# IBM Data Science Capstone Project

( Week 1 , by Uygar Hizal )

## Introduction to the Business Problem:

Hello and welcome to my project introduction.

In this project I will try to solve an imaginary business problem, to present as many skills as possible from the IBM Data Science Capstone multi Course Program in Coursera.

One Foreign Investor wants to invest and open a Clothing Store Business in one of the Germany's big cities. He has a concept in his mind, but as being foreigner, he has not much idea about Germany's City structure and therefore needs help.

He owns already a middle range Clothing Store Chain in USA. And this will be the first Store opened in Germany, therefore, it should meet some criteria to present his brand correctly.

After a meeting with him, he defined his business aim and informed me about the criteria's like following, it should be;

1. Opened in one of the big Cities in Germany (Population over 100.000 and more).
2. Within the max. 15 minutes walking distance from the Geographical coordinates of the City Center
3. As far away from other Clothing Stores as possible
4. As close as possible to Italian Restaurants, because his collections are mostly Italian designs and he thinks, the customers visiting the Italian restaurants can be more interested in store windows as walking
5. As close as possible to Hotels, because guests of the in-city hotels are generally tended to buy clothes nearby.
6. After all He stated honestly that, He needs a city that he can pay less possible salaries as he aims to give 20 employee job.
7. He added also, a city with highest possible unemployment rate would be an advantage for him, as finding personal in a short time. Otherwise he could wait longer to complete all employee team.
8. Population of the city also counts as a positive measure too. (City should be as crowded as possible)

In this point it is very important to interpret investor's desires and convert these sentences to the scientific statements. For example, He saying "…max. 15 minutes walking distance." means for a Data Scientist: with an average walking speed of 5 km/h pedestrian, 1250 meters from the Geocoordinates of that city center. And it will be used in Foursquare Api call as Radius measure (R=**1250**). i.e.:

```
def getNearbyVenues(names, latitudes, longitudes, radius=1250):
```

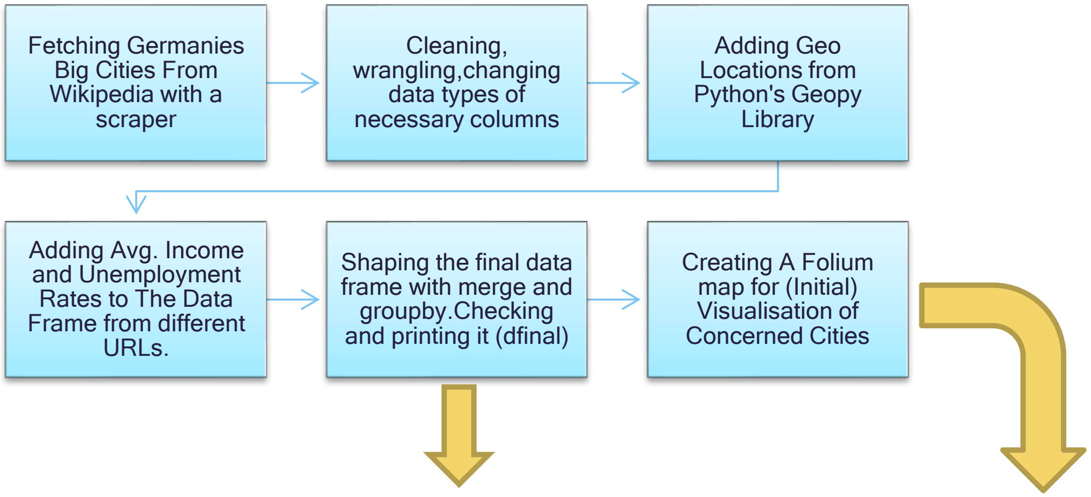
# Necessary Data and its usage in this case:

As you may see from the business problem part above, I decided to add some more complexity to our standard course problem otherwise it could be solved only with foursquare venue data.

But in this case in addition to Venues data of all Major German Cities, we will be adding Socioeconomic information like **Population**, **Average Income /person**, **Average Unemployment Rate** and **Area** in km$^2$ of that city.

### Description of the Data:

Part 1: Socioeconomic Data



Fetching Germanies Big Cities From Wikipedia with a scraper

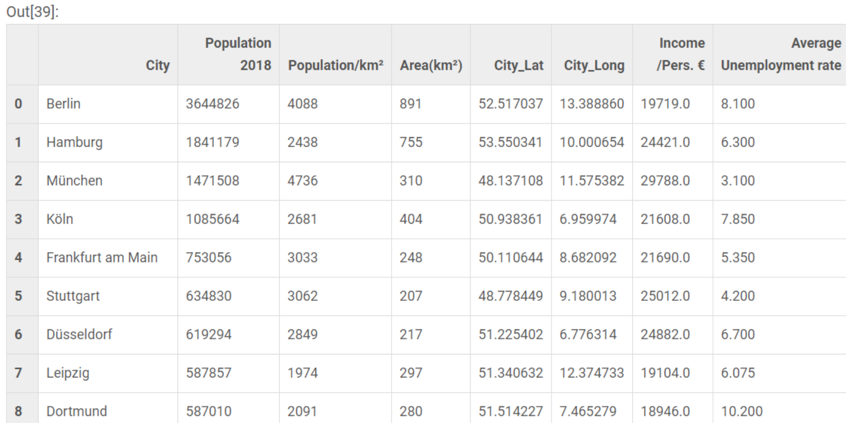
Cleaning, wrangling,changing data types of necessary columns

Adding Geo Locations from Python's Geopy Library

Adding Avg. Income and Unemployment Rates to The Data Frame from different URLs.

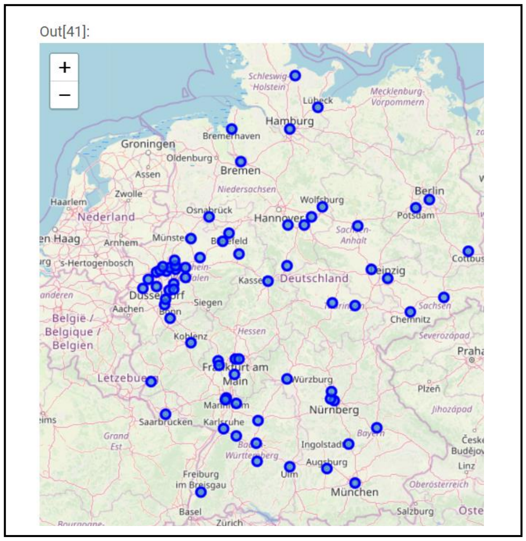
Shaping the final data frame with merge and groupby.Checking and printing it (dfinal)

Creating A Folium map for (Initial) Visualisation of Concerned Cities



Out[39]:

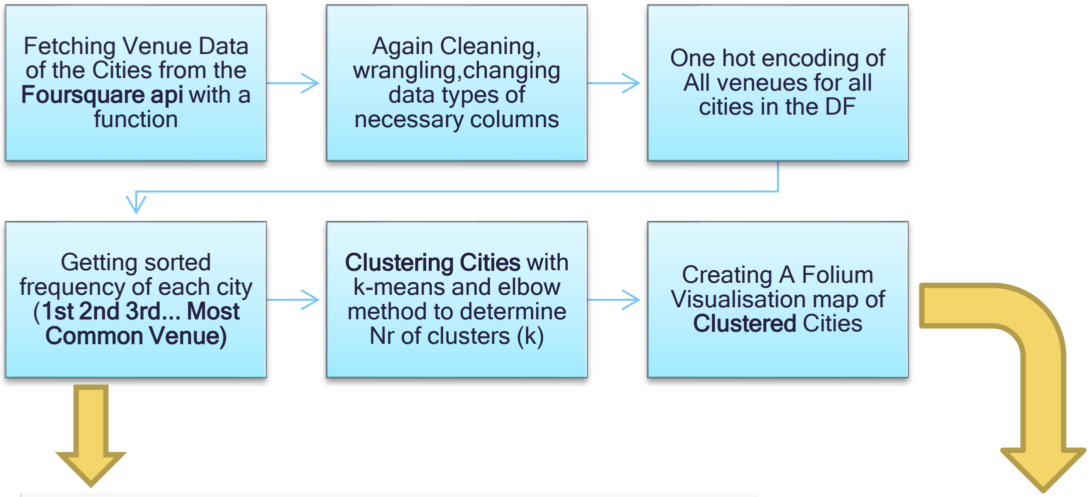
| | City | Population 2018 | Population/km² | Area(km²) | City_Lat | City_Long | Income /Pers. € | Average Unemployment rate |
|---|---|---|---|---|---|---|---|---|
| 0 | Berlin | 3644826 | 4088 | 891 | 52.517037 | 13.388860 | 19719.0 | 8.100 |
| 1 | Hamburg | 1841179 | 2438 | 755 | 53.550341 | 10.000654 | 24421.0 | 6.300 |
| 2 | München | 1471508 | 4736 | 310 | 48.137108 | 11.575382 | 29788.0 | 3.100 |
| 3 | Köln | 1085664 | 2681 | 404 | 50.938361 | 6.959974 | 21608.0 | 7.850 |
| 4 | Frankfurt am Main | 753056 | 3033 | 248 | 50.110644 | 8.682092 | 21690.0 | 5.350 |
| 5 | Stuttgart | 634830 | 3062 | 207 | 48.778449 | 9.180013 | 25012.0 | 4.200 |
| 6 | Düsseldorf | 619294 | 2849 | 217 | 51.225402 | 6.776314 | 24882.0 | 6.700 |
| 7 | Leipzig | 587857 | 1974 | 297 | 51.340632 | 12.374733 | 19104.0 | 6.075 |
| 8 | Dortmund | 587010 | 2091 | 280 | 51.514227 | 7.465279 | 18946.0 | 10.200 |



Out[41]:

## Part 2: Cities Venue Data

After obtaining and cleaning Socioeconomic data in part 1, now it is time to get all the venues for concerned cities.


Fetching Venue Data of the Cities from the **Foursquare api** with a function

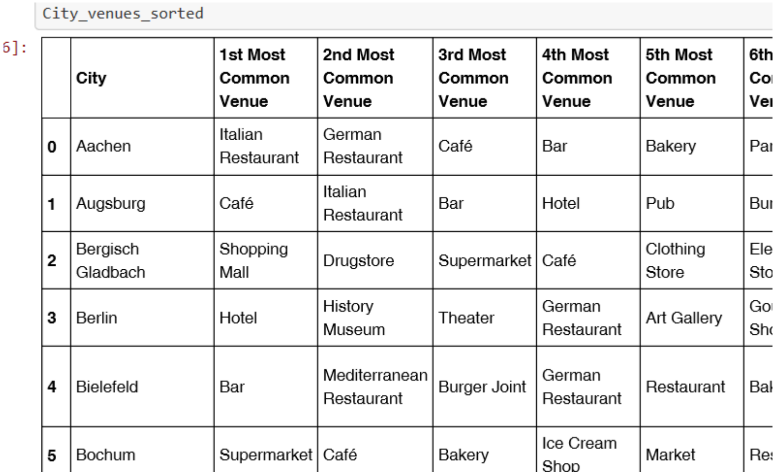
Again Cleaning, wrangling,changing data types of necessary columns

