



The Battle of Neighborhoods

“How to find a suitable location for a Food Truck in Berlin”

By: Håkon Ronold Mathisen

Final project in IBM Data Science Professional Certificate

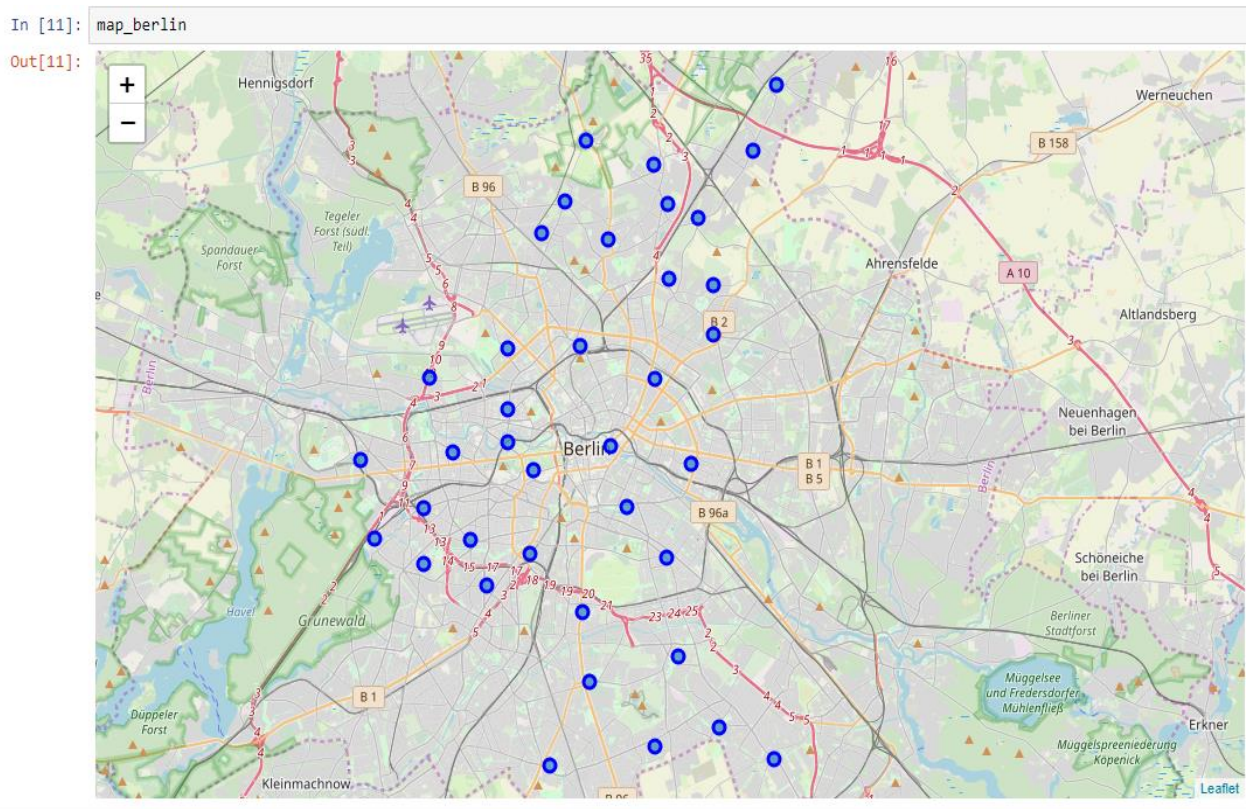
Motivation

- How do can you find the most suitable location for a Food Truck in a large city with fierce competition?
- Business problem: «A client has requested me to find the best possible location for her Food Truck (serving burgers and hot dogs) in Berlin, Germany»
 1. It should be placed in a central borough
 2. It should be in a locality that maximises her potential predicted customer base

Method

1. Find coordinates of all Boroughs and their corresponding localities
2. Find all nearby venues
3. Identify positive and negative drivers for Food Truck placement
4. Assign positive and negative weights
5. Sum up and

