Clustering São Paulo's Cities

Based on COVID-19 metrics

Jorge Messa Junior

June, 2020

Problem

How to group cities based on COVID-19 metrics?

Approach

1. Use data from SEADE – a foundation related to São Paulo State Government [1];

2. Use data from Foursquare API to retrive hospitals close to a location;

3. Create a final dataset from SEADE and Foursquare with new columns;

4. Run a K-Means algorithm to find clusters.

Creating new Columns

New columns are created to be provided to K-Means algorithm:

- **number_days:** number of days since first infection reported in a city.
- **cases_100M:** number of cases for each 100.000 habitants.

$$cases\_100M = \frac{cases}{pop} \times 100.000$$

Creating new Columns

- **IFR:** Infection fatality rate.

$$IFR = \frac{deaths}{cases}$$

- **deaths_100M:** number of cases for each 100.000 habitants.

$$deahts\_100M = \frac{deaths}{pop} \times 100.000$$

Creating new Columns

- **pop_60_rate:** population with age greater or equal 60 year / total population.

$$pop\_60\_rate = \frac{pop\_60}{pop}$$

Foursquare® API

From Fourquare®, the number of hospitals was retrieved from venues search by using hospital **category id** and choosing a radius of search.

To estimate the radius of search, cities were modeled **as circles** and the radius was calculated with the formula below:

$$r = \sqrt{\frac{Area}{\pi}}$$

Number of Hospitals

After retrieving number of hospitals from Foursquare, the following column was created.

number_hospitals_100M: number of hospitals for each 100.000 habitants.

$$number\_hospitals\_100M = \frac{number\_hospitals}{pop} \times 100.000$$

Final Dataset

Example of first 4 rows of the final dataset.

name	cases_100M	IFR	deaths_100M	pop_60_rate	number_hospitals_100M
Adamantina	129,8165	0,068182	8,851124	0,218269	11,8015
Agudos	63,65196	0,043478	2,767477	0,152875	35,9772
Americana	50,54442	0,050847	2,570055	0,172519	6,425139
Amparo	119,1861	0,060241	7,179885	0,182757	4,307931

