26.2.2020

Data science final project report

Andrii Yevtukh

Оглавление

Introduction section

1.1. Kyiv is a biggest Ukrainian city with relatively high population (3 million), square (84 square kilometers) and long history (its age is approximately 1500+ years). The city lies on a so-called crossroad between East and West and experienced a lot of adventures during its history. Many of them influenced on architectural view and venues density. In the end, the city consists of 10 boroughs and conditions in them again, are pretty different- from almost river side height to hills, from concrete jungle to lakes and forests nearby. And all these factors gave certain influence on architecture of the city.

So, as you can see finding a flat in such city could be a tricky question. Furthermore, the density and availability of different public institute is very different too, so finding a flat which would be a perfect match to all your needs is a time-consuming and could be a challenge. Especially if you need something very specific (e.g. dance school nearby). The research and analyze will require a lot of time, and due to active real-estate market good opportunities could be lost.

I'm a resident of Kyiv and at certain point of time my family decided to change a flat. At the start point, the main condition was a price, but after viewing few variant some extra more issues have appeared.

In addition, amount of possible variants didn't make our lives easier. So, I decided to create a project which will allows me and my family member's to spare some time, nerves and get the prefect flat in perfect surrounding as result

1.2. Short problem description: find a flat in Kyiv which will meet following requirements:

- Apartment with 2 rooms
- Total price between 65000-67000 USD
- It should allow us to keep same level of comfort as we have in current apartment
- Safety conditions should be the same as we have now

1.3. Target Audience: I believe this project would be useful for anyone who would like to find the best solution within given price, specific venues and some other parameters. In particular, peoples from other regions which are not familiar with city infrastructure or foreigners with same problem. Plus, this project could be increased to cover more business needs and functionality with relatively small costs and amount of time spent.

Data section

With respect to problem given above, I used the following data in project:

Data types:

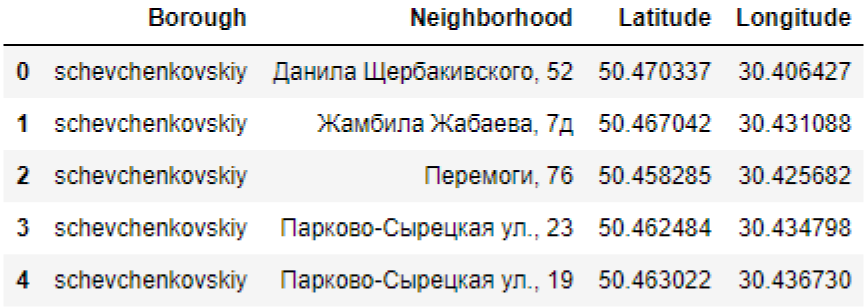
- Manually built data set with all flats that meet initial conditions
- Manually build table with weights used for correlation of results
- Crime rate info

Data sources:

- Real-estate site data taken from https://flatfy.lun.ua/
- Google map geo location
- Forsquare API
- Total crime amount divided by boroughs - https://mvs.gov.ua/
- Density of population- https://kyivcity.gov.ua/

