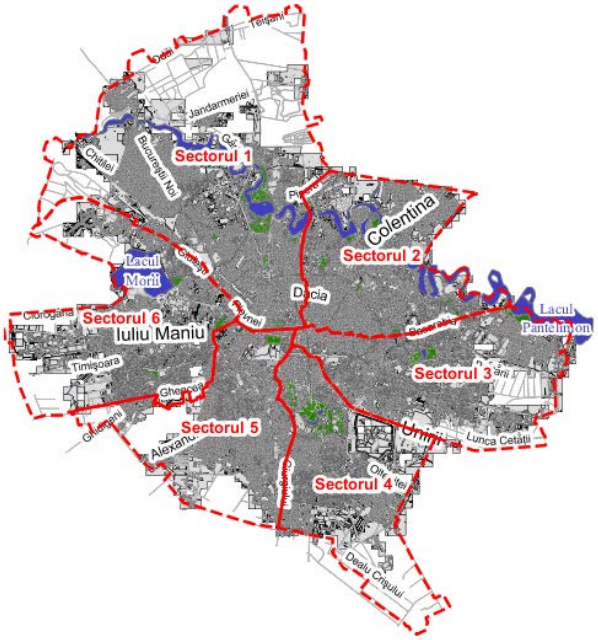
Creating smarter city districts in Bucharest

Introduction

Background

Bucharest, the capital of Romania is currently divided into 6 administrative units, called sectors ("sectoare" in Romanian). Each sector has its own mayor and council who are responsible over local affairs (secondary streets, parks, cleaning services, for example). After World War I, each administrative unit of Bucharest (called "culori" at that time), was first given its own mayor and council. The current divisions of sectors of this city date back to august 1979, and there is an incentive to redefine they way Bucharest is divided, as the territories encompass diverse neighborhoods which translate into diverse needs hard to tackle by the local administrations.



Problem

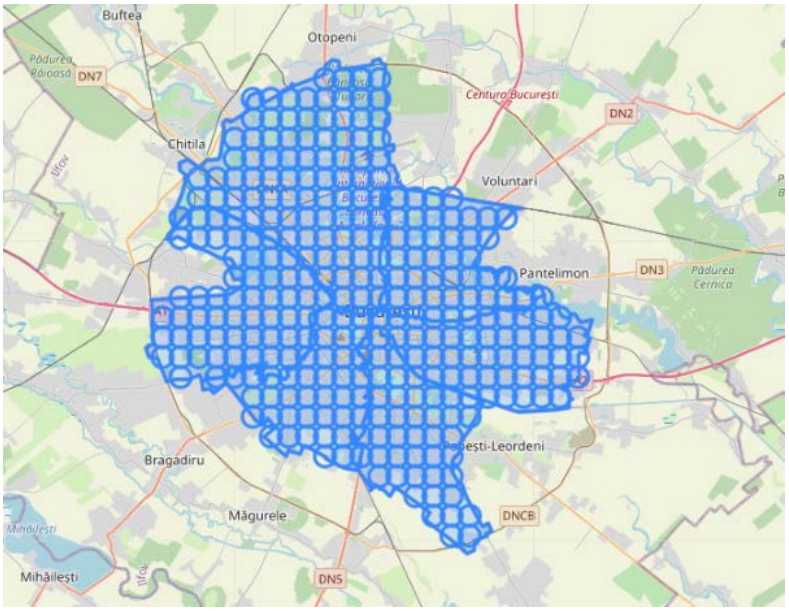
The problem of dividing a territory into coherent divisions demands taking into account a large number of factors regarding what venues are present in the district, what kinds of services are operating and at what level of quality, etc. These factors, if correctly used, will create useful territorial divisions reflecting local needs. Organizing the administration of the city in smaller, more representative units is a necessity at this point, although the question begged by this initiative is how do we do this and take into account enough factors to make it efficient?

My goal is to use machine learning algorithms to try to divide the territory of Bucharest in a more efficient way, with neighborhoods defined by the types of restaurants, parks, museums present. Making the divisions based upon the types of venues which can be found within the territory, will ensure a more accurate depiction of local's needs.

Data

Foursquare venue data will be leveraged in order to help build new clusters of neighborhoods which could potentially replace the present administrative divisions of Bucharest. First step is to define the city's limits and select data for venues inside the city. We will use a grid of small neighborhoods within the city in order to make the calls for venues.

Next step will be to clean the data of potential duplicates and create a new dataset which includes information about the types of venues present within each artificial neighborhood created. See below image for a visualization of the artificial "neighborhoods" created.



