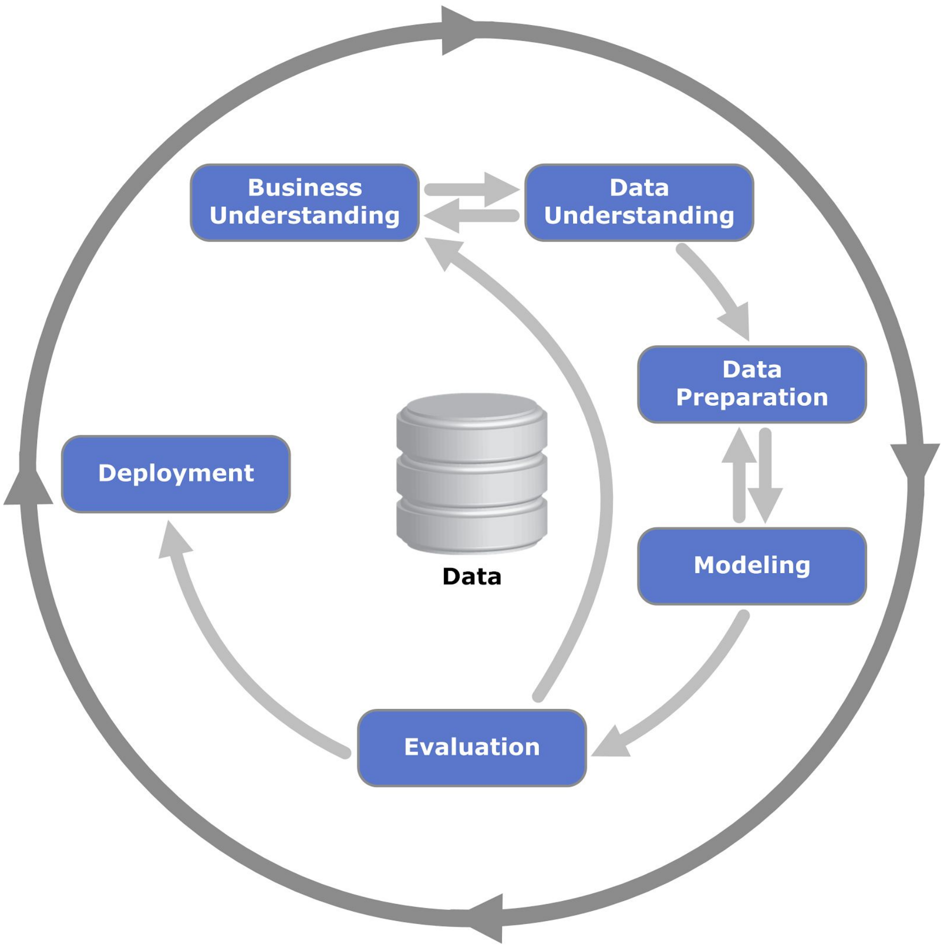
How to use data to find the best spot for a gastronomy?

by Manuel Strigli

Introduction/Business Problem

I live in **Berlin** (Germany) and there is a very hard competition in the **gastronomy** market. Specially, during the COVID-19 lockdown restaurants with a delivery service have an advantage and can make a good stroke of business.
So I want to know where and what kind of restaurant **entrepreneurs as target group** should open now.

Therefore I want to work with the Data Scientist method CRISP:



Description of the data and how it will be used to solve the problem

Therefore, I want to see where is a low density of gastronomy businesses or is there a missing kind of restaurant.
I oriented myself to the New York City Lab, nay it was the base of my notebook. First, I tried to use some data by the **Berlin government** and even found some of **gastronomies** who reported to the government, if they deliver food or if customers can pickup ordered food. The data is in german language, but I translated the necessary information into English. You will see, that I worked with the data for a while. I early recognized that just a few venues have a filled 'neighborhood'-field. So I thought I could just work with the postal codes instead, but after I even created a map with different marker types, I had to realize that not even all venues had a defined postal code.
After that I wanted to use the f**oursquare API** again. Additionally, the geocoder method always just return none to me (even when I use the suggested loop). Hence, I manually searched the latitudes and longitudes of the **berlin districts** (focused on the central districts). With this data I did the same work like in the NYC Lab except that i added different radius to look for venues, because of the different sizes of the berlin districts.