StandardScaler

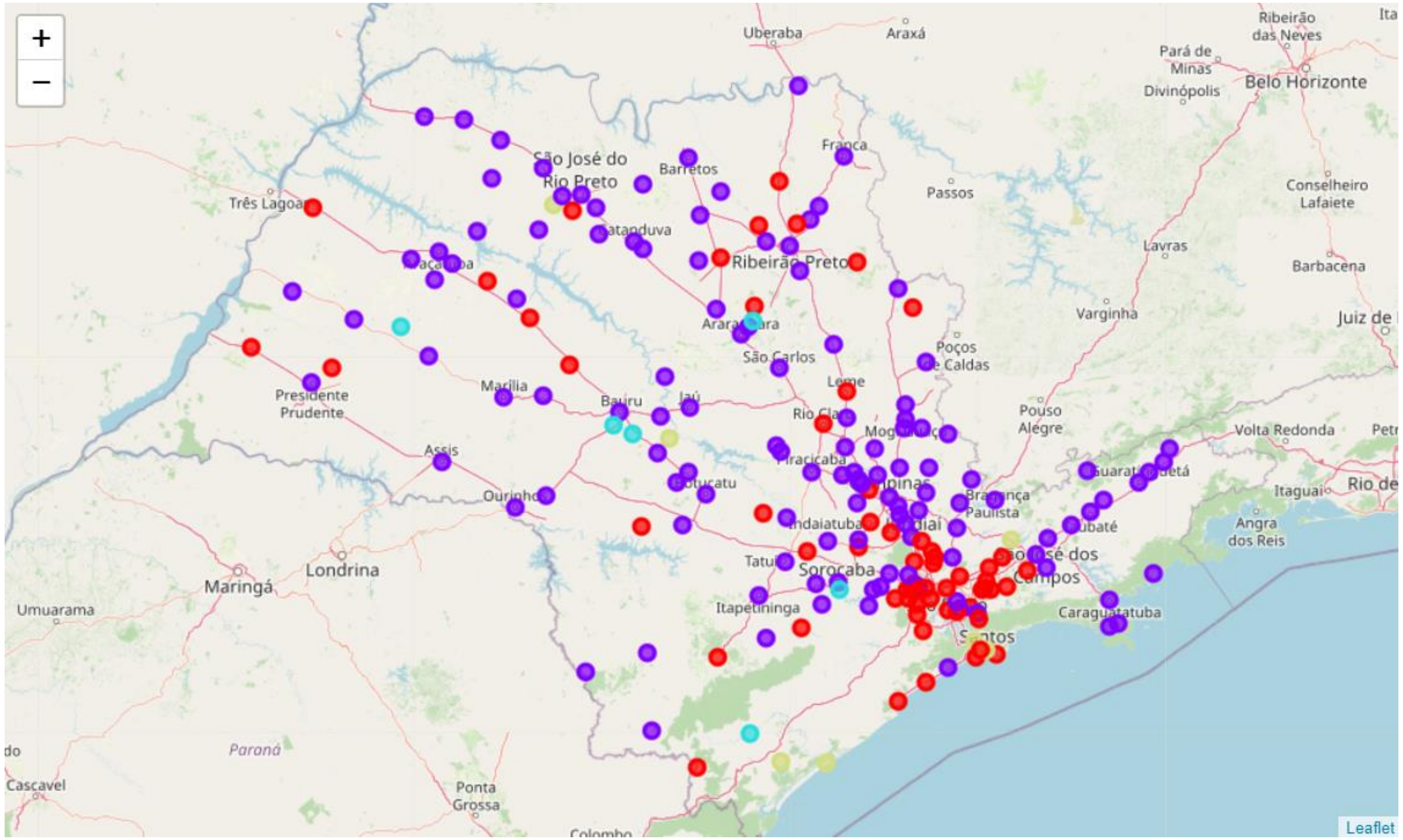
After obtaining the final dataset, a StandardScaler procedure was applied to the dataset.

K-Means

K-Means was executed with the following parameters:

- n_clusters = *4* (The best **k = 4** was set by maximizing the **silhouette score** [2]);
- init = **"k-means++"**;
- n_init = **1000**;
- random_state = **12**.

Final Map



🔴 Cluster 0

🟣 Cluster 1

🔵 Cluster 2

🟡 Cluster 3

Clusters

Cluster	Description
0	Low number of Hospitals, High Deaths per 100M, High IFR
1	Low IFR, Low deaths per 100M, high population with 60 years or more
2	High Number of Hospitals, High Pop with 60 years or more, Low Deaths per 100M
3	High Number of Hospitals, High Deaths per 100M, High Pop with 60 years or more, Low IFR

Clusters

Mean values for each feature and cluster

Cluster	IFR	cases_100M	number_hospitals_100M	pop_60_rate	deaths_100M
0	0,104876	141,977908	3,725253	0,140039	13,383871
1	0,035716	98,502045	5,403868	0,163691	3,481913
2	0,039642	131,207272	37,288564	0,161087	4,912472
3	0,032114	734,979526	15,171242	0.166534	20,511717

Conclusion

Some cluster have been obtained from features like IFR, number of cases per 100M habitants, number of hospitals per 100M habitants, rate of population with 60 years or more and deaths per 100M. One action that could be taken based on cluster was to create more hospitals in cities from **cluster 0** that has low numbers of hospitals and high number of deaths per 100M habitants and high IFR. One important thing to note in this analysis is that foursquare only gives 100 venues from a search, which could make hard to distinguish cities with more than 100 hospitals.

References

[1] SEADE. Github data repository. https://github.com/seade-R/dados-covid-sp

[2] Medium. How to Determine the Optimal K for K-Means. https://medium.com/analytics-vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb