Best spot for a new brewery on São Paulo : An application of scrapping techniques, kmeans classification method and mapping visualizations.

Alvimar Lousada[a]

[a]*Data Science Msc student at Univerisdade do Porto, Quality Engeneer by Univerisdade de Campinas and Master Brewer certified by Davis California Univeristy Mail: alvimar.lousada@gmail.com*

Abstract

One common problem for new business on brewery restaurants is to identify the right spot for their installations, which should be a perfect match between their desirable approach to costumers and the available neighborhood for the choosen commertial points. São Paulo city, the big capital of São Paulo state on Brazil is a huge center where each year dozens of new breweries are set down its presence, hence our scope is indicate best trends for a new brewery placing on São Paulo city using data Science techniques like scrapping, data refinement using python coding, classification methods and mapping visualizations. At the end the target is arrange a method which could be replied to other business and cities as well.

1. Introduction

1.1 Background : we are living in the age of big data (the large data set). The data that are produced by various kinds of sources like social media (Twitter, FaceBook, and Whatsapp), text files, video files, images, machine's log files makes the data growing into a very large amount. Big data has the following characteristics namely, volume, velocity, and variety. The traditional database management systems are not enough to handle these very large datasets. Therefore, there is an essential need to develop the new systems to store and analyze the high volume of data called as big data analytics. A variety of methods at this context can support multiple solutions for problems like the costumer segmentation or identify a best place for a new comercial enterprise.

1.2 The Problem : This paper is mainly concerned with extract information using scrapping techniques to obtain from available addresses guides sites the breweries present on São Paulo city locality data and clustering them based on K-Means algorithm, as well build them in visualizations maps for the final information using Python language. **So the key objective here therefore is enroll a method able to bring clear trends for fixing a new brewery on São Paulo city using Data Science methods.**

Section 2 details the method used for data acquisition and cleaning. Section 3 gives understading about the exploratory analysis of the summary data. Section 4 progress the classification methods and mapping tecniques of visualization, Section 5 deals with the data discussion based on the applied methods here deployed. Section , a brief conclusion.

2. Data acquisition and cleaning

The first step on the chosen process was defined the data to search in possible sites which could supply them for the target analysis. In this case the needed data was:

- Store Name (text);
- Address (text);
- Postal Code (text);
- Locality (city, text).

2.1. Addresses Guides Sites, data acquisition

Three criterias was used to define the search site to acquire the data, the permition checked on the site for scrapping the data, the availability of data looked for and the workability to apply scrapping techniques using Beatiful Soup library on Python.

A simple Python code it was chosen to confirm the site endorsement for scrapping*:

```
if r.status_code==200:
  print (f"{bcolors.OKGREEN}This site accept web scrapping.")
else:
  print(f"{bcolors.FAIL}Scrapping this site is not allowed.")
```

```
This site accept web scrapping.
```

*scrapping: one coding technique to collect available data on chosen sites in automatic fashion, using HTML tags references to identify the interested fragments of information.

2.2. The sctructured data acquired

The collected data was arranged in a dataframe using Pandas library available on Python. Necessary corrections to format the data was needed to achieve the final result:

	Store_Name	Locality	Postal_Code	Street_Address	Street_Name	Number	Neighborhood
0	Lanchonete Choperia Biiu Biiu Ltda - Me	São Paulo	08140-000	Rua Tibúrcio Sousa, 148, Itaim Paulista	Rua Tibúrcio Sousa	148	Itaim Paulista
1	Choperia Takeu Ltda	São Paulo	01510-001	Rua Glória, 523, Liberdade	Rua Glória	523	Liberdade
2	Cervejaria Nacional	São Paulo	54200-01	Avenida Pedroso Morais, 604,	Avenida Pedroso Moraes	604	
3	Toca do Nego Velo Bar & Cervejaria	São Paulo	53520-50	Rua Doutor Teodoro Quartim Barbosa, 114,	Rua Doutor Teodoro Quartim Barbosa	114	
4	Restaurante Choperia Tche Gaúcho Ltda	São Paulo		Estrada Guarapiranga, 884, Guarapiranga	Estrada Guarapiranga	884	Guarapiranga

After a first summary of the data was found some cleaning tasks due to the lack of information or presence of stores with very low affinity with the brewery topic.

