The problem

Even the greatest cities in the world lack a certain balance when it comes to the geographical distribution of their citizens. There are always those neighborhoods where everybody wants to live in, and the real estate is highly valued. On the other hand, there are neighborhoods that are less populated and usually considered as the choice of those who cannot afford the former. This imbalance creates a variance in population density, making some parts of the city overcrowded while others have plenty of space – a major problem when it comes to infrastructure utilization and investments.

Although there are many forces in play here (among which are geographical properties, social economics, pollution etc), one of the key factors is the level of services\goods available in the neighborhood. We all need some essentials in our day-to-day life, and we need them close by, preferably in a walking distance. It's true that we can take a bus or the metro and get there after spending another 20 min (if the public transport is efficient). However, with people being so busy nowadays and so used to instant digital response – a service or a product offered even 20 min away might be too demanding.

Now you might think to yourself, "this is a free market we're talking about! private businesses will seize the opportunity of having some more revenue if they only get the chance to do so. If they are not expanding their operations to these neighborhoods, it means they see no potential there". There's some truth in this saying but the problem is a bit more complex in my opinion and suffer from the 'chicken and egg' effect. Not enough people -> services are not extended -> not enough people in that place.

If we would have been able to list those core services and products and segment the different neighborhoods according to what is missing in them - It would have been possible to offer that information to various stakeholders. They, in turn, can Influence the neighborhoods' value and attract more citizens to it.

Stakeholders

Municipalities are the primary stakeholder in this project. Knowing what exactly makes a neighborhood less desirable is the key in turning it around. For example, if a coffee shop is missing - a tax benefit can be initiated; if it's a park that's missing - the city can sponsor the construction. Naturally, making a neighborhood more desirable is of primary importance to the city. Another stake holder is the private business. Knowing what's missing can give a business an edge in earning customer's loyalty ahead of competition. Again, if it's a potential coffee shop the neighborhood lacks, Starbucks would be happy to know that. Once more services and products become available, the wheel will start turning and the chicken & egg cycle will be broken. This happens naturally all the time, all around the world. Data can make it happen faster.

Data

The city of choice is Toronto. I will be using the same public dataset downloaded from Wikipedia in week 3 peer-graded assignment. That, in order to map all neighborhoods in the city and their GPS coordinates. Each row in the dataset lists a neighborhood, the borough\post-code it belongs to and its coordinates. For example, Rough neighborhood is located in Scarborough under post-code M1B. I will process and enrich this data later on to use it as samples for clustering, since the problem is that of grouping neighborhoods according to the services around them.

The complementary information that is missing are the different services offered in every neighborhood. Here, Foursquare free engine will provide the requirement. It will list every relevant venue, its category and the coordinates. The categories labels are crucial to the project since it will not only determine the value of each sample (borough\post code \ neighborhood) but will allow us to study the variety of categories existing in the city and determine which ones are to be considered as critical services. As an example, we see that M1B post-code has only one venue close to it and it belong to the "restaurants" category: Wendy's at (*43.807448*, *-79.199056*). This is basically the DNA of M1B. Also, since restaurants of all kinds are the most popular category in the city, with more than 500 of them around, it's clear that this needs to be considered as a crucial service.

Methodology

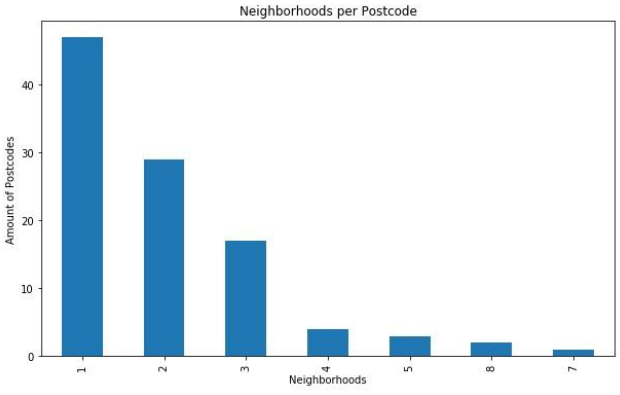
After downloading the public data of Toronto's neighborhoods, I am facing with the most fundamental question: what the basic building block in terms of data. Meaning, what data points will I feed the clustering algorithms with? the answer is comprised of two components:

a. The geographical resolution – Borough or **Post code** or Neighborhood?

b. The service resolution – Venue or **Venue category?**

Exploratory data

A further data analysis is required to determine this. After cleaning the data and removing invalid entries, it is can be seen that in terms of geographical resolution, there are 11 boroughs, 103 post codes and several hundreds of neighborhoods overall. Intuitively, boroughs are too big & few to choose while neighborhoods are too many. Therefore, the natural candidate is the post code. Taking a look at the ratio of neighborhoods to post codes strengthens the assumption. Most post codes include 1-3 neighborhoods, making it a good choice since borough to post code ratio is much lower (~1:10) and there are too



many neighborhoods to receive significant results. Especially when aiming for minimal scanning distance in which case each neighborhood will have very few services around it.

As for the service resolution, here the statistics is clearer. When scanning post codes using a 500-meter radius (my definition of short walking distance) and having a 150 venues per post code limit, foursquare generates 2,234 venues in 274 categories. The choice is clear. I choose the venue categories as the service resolution. Moreover, since I am looking for essential services only, I need to reduce category amount even further. I did this by converting all restaurant types (52) into one category and merging similar services such as coffee shops and Café or Gym and Gym fitness clubs. On top of that I've set a threshold to eliminate relatively rare services which have less than 15% availability in the average post code. After conducting further manual picking of the 10 most essential services, the final data point vector looks as follows:

(Post code, Restaurant, Park, Hotel, Grocery shop, Pub, Pharmacy, Bank, Coffee, Pastries, Workout)

As for the values of the vector, it's the proportion of a given service category out of the total amount of services in the post code. All proportions in a given post code sum up to 1, making them also probabilities of encountering the specific service. One final interesting statistic before I begin with the algorithms is the amount of services in each post code. The statistics show that there's a great variance between different post codes. There are those with about 100 services in them while there are also ones with less than 5. This is important to notice since the entire business problem revolves around the argument that the city is imbalanced in terms of geographical and service distribution.

Machine learning techniques

I started off with K means, being one of the most popular and efficient clustering algorithms out there in use. The number of clusters was determined using the silhouette score due to the fact that the popular elbow method is not reliable enough and does not address the distance to the cluster center, only the between-group variance. The silhouette score was high enough for around 10 cluster-target and therefore the parameter was set to 10 (it's symbolic, could have been set to 11 or 9 easily).

Despite its popularity, K means has some drawbacks, the primary of which is the Euclidean distance metric and its use in the kernel function. The data is split halfway between cluster means leading to overall "softer" clustering. Meaning, weaker correlation between data points within each cluster. This might show the big picture, but - too vaguely. Knowing this, I turned next to run some density-based algorithms. DBSCAN was the first choice. It works differently in concept, advancing from point to point and looking to maximize intra-cluster correlation of data points without having predefined number of clusters to fill assign to. This should perform the clustering in a "harder" manner if the min points per cluster parameter is low enough (which I made sure happens through trial and error). "Harder" means that the common denominator in each group is distinct, allowing for focused conclusions to be drawn.

The second choice in density-based algorithms was Mean Shift. I picked it because I wanted to neutralize the input parameters as much as possible. I might have wrongly chosen them in DBSCAN after all. While DBSCAN requires both a distance (ε) and a minimal amount of points to form a dense region, Mean Shift needs only the distance to be used by the kernel function. That way, I will utilize the density approach with minimal setup intervention.