Berlin



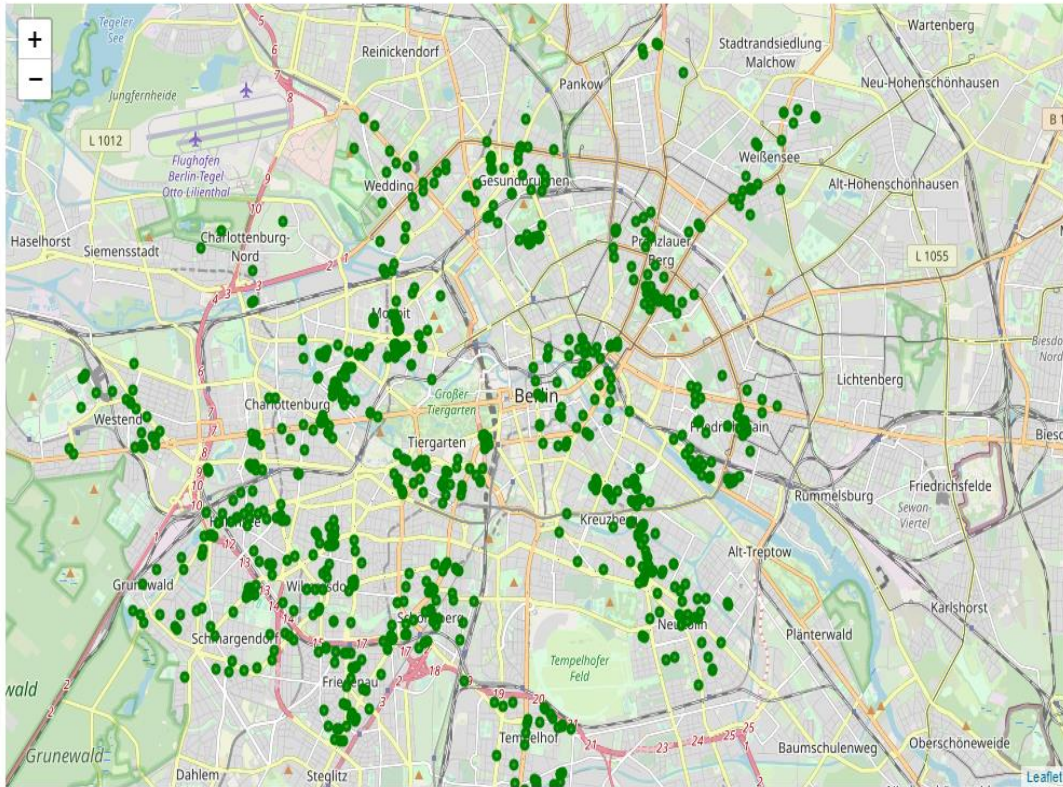
Boroughs and Localities in Central Berlin

- 6 boroughs with 39 corresponding localities
- Given set radius, 4 841 nearby venues

Offices

```
In [27]: map_berlin_offices = folium.Map(location=[latitude, longitude], zoom_start=12)
addToMap(berlin_venues_offices, 'green', map_berlin_offices)
map_berlin_offices
```

Out[27]:

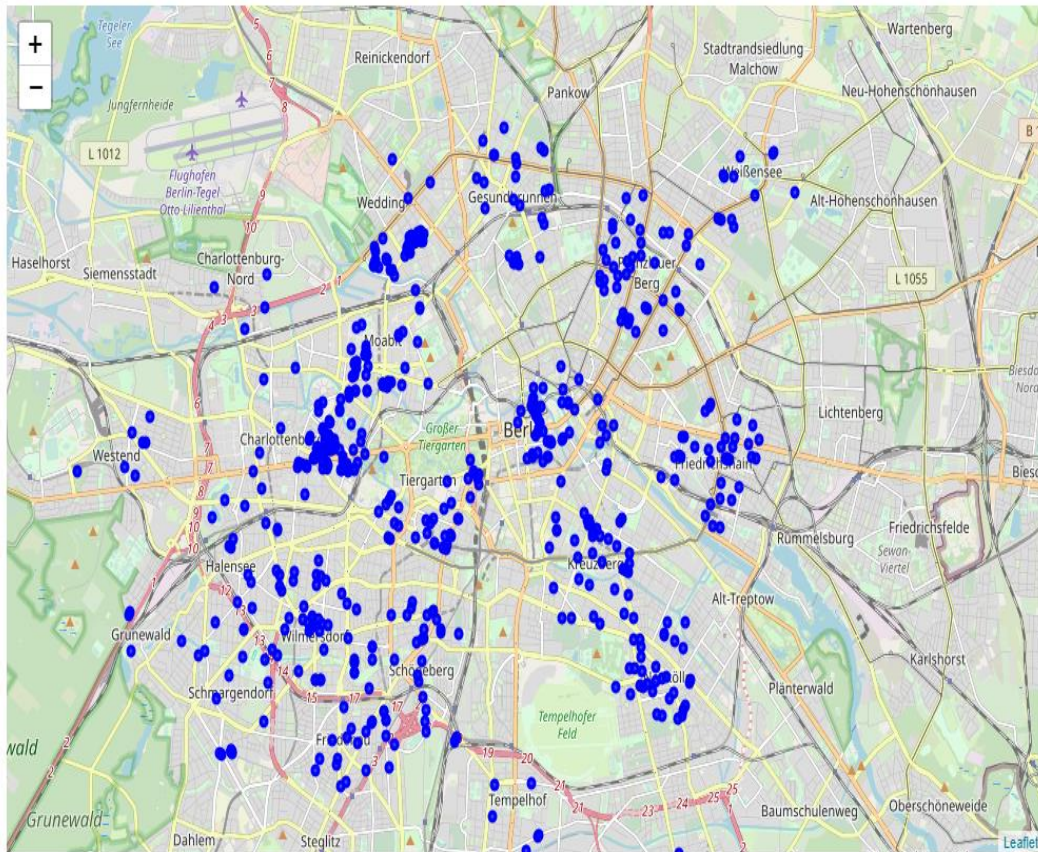


- Total of 902 offices spread over the 39 localities
- Assumption: Nearby offices leads to many customers (lunch, dinner service etc.)
- Positive

University Related Buildings

```
In [30]: map_berlin_universities = folium.Map(location=[latitude, longitude], zoom_start=12)
addToMap(berlin_venues_universities, 'blue', map_berlin_universities)
map_berlin_universities
```

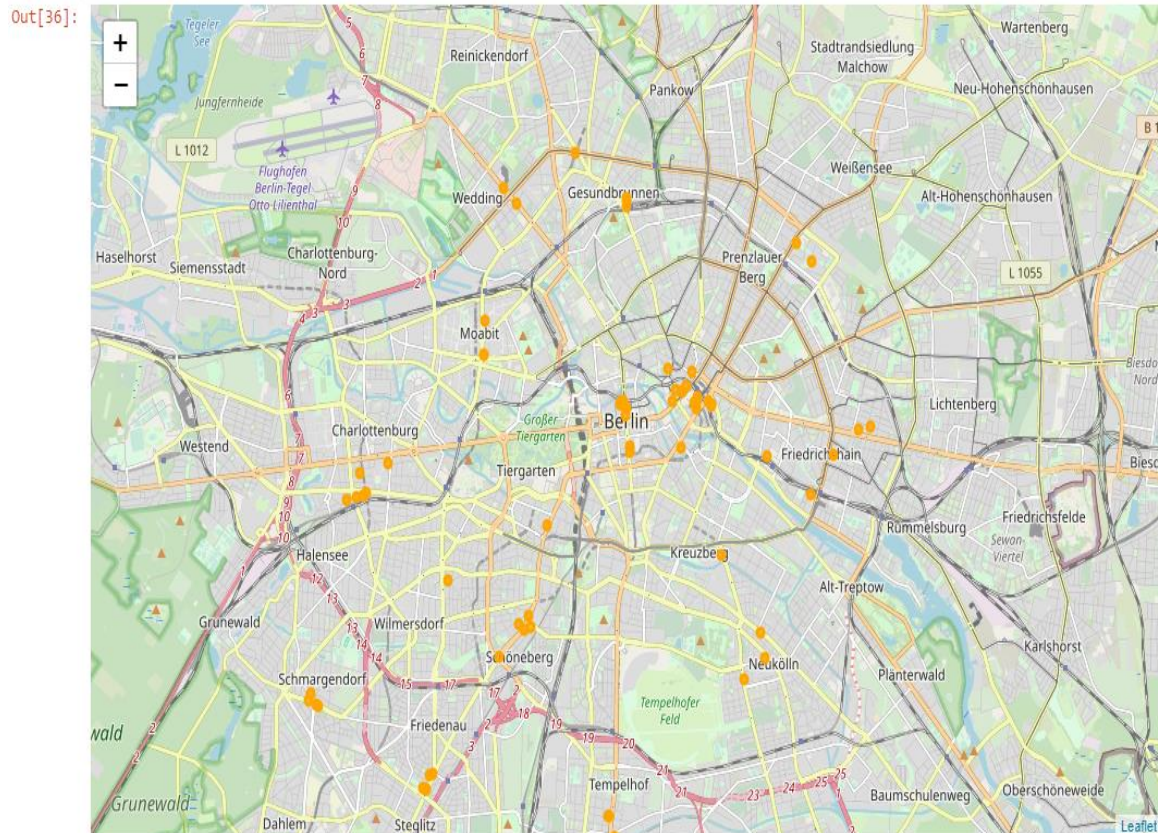
Out[30]:



- Total of 641 University related buildings spread over the 39 localities
- Assumption: University related buildings hold many potential customers
- Positive

Shopping Malls

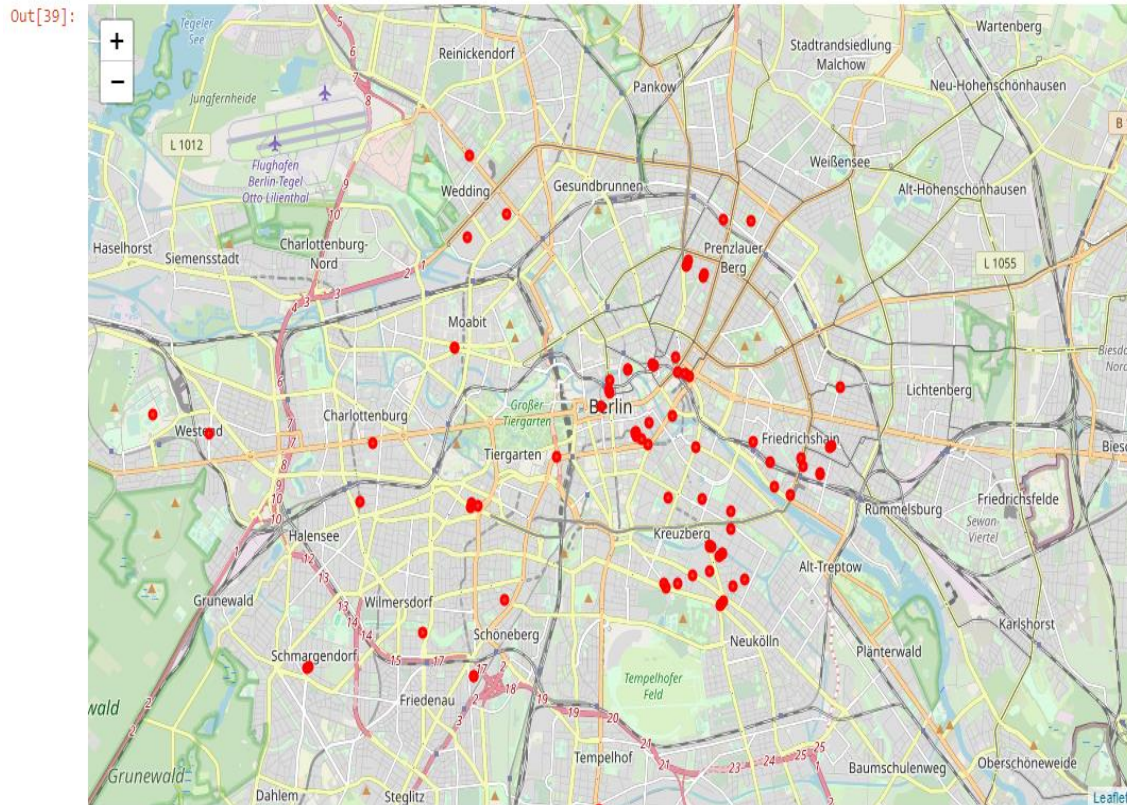
```
In [36]: map_berlin_shopping_malls = folium.Map(location=[latitude, longitude], zoom_start=12)
         addToMap(berlin_venues_shopping_malls, 'orange', map_berlin_shopping_malls)
         map_berlin_shopping_malls
```



