Battle of Neighborhoods: Choosing Location for a New Restaurant in Moscow

(Coursera Applied Data Science Capstone project)

by Anton Startsev, Feb. 2020

1. Introduction

1.1 Background

Moscow is the capital of Russia with total population exceeding 12 mln. The business environment of the city is highly competitive, and any aspiring business proprietor should take into account many factors affecting the likelihood of success. This project focuses on the segment of eating-out services (restaurants, cafes etc.) and aims to facilitate the decision regarding location of a prospective business. To put it simple, the project helps to determine the best district(s) of Moscow to place your future restaurant (café, canteen, snack-bar) in.

1.2 The problem

Evaluate the districts of Moscow in terms of eating-out services availability. The metric used is seats availability (also referred to as "saturation"), which is calculated as aggregated number of seats in restaurants/cafes etc. per 1000 persons of population in each district.

1.3 Potential stakeholders

The findings resulting from my work might be of particular interest to several groups of potential stakeholders, namely:

- investors planning a new business in the eating-out industry,

- current proprietors, whose business strategy might be affected by the findings,

- city authorities, who might consider the project findings when shaping regulative environment and taking investment decisions.

2. Data

2.1 Data sources

(a) The geo-data for mapping Moscow districts: http://gis-lab.info/data/mos-adm/ao.geojson

(Source: GIS-Lab)

(b) Data set containing detailed info on Moscow eating-out businesses: https://raw.githubusercontent.com/RunnerTony/CoureraProjectsRepo/master/moscow_eatout_json_utf8.txt

(Source: Moscow city administration, data.mos.ru)

(c) Population data by district: https://raw.githubusercontent.com/RunnerTony/CoureraProjectsRepo/master/moscow_population_v2.csv

(Source: statistical data portal statdata.ru)

2.2 Data cleaning and preparation

The dataset used for the final analysis here consists of two dataframes: (1) a geodataframe (gdf) and (2) a dataframe with calculated seat-counts for each district (df_data).

The geodata (gdf)

The geodataframe is loaded directly from geo-json file (see (a) in 2.1). The raw data is clean and robust, and the only cleaning neaded was to drop some unnecessary columns. As a result, the head of the geo-dataframe looks like this:

	NAME	geometry
0	Троицкий	MULTIPOLYGON (((36.80310 55.44083, 36.80319 55...
1	Новомосковский	MULTIPOLYGON (((37.08697 55.59036, 37.09492 55...
2	Зеленоградский	POLYGON ((37.13160 56.01645, 37.13266 56.01678...
3	Юго-Западный	POLYGON ((37.45572 55.63705, 37.46372 55.64070...
4	Юго-Восточный	MULTIPOLYGON (((37.66069 55.73070, 37.66082 55...

Where *NAME* corresponds to the district name and *geometry* contains polygonal coordinates shaping each of the district's location on map.

Seats count

The eating-out places dataframe contains 16299 rows and 16 columns. Of the latter, we need the following:

- AdmArea: name of the distict

- SeatsCount: number of seats in the café or restaurant in the corresponding row.

The task is to group the premises by district and calculate the sum of SeatsCount.

Population

The population dataframe looks like this:

	Муниципальное образование	Все население	в т.ч. городское	в т.ч. сельское
2	Город федерального значения - г. Москва	12 197 596	12 054 243	143 353
3	Восточный округ	1 495 835	1 495 835	0
4	в том числе муниципальные образования Восточного округа:			
5	Богородское	106 828	106 828	0
6	Вешняки	121 693	121 693	0
7	Восточное Измайлово	77 698	77 698	0
8	Восточное	12 967	12 967	0
9	Гольяново	160 372	160 372	0

The first column ("Муниципальное образование") includes the district name and the second one ("Все население") – the population figure (number of persons in the district). The tricky part here is that not all rows refer to districts (some refer to neighborhoods within a district) and that the districts are not numerically encoded, which means we have to search for a string (district name) through the table to find the corresponding row. Please refer to the *Methodology* section below to find out how this is resolved.

