CAR RIDE-SHARE POTENTIAL IN MID-SIZE U.S. CITIES FROM GEOGRAPHIC SPREAD

JENS KOEPLINGER

Abstract. This report is part of satisfying the IBM Data Science Specialization on Coursera, Course 9 "Applied Data Science Capstone", assignment "The Battle of Neighborhoods". It describes a ficticious analysis requested by a car ride-share company startup, looking for the best suited places to roll out their service. Cities are selected from the 2017 U.S. census population estimate by having between 300,000 and 400,000 inhabitants. For each city, an iterative algorithm first determines the geographic spread of its densest areas by number of venues. Cities are clustered by similarity of this spread, and then ranked within each cluster by relative count of venues types that would indicate to the ficticious company that the city population has an open attitude to new trends.

1. Introduction

A ficticious car ride-share company startup is looking for suitable cities to roll out their service. Customers will be able to sign up for shared car rides across town. Company research has identified two criteria for a data science analysis to be executed:

(1) The cities need to be geographically spread out, with venues arranged in a non-compact manner as opposed to centered around a single core, and
(2) they need to have a significant number of venues of types that indicate a population open to trying out new transportation arrangements.

The company is targeting cities with 300,000 and 400,000 inhabitants according to the 2017 census estimate. Indicator venues include alternative eating, lifestyle, and work types. This may include coworking spaces, yoga studios, delicatessen, tea rooms, vegan and other nonstandard restaurants. The analysis will list the venue types selected as indicators.

A successful study clusters cities by geographic spread of their venues, and then ranks them within each cluster by relative occurrence of indicator venues as compared to total venues.

2. Data

Geographic location of venues and their spread will be obtained using Foursquare data from U.S. cities with 300,000 to 400,000 according to the 2017 census estimate [1].

A custom algorithm iteratively determines the number of venues in each city, together with their geographic spread. This allows to cluster cities by select key

Date: 20 March 2019.

parameters describing the geographic spread. The details of this algorithm and clustering are decsribed in the methodology section 3 below.

The ficticious company is asking sort each cluster by relative count of venues of types that indicate openness towards trying out new transportation arrangements. Because of the subjective nature of deciding which venue type would be an indicator venue, an ad-hoc proposal will be made for the result (section 4) and defended in the discussion (section 5).

3. Methodology

3.1. **Cities data set.** The data set of cities from the 2017 U.S. census estimate is prepared in Wikipedia [2]. The table is parsed and city name, state, estimated population, and geographic latitude and longitude of a city reference point are extracted.

3.2. **Venues near a given location, indicator venues.** A Foursquare query is used to obtain a list of venues within a certain range around a given geographic location (by latitude and longitude). The query result contains a lot more information, from which the venue type is examined whether a given venue indicates openness of the population towards new transportation arrangements.

Determining whether or not a venue is indicative of a certain population behavior is highly subjective. For the purpose of this Coursera assignment, I'll assume that this can be done using a simple substring match with phrases provided by the ficticious company. The phrases are assumed set as:

> share, sharing, incubat, innovat, coworking, alternative, yoga, salad, bike, fitness, running, jogging, cycling, cycle, athletics, gluten, health, recreation, tennis, vegetarian, vegan, disc golf, pilates

3.3. **Iterative search algorithm for most venues.** An algorithm is built that iteratively finds the regions of highest density of venues. Starting at a given coordinate, it performs six Foursquare queries in a hex grid around that coordinate. It finds the coordinate frmo those six that has not been queried yet, but contains the highest number of venues. The algorithm repeasts until a maximum iteration cutoff is reached.

The rationale behind this algorithm is that it finds and follows regions in a city that have a high density of venues. It favors connectedness in that it will follow the directions of where there are more venues, and disfavor directions in which there are fewer venues. For a car ride-share company, this would indicate that along the directions the algorithm searches there are more venues in general as compared to other directions, giving it a higher chance of finding successive customers at those places.

3.4. **Aggregate geographic spread.** In order to flatten the data set into one row per city, the geographic spread of venues is then aggregated by determining the weighted coordinate center of all venues, calculating the distance distribution for all venues from that center, and obtaining the following descriptive statistics:

- Mean,
- standard deviation,
- mean absolute deviation,
- minimum, 25th-, 50th- (median), 75h-percentile, maximum,

- interquartile range,
- kurtosis,
- skewedness.

3.5. **Feature engineering.** After visually examining the results, the number of features is reduced by eliminating highly correlated data points.

Once a small number of features is determined, they will be standardized by dividing them by their standard deviation and subtracting their mean. This centers all features around 0 and gives them equal weight when comparing cities with one another using a Euclidean distance metric.

In order to not let a single outlier metric dominate the distance function, the features will be bounded in the $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ interval by applying the arctan function. This function leaves values near 0 unchanged, but impacts values further away stronger by bounding them into the interval. Since the data is standardized such that distances $\{-1, 1\}$ correspond to one standard deviation away from the mean, this function has reasonable bounding charactersistics. This is verified by graphing the resulting distributions in histogram form.

3.6. **Clustering.** A simple k-means clustering will be performed, first by finding a proper k that groups the data set into a representative number of clusters, and then by re-running the clustering multiple times to see which cities are stable in a representative cluster.

In order to tag a cluster as representative, the city with the highest number of indicator venues (relative to total venues) is tagged as "most indicative city". Starting with $k = 2$ the number of clusters is increased step by step up to $k = 9$. The best k value is chosen as the smallest k at which the indicator city appears to have been clustered reliably with its similar peers.

3.7. **Final choice.** The cluster with the "most indicative city", as well as the grouping of other cities with a high relative number of indicator venues will be examined. This allows to make a final recommendation to the ficitious company on where to start its ride-share business.

4. Results

For step-by-step details, including tests and trials towards the eventual implementations, see the Jupyter notebooks [3].

4.1. **City selection.** Table 1 shows the selected cities for this study.

4.2. **Geographic distribution of venues.** Figures 7.1 through 7.19 are screenshots of folium maps generated from the iterative venue finder algorithm. The radius of the blue circles is proportional to the number of venues found in the vincinity of that coordinate point. Opacity is used to amplify the visual appearance of density.

4.3. **Aggregates and feature engineering.** Table 2 shows the geographic distribution aggregates for each of the cities under investigation, as determined by the venues algorithm above. The values appear nicely scattered across a wide range, which should make for a meaningful analysis. Figure 4.1 allows to confirm this visually by inspecting the histograms of the raw, unmodified data points.

City	Population	Latitude	Longitude
Arlington, TX	396394	32.7007	-97.1247
New Orleans, LA	393292	30.0534	-89.9345
Wichita, KA	390591	37.6907	-97.3459
Cleveland, OH	385525	41.4785	-81.6794
Tampa, FL	385430	27.9701	-82.4797
Bakersfield, CA	380874	35.3212	-119.0183
Aurora, CO	366623	39.6880	-104.6897
Anaheim, CA	352497	33.8555	-117.7601
Honolulu, HI	350395	21.3243	-157.8476
Santa Ana, CA	334136	33.7363	-117.8830
Riverside, CA	327728	33.9381	-117.3932
Corpus Christi, TX	325605	27.7543	-97.1734
Lexington, KY	321959	38.0407	-84.4583
Stockton, CA	310496	37.9763	-121.3133
St. Louis, MO	308626	38.6357	-90.2446
Saint Paul, MN	306621	44.9489	-93.1041
Henderson, NV	302539	36.0097	-115.0357
Pittsburgh, PA	302407	40.4398	-79.9766
Cincinnati, OH	301301	39.1402	-84.5058

TABLE 1. Cities with population between 300,000 and 400,000 according to the 2017 U.S. Census estimate

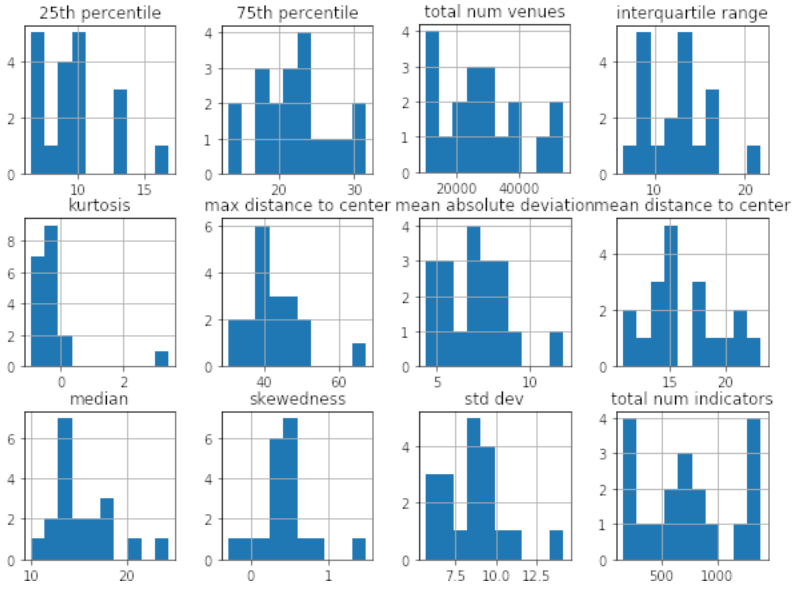

FIGURE 4.1. Histograms of raw data points for all cities.