- Total of 902 offices spread over the 39 localities
- Assumption: Nearby Shopping Malls will lead to many potential customers, lower weight due to potential competition
- Positive

Food Trucks

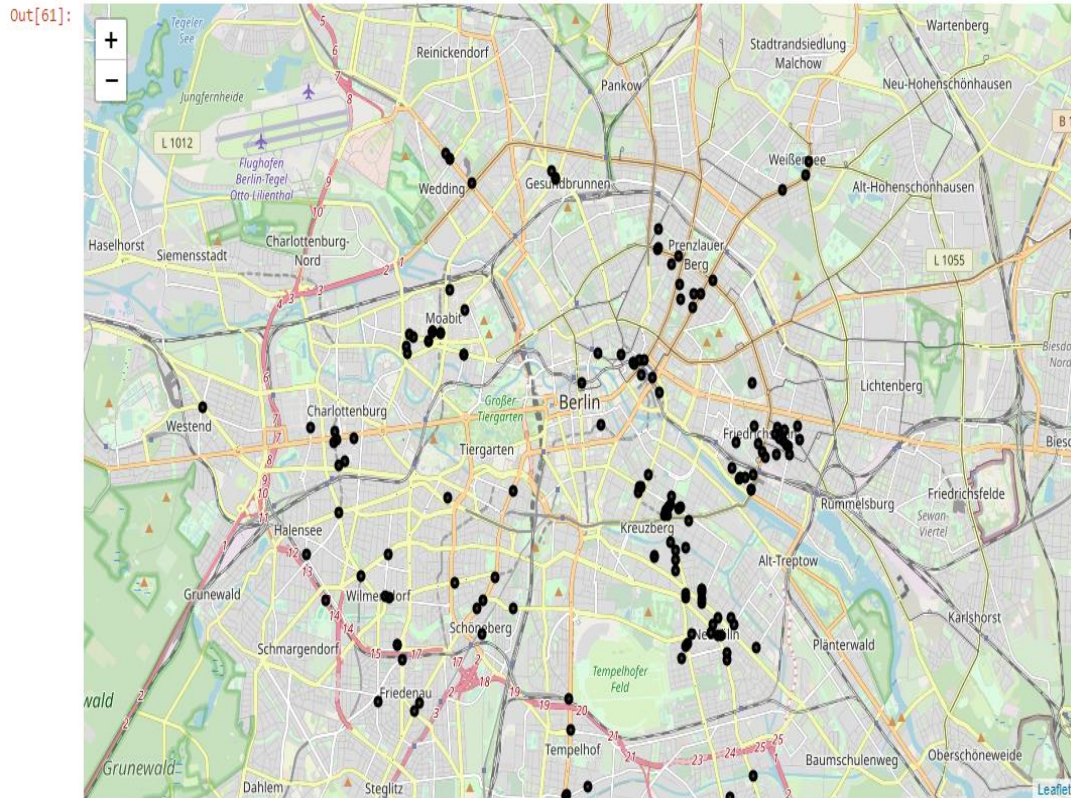
```
In [39]: map_berlin_food_trucks = folium.Map(location=[latitude, longitude], zoom_start=12)
addToMap(berlin_venues_food_trucks, 'red', map_berlin_food_trucks)
map_berlin_food_trucks
```



- Total of 81 offices spread over the 39 localities
- Assumption: Direct competition to the client's Food truck.
- Negative