Finally, I used the **k-means algorithm** to cluster the districts. You will see the result at the end of the notebook.

First I downloaded the data from the website of the Berlin government (https://daten.berlin.de/datensaetze/gastronomien-laden-und-andere-gesch%C3%A4fte-mit-liefer-und-abholservice) as Json data:

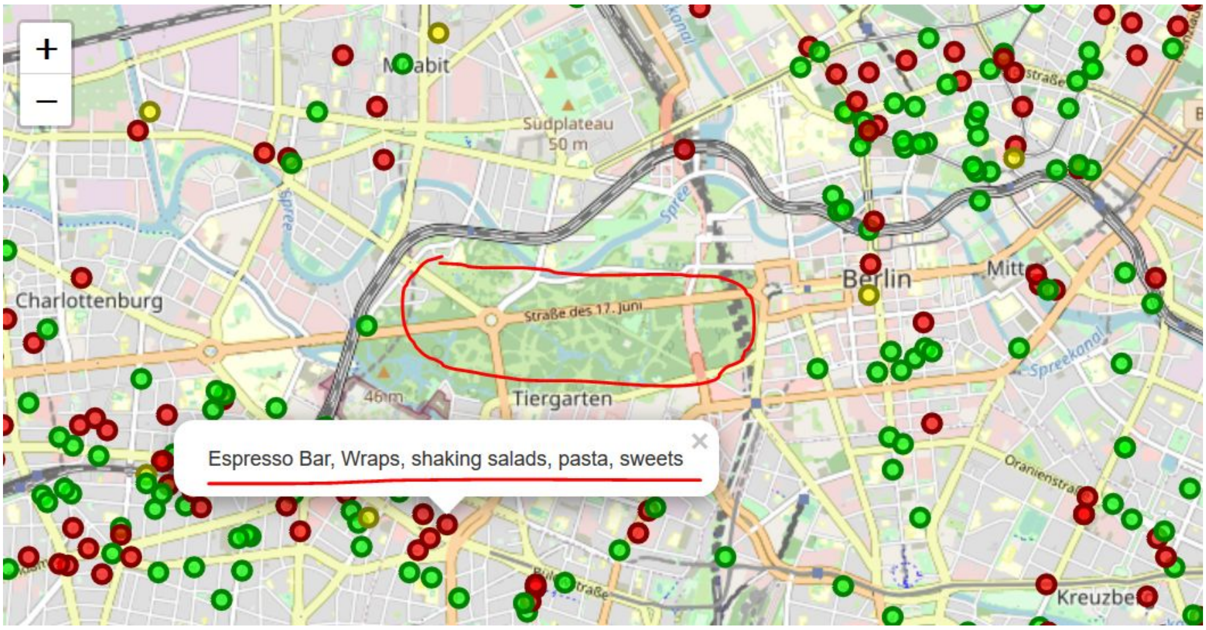

```
{'type': 'FeatureCollection',
 'features': [{'type': 'Feature',
   'geometry': {'type': 'Point', 'coordinates': [13.38212, 52.53116]},
   'properties': {'title': '771',
    'href': '262',
    'description': ' <a href="262">Mehr...</a>',
    'id': '/sen/web/service/liefer-und-abholdienste/index.php/detail/771',
    'data': {'id': '771',
     'unique_id': '262',
     'name': 'Brasserie la bonne franquette',
     'strasse_nr': 'Chausseestraße 110',
     'plz': '10115',
     'art': 'Gastronomie (Café, Restaurant, Imbiss, Lebensmittelhandlung, usw.)',
     'angebot': 'Klassische Französische Küche',
     'lieferung': 'FALSCH',
     'beschreibung_lieferangebot': '',
     'selbstabholung': 'WAHR',
     'angebot_selbstabholung': 'Bestellungen werden jederzeit entgegengenommen. Abholung (später auch Lieferung) immer dienstags
bis samstags in der Zeit von 17 bis 21 Uhr',
     'fon': '+493094405363',
     'w3': 'https://labonnefranquette.de',
     'mail': 'essen@labonnefranquette.de',
     'montag': '',
     'dienstag': '17:00-21:00',
     'mittwoch': '17:00-21:00',
     'donnerstag': '17:00-21:00',
     'freitag': '17:00-21:00',
     'samstag': '17:00-21:00',
     'sonntag': ''}}},
```