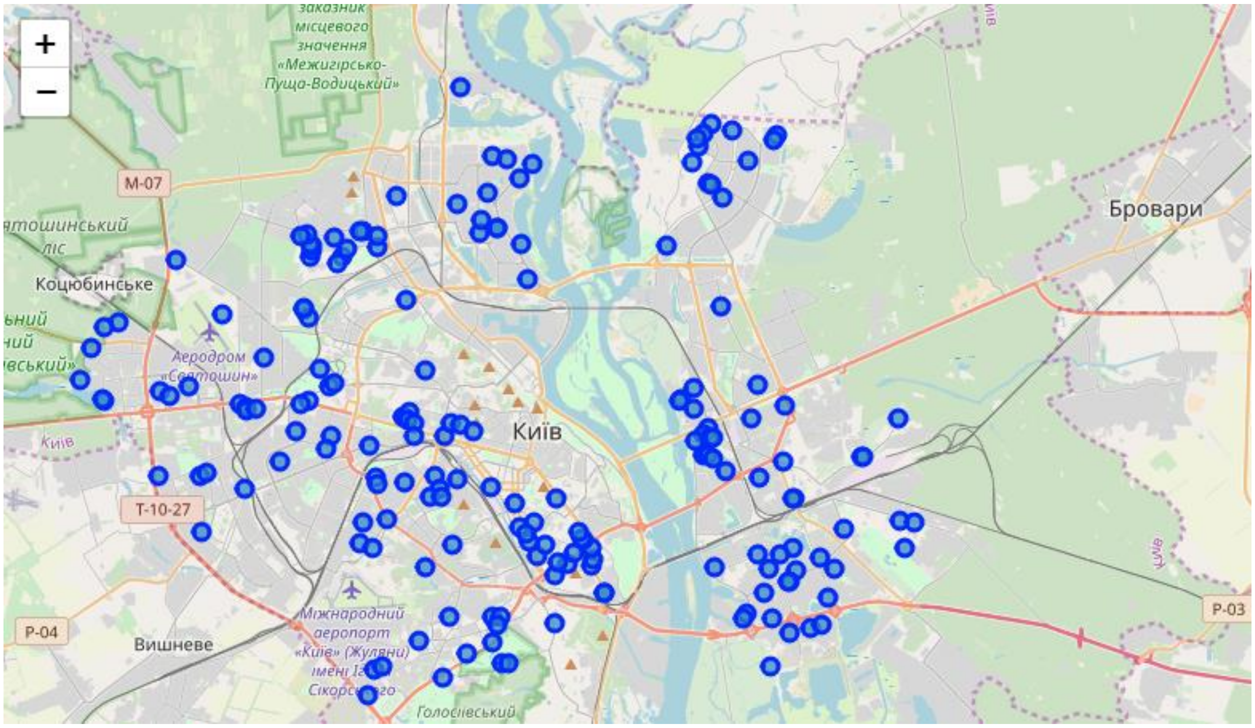
Methodology

As a start point, we made a request based on initial conditions on real-estate site Lun.ua and got 182 results. All these variants were placed in one table with Borough, Address and Geo-data from Google map:



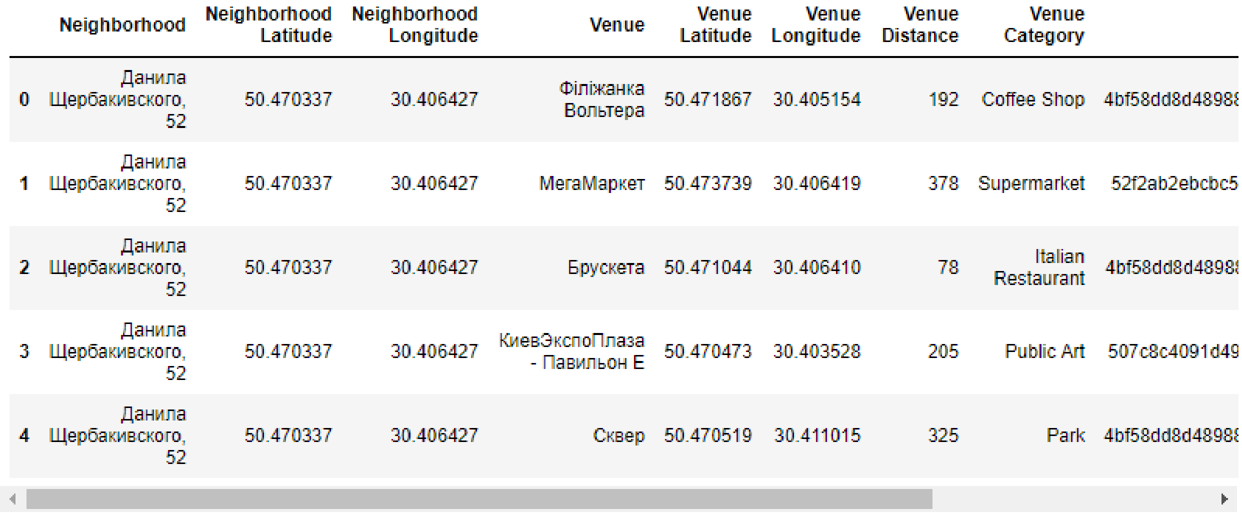
	Borough	Neighborhood	Latitude	Longitude
0	schevchenkovskiy	Данила Щербакивского, 52	50.470337	30.406427
1	schevchenkovskiy	Жамбила Жабаева, 7д	50.467042	30.431088
2	schevchenkovskiy	Перемоги, 76	50.458285	30.425682
3	schevchenkovskiy	Парково-Сырецкая ул., 23	50.462484	30.434798
4	schevchenkovskiy	Парково-Сырецкая ул., 19	50.463022	30.436730

Then I plot all this data to Kyiv map and got following overview:


+
−
заказник місцевого значення «Межигірсько-Пуща-Водицький»
М-07
итошинський ліс
Коцюбинське
Аеродром «Святошин»
Київ
Київ
Т-10-27
Р-04
Вишневе
Міжнародний аеропорт «Київ» (Жуляни) імені Ігоря Сікорського
Голосіївський
Бровари
Р-03

Still, too many options and too much time would be needed to analyze these results.
In the meantime, my family started to raise some specific requirements to nearby venues.