City name, State	#ven	#ind	mean	std dev	25th	median	75th	max	IQR	kurtosis	MAD	skew
Arlington, TX	19691	537	14.3387	7.0843	9.0238	14.2145	18.4895	34.8673	9.4657	-0.2858	5.6521	0.3754
New Orleans, LA	29412	712	15.4791	8.5741	8.9713	13.2711	21.8997	39.3860	12.9284	-0.6378	7.2039	0.5151
Wichita, KS	16443	330	13.6548	6.6745	9.3498	13.3901	17.7241	34.9169	8.3743	-0.2915	5.2644	0.2283
Cleveland, OH	29795	923	17.0707	8.4256	10.4362	16.5191	22.8600	38.7520	12.4237	-0.4959	6.9080	0.3652
Tampa, FL	27350	867	15.1707	6.5010	10.5091	14.7460	19.3944	39.0680	8.8853	-0.0687	5.1626	0.3327
Bakersfield, CA	13099	265	11.3379	5.7561	6.7734	11.3896	14.9125	30.2031	8.1391	-0.3846	4.6722	0.3264
Aurora, CO	38823	1319	15.5712	8.8961	8.4923	14.0925	22.3417	46.4409	13.8494	-0.4944	7.4455	0.4949
Anaheim, CA	50441	1366	18.7794	7.6983	12.9883	17.8040	23.4351	43.2360	10.4467	-0.1643	6.2545	0.5582
Honolulu, HI	22199	518	21.1932	11.2569	12.6960	20.3462	29.5430	49.0593	16.8469	-0.7434	9.2879	0.2719
Santa Ana, CA	54031	1271	20.0972	9.3565	12.7733	18.3937	27.6653	47.2550	14.8920	-0.6433	7.9070	0.4425
Riverside, CA	23727	571	23.2236	9.8502	16.7946	24.3600	30.6846	46.6344	13.8899	-0.6989	8.1771	-0.2850
Corpus Christi, TX	10409	157	13.2069	8.4404	6.4260	11.3794	18.7835	42.7136	12.3575	-0.1420	6.9449	0.7373
Lexington, KY	13018	258	10.7035	5.9801	6.7620	9.9097	13.2641	38.6219	6.5021	3.4513	4.3775	1.4738
Stockton, CA	10320	261	14.7223	9.6108	6.4719	12.8890	22.8703	40.4613	16.3983	-0.8536	8.2666	0.4875
St. Louis, MO	33249	779	17.2145	9.5642	10.0638	16.0151	23.0403	49.0773	12.9764	-0.0928	7.7394	0.5622
Saint Paul, MN	38889	1388	14.1652	5.6409	9.9856	14.2069	18.5735	30.4644	8.5878	-0.6202	4.7007	-0.0357
Henderson, NV	46569	1246	17.2175	9.9413	9.1680	15.5206	25.7430	42.6078	16.5749	-0.9627	8.4417	0.3339
Pittsburgh, PA	28870	687	14.6092	8.7759	7.4238	13.3848	20.9797	41.0456	13.5558	-0.4846	7.3007	0.5579
Cincinnati, OH	25115	719	21.3201	14.1753	9.7997	17.7427	31.5938	67.4332	21.7941	0.1362	11.8521	0.8254

TABLE 2. Aggregate geographic distribution features by city. "#ven" is total number of venues found, "#ind" the number of indicator venues thereof, "25th" and "75th" are percentiles, "IQR" is the interquartile range, and "MAD" the mean average deviation.

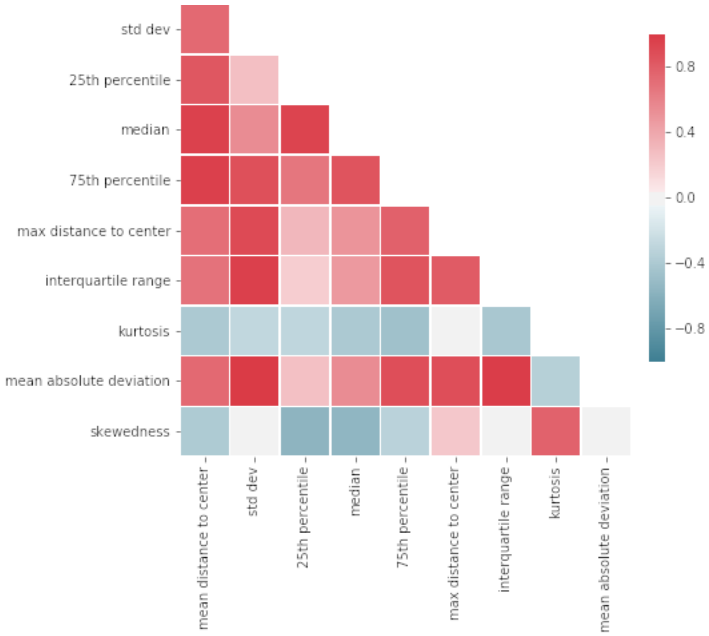

FIGURE 4.2. Correlations between all data points

Some of these data points are highly correlated, which would effectively emphasize these over other data points. In order to create features from about equivalently weighted data points, the ones with the highest degree of correlation are removed. Figure 4.2 shows a heatmap of overall correlations.

The data points for mean absolute deviation, mean distance to center, median, and standard deviation are the highest-correlated overall and are removed from further analysis. Figure 4.3 shows the correlations of the remaining, selected data points.

The upcoming clustering algorithm will compare similarity by Euclidean distance between the selected features. In order to give the six selected data points equal weight, features are generated by dividing the data points by their standard deviation. They are also centralized by subtracting their mean. The resulting distributions are shown in figure 4.4.

Some of the features have outliers that may put an undue emphasis on a single metric from a single city, while reducing the variability between all other cites. After applying the arctan function to bound all values in the $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ interval, the extreme values are pulled closer to the remaining values (which in turn are more spread out). Figure 4.5 shows the value distribution after appling this bounding function, and confirms the desired effect.

4.4. **Clustering.** Saint Paul, MN, has a relative count of about 3.6% indicator venues as compared to total venues. It is temporarily chosen as "most indicative city". The k-means algorithm is then applied for $k = 2, \ldots, 10$ to see how many cities end up in the same cluster as Saint Paul. The result is shown in table 3. Starting at $k \geq 4$ there is at least one outlier city (one sole city for a cluster label), whereas the city distribution for $2 \leq k \leq 5$ is about evenly between cluster labels.

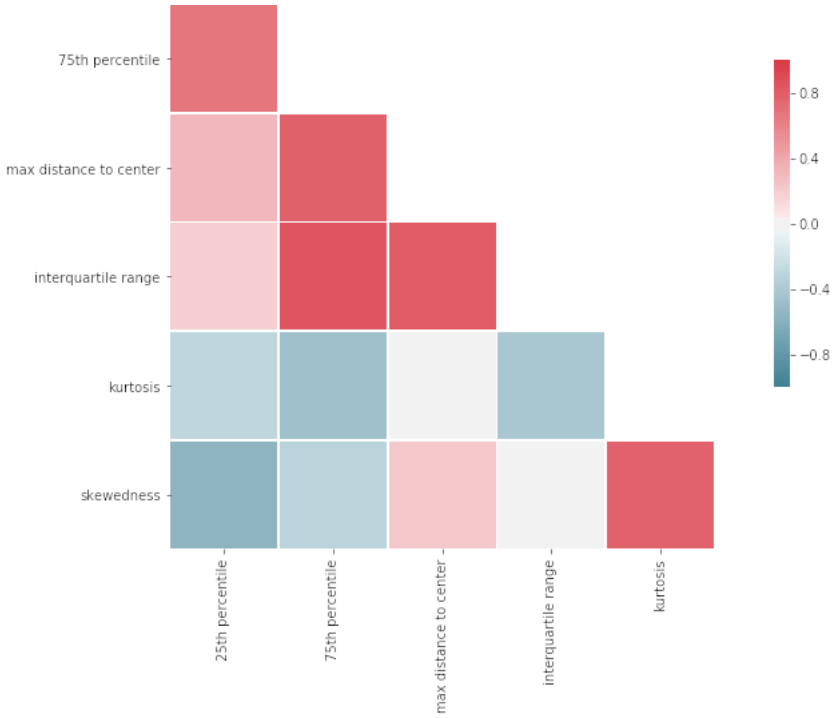

FIGURE 4.3. Correlations of selected data points

k	0	1	2	3	4	5	6	7	8	9
2	[10]	9	-	-	-	-	-	-	-	-
3	9	[6]	4	-	-	-	-	-	-	-
4	6	[5]	7	1	-	-	-	-	-	-
5	7	[5]	3	1	3	-	-	-	-	-
6	[5]	4	5	1	1	3	-	-	-	-
7	[5]	3	5	1	1	2	2	-	-	-
8	2	4	3	1	[5]	1	2	1	-	-
9	2	[5]	2	1	2	1	4	1	1	-
10	2	[5]	2	1	3	1	1	1	1	2

TABLE 3. Cluster sizesfor various k using k-means. The cluster that contains Saint Paul, MN, has square brackets around its size.

