Analysis of top-5 Volga Federal District cities in Russian Federation using Foursquare API

Marat S. Mukhametzhanov

January 2021

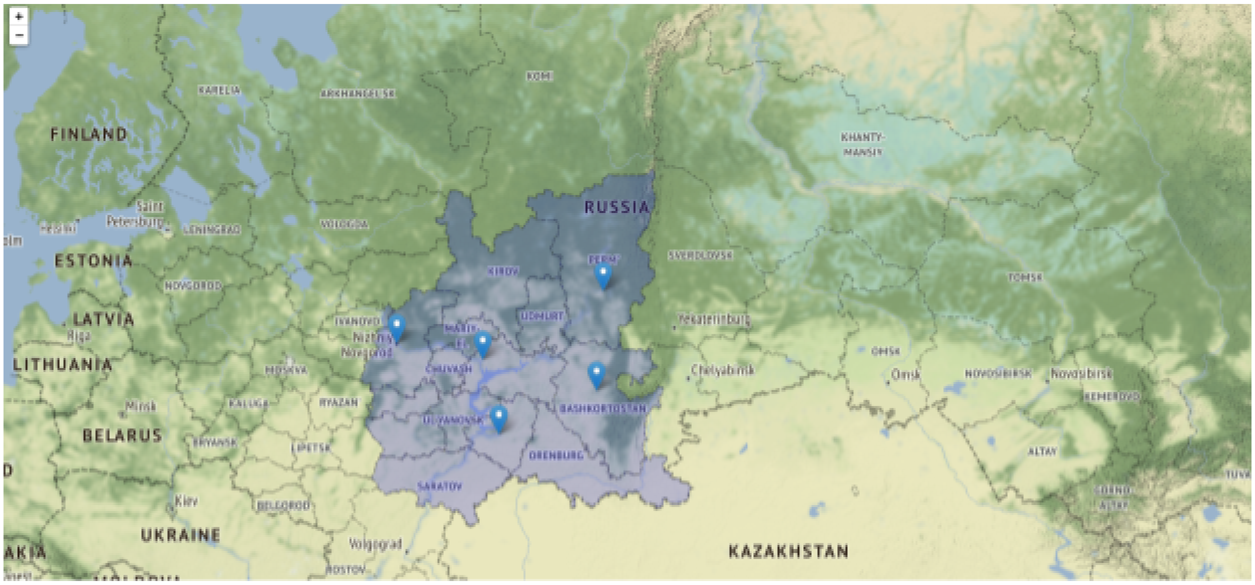
1 Introduction

1.1 Background

Russian Federation consists of the following 8 large federal districts:

Federal district	Area (km2)	Population(2010 census)	Population density(per km2)	HDI(2017)	Federal subjects	Administrative Centre
Central	650200	38438600	59.1	0.838	18	Moscow
Volga	1037000	29900400	28.8	0.797	14	Nizhny Novgorod
Siberian	4361800	17178298	3.9	0.788	10	Novosibirsk
Southern	427800	16141100	37.7	0.793	8	Rostov-on-Don
Northwestern	1687000	13583800	8.1	0.827	11	Saint Petersburg
Ural	1818500	12082700	6.6	0.833	6	Yekaterinburg
North Caucasian	170400	9496800	55.7	0.785	7	Pyatigorsk
Far Eastern	6952600	8371257	1.2	0.801	11	Vladivostok

The Volga Federal District is one of the biggest districts in Russia consisting of 14 federal subjects with almost 30 million people in total. It is mainly placed along Volga, being the longest river in Europe. It has 5 cities with a population over 1 million: Nizhny Novgorod (rus. Нижний Новгород, the administrative centre of the district, population ~1.252 million people), Kazan (rus. Казань, population ~1.257 million people), Ufa (rus. Уфа, population ~1.128 million people), Samara (rus. Самара, population ~1.156 million people), Perm (rus. Пермь, population ~1.055 million people):