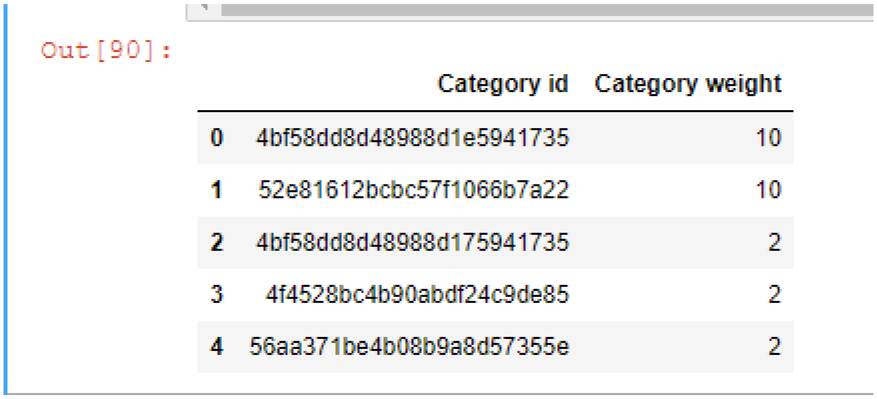
Then I requested data form Foursquare based on geo data for all apartments and set limitation for venues within 500 meters from which of apartment. Example of results is given below:



	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Distance	Venue Category	
0	Данила Щербакивского, 52	50.470337	30.406427	Філіжанка Вольтера	50.471867	30.405154	192	Coffee Shop	4bf58dd8d4898
1	Данила Щербакивского, 52	50.470337	30.406427	МегаМаркет	50.473739	30.406419	378	Supermarket	52f2ab2ebcbc5
2	Данила Щербакивского, 52	50.470337	30.406427	Брускета	50.471044	30.406410	78	Italian Restaurant	4bf58dd8d4898
3	Данила Щербакивского, 52	50.470337	30.406427	КиевЭкспоПлаза - Павильон Е	50.470473	30.403528	205	Public Art	507c8c4091d49
4	Данила Щербакивского, 52	50.470337	30.406427	Сквер	50.470519	30.411015	325	Park	4bf58dd8d4898

However, the results were full of different venues categories which were not relevant for me and my family and amount of data was still too big for analyze.

After brainstorming, we came up with certain amount of categories which seems to be relevant for all of us. And I added weights to all of categories.



Out[90]:

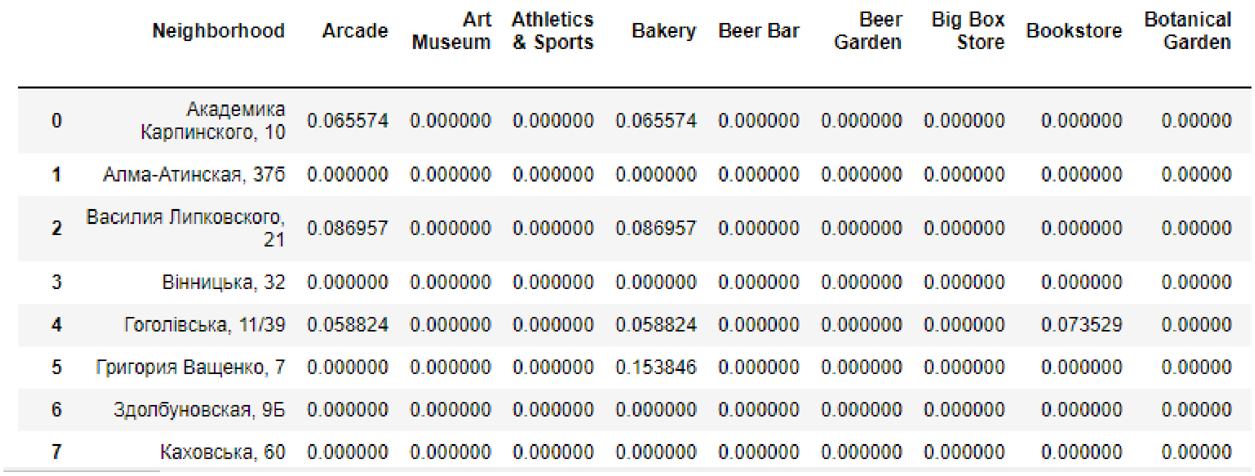
	Category id	Category weight
0	4bf58dd8d48988d1e5941735	10
1	52e81612bcbc57f1066b7a22	10
2	4bf58dd8d48988d175941735	2
3	4f4528bc4b90abdf24c9de85	2
4	56aa371be4b08b9a8d57355e	2

These data were used to filter data set with all categories based on following: if venue belongs to category which has no weights, it should be excluded form data set.