For $k \geq 4$ there consistently are five cities in the same cluster as Saint Paul, MN. This indicates that the cluster that contains this "most indicative city" is stable, independent of number of clusters. This makes $k = 4$ or $k = 5$ good selections for cluster labels, to be the smallest number of clusters from which on the algorithm appears stable for our purpose.

Running k-means multiple times for these cluster sizes consistently has the following cities in the same cluster as Saint Paul, MN, with Lexington, KY being in the same cluster about half of the time:

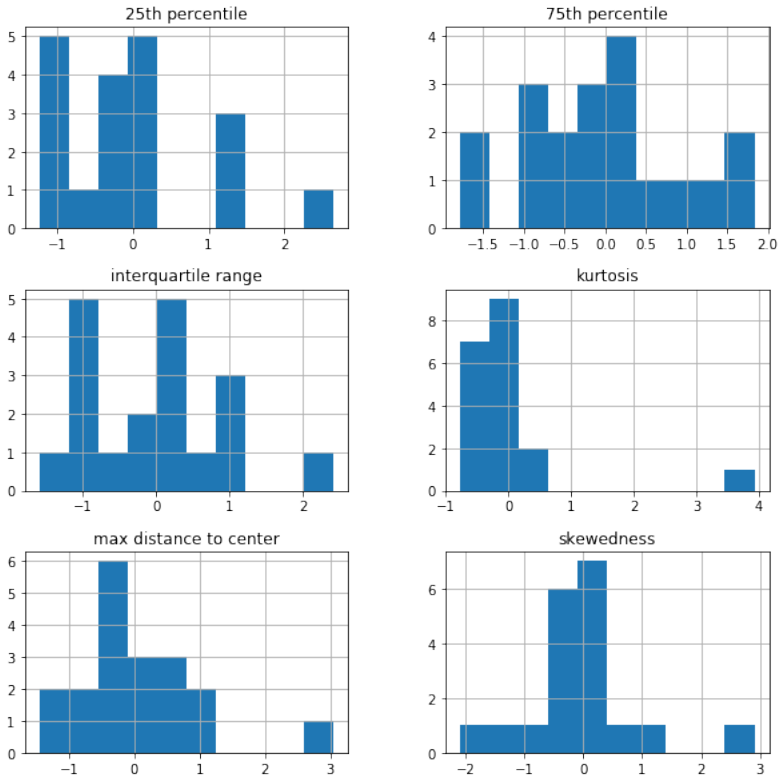

FIGURE 4.4. Feature value distributions after standardizing the data points (divide by standard deviation, center)

- Lexington, KY (2.0% indicator venues),
- Wichita, KS (2.0% indicator venues),
- Bakersfield, CA (2.0% indicator venues),
- Arlington, TX (2.7% indicator venues),
- Tampa, FL (3.2% indicator venues),
- Saint Paul, MN (3.6% indicator venues).

This makes the last three cities (Arlington, Tampa, Saint Paul) candidates for the final result.

Cross-checking against other cities with a high relative number of indicator venues, Saint Paul is highest (#1) and Tampa is third (#3). In between is Aurora, CO (#2) with 3.4% indicator venues. However, as is apparent from its distribution graph (figure 7.7) the algorithm omitted large parts of Aurora itself and went straight to foraging in Denver, CO. This gives the result a weaker emphasis on Aurora, as the premise was to search in cities between 300,000 and 400,000 inhabitants.

With this, it is concluded that the cluster that contains the most indicative city (Saint Paul, MN) is also the most desirable. It can be visually confirmed that indeed venues are spread out across the city, for each of the three candidate cities

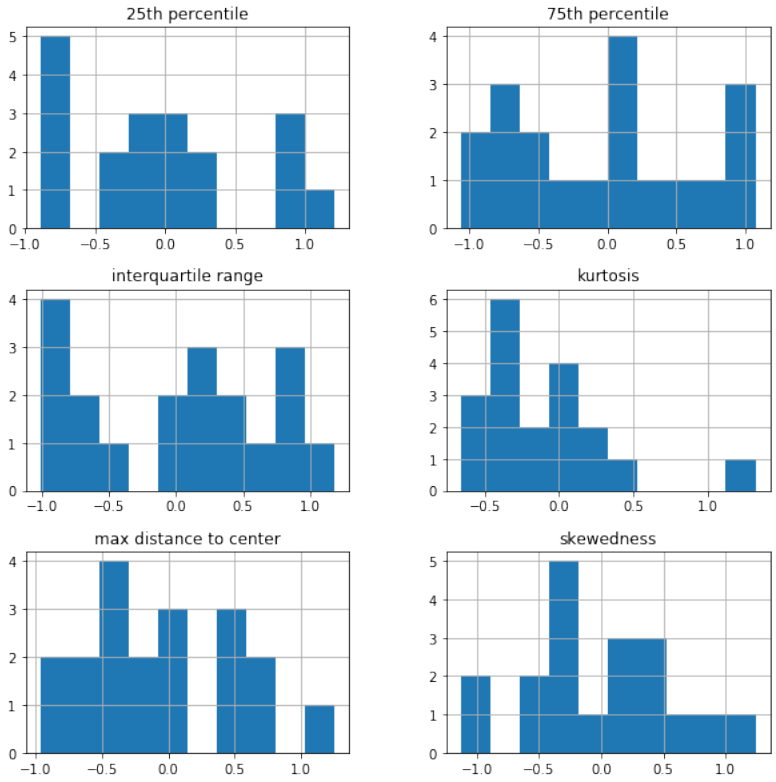

FIGURE 4.5. Final features after bounding the value range using arctan.

Arlington, Tampa, and Saint Paul. The request by the ficticious car-share company is therefore consistent in itself, and can be satisfied with the methodology here.

5. Discussion

The methodology in this investigation appears to yield a stable result, with three cities as the result (Arlington, TX, Tampa, FL, and Saint Paul, MN). There are certain biases, both customer-driven as well as driven by the methodology:

- The determination of whether or not a venue is an indicator venue is driven by substring match from an entirely subjective list. This makes the methodology here sensitive to changing this criterion. Different ways should be investigated that determine how much a city may be open to the service by the ficticious company here.
- Foursquare venues are returned by radius, but then the algorithm forages on a hex grid. This may double-count venues if they are in the same radius circle from two different hex grid coordinates. The algorithm is therefore sensitive to placement of the hex grid across the city, both in terms of absolute position as well as orientation. Different ways of placing the grid should be investigated.
- The number of Foursquare venues depends on the number of Foursquare user willing to put venues into the system, and the granularity of what these

users may consider a "venue". Since this user behavior may vary between cities, as well as between neighborhoods within a city, it intrduces bias that may not easily be quantifiable.

- The algorithm follows the highest densisty of venues within a city, which makes it sensitive to interruptions due to natural obstacles (e.g. a broad river). While people in a city may be used to crossing that obstacle routinely, the algorithm does not and instead follows into neighborhoods that may not be as frequently commuted between. It should be investigated whether changing the algorithm may have a desirable effect here, e.g. by allowing more than one initial coordinate from starters.
- By following the highest density of venues, the algorithm has the effect of foraging into neighboring cities. This is strongly apparent for Aurora, CO (where a good part of Aurora is omitted entirely and the algorithm searched Denver instead), as well as the neighboring cities Anaheim, CA and Santa Ana, CA. This is a systematic bias introduced by the methodology itself, and may warrant another investigation using a different starting point selection (not just city population), or an entirely different methodology.

6. Conclusion

The cities of Arlington, TX, Tampa, FL, and Saint Paul, MN appear to be the best fit for the selection criterion of this study, as prescribed by the ficticious car ride-share company. Due to the sensitivity of the methodology to several nontrivial biases (see section 5), further investigation should be done as recommended prior to establishing business.

References

[1] U.S. Census 2017 estimates for city size: https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?src=bkm (retrieved 1 March 2019).
[2] Wikipedia aggregate of 2017 U.S. Census estimate for city size: https://en.wikipedia.org/w/index.php?title=List_of_United_States_cities_by_population&oldid=883568308 (retrieved 10 March 2019).
[3] Jupyter notebooks with algorithm implementation and raw results: https://github.com/koeplinger/sandbox-ibm-ds/blob/master/ibmDsSpecCourse9capstone.ipynb

7. Appendix: City venue distribution figures

Screenshots from figures created using Python folium.

