Finding optimal bar locations near Groningen city center

Jamo Momand

December 31st 2020

IBM Applied Data Science Capstone

1. Introduction: Business Problem

For the IBM Applied Data Science Capstone,[1] I am going to search for **optimal new bar/restaurant locations** in the student city **Groningen, the Netherlands**. The target audience will be **prospective bar/restaurant owners**.

Groningen is the largest city in the north of the Netherlands. It has a population of approximately 200k, of which 50% are below the age of 30. Since the city's university and higher education hosts more than 60k students, it is a vibrant student city with a multitude of bars and restaurants. This is why there are plenty of opportunities for the nightlife industry and henceforth my choice of business problem.[2]

I will search for bars and restaurants at the edge of the most competitive regions, because I expect that while the Corona virus situation will slowly be resolved in 2021, the city center will be very competitive for new players on the market.

2. Data

Neighborhood data: I will define neighborhoods used for the analysis as is done by the municipality of Groningen. Examples are the names of the neighborhoods, like Binnenstad or A-Kwartier.[3]

Coordinate data: Next, I will search the neighborhood coordinates using the free geolocator Nominatim from GeoPy. The neighborhoods that cannot be found, will be excluded from analysis. If necessary, the neighborhood and coordinate data will be cleaned to discard irrelevant or erroneous neighborhoods.

Venue data: This will be requested from Foursquare. Examples of venues are bars or restaurants and their geographical locations. If necessary, a selection will be made of which venues to select. However, not only bars and restaurants should be included in the analysis, as other venues like supermarkets or gyms can be indicators of livelihood and thus business opportunities.

3. Methodology

This data will then be processed (cleaned/selected) and included in a clustering analysis, similarly to the examples of the IBM course. For this, I will use KMeans and will identify the most competitive region and its nearby neighborhoods, after which I will classify them in competitive order besides the city center. This will then be my advice to the newcomer to the nightlife market as where to start the bar/restaurant business.

4. Results and Discussion

4.1 Data collection, understanding and preparation

First, let's explore Groningen on the map. I will use the free geolocator Nominatim from GeoPy and Folium for coordinates and maps, respectively. If everything works correctly, the output should be a map of Groningen like in Figure 1.

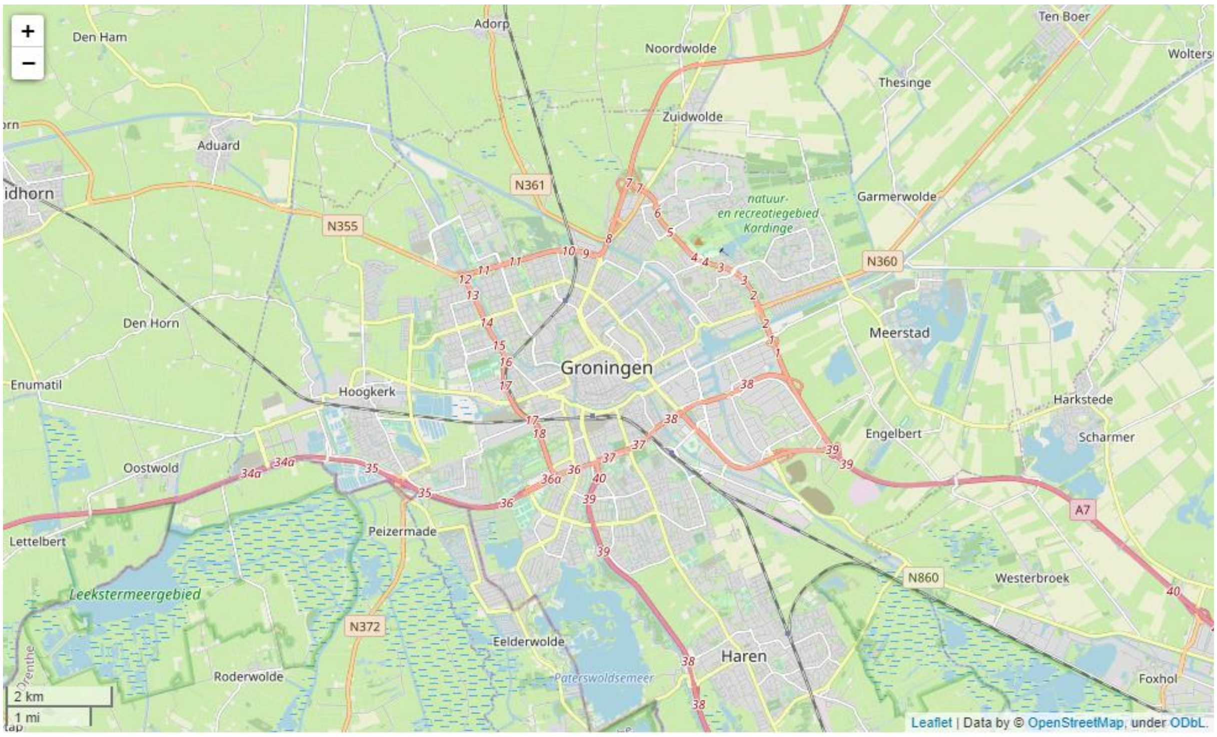
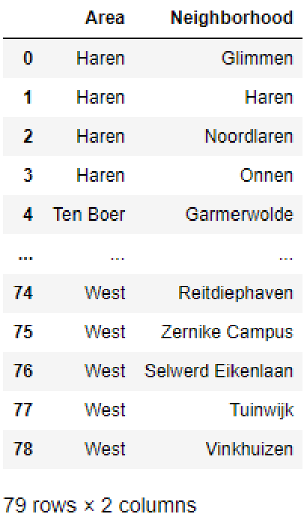