Now our data set is cleaned and sorted, so I started final preparation of data set for clusterization:

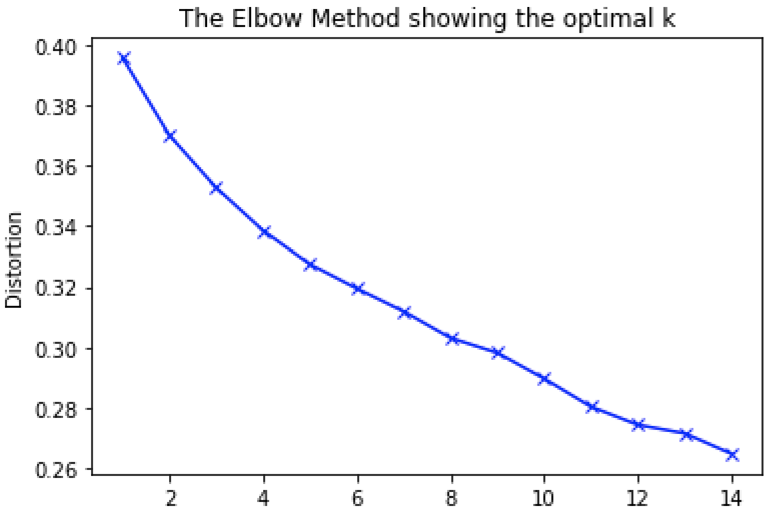
- Using one hot encoding, transform venue.
- Group rows by apartment and by taking the max of occurrence of each category.
- Set weight for each venue.
- Normalize data by rows, (using sklearn.pre-processing.normalize)

The results were the following table:

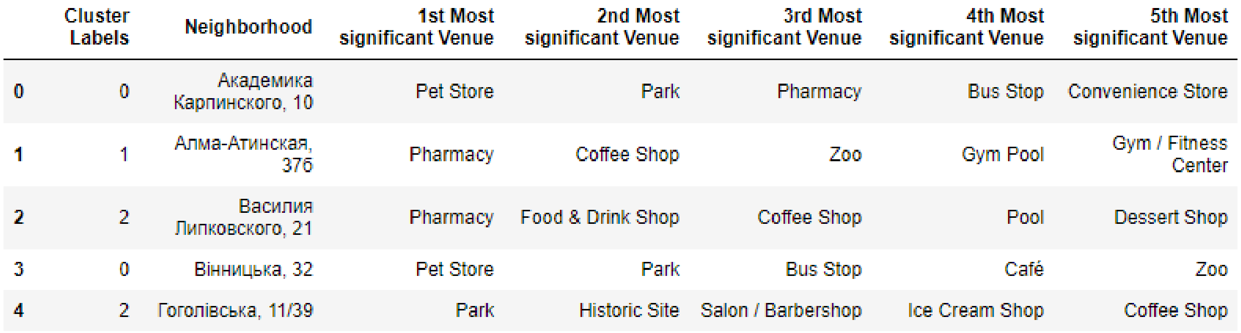


	Neighborhood	Arcade	Art Museum	Athletics & Sports	Bakery	Beer Bar	Beer Garden	Big Box Store	Bookstore	Botanical Garden
0	Академика Карпинского, 10	0.065574	0.000000	0.000000	0.065574	0.000000	0.000000	0.000000	0.000000	0.00000
1	Алма-Атинская, 37б	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
2	Василия Липковского, 21	0.086957	0.000000	0.000000	0.086957	0.000000	0.000000	0.000000	0.000000	0.00000
3	Вінницька, 32	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
4	Гоголівська, 11/39	0.058824	0.000000	0.000000	0.058824	0.000000	0.000000	0.000000	0.073529	0.00000
5	Григория Ващенко, 7	0.000000	0.000000	0.000000	0.153846	0.000000	0.000000	0.000000	0.000000	0.00000
6	Здолбуновская, 9Б	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
7	Каховська, 60	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000