After applying K means and two density algorithms, I should explore another direction combining both. On the one hand, we need to run a "soft" clustering algorithm to be able to group data points with weaker correlation and see the big picture. On the other, we need to be able to form clusters with distinct patterns in them – whatever size they are. The chosen algorithm is GMM (Gaussian Mixture Model), providing both flexibility in terms of cluster shape and also being inherently structured. This can be seen as generalizing K means clustering to also address the covariance structure of the data. That way, if there are significantly different patterns in different parts of the data, the clustering process will recognize this and group accordingly.

One last machine learning technique I decided to use is hierarchical agglomerative clustering. It gives a bit of a different insight. It will show the entire "structure" of the data, building it bottom up and step by step, allowing to monitor the process. I will set the cluster target to 10, as before, and the linkage type to "complete" such that intra-cluster correlation will be maximized again.

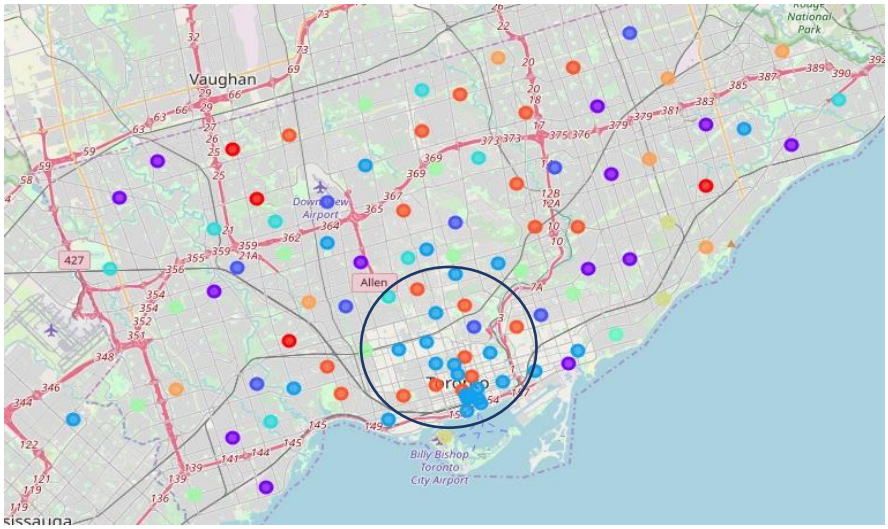
Results

K means

When looking at the clustering results of K means, as expected, it mostly grouped together data points somewhat similar to one another having lots of services but no clear pattern. The only unambiguous consistency that can be observed are clusters with post codes having only 1 or 0 services in them (marked in red below). I call this a "single service" cluster and an example can be found in #8, having only restaurants in it. The summary table containing the average data point in the cluster portrays this well:

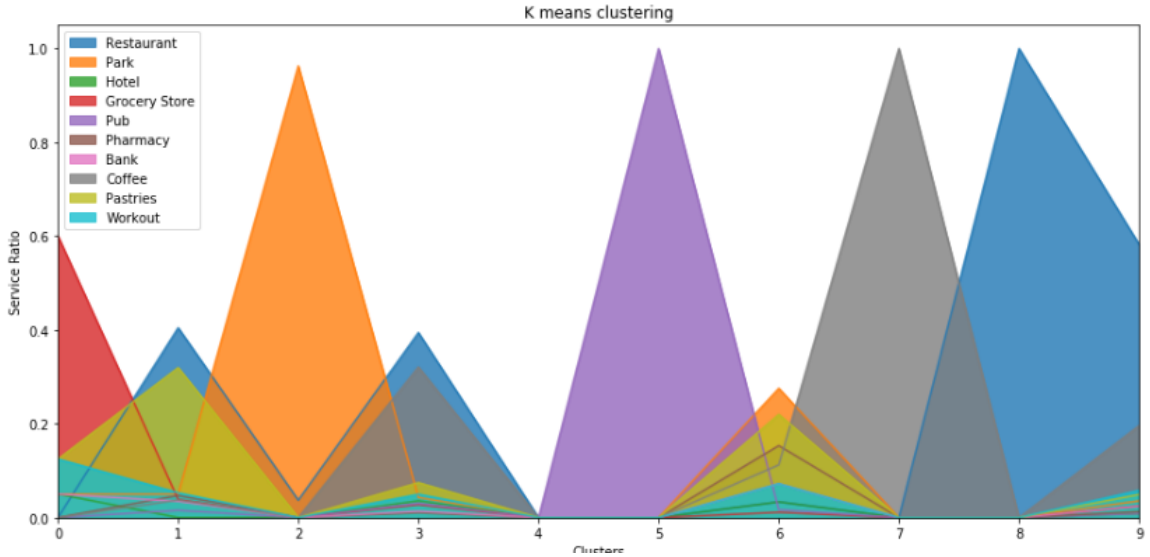
	Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
K Cluster Labels										
0	0.000000	0.050000	0.050000	0.600000	0.000000	0.000000	0.050000	0.000000	0.125000	0.125000
1	0.404701	0.050000	0.000000	0.038462	0.015385	0.047009	0.034615	0.037393	0.319658	0.052778
2	0.037037	0.962963	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	0.394472	0.042878	0.035571	0.027552	0.030253	0.010362	0.013526	0.321042	0.074536	0.049809
4	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
5	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
6	0.033333	0.275556	0.033333	0.011111	0.016667	0.153333	0.072222	0.112222	0.220000	0.072222
7	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000
8	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
9	0.582196	0.035457	0.008761	0.024398	0.008418	0.012822	0.024379	0.196235	0.049131	0.058202

When visualizing all the post codes on the map and painting each one according to its K means cluster, I am able to easily spot the dense Toronto areas which are closer to the shoreline. It is even possible to distinguish between two kinds: super-dense and mildly dense. They are included in cluster #3 (light blue) & #9 (orange) respectively:



What K means did for us was basically to find the city center. As you can see in the table above, clusters 3 & 9 are full of all 10 services, with restaurants and coffees being the dominant.

It is also interesting to use an area chart to visualize the trait I mentioned earlier: softness degree, or intra-cluster similarity. The single-color high spikes show clusters with one dominant service in total and a very distinct pattern (not soft). The lower ones, having multiple colors are the best expressions to soft results. The more lines a triangle has inside it the heavier its density and abundance of services, making it softer and with a less distinct pattern:

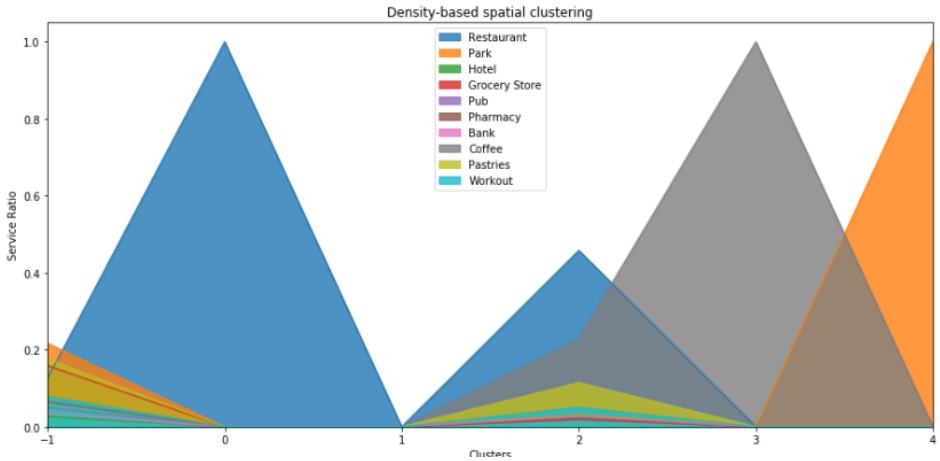


DBSCAN

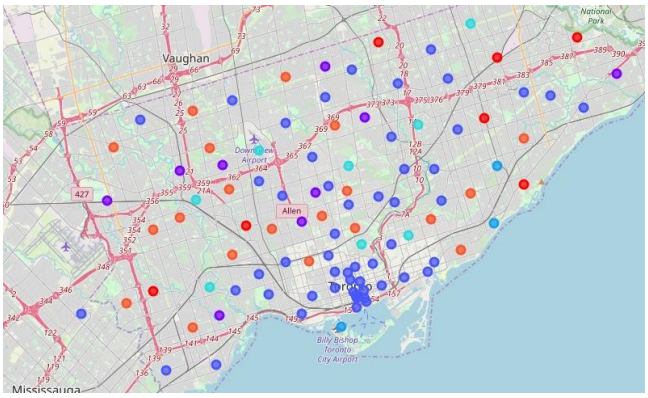
The results exhibit more powerful patterns in each cluster. The hard clustering can be easily noticed when compared to K means. Unlike with latter, DBSCAN grouped all the dense post codes in a single cluster (#2). The other clusters include, as before, single service post codes and cluster #1 contains outliers:

	Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
DBSCAN Cluster Labels										
-1	0.125758	0.216591	0.026667	0.158636	0.050000	0.064167	0.035000	0.067348	0.176667	0.079167
0	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	0.457974	0.032017	0.020209	0.018857	0.024318	0.025447	0.027539	0.228462	0.114697	0.050479
3	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000
4	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

Interesting to note that K means found post codes containing only a Restaurant\Pub\Coffee while here Restaurant\Park\Coffee are the single service clusters - so they do not completely overlap. Bottom line though, DBSCAN delivered much more "concentrated" ("hard") clustering all across the board generating limited added value on top of K means. The resulting dichotomy can be also seen in the area chart. Either an "empty" spike (single service) or a filled one (dense):



Visualizing the results on the map demonstrates that the grouping of post codes has little geographical context in it. This is because we have a small number of clusters and most post codes having more than one service are simply grouped together in #2:



Mean Shift

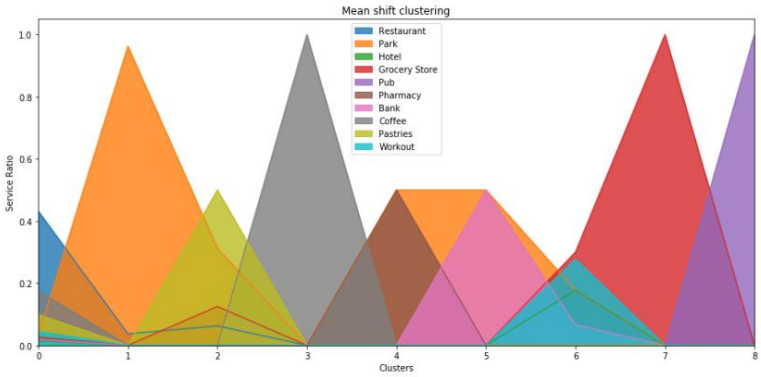
Now the following algorithm, Mean Shift, was trusted to generate results resembling DBSCAN since both are powered by density logic. Surprisingly, it did much better than its colleague, uncovering a new type of clusters in the middle of the "softness spectrum" – something not yet seen in neither DBSCAN nor K means results. This can be demonstrated by the blue marks in the following summary table of Mean Shift clusters:

	Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
Mean Shift Cluster Labels										
0	0.431570	0.033471	0.013991	0.022962	0.016836	0.02766	0.019065	0.175435	0.099064	0.044563
1	0.037037	0.962963	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000
2	0.062500	0.312500	0.000000	0.125000	0.000000	0.00000	0.000000	0.000000	0.500000	0.000000
3	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.000000	1.000000	0.000000	0.000000
4	0.000000	0.500000	0.000000	0.000000	0.000000	0.50000	0.000000	0.000000	0.000000	0.000000
5	0.000000	0.500000	0.000000	0.000000	0.000000	0.00000	0.500000	0.000000	0.000000	0.000000
6	0.000000	0.177778	0.177778	0.300000	0.000000	0.00000	0.066667	0.000000	0.000000	0.277778
7	0.000000	0.000000	0.000000	1.000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000
8	0.000000	0.000000	0.000000	0.000000	1.000000	0.00000	0.000000	0.000000	0.000000	0.000000

Zoom in to cluster #2 for example, having a combination of few services in each post code I contains. This is what I call a balanced result: not too general and not too rigid. One can learn a whole lot from looking at such clusters as #2 (will be further analyzed in conclusions chapter):

Postcode	Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
M1L	0.25	0.25	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0
M6L	0.00	0.50	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0
M6N	0.00	0.00	0.0	0.5	0.0	0.0	0.0	0.0	0.5	0.0
M9R	0.00	0.50	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0

Area chart displays the fact the results are balanced as well. You can see that a lot of triangles are of medium height and contain several colors:



GMM

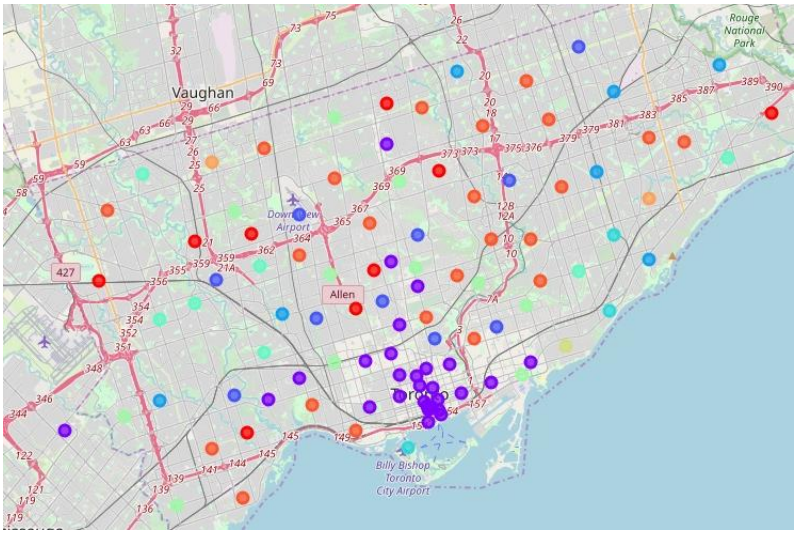
Next, GMM also delivered balanced results (marked in blue) while including the previous patterns of single service clusters (marked in red) and dense ones (marked in green):

	Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
GMM Cluster Labels										
0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.444956	0.046018	0.038975	0.020076	0.030392	0.007495	0.005832	0.278529	0.083197	0.044530
2	0.083333	0.916667	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
4	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000
5	0.214286	0.178571	0.000000	0.071429	0.000000	0.000000	0.000000	0.035714	0.500000	0.000000
6	0.103030	0.214141	0.048485	0.046465	0.033333	0.139394	0.083838	0.102020	0.145455	0.083838
7	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
8	0.000000	0.000000	0.000000	0.750000	0.000000	0.000000	0.000000	0.000000	0.000000	0.250000
9	0.492188	0.000000	0.000000	0.048599	0.004155	0.039752	0.047895	0.196211	0.099890	0.071311

When comparing between GMM and Mean Shift, specifically the balanced clusters (blue), one can see that all combinations that GMM generated are new and do not exist in Mean Shift (vice versa is all true). That's great. For example, cluster #5: post codes there usually include a restaurant, pastries and coffee \ grocery \ park. However, there are two other post codes that lack a restaurant: M6L and M6N. If we know this combination works well in practice somewhere else, isn't this a potential lead?