Volga Federal District and its top-5 cities (indicated by markers)

1.2 Business Problem

A series of restaurants in Nizhny Novgorod has become successful, so its stakeholders are interested to extend their business into other cities. However, Russian Federation is a country with large distances between different cities and each region has its own cultural, economic and even natural or weather specifics. So, the stakeholders first consider opening a restaurant in a city of the same district. It is reasonable to consider only cities similar to Nizhny Novgorod, where their business has already become successful. One can see that there are 4 cities similar to Nizhny Novgorod in Volga federal district: Kazan, Ufa, Samara, and Perm. These cities are administrative centres of their regions and each of them has more than one million residents.

The main problem considered in this study can be briefly described as follows.

1. Find the city of the Volga Federal District closest to Nizhny Novgorod in terms of public venues availability and their categories.
2. In the chosen city, find the best places, where it is reasonable to open a restaurant based on known locations of the restaurants in the original city (Nizhny Novgorod).

1.3 Interest

Since Nizhny Novgorod is a big, grown and economically developed city, then the present study can be interesting to different businessmen who consider extending their business to other cities. In particular, there are several series of food restaurants, sushi bars, pizzerias, bars, etc. (they are not listed here only for privacy reasons). Moreover, the methodologies presented in this study are easily extendable to other regions and districts not only in Russia, but in other countries as well (e.g., in Europe, USA or Canada). In this case, only the Data section of the Jupyter notebook related to this study will differ.

2 Data

2.1 Data wrangling

2.1.1 Postal codes and their coordinates.

The main methodology used in the present research is clustering of a city based on the information of all venues available for a neighborhood in a fixed radius. Since all chosen cities are well divided by neighborhoods based on their postal codes, which is a unique common criterion for different cities in Russian Federation, then we will use postal codes of each city as places around which the venues are searched.

In this case, first, we need the following data for each chosen city:

1. List of all postal codes
2. Latitude and longitude associated with each postal code in order to make queries using Foursquare API.

The list of all postal codes in Russia is available from the following link: http://download.geonames.org/export/zip/RU.zip

The information available in this zip-archive is well described in the respective readme file. In particular, for each postal code (obviously, unique), its place name (mostly consisting of the city's name), estimated latitude and longitude are provided:

	country code	postal code	place name	admin name1	admin code1	admin name2	admin code2	admin name3	admin code3	latitude	longitude	accuracy
0	RU	385000	Майкоп	Адыгея Республика	1.0	NaN	NaN	NaN	NaN	44.8802	40.2166	1.0
1	RU	385001	Майкоп 1	Адыгея Республика	1.0	NaN	NaN	NaN	NaN	44.8802	40.2166	1.0
2	RU	385002	Майкоп 2	Адыгея Республика	1.0	NaN	NaN	NaN	NaN	44.8802	40.2166	1.0
3	RU	385003	Майкоп 3	Адыгея Республика	1.0	NaN	NaN	NaN	NaN	44.8802	40.2166	1.0
4	RU	385006	Майкоп 6	Адыгея Республика	1.0	NaN	NaN	NaN	NaN	44.8802	40.2166	1.0

However, the latitude and longitude for each postal code are estimated very roughly, because they coincide for many postal codes in each city. To solve this issue, let us use the Python's Nominatim tool in geopy.geocoders library for retrieving the real coordinates for each postal code (and for each city location as well). An example for the postal code '603076' in Nizhny Novgorod is presented below:

```
geolocator = Nominatim(user_agent="PFO_explorer")
pcode = '603076'
address = pcode+', '+'Nizhny Novgord'+', Russia'
try:
        location = geolocator.geocode(address)
except:
        print('error in '+address+', coordinates have not been changed')
```

Since the *geolocator.geocode(address)* for several postal codes returned *None,* then for these postal codes, the original roughly measured values have been used (for each city, there were no more than 5% postal codes with this issue, so the obtained accuracy is acceptable).

