IBM Data Science Professional Certification **June 2020**

Capstone Project Report: IBM Data Science

Course: Applied Data Science Capstone *Author: Rishabh Jain*

1 Introduction

Foursquare API [2] is a worthwile investment of time and effort. The purpose of this project is to use this API to solve a real world problem. As such, I have attempted to find the most optimal locations for opening a restaurant in the city of Toronto. This problem has many layers which require deliberate inquisitiveness. Deciding what data to look at, how to obtain said data, what analysis to perform, what hypothesis to check, which inferences to obtain, what recommendations to make, etc. are all worthwhile questions to ask and I will attempting to duly answer all these questions through this report and the attached (.ipynb) notebook. (see [3])

Anytime any end-user wants to open a new business (a restaurant in this case), a primary question associated with the task is where the business should be opened? Through this model, I attempt to provide a proof of concept that certain standard analysis can always be done no matter which business one is trying to open, should the corresponding data be available.

My model looks at all the neighborhoods in Toronto to come up with the neighborhoods best suited to particular end-user needs. For each neighborhood, I look at parameters like population, area (sq. km), density, per-capita income, number of pre-existing restaurants, etc. to come up with an attribute proportional to the "potential" of each neighborhood. Furthermore, I look at how some of these places are similar/different from each other through clustering and how some clusters might be good for some people but nit for others and vice-versa!

I will first describe the data I used along with its sources in section 2. I will then go on to describe the methodology I have used for developing this model and all the results I obtained along the way in section 3. I will then discuss potential ramifications of all the obtained results in section 4, giving deeper insight into the meaning behind all these numbers. I'll end with a brief conclusion of what I believe to have accomplished through this endeavor in section 5

2 Data

To get most of the neighborhood characteristics, I used [1]. The table available on this page tells a great deal about the attributes of each neighborhood. Then, in order to obtain geo-spatial data about all these neighborhoods, I used the open source python library (geopy.geocoders.Nominatim). This is pretty intuitive to use and had most of th data I was looking for. Furthermore, to obtain data about restaurants in each of these neighborhoods, I used the Foursquare API. However, since I had a personal account which allows only 450 premium calls a day, I hd to rely upon data available from regular calls to make the model. As such, I couldn't include parameters like average restaurant rating, number of likes, tips, etc. Anyhow, since I am simply providing a proof of concept, the parameters obtainable from regular calls shall suffice!

3 Methodology & Results

1. First, I downloaded the data from [1] into a pandas dataframe, followed by some data wrangling to make the data cleaner and to do away with the data I didn't need.

2. I then used the (geopy.geocoders.Nominatim) library (see [4]) to obtain cordinates for each neighborhood in the dataframe followed by some more wrangling to combine this data with the original dataframe.

Name	Population	Land area (km2)	Density (people/km2)	Average Income	latitude	longitude
Agincourt	44577	12.45	3580	25750	43.785353	-79.278549
Alderwood	11656	4.94	2360	35239	43.601717	-79.545232
Alexandra Park	4355	0.32	13609	19687	43.650787	-79.404318
Allenby	2513	0.58	4333	245592	14.595343	121.035220
Amesbury	17318	3.51	4934	27546	42.857954	-70.930092

Figure 1: Preliminary Dataframe (.head())

3. I then created a pre-liminary map using the folium library (see [5]) to spatially show the distribution of all the neighborhoods.

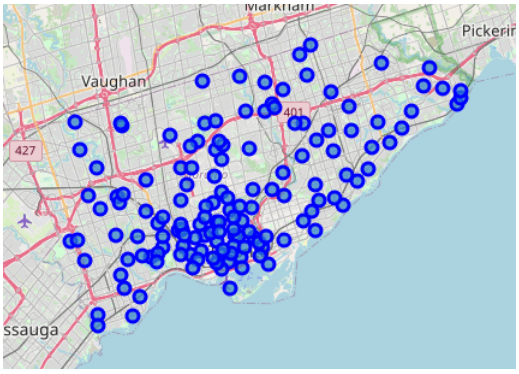

Figure 2: Preliminary Map Snippet

4. I used Foursquare API to obtain venue information at all the neighborhoods using the key word 'restaurant'. Since this is a regular request call, the number of calls made wasn't a major concern.

5. A lot of data wrangling later, I came up with a compiled dataframe inculcating all the important information up to this point.