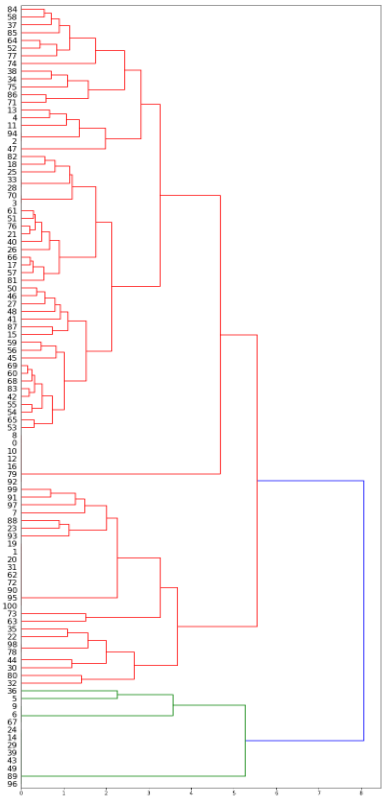
Restaurant	Park	Hotel	Grocery Store	Pub	Pharmacy	Bank	Coffee	Pastries	Workout
0.50	0.00	0.0	0.0	0.0	0.0	0.0	0.00	0.5	0.0
0.25	0.25	0.0	0.0	0.0	0.0	0.0	0.00	0.5	0.0
0.00	0.50	0.0	0.0	0.0	0.0	0.0	0.00	0.5	0.0
0.00	0.00	0.0	0.5	0.0	0.0	0.0	0.00	0.5	0.0
0.50	0.00	0.0	0.0	0.0	0.0	0.0	0.00	0.5	0.0
0.25	0.00	0.0	0.0	0.0	0.0	0.0	0.25	0.5	0.0

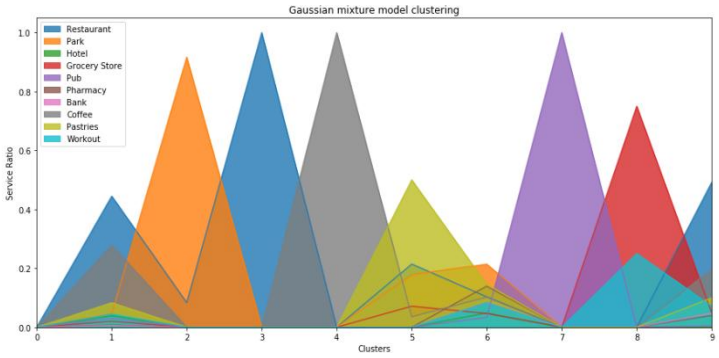
Also, while Mean Shift has clustered the post codes with no noticeable geographical context, GMM did. Look at the colors, it seems that there's a correlation between service supply, area density and location. The purple post codes belong to the densest areas, next are the orange \ turquoise and then the red. This means we are seeing some kind of a "service function" here that depends on the distance from the shoreline: the farther away, the less services the post codes have:

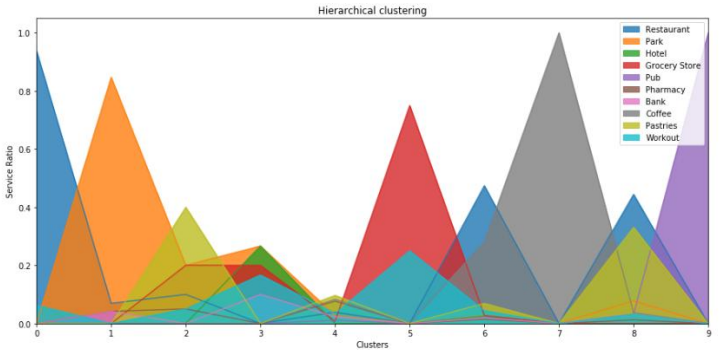


Hierarchical clustering

Last but not least is the agglomerative hierarchical clustering. The unique thing about it is the ability to display the entire clustering process visually using the hierarchical tree chart – a dendrogram. Even if to judge only by the structure of the dendrogram, it is already possible to recognize the nature of the clustering. It is similar to GMM and Mean Shift in the way that there are several types of clusters. The service-dense ones are at the top of the tree image while the sparse are in the bottom. There's a hidden assumption here that a cluster with more post codes contains more services as well. This is not only intuition but also inferred from the previously applied algorithms. I've also added the area charts of GMM and hierarchical clustering at the right side of the screen to show this similarity via an additional medium (upper is GMM, lower is hierarchical):







Discussion

Eventually, when looking at the results of all 5 algorithms that I used throughout the final project (K means, DBSCAN, Mean Shift, GMM and hierarchical) there seem to be 3 major types of clusters:

1. Clusters representing geographically highly dense areas are the 1st type. These are clusters where most post codes have a huge load of essential services in them. Unsurprisingly, the location of this type of post codes clustered together was quite similar - near the center of the shoreline. It is basically the city center and almost all algorithms spotted it without "having a clue" in terms of coordinates! This is impressive and proves two points. One, level of services is strongly correlated to location. Two, the initial problem I wished to solve is real. Meaning, the density of services, which is assumed to be a mirror of overall density, is very imbalanced. Judging by visual scale only, the dense areas are spread over ~20% of total territory, having ~80% of all services. This type of clusters contained most of the post codes in every algorithm I used. Few examples for such post codes are: M2K, M2J, and M5B. There is not much value to be extracted for the stake holders in this type of clusters.

2. More insights can be drawn from the 2nd type, consisting of clusters with post codes that have a single service in them at most. This is the sort of areas that require municipal intervention. It is obvious they are those who should take the load of the dense areas. These are not necessarily post codes on the outskirts of the city but simply areas that could use more infrastructure

development and various incentives. Unlike the previous type, this one is not necessarily homogeneous in terms of post code locations. For example, the following post codes are not far away from the center: M6M, M1N, and M4E. M6M has only restaurants, M1N has only coffees and M4E has only a pub. Toronto's municipal authorities can slightly stimulate these post codes to encourage growth in activity. For businesses, it will be harder to utilize this information since the development of such areas is uncertain and setting up a business there is still risky.

3. Businesses can profit mostly from the 3rd type of clusters since they are comprised of post codes having only few services, usually between 2 to 3. This information can serve as a business hint for opening up new branches without taking too much risk. For example: in post code M8Z and M9V you have Restaurants + grocery stores + pastries shops while in post code M6N you have only grocery and pastries. If the Restaurant\grocery\pastries is a proven combination in practice, wouldn't it be much safer for restaurant owner to open up there?

All in all, the different algorithms uncovered gradually the 3 types shaping the entire picture: K means granted some top-down perspective without being too precise (too soft). DBSCAN turned out to be too rigid without adding extra value on top of K means. Eventually, the golden middle was achieved with Mean Shift, GMM and Hierarchical clustering. Hierarchical clustering was good mostly for visual explanation, but the optimal balanced results were delivered by Mean Shift and GMM, each showing all three types of clusters.

Conclusion

When describing the entire project step-by-step, what I can say is that it was gradually built from raw meaningless data, to vectors, to clustering results, to insights. I tries using 5 different machine learning algorithms mostly because the first ones I tried did not deliver the product I hoped they will. In each step I tried to use a new tool such that it will open up a dimension not yet revealed by the ones already used. Eventually, the picture became clearer and it was proven by actual data that the business problem I set out to solve existed indeed. I believe that the results generated significant value in relation to the business problem, pointing at trends and insights that can help stakeholders achieve their goals.

One final interesting thought about additional directions for research is the addition of time-dependent data. In my opinion, much more value can be created by looking at actual historical trends. There are a lot of assumptions, even though logical, in my recommendation. They can be tested in reality using historical data.