3. Methodology

I had to aggregate both the population figures and the SeatsCount figures by district. The tricky part was that districts were not numerically coded in the datasets (had to search by district name) and the district names are not perfectly unique strings. I.e., “Западный” is a district name, but it is also part of the “Северо-Западный” district name. To address this issue, I sorted the districts by the length of the name:

```
df['name_length'] = df['NAME'].str.len()
df.sort_values('name_length', ascending=False, inplace=True)
df.head(12)
```

	NAME	OKATO	ABBREV	name_length
8	Северо-Восточный	45280000	СВАО	16
7	Северо-Западный	45283000	СЗАО	15
1	Новомосковский	45297000	Новомосковский	14
2	Зеленоградский	45272000	ЗелАО	14
4	Юго-Восточный	45290000	ЮВАО	13

Then, I moved top-down when grouping the population and SeatsCount data, searching for longer names first (and eliminating data already counted).

Here’s how it works with the SeatsCount grouping:

```
df_tmp = df_cafes
Scount = []
print('Seats count by district:')
for distr in df['NAME'].values:
    df2 = df_tmp[df_tmp['AdmArea'].str.contains(distr)]
    df_tmp = df_tmp[df_tmp['AdmArea'].str.contains(distr)==False]
    s = df2['SeatsCount'].sum()
    print(distr,': ', s)
    Scount.append(s)

df['SeatsCount']=Scount
```

```
Seats count by district:
Троицкий :  6901
Новомосковский :  17075
Зеленоградский :  17159
Юго-Западный :  66151
Юго-Восточный :  65867
Центральный :  245833
Северный :  72842
Северо-Западный :  44228
Северо-Восточный :  90083
Южный :  103701
Восточный :  74862
Западный :  91702
```

…and with the population data:

```
df_tmp = df_pop
Pcount = []
print('Population by district:')
for distr in df['NAME'].values:
    df2 = df_tmp[df_tmp['Муниципальное образование'].str.contains(distr)]
    df_tmp = df_tmp[df_tmp['Муниципальное образование'].str.contains(distr)==False]
    s = df2['Все население'].sum()
    print(distr,': ', s)
    Pcount.append(s)

df['Population']=Pcount
df.head()
```

```
Population by district:
Троицкий :  108 063
Новомосковский :  183 591
Зеленоградский :  232 489
Юго-Западный :  1 414 510
Юго-Восточный :  1 363 859
Центральный :  760 690
Северный :  1 151 160
Северо-Западный :  979 614
Северо-Восточный :  1 402 928
Южный :  1 760 813
Восточный :  1 495 835
Западный :  1 344 044
```

As a result, I got a dataframe with SeatsCount and Population figures for each district:

	NAME	OKATO	ABBREV	SeatsCount	Population
0	Троицкий	45298000	Троицкий	6901	108 063
1	Новомосковский	45297000	Новомосковский	17075	183 591
2	Зеленоградский	45272000	ЗелАО	17159	232 489
3	Юго-Западный	45293000	ЮЗАО	66151	1 414 510
4	Юго-Восточный	45290000	ЮВАО	65867	1 363 859

The next step was calculating the Sauration measure, which is equal to 1000*SeatsCount/Population for each district:

```
df['Saturation']=1000*df['SeatsCount']/df['Population']
df.head()
```

:

	NAME	OKATO	ABBREV	SeatsCount	Population	Saturation
0	Троицкий	45298000	Троицкий	6901	108063	63.860896
1	Новомосковский	45297000	Новомосковский	17075	183591	93.005648
2	Зеленоградский	45272000	ЗелАО	17159	232489	73.805642
3	Юго-Западный	45293000	ЮЗАО	66151	1414510	46.766018
4	Юго-Восточный	45290000	ЮВАО	65867	1363859	48.294582

After dropping the irrelevant columns and sorting the dataframe by Saturation the dataframe turns into:

	NAME	Saturation
5	Центральный	323.171068
1	Новомосковский	93.005648
2	Зеленоградский	73.805642

It can be easily seen that the central district ("Центральный") is an outlier with Saturation measure far above the next highest. So, to get a less distorted picture, I've set the central's district Saturation value to zero and focused on the rest of the districts.