6. I created a new attribute for each neighborhood by the name of "potential". I used the following formula to calculate potential for each neighborhood:

$$Potential = (\frac{Population}{Area}) \times (per - capita\ income) \times (\frac{1}{pre - existing\ restaurants})$$

7. Here is what the combined dataframe looked like at this point: (see figure 3)

Name	Population	Land area (km2)	Density (people/km2)	Average Income	frequency	potential	latitude	longitude
High Park North	22746	2.18	10434	46437	1	4.845237e+08	43.657383	-79.470961
Old Mill/Baby Point	4010	1.07	3748	110372	1	4.136743e+08	43.649826	-79.494334
The Kingsway	8780	2.58	3403	110944	1	3.775424e+08	43.647381	-79.511333
Bracondale Hill	5343	0.62	8618	41605	1	3.585519e+08	43.676195	-79.428016
Humewood–Cedarvale	27515	3.19	8624	40404	1	3.484441e+08	43.690248	-79.422097
Wallace Emerson	10338	0.88	11748	25029	1	2.940407e+08	43.666733	-79.446478
Forest Hill	24056	4.35	5530	101631	2	2.810097e+08	43.693559	-79.413902
Crescent Town	8157	0.40	20393	23021	2	2.347336e+08	43.695403	-79.293099

Figure 3: Dataframe with 'potential' (.head())

8. I then created a folium map to get a spatial glimpse of the potential in each neighborhood. For this purpose, I created a bubble map with the radius of each bubble proportional to the restaurant opening potential in that neighborhood. (see figure 4)

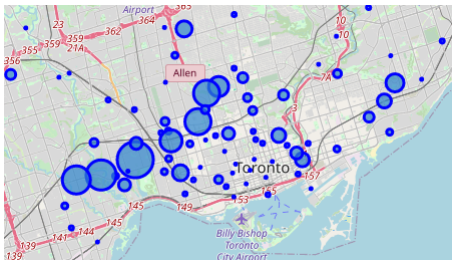

Figure 4: Map Snippet with each neighborhood's 'potential' as bubble radius

9. Upon some investigation, a problem surfaced. One apparent shortcoming of this model was the fact that 'potential' was very sensitive towards neighborhoods with only 1 or 2 documented restaurants, since even if only one or two more restaurants were to be added in these places, the potential would have halved. This made the model very vulnerable if there was even one restaurant undocumented in these places.

10. So, to counter this issue and add a scent of robustness to the model, I decided to consider only those neighborhoods which had more than 10 documented restaurants. This would make 'potential' significantly less sensitive even if more restaurants were documented at a later stage.

11. After making this change, another problem became apparent. Land area seemed to dominate 'potential'. A way around was to scale all parameters between 1 and 2 and then recalculate 'potential'. That would give each parameter equal opportunity to affect the final potential. (see figure 5)

Name	Population	Land area (km2)	Density (people/km2)	Average Income	frequency	potential	latitude	longitude	Land area (km2) inv	Population pro	Average Income pro	frequency inv
Newtonbrook	36046	8.77	4110	33428	10	4.208387	43.793886	-79.425679	1.003740	1.806241	1.160616	2.000000
Harbourfront / CityPlace	14368	1.87	9228	69232	14	3.572285	43.640080	-79.380150	1.050434	1.313884	1.579121	1.639098
Niagara	6524	0.55	11862	44611	11	3.298033	43.644075	-79.408698	1.192869	1.135729	1.291332	1.885167
Yorkville	6045	0.56	10795	105239	27	3.223104	43.671386	-79.390168	1.189265	1.124850	2.000000	1.204678
Bay Street Corridor	4787	0.11	43518	40598	29	3.198895	43.667342	-79.388457	2.000000	1.096277	1.244424	1.172414
Agincourt	44577	12.45	3580	25750	18	3.081097	43.785353	-79.278549	1.000000	2.000000	1.070869	1.438596
Milliken	26272	7.19	3654	25243	13	2.901275	43.823174	-79.301763	1.006521	1.584251	1.064943	1.708502
Christie Pits	5124	0.64	8006	30556	11	2.731306	43.667139	-79.422766	1.164493	1.103931	1.127046	1.885167

Figure 5: Dataframe Snippet with updated 'potential'

12. with that in mind, I re-calculated the potential using scaled parameters. This time results made a lot more strategic sense. I then went on to recreate the previous folium map, only this time with more accurate 'potential' values. (see figure 6)

Figure 6: Map Snippet with updated 'potential'

13. Now that we have individual data points were readily available, I could find optimal locations as per my needs from the analysis above. Extending this idea, I created clusters of all these locations to find out which of them were similar and in what way. This would further help in combining analytical findings with geospatial data!

14. For this purpose of clustering, I used the KMeans library in python (see [6]) to assign cluster labels to each neighborhood. I used 4 clusters with an n_{init} value of 12. To check out the allotted cluster labels, see figure 7

Name	potential	latitude	longitude	Land area (km2) inv	Population pro	Average Income pro	frequency inv	label
Newtonbrook	4.208387	43.793886	-79.425679	1.003740	1.806241	1.160616	2.000000	1
Harbourfront / CityPlace	3.572285	43.640080	-79.380150	1.050434	1.313884	1.579121	1.639098	2
Niagara	3.298033	43.644075	-79.408698	1.192869	1.135729	1.291332	1.885167	2
Yorkville	3.223104	43.671386	-79.390168	1.189265	1.124850	2.000000	1.204678	3
Bay Street Corridor	3.198895	43.667342	-79.388457	2.000000	1.096277	1.244424	1.172414	0
Agincourt	3.081097	43.785353	-79.278549	1.000000	2.000000	1.070869	1.438596	1
Milliken	2.901275	43.823174	-79.301763	1.006521	1.584251	1.064943	1.708502	1
Christie Pits	2.731306	43.667139	-79.422766	1.164493	1.103931	1.127046	1.885167	2
Church and Wellesley	2.495469	43.665524	-79.383801	1.192869	1.291830	1.210001	1.338346	0

Figure 7: Dataframe Snippet with cluster labels

15. I then re-mapped all the neighborhoods, this time color coded with respect to their cluster labels. Then, looked at each neighborhood cluster individually to get an idea of what it was that made them similar and how those characteristics eventually affected their potential. (see figure 8)

16. Finally, as a last bit of analysis, I compared the statistics of each cluster by wrangling them into a common dataframe. This provided some useful insights and findings which have been been disussed in section 4

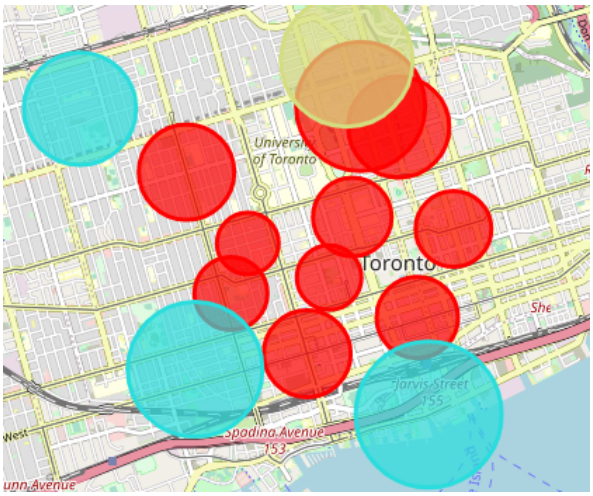

Figure 8: Map Snippet with color scheme w.r.t cluster labels

4 Discussion

As per figure 5, if one wishes to open a restaurant anywhere in Toronto, Newtonbrook is the place for them. It has a significant population with decent per capita income and relatively low number of pre-existing restaurants!

However, if one insists on opening a restaurant in downtown Toronto because of tourist attractions and networking, Harbourfront(City Place) is the optimum location. If one wants to open a restaurant with lowest risk of failure, i.e. a place where many restaurants are known to flourish, Bay street corridor is the optimum location. Basically, any end-user using this model can apply their own filters on the final results according to their needs and wishes and the model will give them the optimum location leveraging which parameters they value and which they don't. Cool!

Post clustering, I got the following dataframe: (see figure 9)

	mean	Cluster 0	Cluster 1	Cluster 2	Cluster 3
label	-	0	1	2	3
count	17	10	3	3	1
Population	11814.1	6188.4	35631.7	8672	6045
Land area (km2)	2.20471	0.545	9.47	1.02	0.56
Density (people/km2)	11286.9	14064.2	3781.33	9698.67	10795
Average Income	43720.4	40918.8	28140.3	48133	105239
frequency	24.7059	31.6	13.6667	12	27
potential	2.58268	2.089	3.39692	3.20054	3.2231

Figure 9: Combined Dataframe with statistics for all clusters

After scrutinizing figure 9, following inferences can be obtained:

1. Group 0: Places with low population, extremely low area, below average per-capita income with large number of pre-existing restaurants. Overall, these places have pretty low potential for opening up a new restaurant.
2. Group 1: Places with very high population & even higher area, low per-capita income with low number of pre-existing restaraunts. Overall, these places have pretty high potential for opening up a new restaurant. Having said that, these places are located in relatively remote areas and thus networking and transportation might be an issue.
3. Group 2: Places with slightly below average population & area, above average per-capita income with low number of pre-existing restaraunts. Overall, these places have pretty high potential for opening up a new restaurant.
4. This group only has one place, an outlier with extremely low population & area, exceptionally high per-capita income with large number of pre-existing restaurants. Though this place does have a high potential, this analysis might not model this place well, you know, this place being an outlier and everything.

5 Conclusion

As I said at in section 1, the aim of this project is to serve as a proof of concept for this kind of analysis. Through the example of restaurant location scouting, I have shown how one can leverage Foursquare API's resources to compare different locations to obtain either one optimum location or a cluster of similar close-to-optimum locations inculcating end-user values and needs. As far as the case of finding an optimum restaurant location is considered, in terms of further possible improvements, one can look at including average restaurant rating, number of likes on each restaurant, tips, etc. to calculate 'potential' for each location. However, this further analysis couldn't be done with my personal account on Foursquare. If someone with a premium account would like to venture forth with this aspect, be my guest!

References

[1] $https://en.wikipedia.org/wiki/demographics\_of\_toronto\_neighbourhoods$

[2] $https://foursquare.com/$

[3] $https://dataplatform.cloud.ibm.com/analytics/notebooks/v2/ce82113e-fd12-423f-84eb-fd6e1b9fc243/view?access\_token=fba30a82f533337e811ec0c21e98e7d0ee980b06f6fe83f050da823f74d954df$

[4] $https://geopy.readthedocs.io/en/stable/$

[5] $https://python-visualization.github.io/folium/$

[6] $https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html$