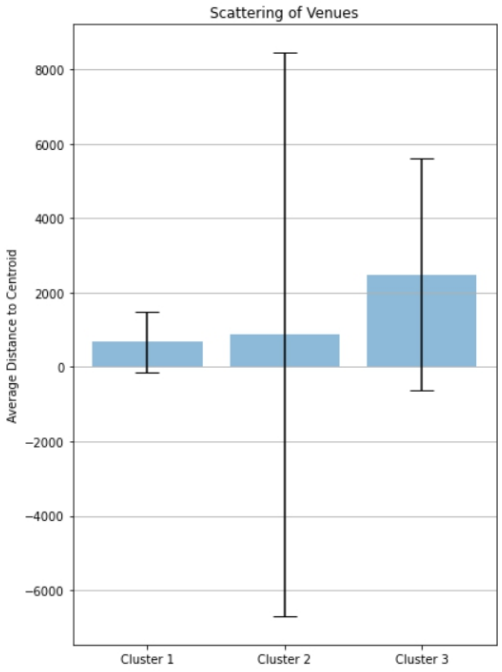

Figure 9 - Average distances of venues to final clusters' centroids (blue bars) and their standard deviation.

5. Discussion

It is clear from the figure 9 above that neighborhoods in Cluster 1 presented a higher density of venues, so people who choose to live in them will have a range of venues in a walking or biking distance. Average distance to venues is around 680 m, whith a standard deviation of 814, meaning the neigborhoods are close together and there aren't venues very far away.

In the other hand, Cluster 3 presented a low density of venues. That means a person needs to travel long distances to get to service building, schools, shops, restaurants and recreational sites. This implies that individuals who chose to live in those areas probably have their own means of transportation. These might also be suitable areas for opening delivery business.

Interestingly, Cluster 2 presented average distance (875 m) not so difference from Cluster 1, but the standard deviation is the highest from all clusters (7572 m), indicating the presence of possible outliers.

6. Conclusion

We identified how venue density is geographically spread over the city. There were three main cluster conformations: low mean and low standard deviation, low mean and very high standard deviation and high mean and high standard deviation.

The first configuration describes venues that are near each other and in a homogeneous manner. The second describes venues that are mostly close to each other, but some relevant venues located very distant from the center. The third configuration describe venues that are spread apart.

For further analysis, we recommend to identify and treat the outlier venues, deciding carefully wether these venues should be included or not based on their relevancy. Moreover, we recommend to include venues cathegory to further understand the geographical distribution of venues within the city and with that, provide information support for better decisions regarding this issue.