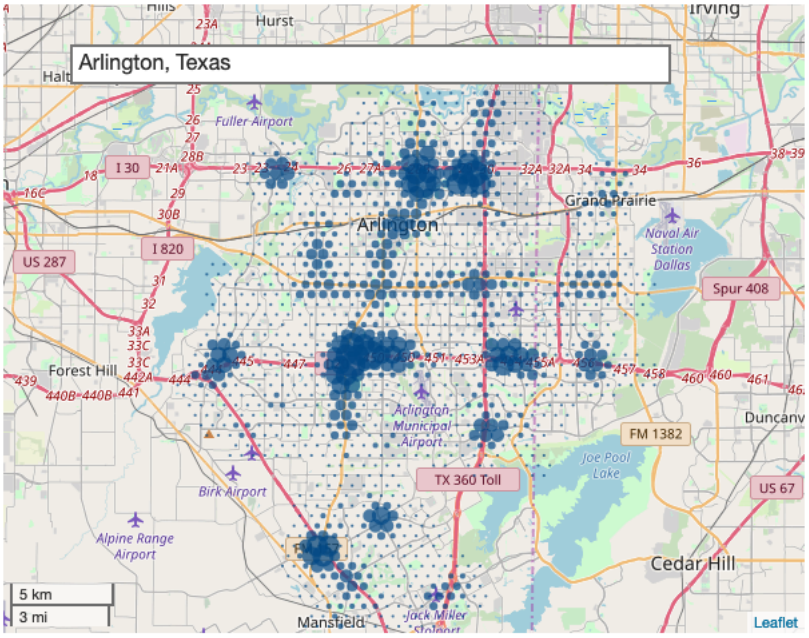

FIGURE 7.1. Arlington, TX

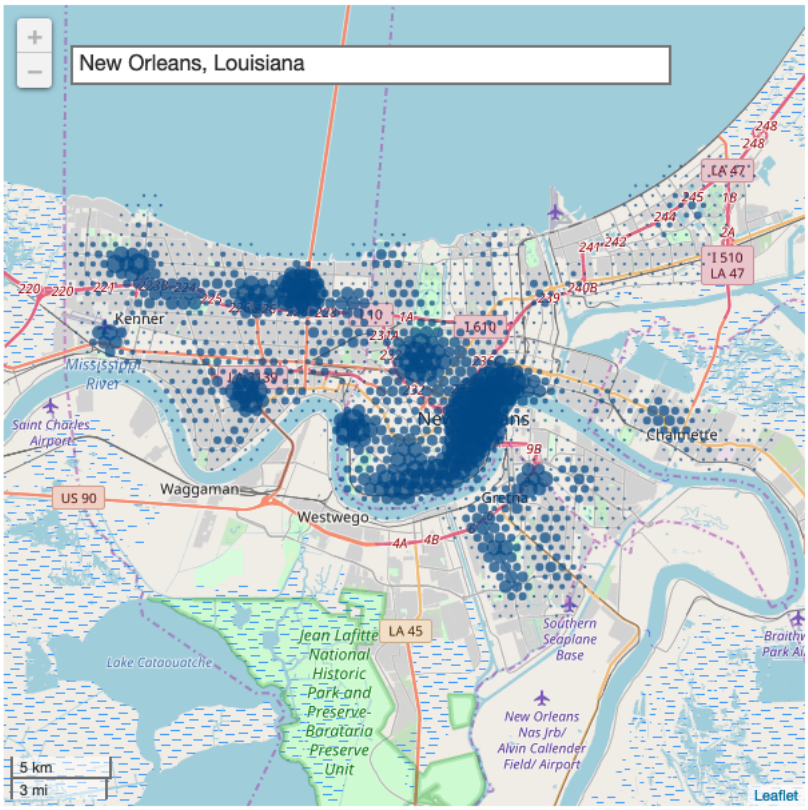

FIGURE 7.2. New Orleans, LA

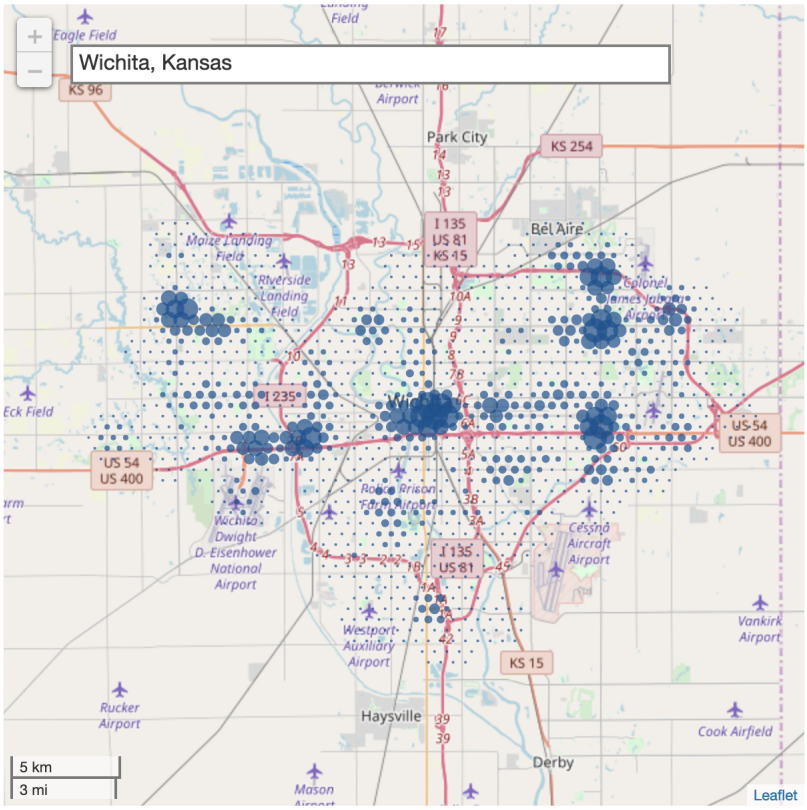

FIGURE 7.3. Wichita, KS

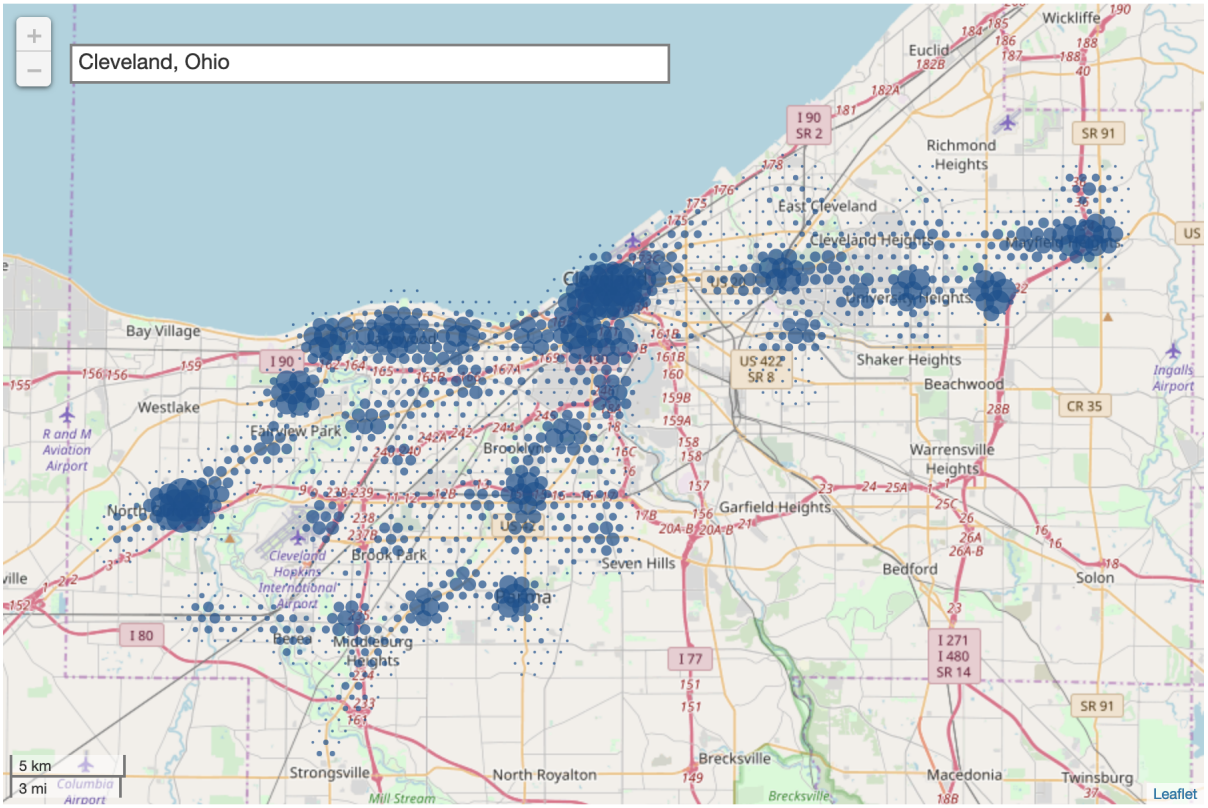

FIGURE 7.4. Cleveland, OH