Each circle represents an artificial neighborhood which will contain information about venues present within it. These neighborhoods will then be used together with their location data in order to be clustered and create new "sectors" for the city.

Methodology

After extracting venue data through Foursquare's API, the dataset resulted from this process included data for each venue found, each line representing a venue. I present here the first few rows of the dataset which has in total 3124 rows.

	Neighborhood index	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	43	44.341705	26.153622	Baza Steaua Bucuresti(Berceni)	44.341298	26.153184	Soccer Field
1	90	44.358354	26.121788	Jumbo	44.361887	26.124094	Toy / Game Store
2	90	44.358354	26.121788	Auchan	44.361260	26.122536	Department Store
3	90	44.358354	26.121788	Hasco Fashion	44.360040	26.123019	Shopping Mall
4	90	44.358354	26.121788	Orange store	44.361326	26.122492	Electronics Store

Next step in the data cleaning process was deleting duplicate rows, which this set had none. Using one hot encoding, I added categories for every type of venue in the dataset, in order to be able to group them by neighborhoods, as seen below.

	Neighborhood index	ATM	Accessories Store	Airport Terminal	American Restaurant	Amphitheater	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	...	Vietnamese Restaurant	Warehouse Store	Water Park	Wate
0	113	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	...	0.0	0.0	0.0	
1	114	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	...	0.0	0.0	0.0	
2	115	0.0	0.0	0.0	0.0	0.0	0.333333	0.0	0.0	0.0	...	0.0	0.0	0.0	
3	116	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	...	0.0	0.0	0.0	
4	117	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	...	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	

In the first stages of the project, I decided to use K-means clustering in order to explore our neighborhoods and their similarity. For this analysis I used just the venues data presented above. Each neighborhood was characterized by the number of venues present within it from each category of possible venues. The resulting clusters, where of course spread geographically all over the city, and did not provide a very good insight. Additionally, I have decided do go further into the analysis using DBSCAN clustering algorithm, because I am not sure how many districts should the city have, and KMeans needs a fixed number of clusters in order to work.

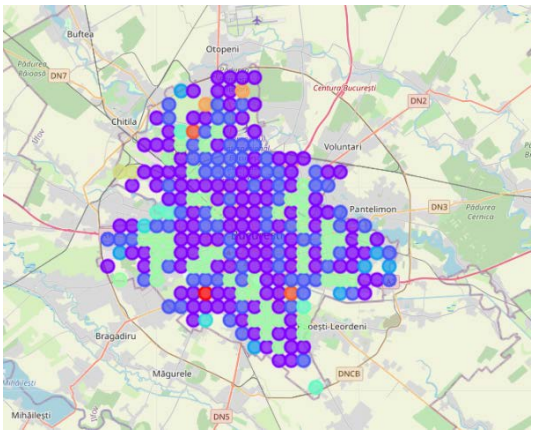
In order to use DBSCAN clustering algorithm, I have decided to add the geographical coordinates of each neighborhood in the analysis, and to reduce the dimensions of existing variables. I have decided to reduce the 302 columns representing venues to 10 principal components, using PCA Analysis.

After reducing our venues data, I added the coordinates of each neighborhood back into the dataset and standardized all of them in order to run the clustering algorithm. The resulted dataset looked like this before using StandardScaler on the whole dataset.

	principal component 1	principal component 2	principal component 3	principal component 4	principal component 5	principal component 6	principal component 7	principal component 8	principal component 9	principal component 10	LAT	LON
0	-1.188686	0.222272	0.185851	-0.477468	0.412854	0.132388	0.396173	0.589247	-0.067252	0.210014	44.366679	26.100566
1	-1.212115	0.334354	0.317480	-0.415647	0.424840	0.363880	0.297824	0.678941	-0.143923	0.116646	44.366679	26.111177
2	-1.254399	0.131917	0.490719	-0.490620	0.549470	-0.021312	0.510665	0.616766	-0.012034	-0.347121	44.366679	26.121788
3	-1.183395	0.255866	0.326575	-0.437731	0.412390	0.226118	0.367707	0.627006	-0.063849	0.094994	44.366679	26.132399
4	-0.290958	0.028854	0.392725	0.139747	-0.540154	-0.335909	1.080192	0.728426	-0.612113	-0.459225	44.366679	26.143010
...	...	...	...	...	...	...	...	...	...	...	...	...
267	-1.174844	0.238706	0.266944	-0.387348	0.312172	0.204702	0.593901	0.661348	-0.163267	0.176935	44.533175	26.089954
268	-1.426632	0.208617	0.485592	-0.605706	0.623290	0.118338	0.687633	0.776213	-0.630681	0.329806	44.533175	26.100566
269	-1.078632	0.433739	1.240629	-0.869439	0.305050	0.116106	0.918119	1.015051	-0.221664	-0.119576	44.358354	26.121788
270	-1.481491	0.310229	0.788381	-0.605131	0.564863	0.108351	0.577781	0.811739	-0.276436	0.040928	44.358354	26.132399
271	-1.288875	0.272317	0.468213	-0.490964	0.428280	0.305833	0.359454	0.748325	-0.119022	0.052103	44.358354	26.143010

