Where would it be the best place for brazilians to live abroad?

Emanuel Monção Lima

April 17, 2020

2. Data Management

2.1 Data Sources

Firstly data has been scraped from a *Wikipedia* page (https://en.wikipedia.org/wiki/Brazilian_diaspora) containing countries in the world which have got a significant number of born Brazilians living it. Secondly, it was scraped from another webpage that provides a bunch of data about US cities and their citizens (http://www.city-data.com/top2/h153.html) but here it indicates the top 101 cities in US with the most residents born in Brazil. Finally, the *Foursquare* API was used to get some data about places related to Brazil in the chosen cities through a query call in a *Jupyter Notebook* that has been generated.

2.2 Data Cleaning

The pages were downloaded using *urlopen* method from *urllib.request* module and parsed from HTML to a *BeautifulSoup* instance so it was possible to better look into tags within the page and have a easier access to data therein.

In first page, the data was within a table tag with a specified class called *"infobox vcard"*, so it was a nice try to use the *find()* method since there was only one table with that class name. Next step was to remove some links that were on our way to get a clean data from the table and it was observed that all those links had a *"sup"* tag in common which has made it easier to remove with the *find_all()* and *decompose()* methods in such order. Then all table rows have been taken from it then selected which rows had useful data and put into variables lists. With good rows in hand, it was parsed to a dataframe from *pandas* module to be better

manipulated such as taking some remaining brackets out and converting necessary rows to the right data types to be manipulated correctly afterwards.

The second webpage had the same scraping strategy and methods and the table which had the desired data was tagged as *table class="tabBlue tblsort tblsticky"*. Like the data treating of the first page, the same methods for removing not useful tags in the table, taking the good rows out of the table and parsing them to a *pandas.DataFrame()* object have been used. Both scraped data were treated to their better layout so graphs were to be taken out of them in a easier way and so data were better visualized.

Finally the third dataset was the *Foursquare API* data which has been requested (*GET)* using a search query and a radius of interest alongside with the developer credentials such as client ID, client secret as well as the *Foursquare* version set to "20180604" and a specific limit number set to 50. Once all of these were set up, they have been formatted such as

"**https://api.foursquare.com/v2/venues/search?client_id={}&client_secret={}&ll={},{}&v={}&query={}&radius={}&limit={}**".

Its result were normalized to .json and had their rows selected as so convenient. These contain Brazil-related places such as restaurants and communities that may be useful as support for newcomers to a new city with such different culture and habits. So, if clustered, this may indicate potential good areas to live or stay in and yet feel as comfortable as possible at least at first.