Starting from a initial dataset with 2234 rows, each row being one sellected store, after the sequential filters application was remained 91 stores, such filters used was:

- ✓ Remove not valide rows with error on the scrapping.
- ✓ Remove rows with none affinity using stoping words.
- ✓ Remove rows with zero score in one of the three searching key topic words (beer, brewery, good beer).
- ✓ Remove rows with incomplete information (like not complete or absente postal code).

Still to select stores with the brewery theme affinity was,

I. used scraping tools joined with yahoo searching engines ranking, collecting the key references words obtained from the search based on each store name on the dataset.

II. applied a similarity algorithm using Jacarhd distance [1] to evaluate if the collected words obtained in the step I has significant affinity to the topic words chosen for data refinement : beer, brewery and good beer* (searching performed in portuguese language: cerveja, cervejaria, boas cervejas).

The similarity measurement let on define rankings for each store on the dataset according to the seen quotes for each store searched in yahoo using scrapping approach with Beatiful Soup.

In that way some main results were feasible,

- ✓ excluding the stores with none reference to the topic (with zero score).
- ✓ quantifying the similarity of the beer stores selected with chosen topic words using scores helping further to run the classification of the stores.

Postal_Code	Street_Address	Street_Name	Number	Neighborhood	Num_oc_geral_1	Score_geral_1	Num_oc_geral_2	Score_geral_2	Num_oc_espec_1	Score_espec_1
08140-000	Rua Tibúrcio Sousa, 148, Itaim Paulista	Rua Tibúrcio Sousa	148	Itaim Paulista	1	0.000000	0	0.000000	1	0.000000
01510-001	Rua Glória, 523, Liberdade	Rua Glória	523	Liberdade	0	0.000000	13	0.008721	13	0.008721
54200-01	Avenida Pedroso Morais, 604,	Avenida Pedroso Morais	604		0	0.000000	0	0.000000	14	0.010906

```
[ ] key_words = 'cerveja beer artesanal cervejaria'
    text=""
    minhash_1=[]
    count_f=[]
    score_f=[]

    opt_keyw    = ["cerveja","cervejaria","boas cervejas"]
    opt_nam_col = ["geral_1","geral_2","espec_1"]

    url1 = 'https://br.search.yahoo.com/search?p='

    k=0
    while k<3:

      for j in tqdm(range(df.shape[0])):

        #print(opt_nam_col[k])

        score_1_médio=0
        score_1=[]
        count=0

        key_word=opt_keyw[k]

        text = '"' + str(df["Store_Name"].loc[j:j]) + " " + str(df["Locality"].loc[j:j]) + " " + key_word + '"'
        soup = browse(url1,text,j)
```

2.3. Geolocation data

After refining the dataset accomplished, aiming the objective of this work with a visualization map of the present ranked breweries in São Paulo based on the engined algorithm, it was directed to find the geolocation coordinates for each filtered store brewed collected.
This was done using the Google API Geolocator [2], one usual tool that given specific data (postal code, adress, neighborhood and locality) it returns the linked latitude and longitude coordinates.

```
[ ] #found latitudes and longitudes on final dataset

    from time import sleep
    from geopy.geocoders import GoogleV3

    ini=df.shape[0]
    lati_ = []
    long_ = []

    for i in tqdm(range(0,ini)):

      CEP    = df['Postal_Code'].loc[i:i].to_list()[0]
      RUA    = df['Street_Name'].loc[i:i]
      NUM    = df['Number'].loc[i:i]
      BAIRRO = df['Neighborhood'].loc[i:i]
      CIDADE = df['Locality'].loc[i:i]

      if len(CEP) == 9 and CEP[8]!=" ":

        line = CEP + " , " + RUA + " , " + NUM + " , " + BAIRRO + " , " + CIDADE

        sleep(2)
        geolocator = GoogleV3(api_key='      YOR KEY OBTAINED AT GOOGLE CLOUD      ') #Chave adquirida através da Google
        location = geolocator.geocode(str(line), timeout=100)
        if location:
          lat=location.latitude
          lon=location.longitude
        else:
          lat = None
```

As said, this is quite necessary for the final visualization step.

3. Exploraty analysis of the collected data

In this section it is analysed the results summarised on the final dataset obtained. Using the method describe in Python it was found the following statistics resume of the numeric data.

	Num_oc_geral_1	Score_geral_1	Num_oc_geral_2	Score_geral_2	Num_oc_espec_1	Score_espec_1	latitude	longitude
count	91.000000	91.000000	91.000000	91.000000	91.000000	91.000000	91.000000	91.000000
mean	10.978022	0.008693	10.780220	0.008815	10.802198	0.008728	-23.544906	-46.633162
std	3.319900	0.009333	3.418259	0.008625	3.383686	0.008962	0.161395	0.057926
min	2.000000	0.000999	2.000000	0.000999	2.000000	0.001054	-23.713724	-46.741370
25%	8.500000	0.003311	8.000000	0.003940	8.000000	0.003311	-23.580074	-46.679869
50%	12.000000	0.006209	12.000000	0.006534	12.000000	0.006534	-23.559216	-46.646123
75%	14.000000	0.010906	14.000000	0.010701	14.000000	0.010906	-23.534832	-46.587967
max	16.000000	0.061224	14.000000	0.061224	14.000000	0.061224	-22.070986	-46.458516

As already detailed it was used three keywords to calculate scores for each selected store with the focus topic brewery. In consequence the three scores found was:

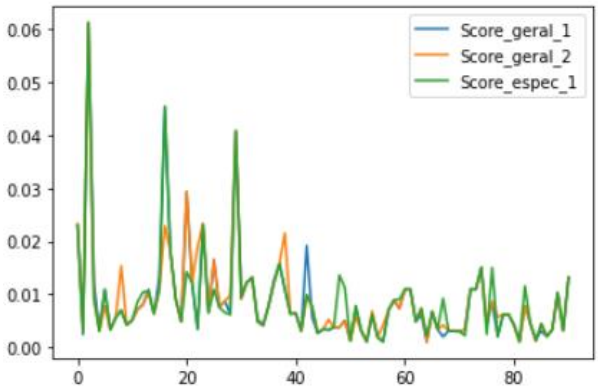
- Score_Geral_1 (related to the word beer, “cerveja”).
- Score_Geral_2 (related to the word brewery, “cervejaria”).
- Score_espec_1 (related to the words : good beer, “boas cervejas”).

The score was calculated based on the Jacard distance, which means it measures the expected distance between each store name plus the topic words chosen (beer, brewery and good beer, translated to portuguese) to the results obtained in a yahoo searching results quotes. Also, and complementary, it was measured the number of quotes found in such search.

3.1 Results found in graphic resume

The ranking found for the 91 brewery stores in São Paulo city to the filetered dataset was,

Figure 1 : scores representation for the stores selected.

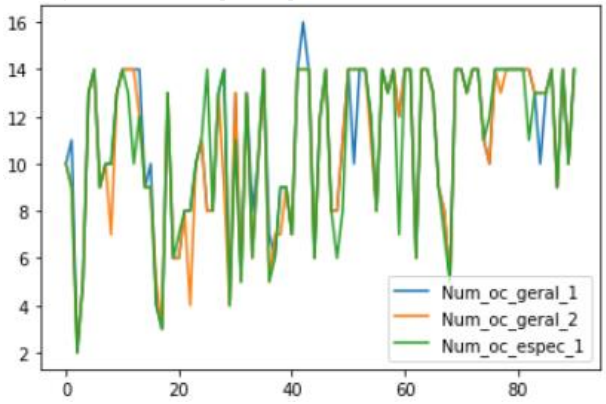


Score_geral_1 : store name + the word beer* in yahoo search.
Score_geral_2 : store name + the word brewery* in yahoo search.
Score_espec_1 : store name + the word good beer*.

*all topic words translated to portuguese.

And the number of quotes settled,

Figure 2 : number of quotes representation for the stores selected.



4. Methodology : K-Means classification and mapping visualizations

4.1 K-Means clustering

K-means is one of the most widely used and simple unsupervised clustering algorithms, which allocates the instances (unlabeled data) to different clusters based on their similarity with each other [3].

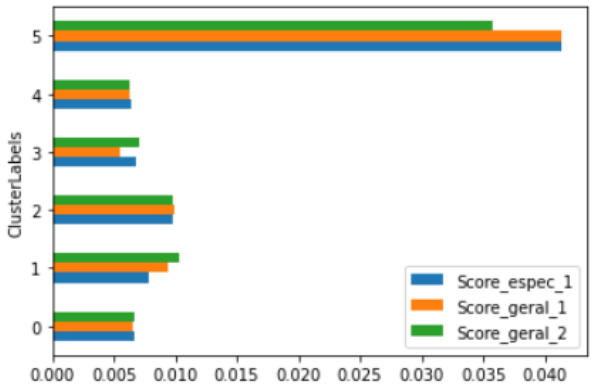
The similarity is calculated based on the distance between the unlabeled distance. K-Means is intuitive, easy to implement, and fast. It works by chosen one number of desirable clusters, and the calculation will proceed finding non overlaping groups of data joined by their distance from each other, or similiraty.

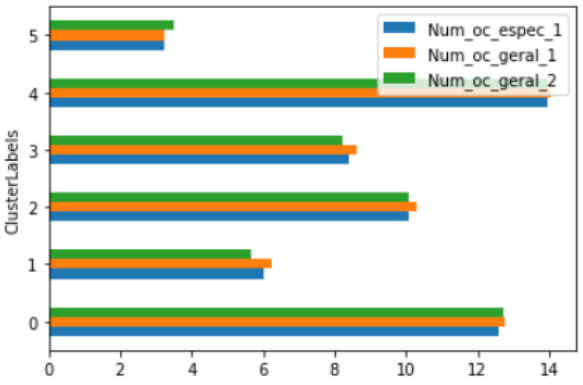
The kmeans implementation on Python used at end was,

It was tested and validated the number of cluster as 6 as the statisticals results show groups with specifics charecteristics detailed bellow.

Therefore the clustering statitiscs resume found were,

Figure 3,4 : score and number of quotes calculated for each cluster.



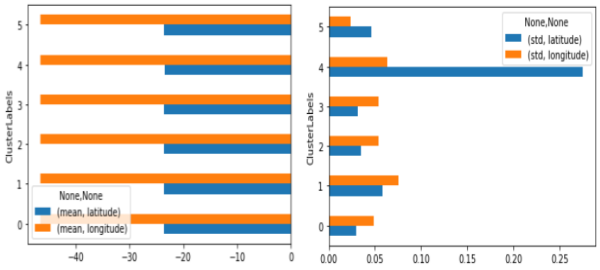


The six clusters have clear difference towards the topic word scores and number of citations. Cluster number 5 has a highlighted better score, while keeps a lower number of citations on yahoo search.

Another view shows how simmilarities of the clusters are arranged front of their geolocations parameters.

In average the clusters has the same location, however When we check the deviation on their positions inside the clusters for each brewery store there is visible differences. In some clusters the breweries are closer from each other, as in another ones they are far from each other.

Figure 5 : profile of the clusters geocordinates average and standard deviation.



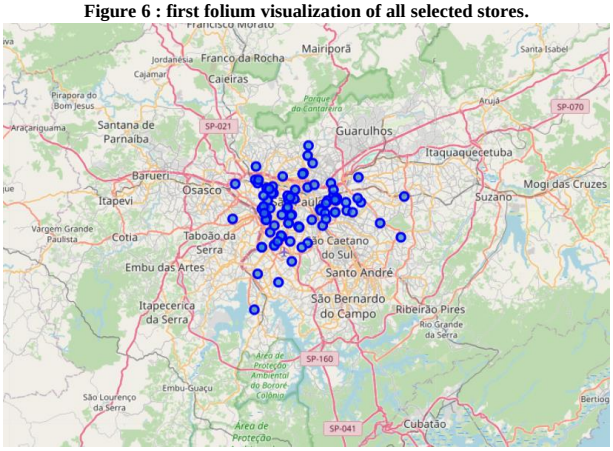
4.2 Mapping visualizations with Folium

Folium is a Python library used for visualizing geospatial data. It is easy to use and yet a powerful library. Folium is a Python wrapper for Leaflet. js which is a leading open-source JavaScript library for plotting interactive maps [4].

In this paper this library it is used to show the results obtained, as the target of the paper is present visual trends for new breweries placing their new breweries.

Bellows, it is show the final breweries stores position on São Paulo city.

Figure 6 : first folium visualization of all selected stores.



The breweries are spreaded accross the city, some of them reaching the vicinity cities part of Grande São Paulo, the metropolitan area.