Afterwards I prepared the data as a pandas dataframe:

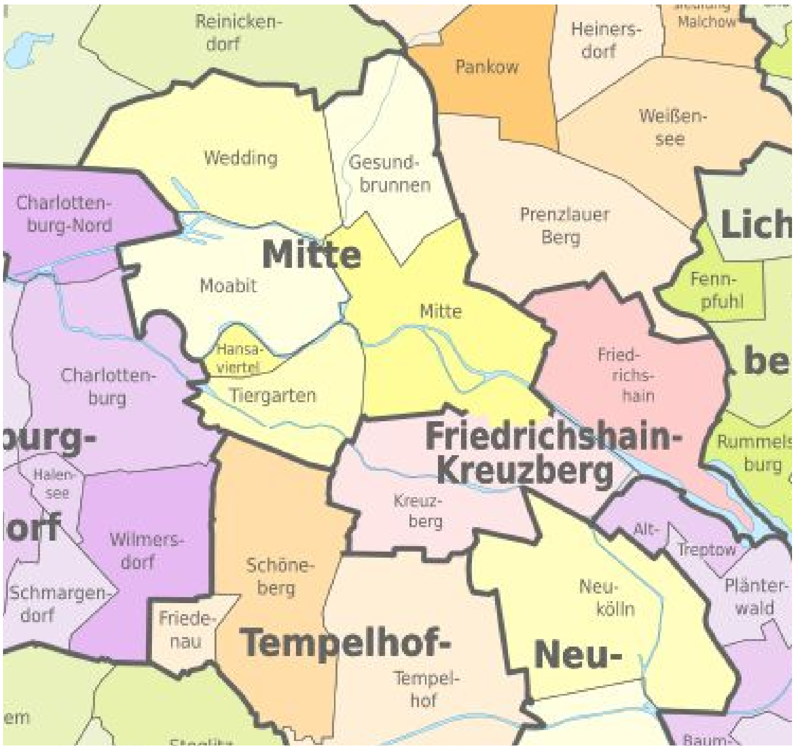
	Name	Speciality	Delivery	Pickup	Postal_code	Latitude	Longitude
0	Brasserie la bonne franquette	Klassische Französische Küche	FALSCH	WAHR	10115	52.53116	13.38212
1	Alpenstück	Süddeutsche Spezialitäten wie Maultaschen Köni...	FALSCH	WAHR	10115	52.53033	13.39184
2	sagrantino 136	Italian fusion kitchen 3 -course menus by Mat...	FALSCH	WAHR	10115	52.52633	13.38897
3	Risorante Bonfini	Italienische Küche, Pasta und Pizza	FALSCH	WAHR	10115	52.52953	13.38480
4	CAFE RIBO	Maultaschen, schwäbische küche, cafe	FALSCH	WAHR	10115	52.53089	13.39654

and then created a Folium Map with Classification if the venues deliver or customers can pick up ordered food there:

Then I recognized that my approach to use the density of the venues would be biased by gaps from e.g. parks:



Additionally, even some postal codes were missing. This is why I decided to work further with foursquare again and work through the city district by district, but had to keep in mind that the districts have different sizes.



Then I prepared the data as data frame and added a searching radius for each district. After I got the venues for each district I used one hot encoding to prepare the data for modeling:

	District	Latitude	Longitude	Radius
0	Mitte	52.522295	13.362587	2000
1	Tiergarten	52.510807	13.336607	1500
2	Hansaviertel	52.518247	13.333539	500
3	Moabit	52.524654	13.322123	1500
4	Charlottenburg	52.517217	13.345421	3000
5	Halensee	52.495719	13.275483	500
6	Wilmersdorf	52.485248	13.289401	2000
7	Schöneberg	52.479915	13.321338	2500
8	Tempelhof	52.465193	13.356444	2000
9	Neukölln	52.477365	13.407260	2000
10	Kreuzberg	52.496636	13.375818	2000
11	Alt-Treptow	52.492591	13.399446	500
12	Friedrichshain	52.508585	13.420751	2000
13	Prenzlauer Berg	52.539299	13.399509	2000
14	Gesundbrunnen	52.550445	13.349664	1500
15	Wedding	52.550123	13.302204	1500

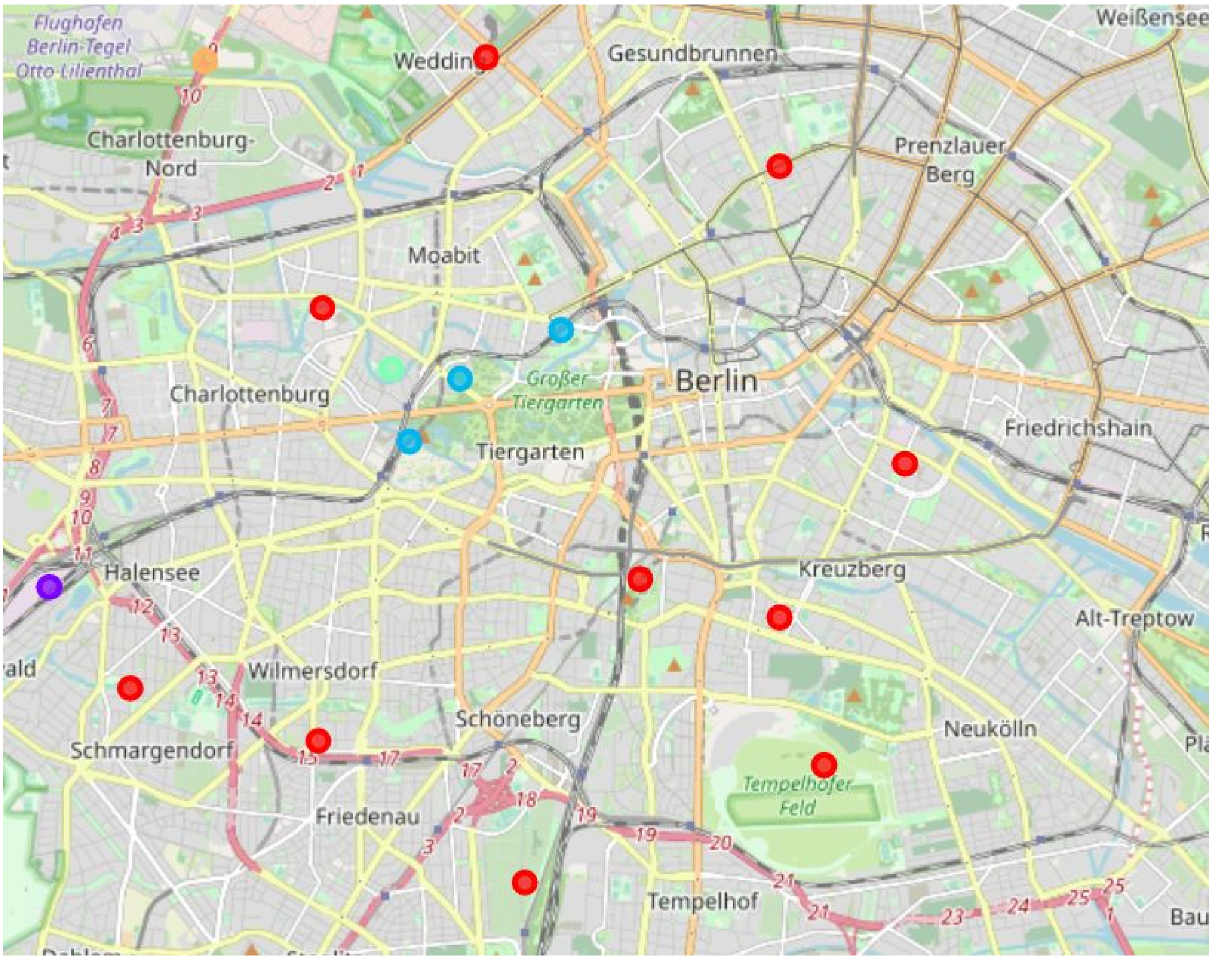
	District	African Restaurant	Airport Lounge	Airport Service	Aquarium	Art Gallery
0	Alt-Treptow	0.017857	0.00	0.000	0.00	0.00
1	Charlottenburg	0.000000	0.00	0.000	0.00	0.01
2	Friedrichshain	0.010000	0.00	0.000	0.00	0.03
3	Gesundbrunnen	0.010000	0.00	0.000	0.00	0.00
4	Halensee	0.000000	0.00	0.000	0.00	0.00
5	Hansaviertel	0.000000	0.00	0.000	0.00	0.00
6	Kreuzberg	0.000000	0.00	0.000	0.00	0.01
7	Mitte	0.000000	0.00	0.000	0.00	0.01
8	Moabit	0.000000	0.00	0.000	0.00	0.01
9	Neukölln	0.010000	0.00	0.000	0.00	0.02
10	Prenzlauer Berg	0.000000	0.00	0.000	0.00	0.00
11	Schöneberg	0.000000	0.00	0.000	0.00	0.00
12	Tempelhof	0.010000	0.00	0.000	0.00	0.00
13	Tiergarten	0.000000	0.00	0.000	0.01	0.01
14	Wedding	0.000000	0.05	0.175	0.00	0.00
15	Wilmersdorf	0.000000	0.00	0.000	0.00	0.00