	NAME	Saturation
5	Центральный	0.000000
1	Новомосковский	93.005648
2	Зеленоградский	73.805642
11	Западный	68.228421
8	Северо-Восточный	64.210708
0	Троицкий	63.860896
6	Северный	63.277042
9	Южный	58.893818
10	Восточный	50.046964
4	Юго-Восточный	48.294582
3	Юго-Западный	46.766018
7	Северо-Западный	45.148395

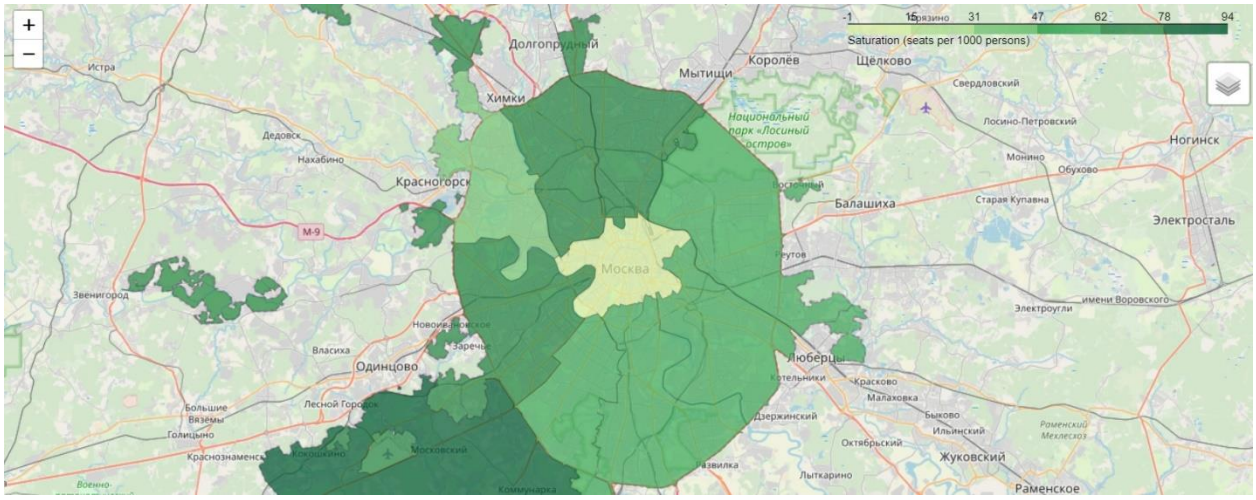
Just sorting the data by the Saturation measure reveals the most promising districts in terms of business potential, the ones with the lowest Saturation values, namely, "Северо-Западный", "Юго-Западный" and "Юго-Восточный".

To get more intuitive results, I've used the data above to create a "heat-map" of the districts (using Folium library and its choropleth map):

```
seats_map = folium.Map(location=[55.709, 37.627], zoom_start=10)

seats_map.choropleth(geo_data=gdf, name='choropleth', data=df_data, columns=['NAME', 'Saturation'], key_on
='feature.properties.NAME', fill_color='YlGn', fill_opacity=0.7, line_opacity=0.2, legend_name='Saturation
(seats per 1000 persons)')
folium.LayerControl().add_to(seats_map)

seats_map
```



4. Results

The project's primary results are (1) the data revealing the Moscow's eating-out services saturation by district (the dataframe above) and (2) the "heat map" of Moscow giving a visual representation of the data. The main finding that should be of interest to the stakeholders is the list of the least saturated (in terms of seats availability) districts of Moscow: "Северо-Западный", "Юго-Западный", "Юго-Восточный" (which can easily be seen on the map).

5. Discussion

The work revealed the districts of Moscow with the highest business potential in terms of eating-out market saturation. Further steps to enhance the insight (are out of the scope of this project, but) might include:

- getting down to the next level of detail and performing similar analysis with neighborhoods data (each district contains several neighborhoods),

- refining the analysis by adding data on consumer traffic, population income levels etc.

6. Conclusion

In this project, I aggregated data reflecting Moscow's eating-out service sector, calculated a measure of market saturation for each district and provided a visual representation of the output data as a map. The findings of this project may be used as a supportive tool by decision-makers in the eating-out industry, as well as by the regulators and the city authorities. Also, you are welcome to use my map when choosing a place for eating out in Moscow next time!