Figure 1: Map of Groningen, as found with Nominatim from GeoPy and displayed with Folium.

Next, I will scrape neighborhood names from Groningen municipality website.[2] Some data cleaning steps are performed, because known neighborhoods are omitted and there are some typos on the website. As this is not a robust function against website updates, I store and load previous results. The results will be stored in a Pandas DataFrame, as shown in Figure 2.

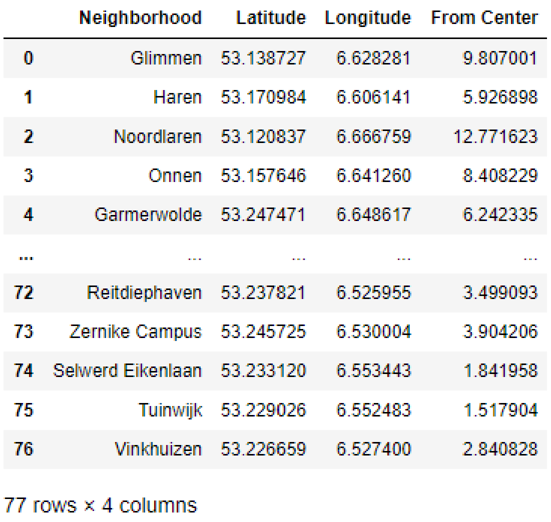


	Area	Neighborhood
0	Haren	Glimmen
1	Haren	Haren
2	Haren	Noordlaren
3	Haren	Onnen
4	Ten Boer	Garmerwolde
...	...	...
74	West	Reitdiephaven
75	West	Zernike Campus
76	West	Selwerd Eikenlaan
77	West	Tuinwijk
78	West	Vinkhuizen

79 rows × 2 columns

Figure 2: Scraped neighborhood names.[2]

As stated previously, I will search the neighborhood coordinates using the free geolocator Nominatim from GeoPy and store the results in a Pandas DataFrame. In addition, I will add the distance from the city center. The output should be as shown in Figure 3.



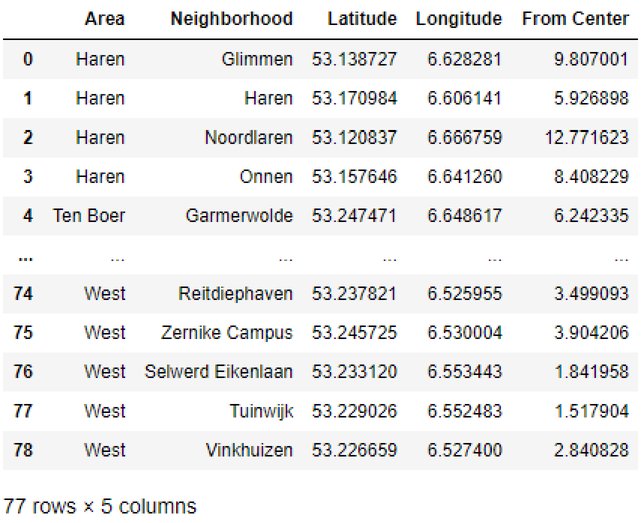
	Neighborhood	Latitude	Longitude	From Center
0	Glimmen	53.138727	6.628281	9.807001
1	Haren	53.170984	6.606141	5.926898
2	Noordlaren	53.120837	6.666759	12.771623
3	Onnen	53.157646	6.641260	8.408229
4	Garmerwolde	53.247471	6.648617	6.242335
...	...	...	...	...
72	Reitdiephaven	53.237821	6.525955	3.499093
73	Zernike Campus	53.245725	6.530004	3.904206
74	Selwerd Eikenlaan	53.233120	6.553443	1.841958
75	Tuinwijk	53.229026	6.552483	1.517904
76	Vinkhuizen	53.226659	6.527400	2.840828