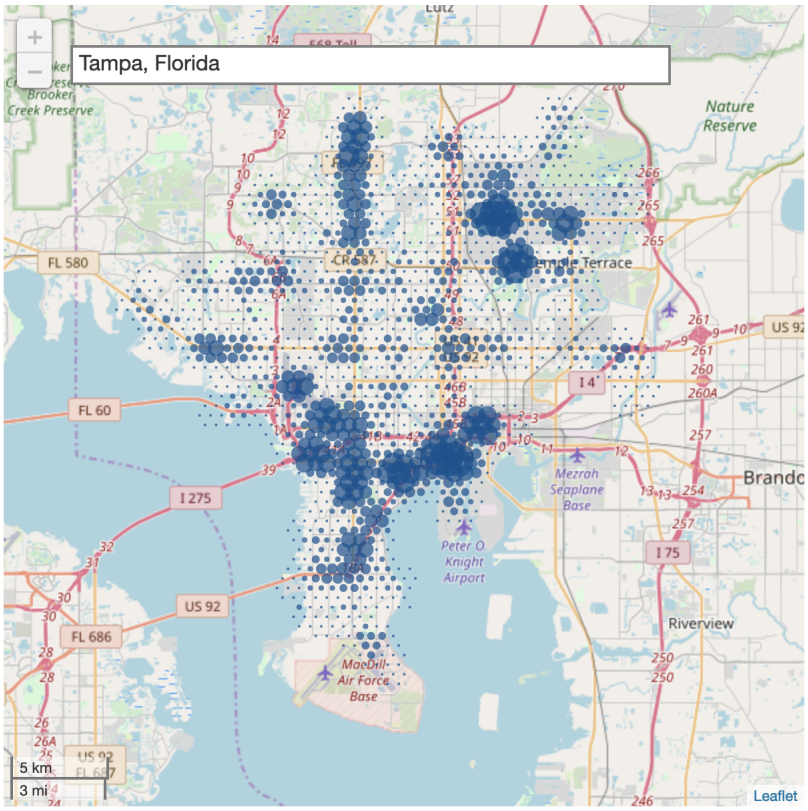

FIGURE 7.5. Tampa, FL

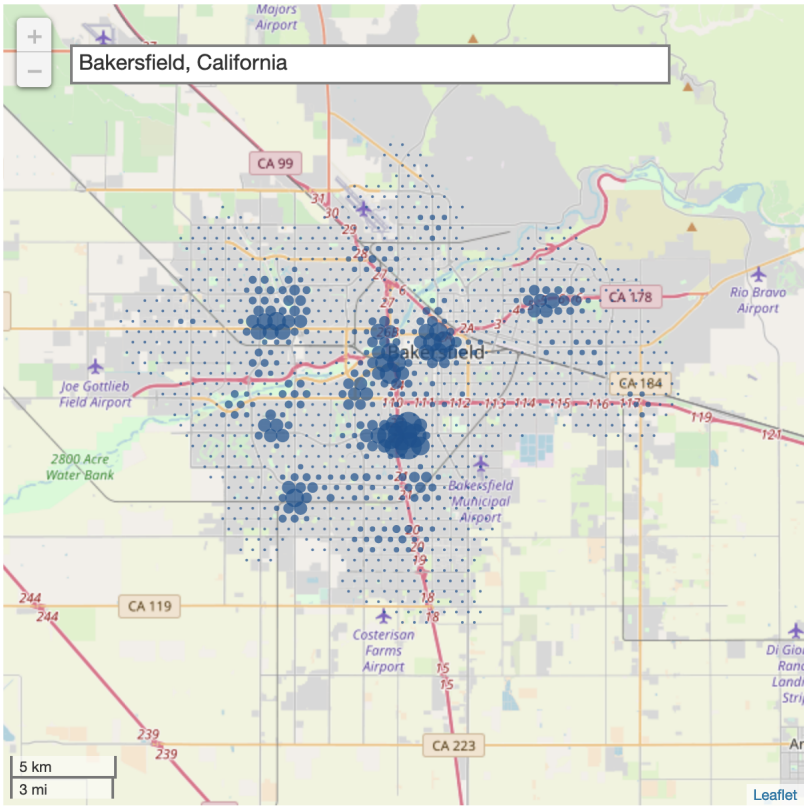

FIGURE 7.6. Bakersfield, CA

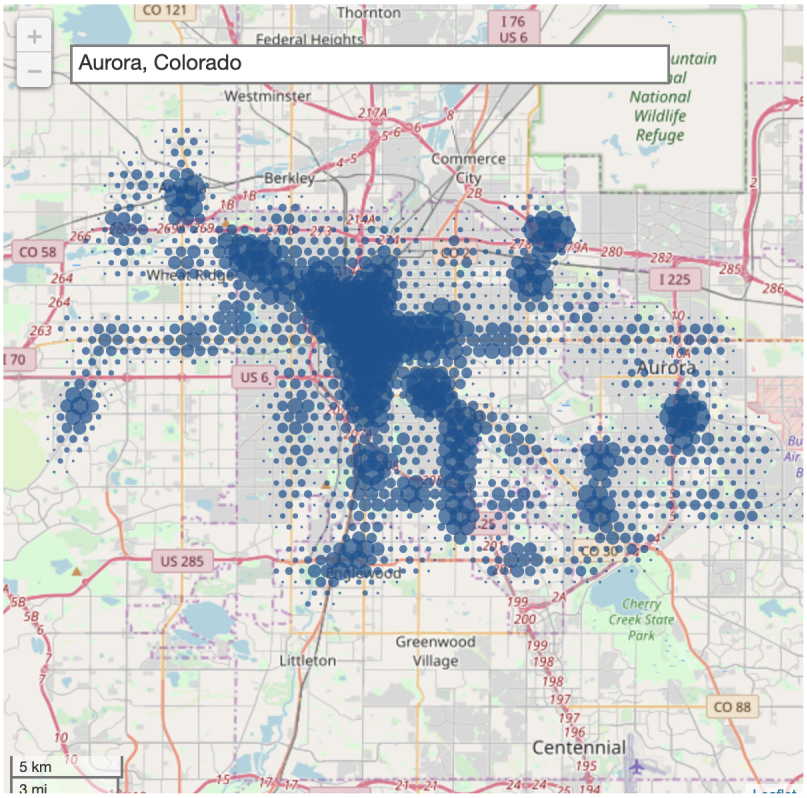

FIGURE 7.7. Aurora, CO

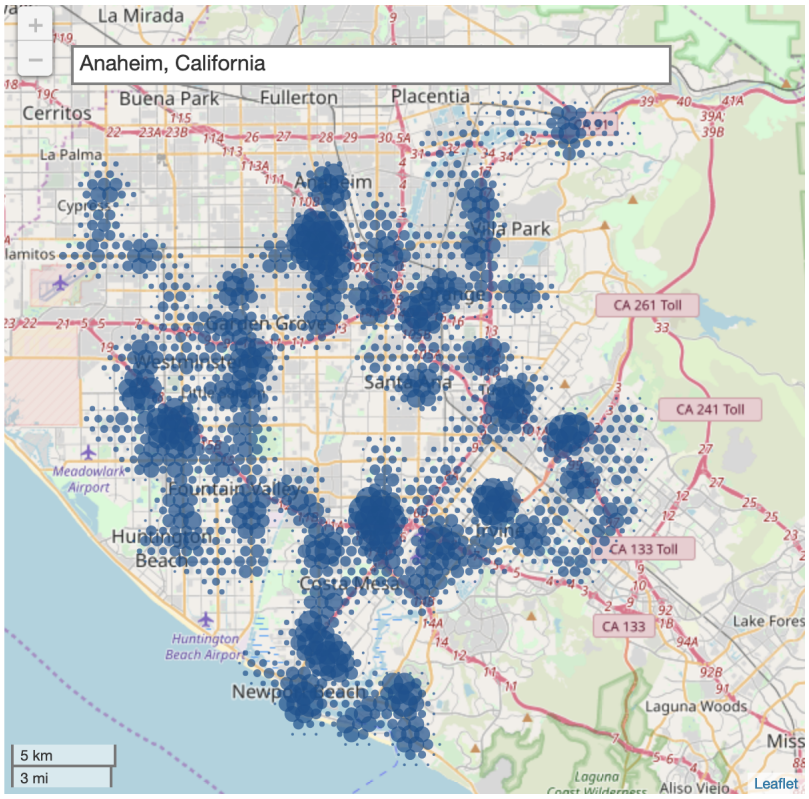