One hot encoding of All veneues for all cities in the DF

Getting sorted frequency of each city **(1st 2nd 3rd... Most Common Venue)**

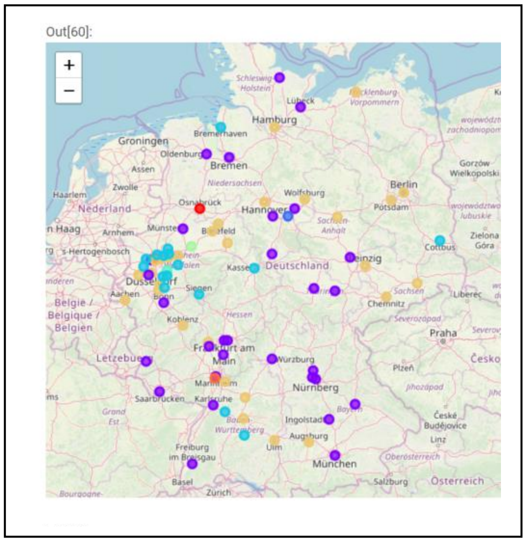
**Clustering Cities** with k-means and elbow method to determine Nr of clusters (k)

Creating A Folium Visualisation map of **Clustered** Cities


```
City_venues_sorted
```

6]:

| | City | 1st Most Common Venue | 2nd Most Common Venue | 3rd Most Common Venue | 4th Most Common Venue | 5th Most Common Venue | 6th Co Ve |
|---|---|---|---|---|---|---|---|
| 0 | Aachen | Italian Restaurant | German Restaurant | Café | Bar | Bakery | Pa |
| 1 | Augsburg | Café | Italian Restaurant | Bar | Hotel | Pub | Bu |
| 2 | Bergisch Gladbach | Shopping Mall | Drugstore | Supermarket | Café | Clothing Store | Ele Sto |
| 3 | Berlin | Hotel | History Museum | Theater | German Restaurant | Art Gallery | Go Sho |
| 4 | Bielefeld | Bar | Mediterranean Restaurant | Burger Joint | German Restaurant | Restaurant | Bal |
| 5 | Bochum | Supermarket | Café | Bakery | Ice Cream Shop | Market | Re |

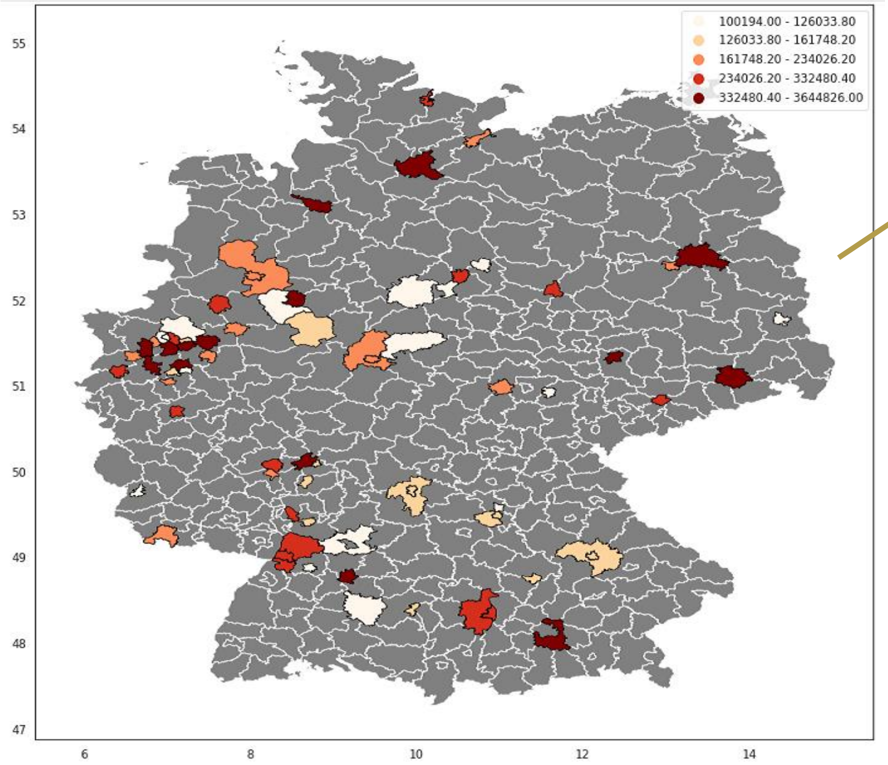

Out[60]:

**List of Data Sources:**

- Foursquare API data based on free API calls
- Wikipedia site for Major Cities List in Germany
- Federal Statistical Office of Germany – (Statistisches Bundesamt) https://www.destatis.de/

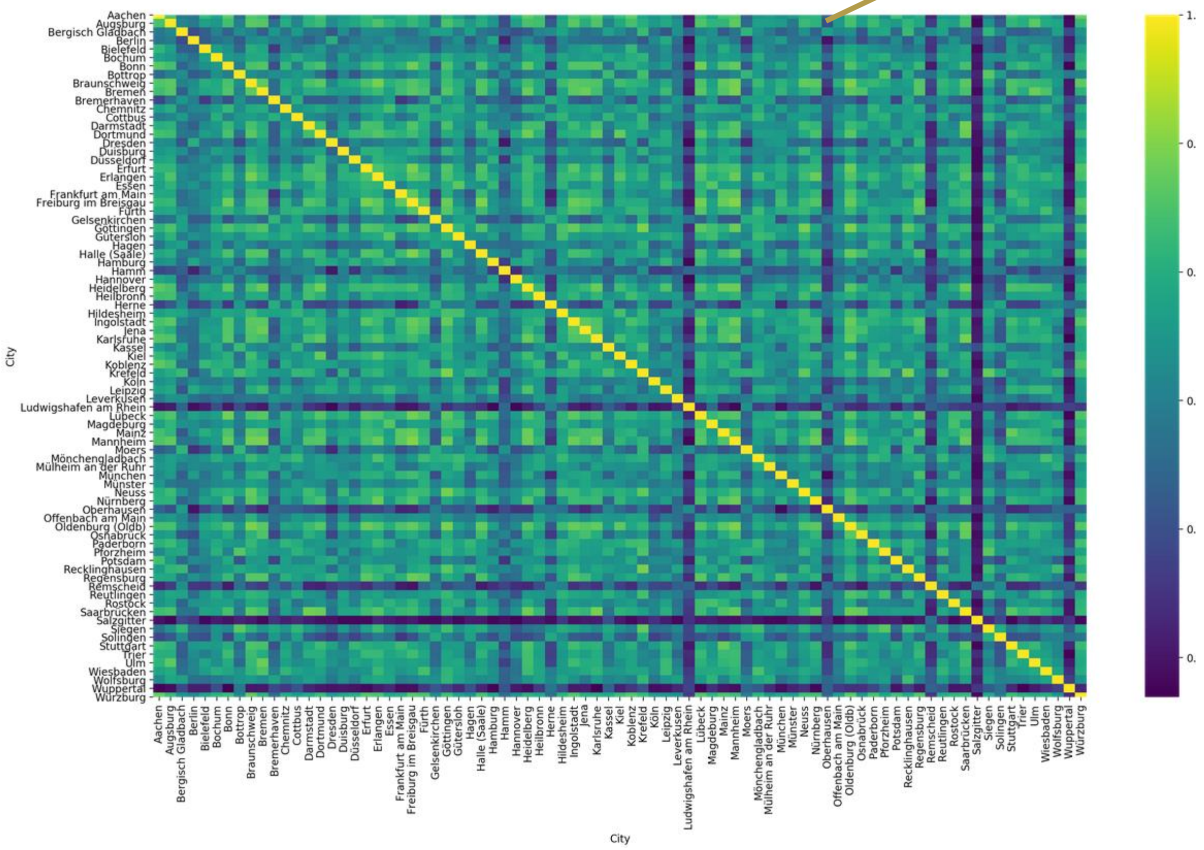
# Some Extra informative Visualizations:

I have already added some extra visualizations to present achieved data frames better. They are not in the scope of the assignment but I hope, readers can find useful information and see how to apply some other methods to their projects.



Geo Pandas Library as an alternative map (Choropleth Map of Cities population)

In [87]:



Seaborn Correlation Matrix of Cities (Venue Data)

After the preparation of both Socioeconomic and Hot encoded Venue information into the two different data frames, we still should combine / interpret these two data frames for a logical solution. At this point We will use the weighted properties matrix of investor desires.
Every feature that investor prerequisites from us, will be weighted to a scale from 1 to 10 in a manner of importance:

1. Opened in one of the big Cities in Germany (Population over 100.000 and more)

   This is already satisfied because we have fetched cities only with population >100k

2. Within the max. 15 minutes walking distance from the Geographical coordinates of the City Center

   This request is converted to an variable to use in Foursquare Api call (Radius = 1250 meters in search )

3. As far away from other Clothing Stores as possible

   This request is very important for him and weighted as 9 points over 10 points. **(0,9)**

4. As close as possible to Italian Restaurants….

   This request is somehow second degree and weighted as 5 points over 10 points. **(0,5)**

5. As close as possible to Hotels ….

   This request is second degree in importance but still weighted as 7 point over 10 points. **(0,7)**

6. Cities that statistically less possible salaries are paid ….

   For the investors salary issues are always important (☹), So it is weighted as 8 point. **(0,8)**

7. Cities with highest possible unemployment rate …

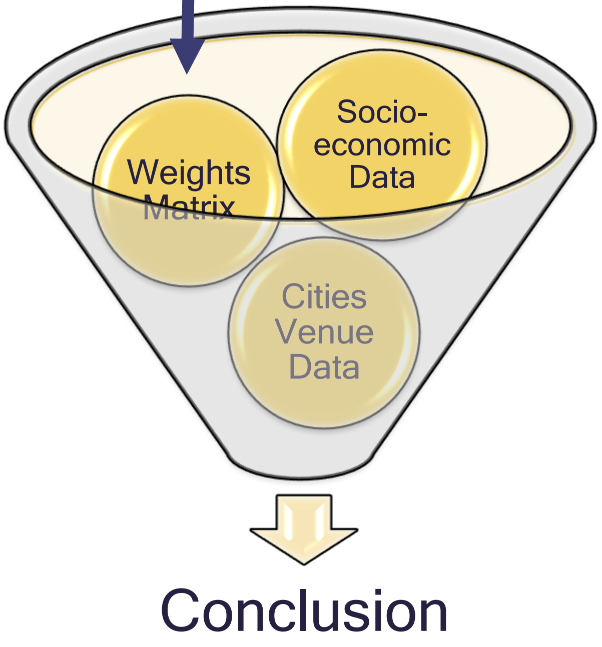
   This request is also important but not more than salaries. So, gets 7 points over 10 pts. **(0,7)**

8. Population of the city also counts as a positive measure too. (City should be as crowded as possible)

   Population value is directly related to the number of potential customers therefore it will be weighted as 8 points over 10. **(0,8)**

$$\overset{\underline{W1}}{\left\{\begin{matrix}\text{Request 3} & 0{,}9\\ \text{Request 4} & 0{,}5\\ \text{Request 5} & 0{,}7\\ \text{Request 6} & 0{,}8\\ \text{Request 7} & 0{,}7\\ \text{Request 8} & 0{,}8\end{matrix}\right\}} \text{ x } \overset{\underline{C1}}{\left\{\begin{matrix}-1\\ 1\\ 1\\ -1\\ 1\\ 1\end{matrix}\right\}} = \text{Weights Matrix}$$

Correlation bit: (-1 for negatively correlated variables and +1 is for positive correlation) For example increase on value of the Request 3 variable (Amount of Clothing stores in the area ) will be added as punishment to end Data Frame , as it is not desired. But increase on population is positively counts (+) to end Data frame, as it is an advantage)



## Conclusion and final words will be added in week2

(not the scope of this week’s assignment)

Thank You for your Reading

Uygar Hizal