77 rows × 4 columns

Figure 3: Coordinates of found neighborhoods.

These DataFrames are then merged, omitting neighborhood names that are not found. This results in the DataFrame in Figure 4 and the map in Figure 5. The average distance and its standard deviation are: 3.95 km and 3.63 km.

Visual inspection of the neighborhoods shows that there is a good agreement (which was partially the result of cleaning of neighborhood names). One could choose to discard the neighborhoods far from the center in later steps, not being relevant to the city culture, but I chose to keep it as to see what the analysis will show for it.



	Area	Neighborhood	Latitude	Longitude	From Center
0	Haren	Glimmen	53.138727	6.628281	9.807001
1	Haren	Haren	53.170984	6.606141	5.926898
2	Haren	Noordlaren	53.120837	6.666759	12.771623
3	Haren	Onnen	53.157646	6.641260	8.408229
4	Ten Boer	Garmerwolde	53.247471	6.648617	6.242335
...	...	...	...	...	...
74	West	Reitdiephaven	53.237821	6.525955	3.499093
75	West	Zernike Campus	53.245725	6.530004	3.904206
76	West	Selwerd Eikenlaan	53.233120	6.553443	1.841958
77	West	Tuinwijk	53.229026	6.552483	1.517904
78	West	Vinkhuizen	53.226659	6.527400	2.840828

77 rows × 5 columns

Figure 4: Resulting DataFrame from the neighborhood names and coordinates.