Then I found the number of clusters using ‘elbow method’. This

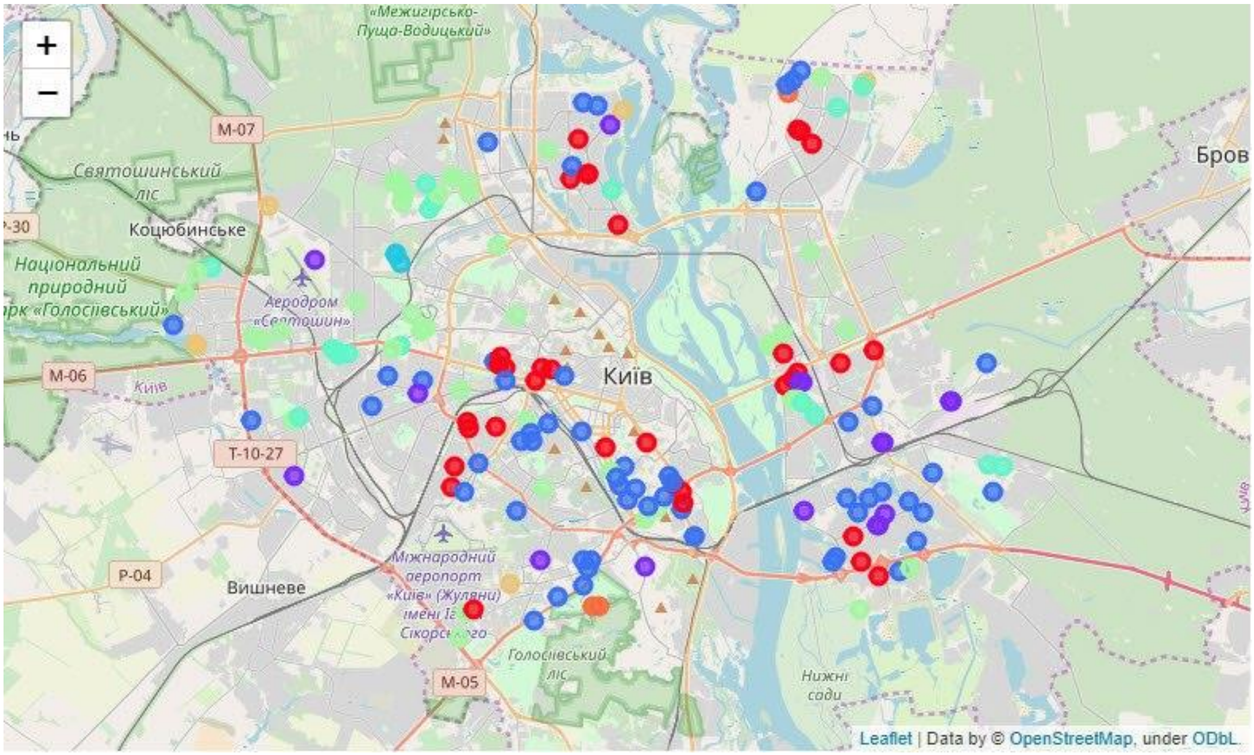


And create a clusters themselves using k-means clustering



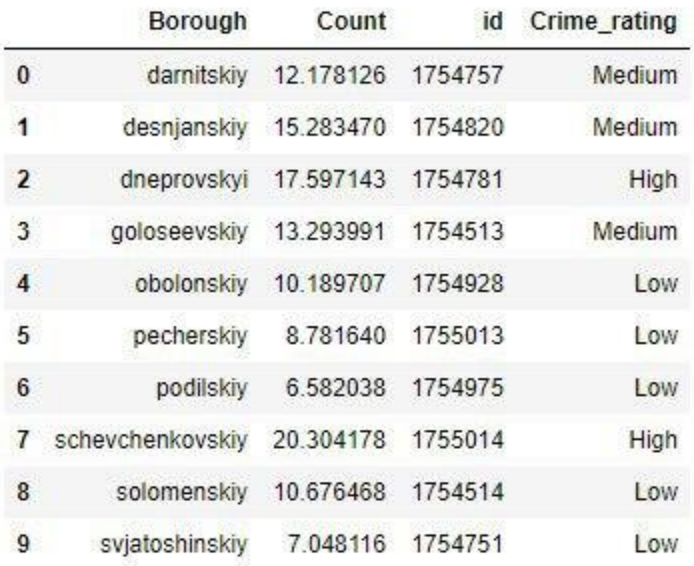
	Cluster Labels	Neighborhood	1st Most significant Venue	2nd Most significant Venue	3rd Most significant Venue	4th Most significant Venue	5th Most significant Venue
0	0	Академика Карпинского, 10	Pet Store	Park	Pharmacy	Bus Stop	Convenience Store
1	1	Алма-Атинская, 37б	Pharmacy	Coffee Shop	Zoo	Gym Pool	Gym / Fitness Center
2	2	Василия Липковского, 21	Pharmacy	Food & Drink Shop	Coffee Shop	Pool	Dessert Shop
3	0	Вінницька, 32	Pet Store	Park	Bus Stop	Café	Zoo
4	2	Гоголівська, 11/39	Park	Historic Site	Salon / Barbershop	Ice Cream Shop	Coffee Shop

And here is a clustered representation of clustered data:



The choice isn't optimal yet, so I decided to add one more parameter, very important one – safety. The estimation of safety could become another problem, but I decided to use quite common method- namely, take the total rate of crimes per boroughs and divide it on density of population per borough.