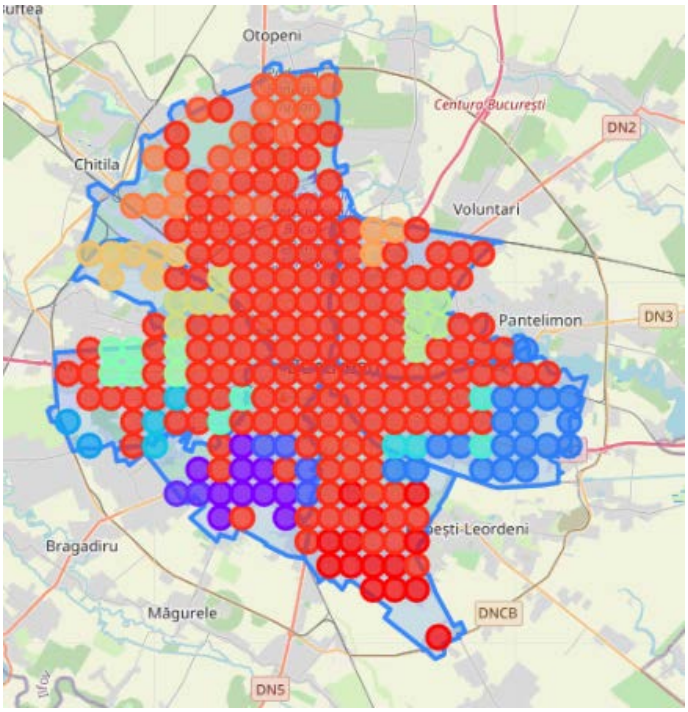
Results

First try at clustering the neighborhoods was done using KMeans clustering algorithm, which requires as input an established number of clusters. I used as input 10 clusters, to explore how the algorithm will work on this set of data. The results were not promising, as the algorithm was agnostic about the locations of each neighborhood and could not cluster them geographically. Below, there is a picture of the results of KMeans.



As seen here, we need to take into account the latitudes and longitudes of each neighborhood and give them a higher weight in the analysis. This is one of the reasons for which I decided to reduce the dimensionality of the dataset by using 10 principal components of venue types, and additionally two variables representing latitudes and longitudes of each neighborhood.

Using DBSCAN, which was chosen for its ability to cluster points without the necessity to input a number of predetermined clusters, I analyzed the new dataset, and after tweaking a bit with the parameters of the algorithm, I reached the following distribution of clusters in Bucharest.



A total number of clusters of 19, distributed in such a way that they can arguably become “new neighborhoods”. A note here regarding the central cluster – those points do not belong to any cluster – they are unlabeled, and I will discuss this aspect below.

Discussion

As one could expect, the newer areas where Bucharest has expanded over the last years, have a distinct composition of local venues, which in turn will be reflected in the needs of their residents. This aspect of our results clearly shows a need for these areas to be administered more independently, with solutions targeting their specific needs, and not be included just in umbrella projects aimed for the entire sector.

As noted above, the algorithm wasn't able to cluster points in the center of the city, which suggests that venues in central Bucharest tend to be fairly homogenously distributed. This may tempt us to conclude that central Bucharest does not need to be re-analyzed, however I argue that this needs to be done adding additional information about other aspects that may affect the quality of life of citizens.

Conclusion

Local administrations are struggling every year to tackle problems in these big areas, and to balance their budget in a way that is fair for every part of the sector. This problem could be greatly improved by creating smaller districts, centered around neighborhoods which present similar problems. This small study focuses on venues which are available on Foursquare for Bucharest, and creates an argument for rethinking the way Bucharest is administered today.

For future projects, there are lots of additional data which can be added in order to create smarter city districts, and more efficient local administrations. In order to really take into account all the needs of residents of a neighborhood, structured and unstructured data regarding water quality, heating conditions, sewage system, public transport should be taken into consideration when redesigning a city's administrative units.