FIGURE 7.8. Anaheim, CA

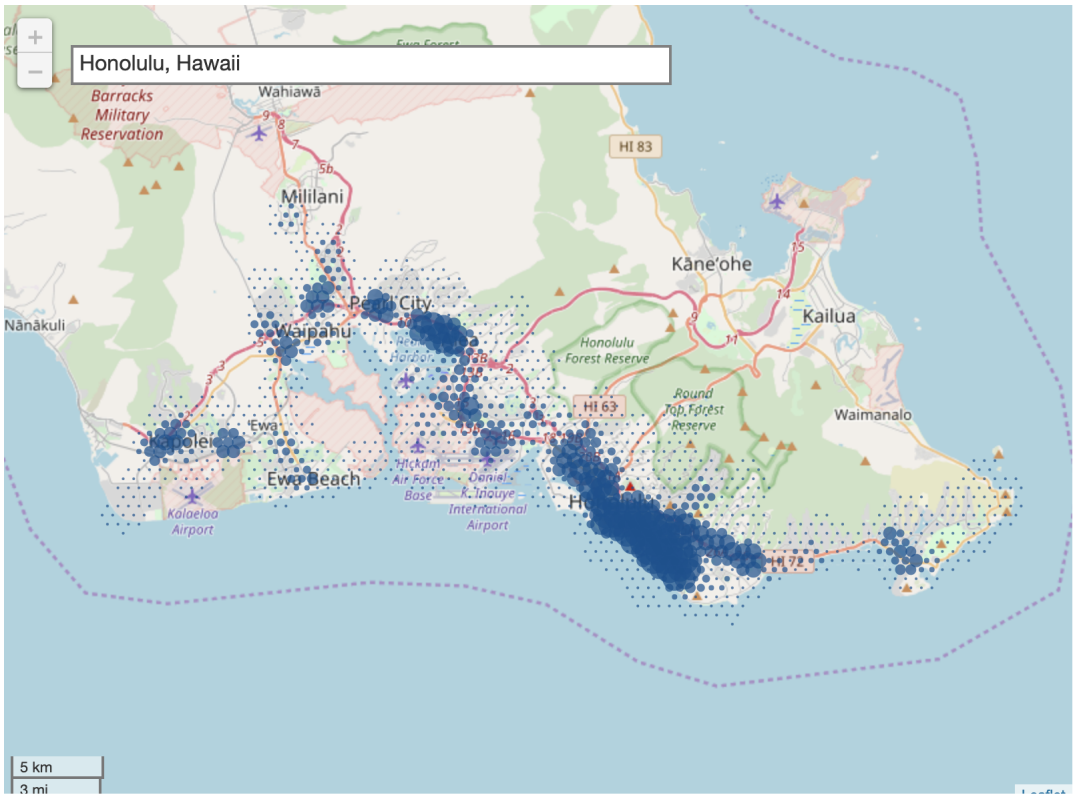

FIGURE 7.9. Honolulu, HI

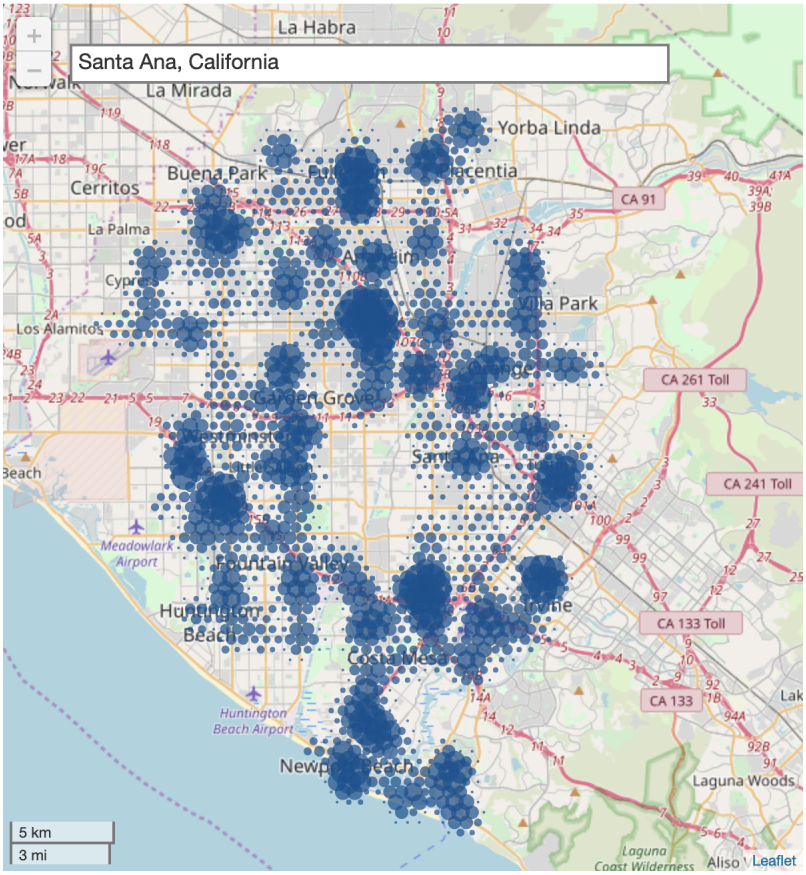

FIGURE 7.10. Santa Ana, CA

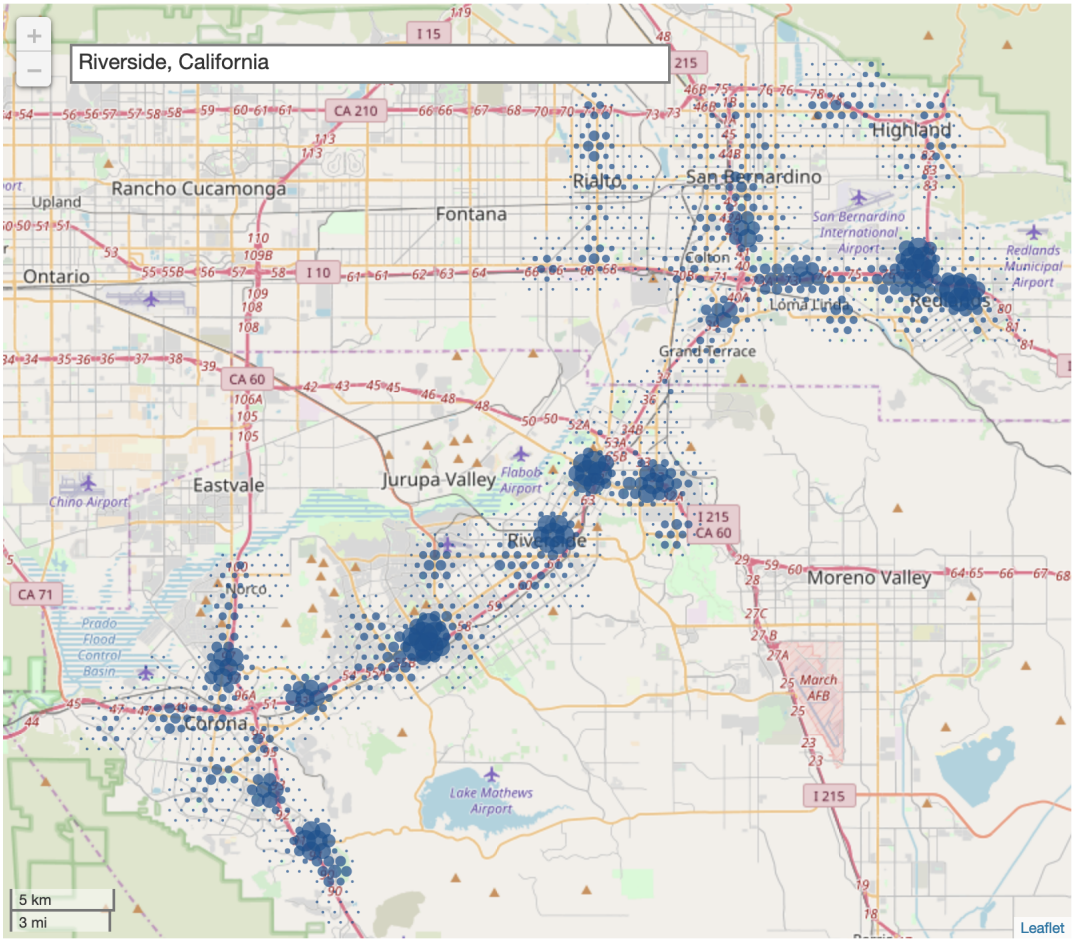

FIGURE 7.11. Riverside, CA

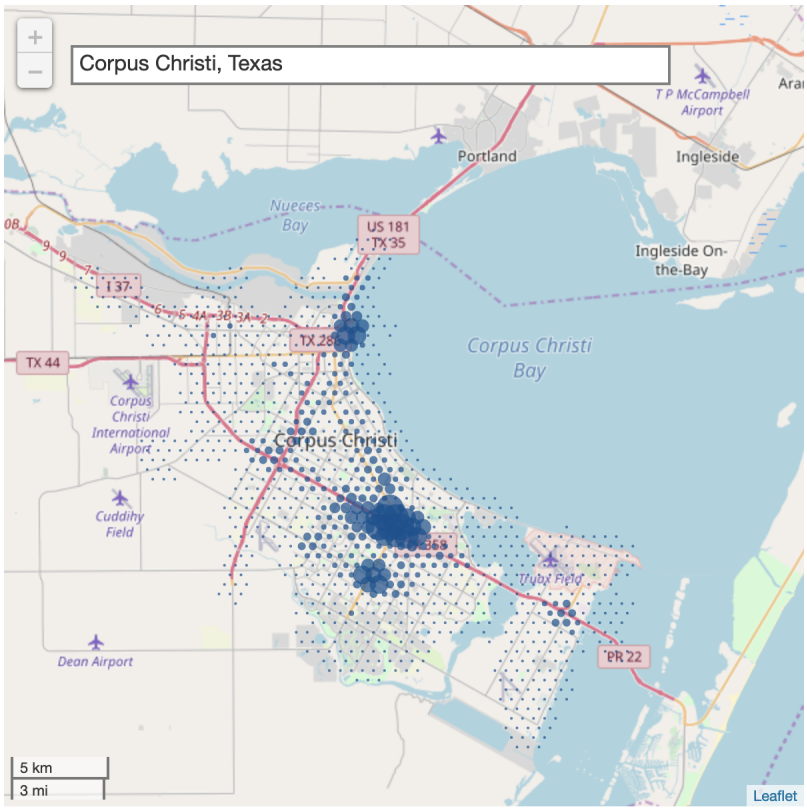

FIGURE 7.12. Corpus Christi, TX

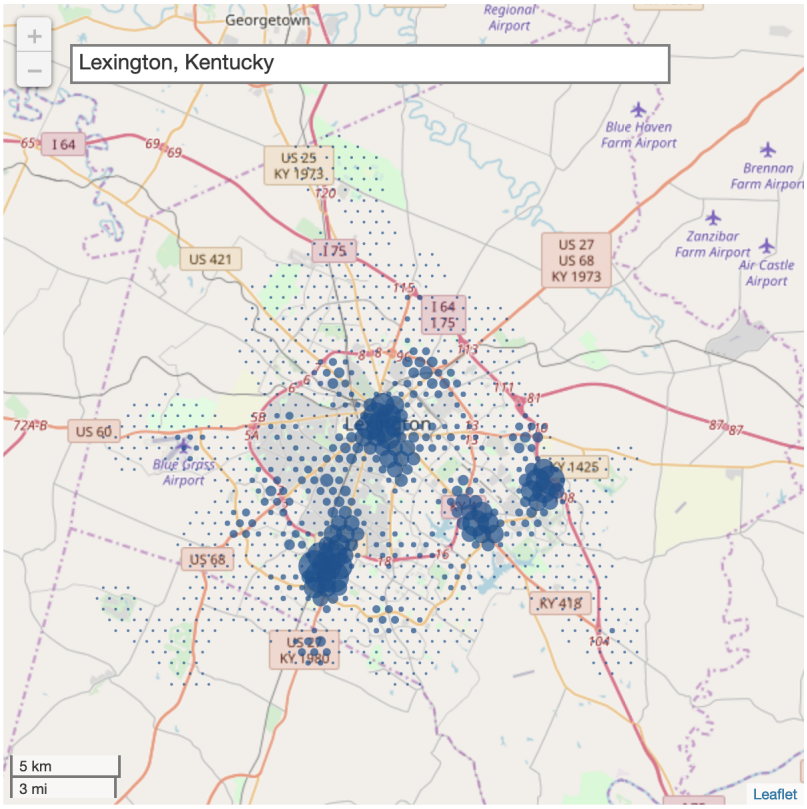

FIGURE 7.13. Lexington, KY

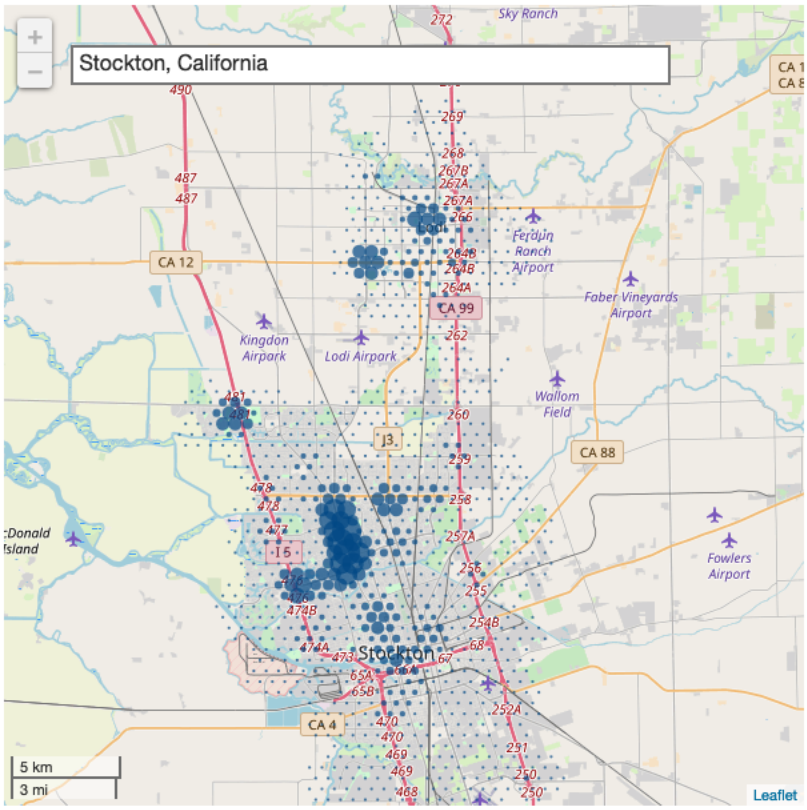

FIGURE 7.14. Stockton, CA

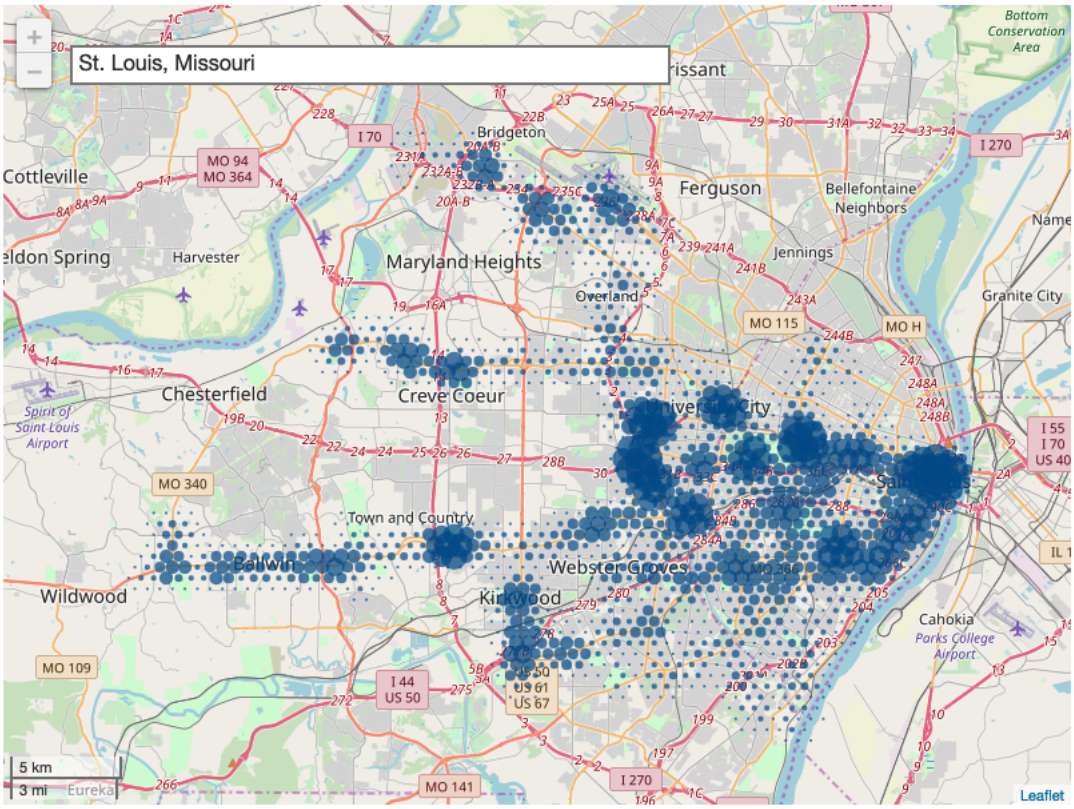

FIGURE 7.15. St. Louis, MO

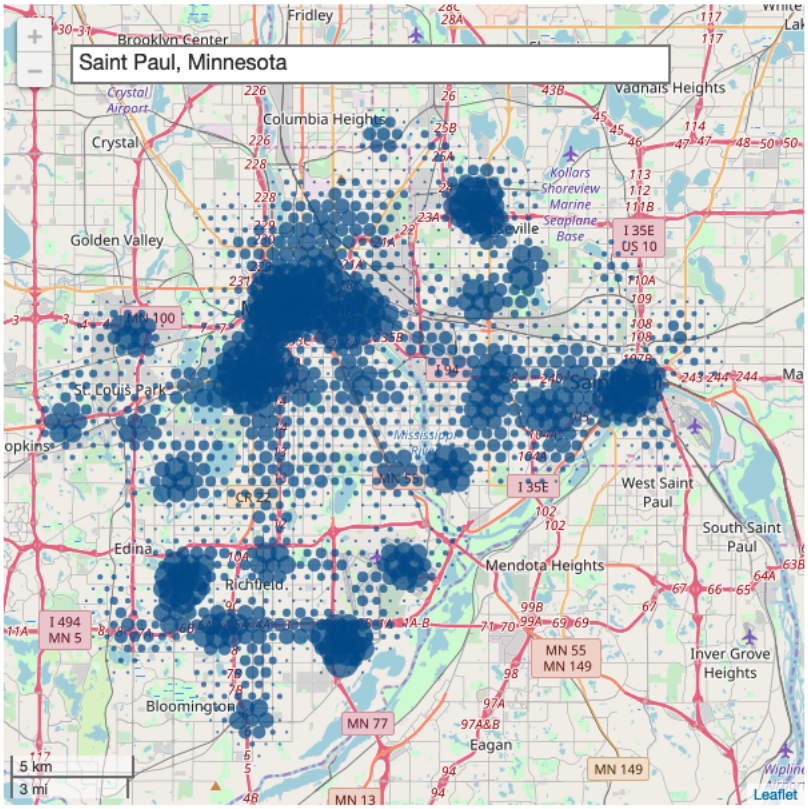

FIGURE 7.16. Saint Paul, MN

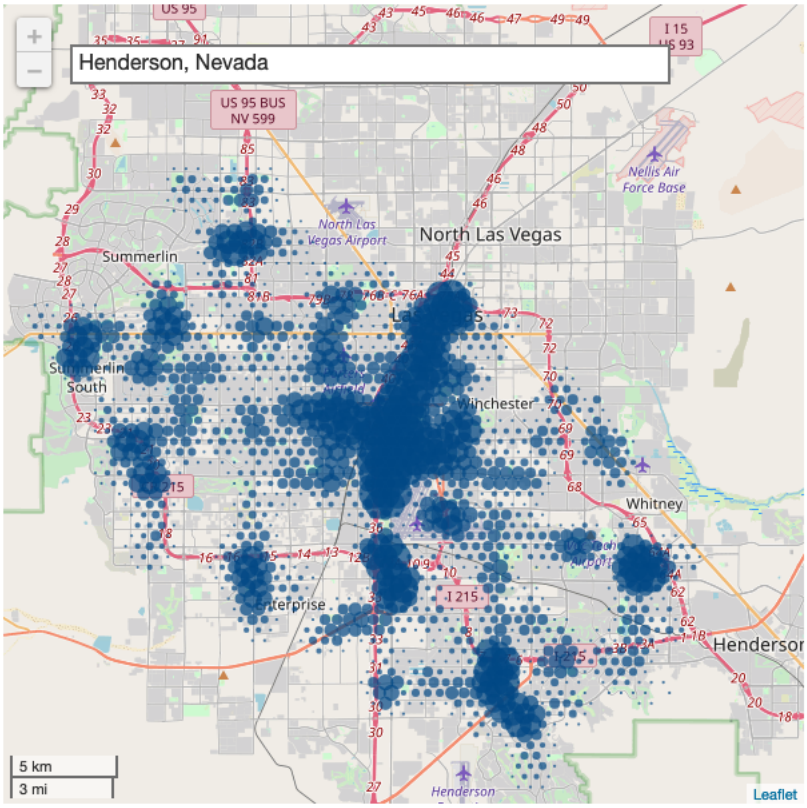

FIGURE 7.17. Henderson, NV

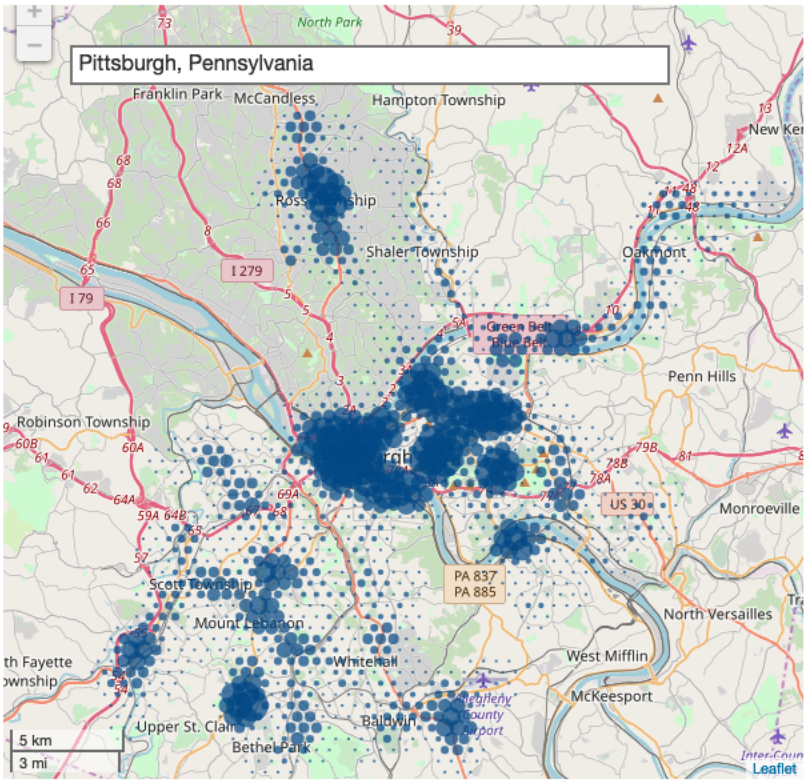

FIGURE 7.18. Pittsburgh, PA

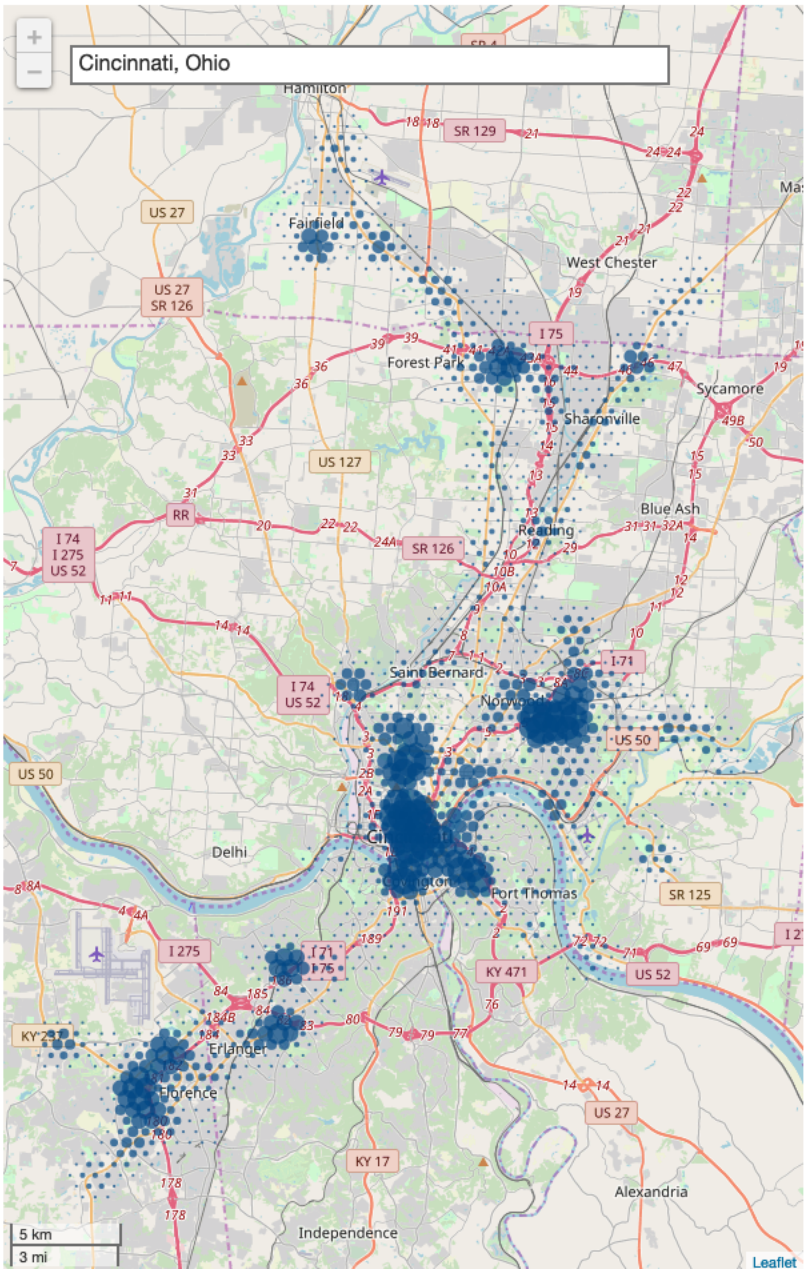

FIGURE 7.19. Cincinnati, OH