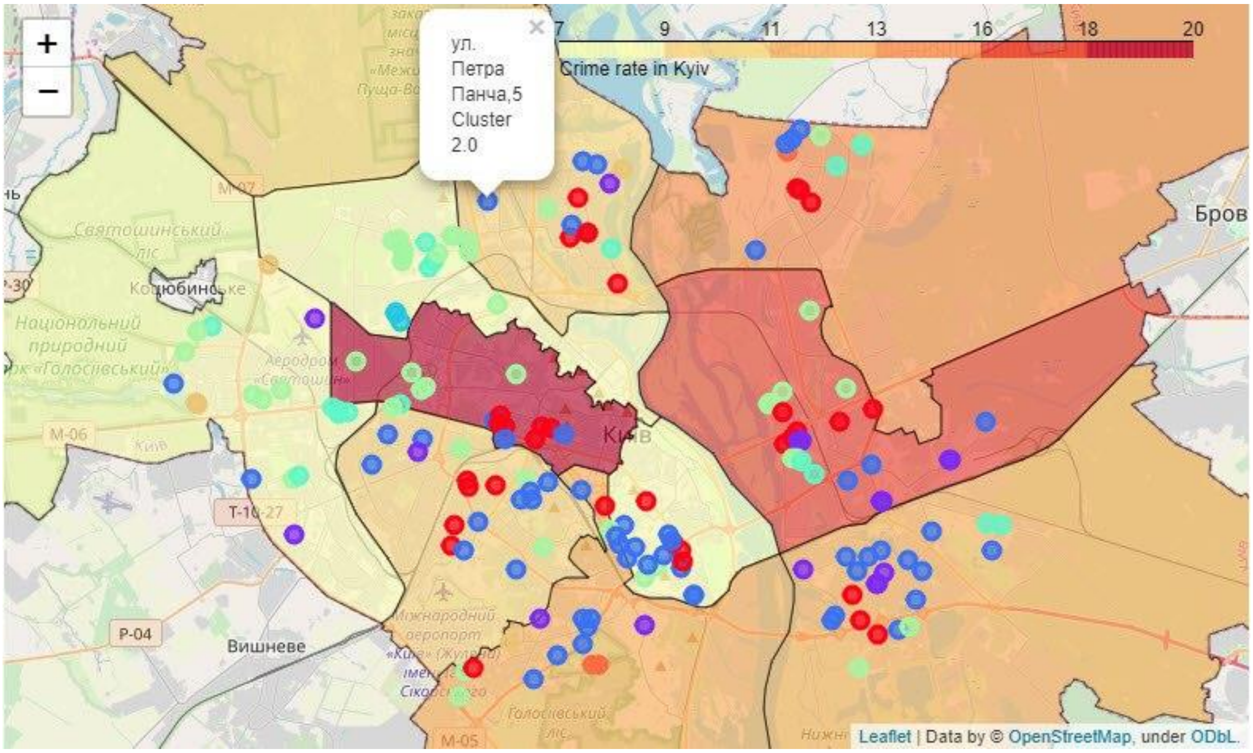
This action gave me an approximate average crime rate per borough which I applied to my table



	Borough	Count	id	Crime_rating
0	darnitskiy	12.178126	1754757	Medium
1	desnjanskiy	15.283470	1754820	Medium
2	dneprovskyi	17.597143	1754781	High
3	goloseevskiy	13.293991	1754513	Medium
4	obolonskiy	10.189707	1754928	Low
5	pecherskiy	8.781640	1755013	Low
6	podilskiy	6.582038	1754975	Low
7	schevchenkovskiy	20.304178	1755014	High
8	solomenskiy	10.676468	1754514	Low
9	svjatoshinskiy	7.048116	1754751	Low

Results

After applying Crime ratio to previously clustered data, I got following view in results:

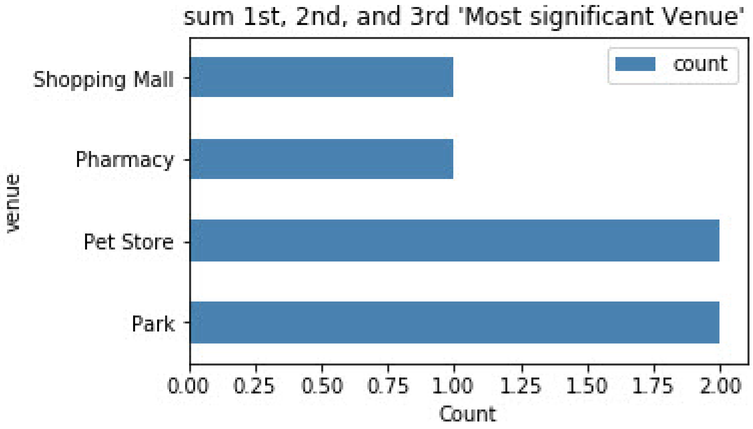


This map provides us with not only visual representation of all available parameters but also with extra data, namely:

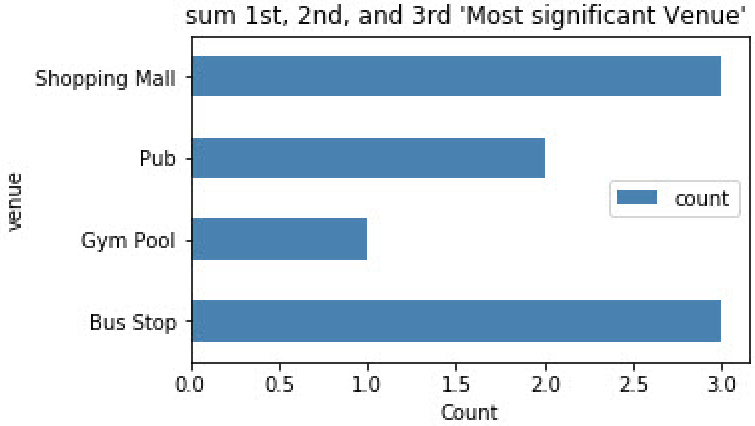
- Data cluster – we can quite easily check what's in and what's not
- Safety parameter