Burger Joints

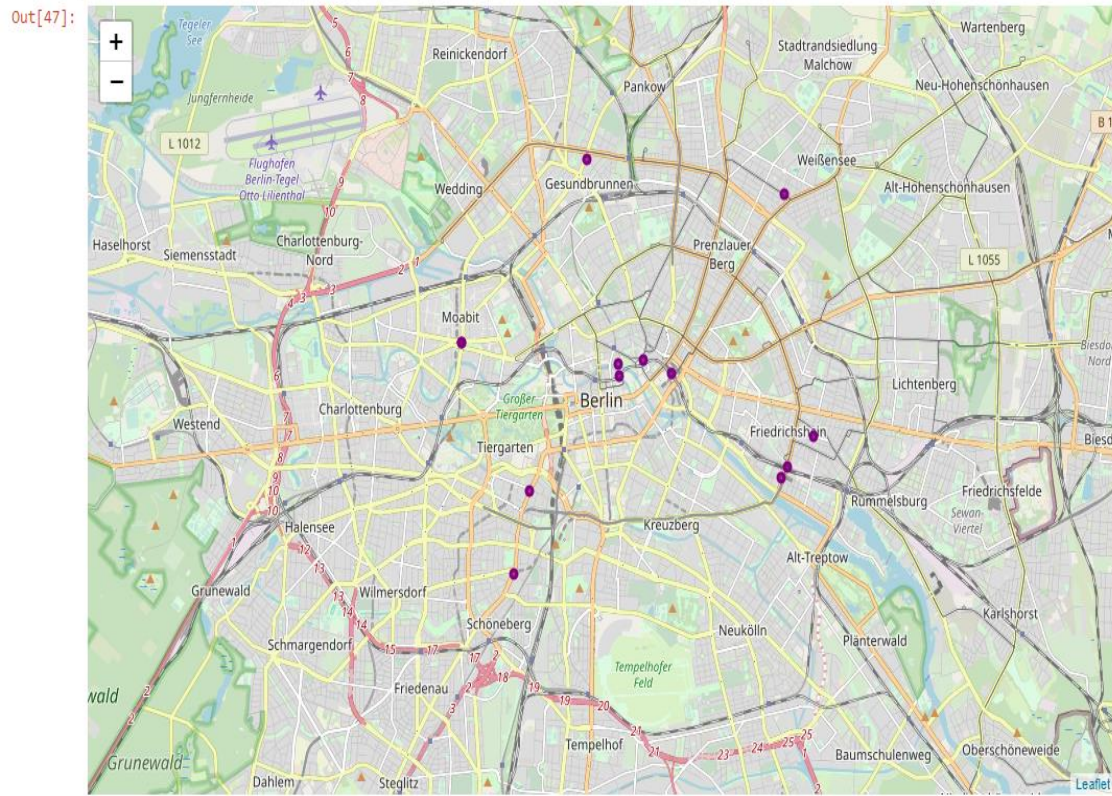
```
In [61]: map_berlin_burger = folium.Map(location=[latitude, longitude], zoom_start=12)
addToMap(berlin_venues_burger_joints, 'black', map_berlin_burger)
map_berlin_burger
```



- Total of 169 burger joints spread over the 39 localities
- Assumption: Direct competition to client's food truck.
- Negative

Hot Dog Joints

```
In [47]: map_berlin_hot_dogs = folium.Map(location=[latitude, longitude], zoom_start=12)
addToMap(berlin_venues_hot_dogs, 'purple', map_berlin_hot_dogs)
map_berlin_hot_dogs
```



- Total of 902 offices spread over the 39 localities
- Assumption: Direct competition to client's food truck
- Negative (less)

Weights

Driver	Effect
Nearby Offices	+1.5
University related buildings	+1.0
Shopping Malls	+0.5
Other food trucks	-1.5
Burger joints	-1.0
Hot Dog joints	-0.5

Results

Using the weights to calculate optimal neighborhood

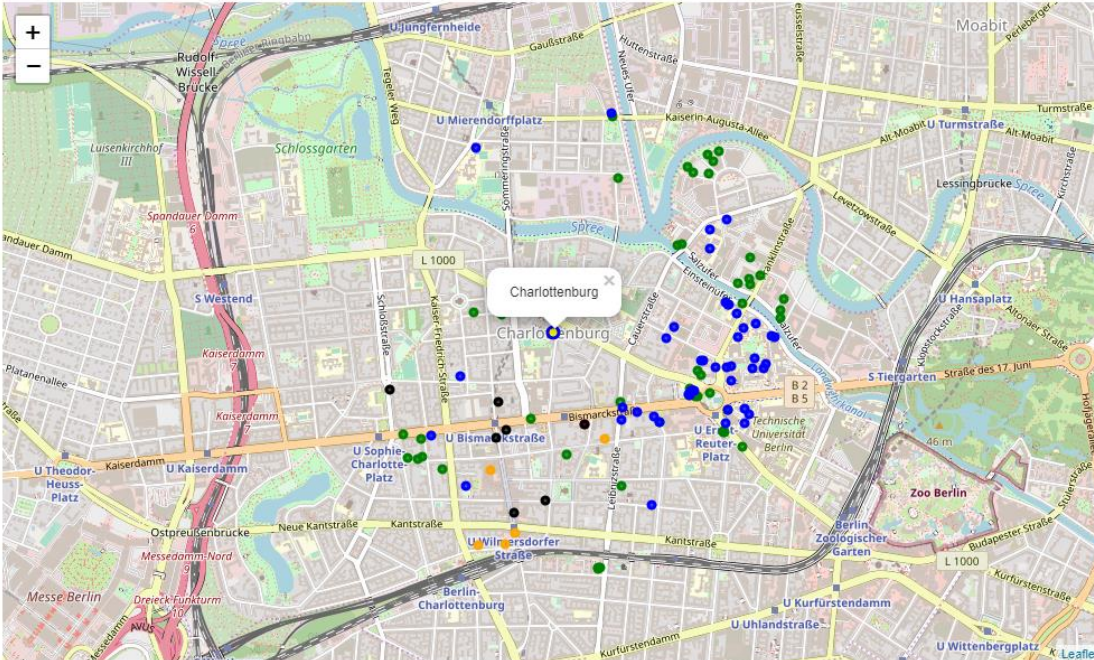
```
In [41]: df_weighted = df_data.copy()
df_weighted['Score'] = df_data['Offices']*weight_offices + df_data['Universities']*weight_universities + df_data['Shopping Malls']
df_weighted = df_weighted.sort_values(by=['Score'], ascending=False)
df_weighted.head()
```

Out[41]:

	Borough	Locality	Latitude	Longitude	Offices	Universities	Shopping Malls	Food Trucks	Burger Joints	Hot Dogs Joints	Score
21	Charlottenburg-Wilmersdorf	Charlottenburg	52.515747	13.309683	48.0	47.0	5.0	1.0	7.0	0.0	113.0
2		Mitte	52.519123	13.341872	47.0	44.0	1.0	1.0	6.0	1.0	107.0
5		Gesundbrunnen	52.550920	13.384846	49.0	27.0	4.0	0.0	3.0	1.0	99.0
3		Mitte	52.509778	13.357260	47.0	36.0	1.0	4.0	2.0	1.0	98.5
22	Charlottenburg-Wilmersdorf	Wilmersdorf	52.487115	13.320330	43.0	36.0	1.0	1.0	5.0	0.0	94.5

map_berlin_results

Out[60]:



- Charlottenburg (Charlottenburg-Wilmersdorf) most suitable locality given assumptions
- Many offices, university related buildings while having little to no competition

Discussion

- Model clearly finds Charlottenburg as most suitable location
- Limited model with weaknesses
 - Excludes important aspects (e.g. population, average income, competitive menu etc.)
 - Able to place the Food Truck anywhere I want
 - Weighting values decided arbitrarily
 - Using a smaller geographical area (e.g. neighborhoods) might improve the accuracy

Conclusion

- Have by using the Foursquare API found the most suitable locality for Food Truck placement (out of 39 localities)
- The same logic can be applied to many situations (e.g. where to open a store, apartments etc.).