Then I build up a dataframe with the most common venues for each district:

District	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
Alt-Treptow	Coffee Shop	Café	Italian Restaurant	Bakery	Cocktail Bar	Mediterranean Restaurant	Bookstore	Gift Shop	Kebab Restaurant	Breakfast Spot
Charlottenburg	Hotel	Café	Zoo Exhibit	Coffee Shop	Beer Garden	Art Museum	Concert Hall	Cocktail Bar	Monument / Landmark	Gym / Fitness Center
Friedrichshain	Bar	Coffee Shop	Café	Turkish Restaurant	Italian Restaurant	Bakery	Hotel	Art Gallery	Plaza	Organic Grocery
Gesundbrunnen	Café	Bar	Park	Coffee Shop	Ice Cream Shop	Supermarket	Drugstore	Turkish Restaurant	Chinese Restaurant	Fast Food Restaurant
Halensee	Historic Site	Italian Restaurant	Light Rail Station	Lake	Automotive Shop	Beach	Farmers Market	Falafel Restaurant	Fair	Fabric Shop

and run a k-means algorithm on it and created another Folium map:

	District	Latitude	Longitude	Radius	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue
0	Mitte	52.522295	13.362587	2000	2	Hotel	Monument / Landmark	Spa	Park
1	Tiergarten	52.510807	13.336607	1500	2	Zoo Exhibit	Hotel	Art Museum	Jazz Club
2	Hansaviertel	52.518247	13.333539	500	3	Hotel	Pub	Hotel Bar	Gastropub
3	Moabit	52.524654	13.322123	1500	0	Hotel	Bakery	Supermarket	Asian Restaurant
4	Charlottenburg	52.517217	13.345421	3000	2	Hotel	Café	Zoo Exhibit	Coffee Shop



In the end I had clustered the districts with spots for each district and could see which kind of businesses are already common in each district:

Cluster 1

```
berlin_merged.loc[berlin_merged['Cluster Labels'] == 0, berlin_merged.columns[[0] + list(range(5, berlin_merged.shape[1]))]]
```

	District	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Mitte	Hotel	Monument / Landmark	Spa	Park	Science Museum	Art Museum	Plaza	Concert Hall	Indie Movie Theater	Bookstore
1	Tiergarten	Zoo Exhibit	Hotel	Jazz Club	Monument / Landmark	Art Museum	Restaurant	Theater	German Restaurant	Furniture / Home Store	Fried Chicken Joint
2	Hansaviertel	Hotel	Hotel Bar	Gastropub	Pizza Place	Pub	Coffee Shop	Cocktail Bar	Clothing Store	Restaurant	River
4	Charlottenburg	Hotel	Café	Zoo Exhibit	Coffee Shop	Beer Garden	Art Museum	Concert Hall	Cocktail Bar	Monument / Landmark	Gym / Fitness Center

Cluster 2

```
berlin_merged.loc[berlin_merged['Cluster Labels'] == 1, berlin_merged.columns[[0] + list(range(5, berlin_merged.shape[1]))]]
```

	District	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
3	Moabit	Bakery	Hotel	Supermarket	Bar	Asian Restaurant	Coffee Shop	Italian Restaurant	Doner Restaurant	Café	Vietnamese Restaurant

For my district Moabit I can see now where to place the gastronomy and see that vegan restaurants are still pretty uncommon. So my recommendation to all brave entrepreneurs is to open a vegan restaurant directly at the Spree-river. So it is attractive to sit there and offer a delivery service as well.