We're mostly interested in 0,1,3 and five – below you can more data about our clusters:

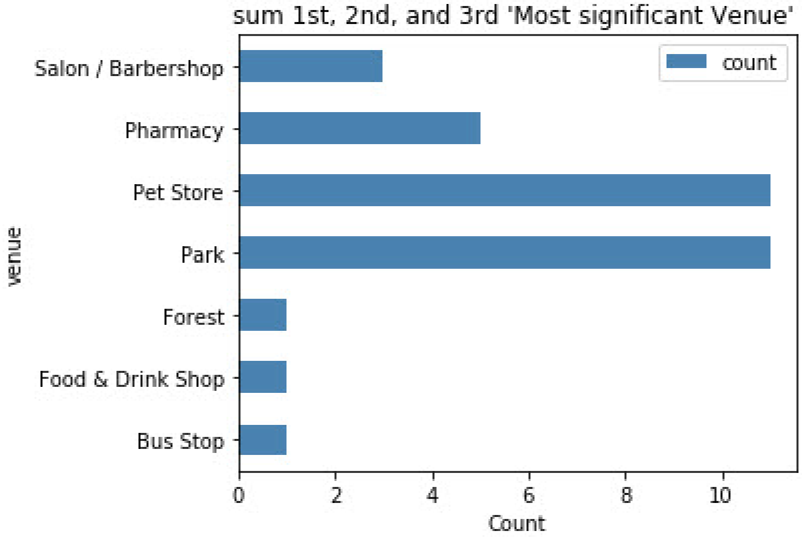
```
(2, 8)
```



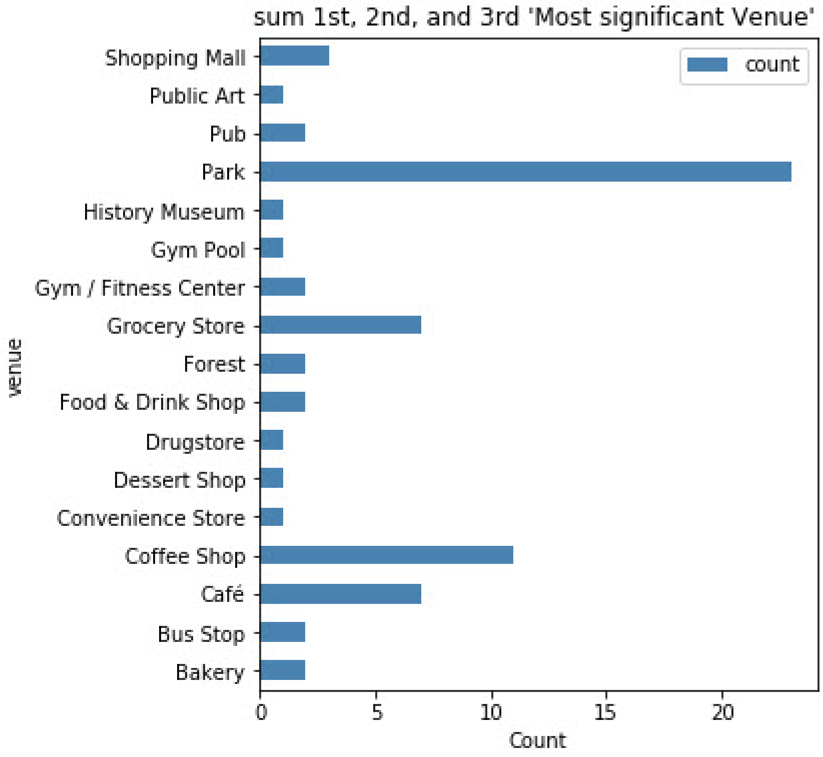
```
(3, 8)
```


sum 1st, 2nd, and 3rd 'Most significant Venue'
Shopping Mall
Pub
Gym Pool
Bus Stop
venue
count
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Count