The Python implementation used for this result was,

```
# create map
map_clusters = folium.Map(location=[-23.562477,-46.630399],zoom_start=11)

# set color scheme for the clusters
x = np.arange(k)
ys = [i + x + (i*x)**2 for i in range(k)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, neighbourhood, cluster in zip(df_f['latitude'], df_f['longitude'], df_f['Neighborhood'], df_f['ClusterLabels']):
    label = folium.Popup(' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)
```

The next step is mapping the same points (each one a brewery spot), but this time showing the K-means clusters label found.

Figure 7 : folium visualization of the clusters obtained.

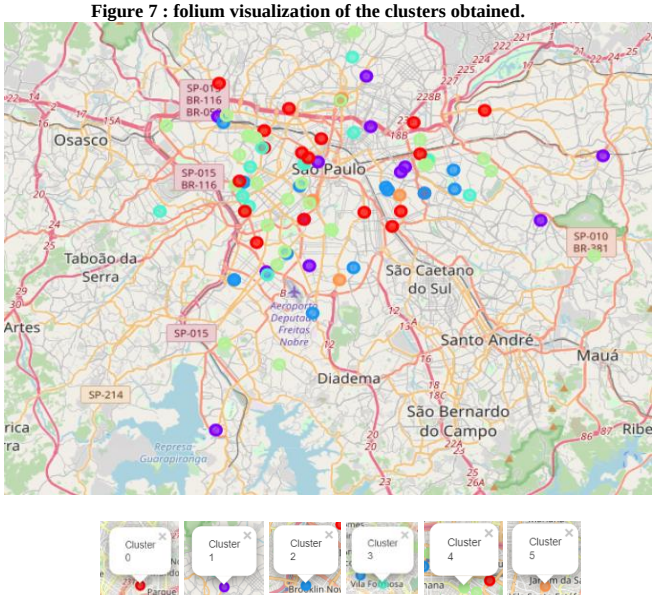


Table1- Clusters resume

Cluster	Number of Outliers*	Number of breweires on the cluster	Overall Scores Sum % **
0	3	19	9%
1	1	12	12%
2	2	13	13%
3	2	13	8%
4	1	30	8%
5	0	4	51%

*outliers in this case is stores that has no relation with the topic brewery.

**the sum of the three generated scores using key topic yahoo searching plus the store name.

6. Discussion

The k-means algorithm shows a 90% level of confiability at the end, which could be meanfully improved using artifitial intelligence approachs.

However with the final processed data it is possible to see trends for the present breweries installed, which is a support tool to a decision taken for a new commertial point.

If seen one specific portion of the map, bellows, it is possible to check where is available some gaps for new business points positioning considering empty spaces, placement near lower ranking breweries or closer to a specific one better ranked. The commertial strategy will conduct the decision on the available data.

Figure 8 : folium details of one map segment.

At the end Data Science is a tool that will support a strategic decision aware of the error boundaries of the used model.

Another positive point it was observed that the ranking method using a search engine with scrapping techniques could bring specific charecteristics for each brewery sampled together with its geolocation data.

The geolocation data was also a crutial part on this process, between the possibilities looked for the google geolocator API show the best option between the options searched.

7. Conclusion

In this paper an approach has been made to implement an Python code which is possible to deliever possible trends to support a decision to determine the better place for a new brewery in São Paulo city, Brazil.

It was used k-means algorithm and Folium map visualizations to have a final picture as map of São Paulo with the brewerie positions labeled by clusters.
The clusterization was possible introducing a ranking method, where each store name was crossed with some key topic words using an automatic scrapping searching engine.
Finally it can being affirmed that the 90% confiability found on this model for the methodology is motivating, as a intial approach with limted optmizations implemented, it opens space to improve the model with more robust methods using another machine learning paths or artifitial inteligence methods.

The Python code with details of all implementation are available on github [5].

REFERENCES

Mahmoud Harmouch, “17 types of similarity and dissimilarity measures used in data Science.” Towards Data Science.

Bruna Candeia, (2021). Geocodificação de Endereços: a melhor geotecnologia de todos os tempos da última semana, Linkedin.

S. Josephine Isabella, Sujatha Srinivasan (2018). Analysis of k-means algorithm for big data analytics using R language, International Journal of Advanced Stuides of Scientific Reasearsh), pag.233-235.

Abishek Sharma, (2020). Your Guide to Getting Started with Geospatial Analysis using Folium (with multiple case studies), Analytics Vidhya.

Alvimar Lousada (2021). Python Code – a new spot for a brewery on São Paulo. *Github*.