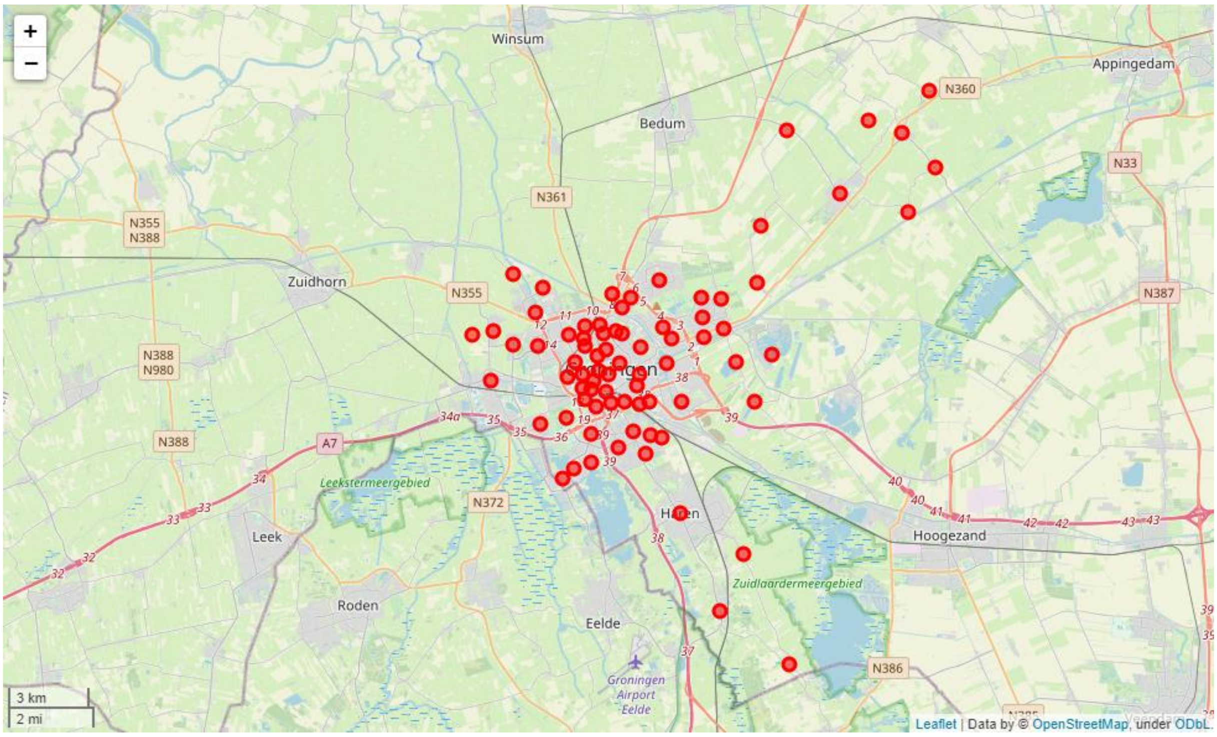
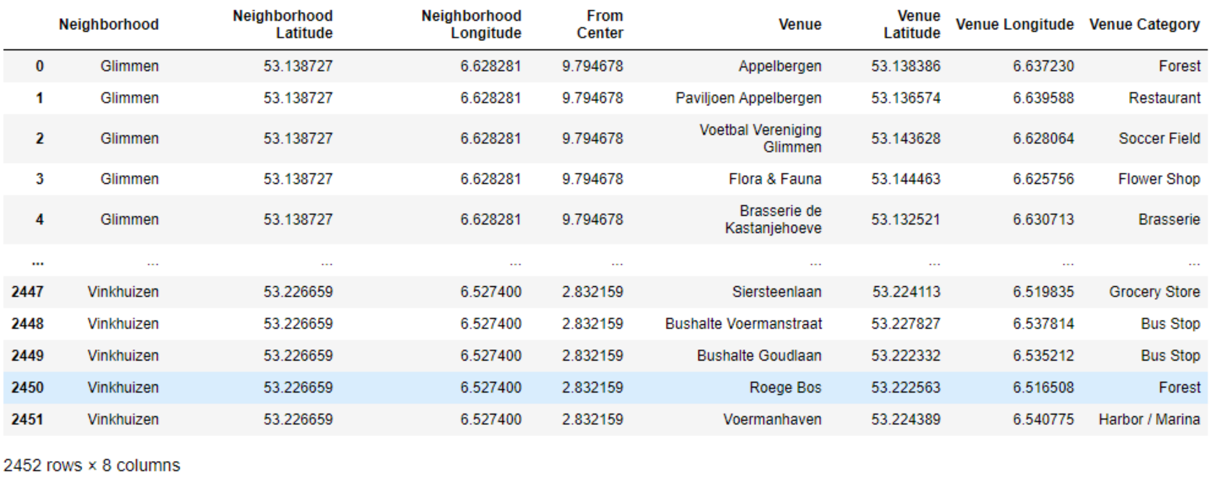


Figure 5: Folium map showing the found neighborhoods. Visual inspection shows that there is a good agreement.

Finally, I use Foursquare to find venues within a radius of 1 km of each neighborhood. Examples of venues are bars and restaurants with their coordinates, but also supermarkets, gyms, etc. I chose 1 km because of the Groningen cycling culture, where 1 km is approximately a 4 min bicycle ride and 12 minute walk. The results are stored in a Pandas DataFrame, as in Figure 6.



	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	From Center	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Glimmen	53.138727	6.628281	9.794678	Appelbergen	53.138386	6.637230	Forest
1	Glimmen	53.138727	6.628281	9.794678	Paviljoen Appelbergen	53.136574	6.639588	Restaurant
2	Glimmen	53.138727	6.628281	9.794678	Voetbal Vereniging Glimmen	53.143628	6.628064	Soccer Field
3	Glimmen	53.138727	6.628281	9.794678	Flora & Fauna	53.144463	6.625756	Flower Shop
4	Glimmen	53.138727	6.628281	9.794678	Brasserie de Kastanjehoeve	53.132521	6.630713	Brasserie
...	...	...	...	...	...	...	...	...
2447	Vinkhuizen	53.226659	6.527400	2.832159	Siersteenlaan	53.224113	6.519835	Grocery Store
2448	Vinkhuizen	53.226659	6.527400	2.832159	Bushalte Voermanstraat	53.227827	6.537814	Bus Stop
2449	Vinkhuizen	53.226659	6.527400	2.832159	Bushalte Goudlaan	53.222332	6.535212	Bus Stop
2450	Vinkhuizen	53.226659	6.527400	2.832159	Roege Bos	53.222563	6.516508	Forest
2451	Vinkhuizen	53.226659	6.527400	2.832159	Voermanhaven	53.224389	6.540775	Harbor / Marina

2452 rows × 8 columns

Figure 6: Foursquare venues for each neighborhood within a radius of 4 km.

A couple of notes should be made on the data here. First, one could do a lot of data cleaning here. E.g. one could choose to discard a Forest or merge Restaurant with Brasserie in Glimmen. This could be done when finetuning the analysis, but I leave it as is for the course assignment. Second, the free Foursquare account has a limit of 100 venues per request, which is clearly visible for neighborhood near the city center. I leave it as is, but for finetuning the analysis one could define denser location data when this happens.

4.2 Modelling and Evaluation

Now that the data is complete, I perform the analysis. The venue data is One Hot coded, grouped by neighborhood and summed. Here is an additional point where one can finetune the analysis. My choice was just summing the number of venues with equal weight for simplicity, but also here one could give bars and restaurants a bigger weight than e.g. a forest.

As stated before, I will use KMeans for clustering. To find the optimal number of clusters I use the elbow method as implemented in the Yellowbrick library, see Figure 8. Several runs gives values between 5 and 11 clusters, so I choose to set the number of clusters to 8 and perform KMeans.

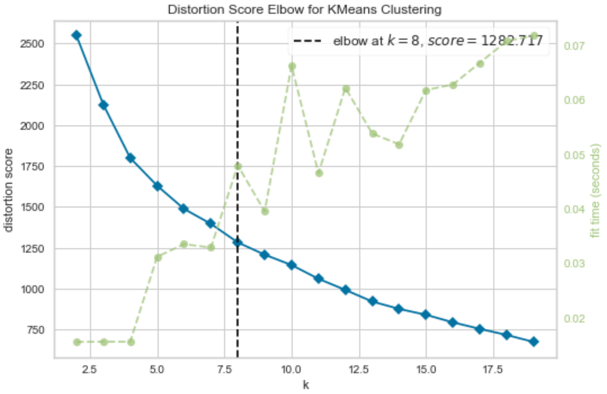


Figure 7: Elbow method with Yellowbrick, showing an optimal cluster size of 8.

Combining everything, the data and results give the map in Figure 8. Visual inspection of the results shows that cluster 7 is the inner city center, which will be used to answer the business problem in the introduction. Cluster 5 and 0 are living areas or areas where there are very few venues, respectively. One can omit these clusters to get the final recommendation results in Figures 9 and 10.

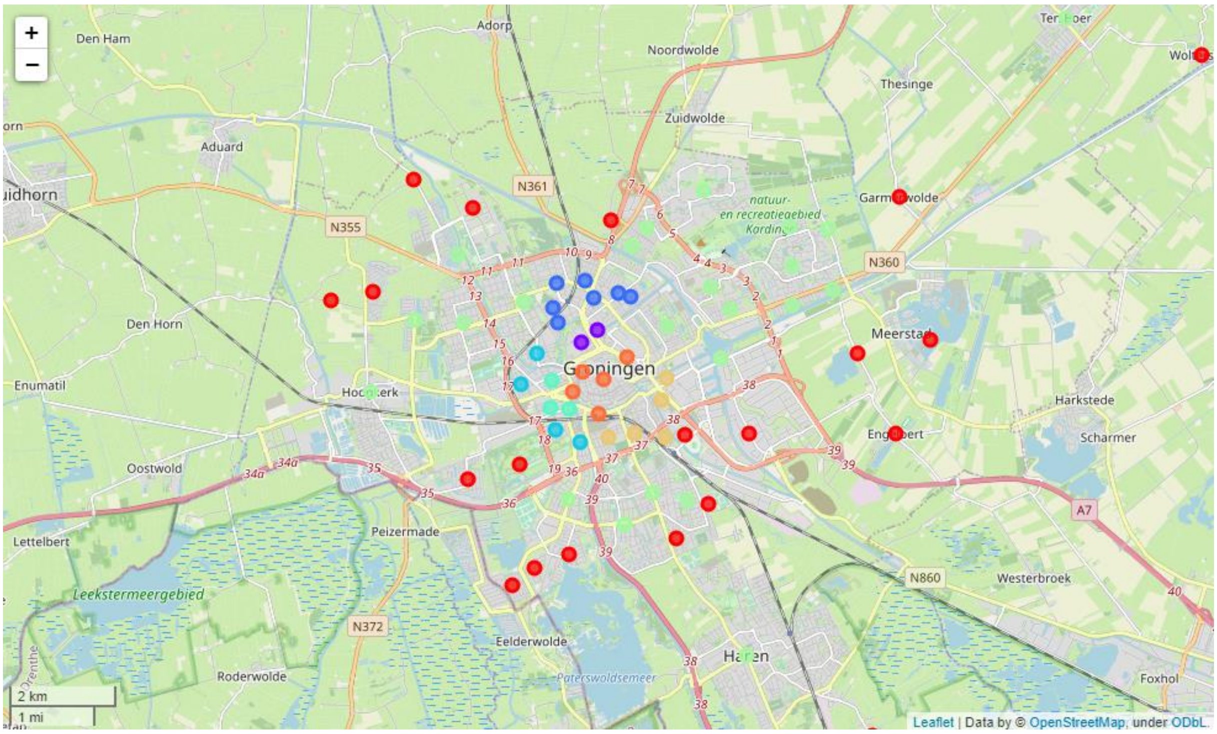
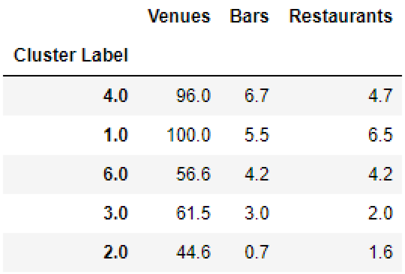
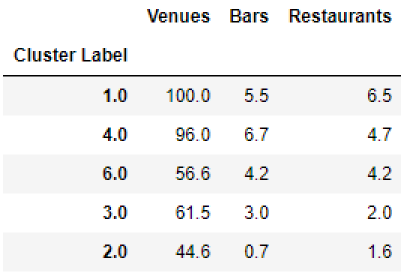


Figure 8: Results of clustering the neighborhoods based on venue data. The colors indicate: Purple – 0, Red – 1, Dark Blue – 2, Blue – 3, Light Blue – 4, Green – 5, Beige – 6, Orange – 7.



Cluster Label	Venues	Bars	Restaurants
4.0	96.0	6.7	4.7
1.0	100.0	5.5	6.5
6.0	56.6	4.2	4.2
3.0	61.5	3.0	2.0
2.0	44.6	0.7	1.6

Figure 9: Optimal Bar locations.



Cluster Label	Venues	Bars	Restaurants
1.0	100.0	5.5	6.5
4.0	96.0	6.7	4.7
6.0	56.6	4.2	4.2
3.0	61.5	3.0	2.0
2.0	44.6	0.7	1.6

Figure 10: Optimal Restaurant locations.

Of course it is up to the prospective bar or restaurant owner here to decide whether more venues will be beneficiary (more livelihood) or whether fewer would be better to open a niche bar or restaurant. Such a question could be further explored if this is a requirement for the business problem.

5 Conclusion

In summary, for the IBM Applied Data Science Capstone I have defined a business problem to find optimal bar and restaurant locations in the city of Groningen, the Netherlands. For this, I have used neighborhood data from the municipality website and Foursquare venue data. Using KMeans clustering from ScikitLearn and an optimal cluster size from Yellowbrick, the optimal bar and restaurant locations are found and given in Figures 9 and 10, respectively. Different data cleaning and optimizing steps, as well as analysis steps, are discussed in between. With this, the business problem is answered and an advise to prospective bar or restaurant owners can be given.

Acknowledgements

The Faculty of Science and Engineering from the University of Groningen is gratefully acknowledged for funding the IBM Data Science Professional Certificate program from Coursera.

References

1. https://www.ibmbigdatahub.com/blog/why-we-need-methodology-data-science
2. https://www.youtube.com/watch?v=05Iiuz7A7E0
3. https://gemeente.groningen.nl/wijken-dorpen-wijkwethouders-en-gebiedsteams