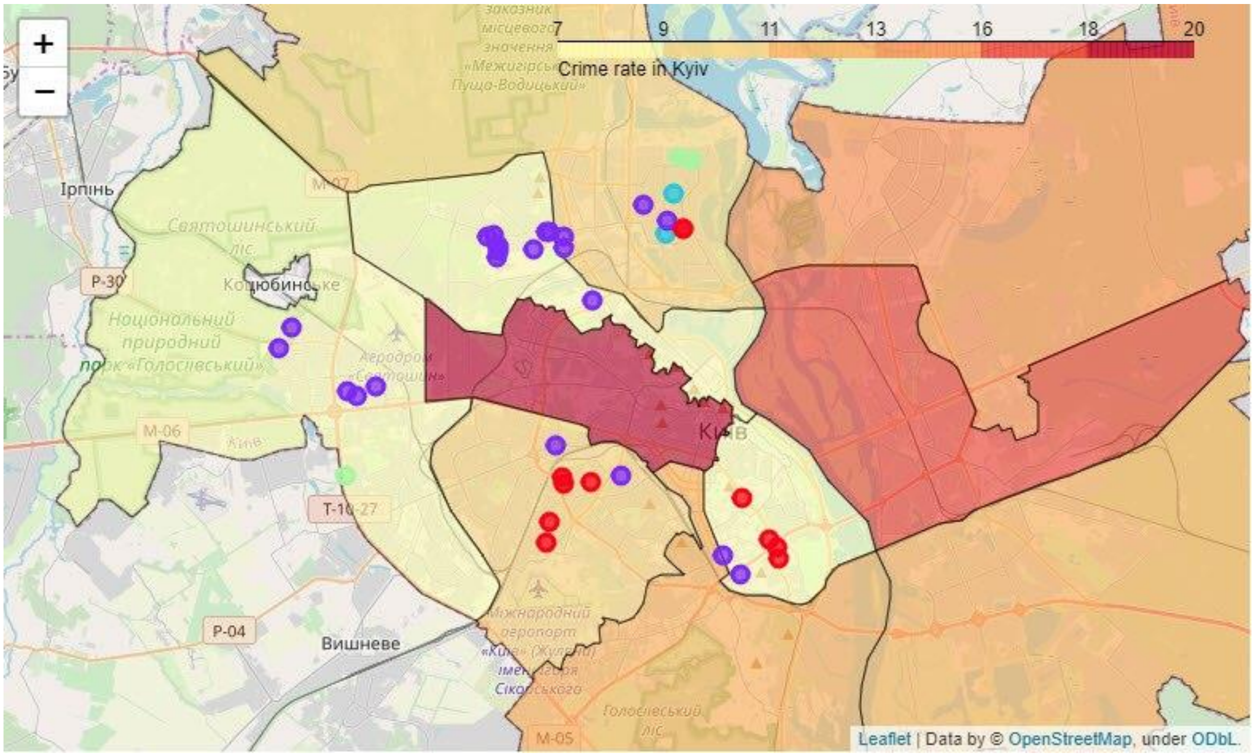
```
(11, 8)
```


sum 1st, 2nd, and 3rd 'Most significant Venue'
Salon / Barbershop
Pharmacy
Pet Store
Park
Forest
Food & Drink Shop
Bus Stop
venue
count
0
2
4
6
8
10
Count

```
(23, 8)
```



In total they gave us 39 apartments while the initial amount of choices was 182. Taking into account this information, the map will look like following:


+
−
7
9
11
13
16
18
20
Crime rate in Kyiv
Ірпінь
Святошинський ліс
Коцюбинське
Національний природний парк «Голосіївський»
Аеродром «Святошин»
P-30
M-07
M-06
T-10-27
P-04
Вишневе
Київ
Міжнародний аеропорт «Київ» (Жуляни) імені Ігоря Сікорського
Голосіївський ліс
M-05
Leaflet | Data by © OpenStreetMap, under ODbL.

So, after doing this, I got a possibility to select the best option where all needed conditions are met and do it in easy and smart way.

Discussion

First of all, based on the results of this project I was able to find the most suitable results from huge variety of options. And even though there are still plenty of data which I'll have to check and analyze, the variety of choices were decreased dramatically.

Secondly, this project gave me unique possibility to take a part in overall process- from defining business problem to the very end of analyzing it. This is not something what you could easily do on daily basis.

And last, but not least, I would say that project described in this report has a huge potential to grow into something much more useful and powerful.

Just imaging that you could not only find a general info about a flat on the real-estate site, but also :

- check what's nearby this flat – and not only in general but also select something specifically relevant for you (dance school? Yoga? Anything else?)
- compare air clearance and crime rate, check if the territory where the building placed isn't overheating/under heating, and so on.
- Furthermore, I'd be happy to see a recommendation and likes system applied to everything- starting from house keeper quality of service and ending with water supply/electricity isuses if there are any.

However, there are also weak points which I'd like to improve:

- would be nice to make all data gathering and clearance automatically- because in my project certain amount of manual work was involved.

- Get more sources of data - because in my project I used one real-estate site (actually, the biggest one in UA).
- Use more APIs to collect relevant data – because Foursquare is very good for certain purposes, but some venues (especially public institutes) have relatively poor presence in this system.

Conclusion

- There are plenty of variants which suits our initial requirements, so the choice was not so obvious
- Adding extra criteria helps me to reduce the amount of options from 182 to 39 which is quite good results
- The best option within criteria we defined are parts of clusters 0,1,3 and 5
- However, we still need to visit these places to make sure that they will meet our needs
- At the end, this project was sometimes demanding and challenging, but very interesting experience. I hope I'd be able to make a good use of knowledge I got during the course