The folium map of Nizhny Novgorod covered by the neighborhoods defined through the postal code is presented below. Here, each neighborhood is represented by a circle of a fixed radius (equal to 500 metres). However, one can see that the radius of 500 metres is not sufficient for covering the city. For this reason, the radius of 1000 metres has been chosen for each neighborhood (this radius will be used in the Foursquare API for searching the venues), since, on the one hand, it becomes sufficient to cover the city, and, on the other hand, a venue located in 1000 metres is accessible even by walking.

(a) (b) (c)

The folium maps of Nizhny Novgorod covered by the neighborhoods defined by the postal codes. Each neighborhood is represented by a circle of a fixed radius. (a) The map of the whole city using the radius = 500m. (b) The map of a part of the city that includes a park, cinema, metro station, mall, etc. showing that the radius of 500m. is not sufficient (this part of the city is not covered even it is easily accessible). (c) The map of the whole city using the radius of 1000m. In this case, all parts of the city are covered well.

2.1.2 Venues available from each neighborhood

Once the coordinates of each neighborhood have been obtained, the Foursquare API can be used for retrieving the available nearby venues for each neighborhood by the following commands:

url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET, VERSION,lat, lng, radius, LIMIT)
results = requests.get(url).json()["response"]['groups'][0]['items']

By default, the Foursquare API returns the following information for each neighborhood (other fields are omitted): Neighborhood, Neighborhood Latitude, Neighborhood Longitude, Venue, Venue Latitude, Venue Longitude, Venue Category. However, using the default venue categories can be redundant: e.g., in our case, the venue categories “chinese restaurant” and “italian restaurant” are similar, we are not interested in detailed categories of each venue. For this reason, we will use macro categories for each venue:

- Arts & Entertainment (id 4d4b7104d754a06370d81259),
- College & University (id 4d4b7105d754a06372d81259),
- Event (id 4d4b7105d754a06373d81259),
- Food (id 4d4b7105d754a06374d81259),
- Nightlife Spot (id 4d4b7105d754a06376d81259),
- Outdoors & Recreation (id 4d4b7105d754a06377d81259),
- Professional & Other Places (id 4d4b7105d754a06375d81259),
- Residence (id 4e67e38e036454776db1fb3a),
- Shop & Service (id 4d4b7105d754a06378d81259),
- Travel & Transport (id 4d4b7105d754a06379d81259),

First, for each macro category, we will retrieve the list of all venue categories belonging to it:

url='https://api.foursquare.com/v2/venues/categories?&client_id={}&client_secret={}&v={}'.format(
CLIENT_ID,CLIENT_SECRET, VERSION)
results = requests.get(url).json()['response']['categories']

The *results* variable contains now the list of all macro categories. Each macro category is represented by a dictionary with the key *‘categories’* containing the list of all its sub-categories (which in their turn are represented by the dictionaries in the same way). After retrieving the list of all sub-categories for each macro-category, it is easy to assign a macro category for each venue (let us denote ‘venue category’ as ‘micro category’ and ‘venue macro category’ simply as ‘venue category’, hereinafter, just for simplicity, but the data about the micro categories will be also used later in order to determine the best places in the chosen city):

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Micro Category	Venue Category
0	603000	56.317042	43.994228	Burrito Family	56.316153	43.992785	Burrito Place	Food
1	603000	56.317042	43.994228	Franky Bar	56.316330	43.994536	Cocktail Bar	Nightlife Spot
2	603000	56.317042	43.994228	Surf Coffee	56.317038	43.994256	Coffee Shop	Food
3	603000	56.317042	43.994228	Большая Покровская улица	56.320291	43.998442	Road	Travel & Transport
4	603000	56.317042	43.994228	Бикрам-Йога | Нижний Новгород	56.319637	43.994256	Yoga Studio	Outdoors & Recreation

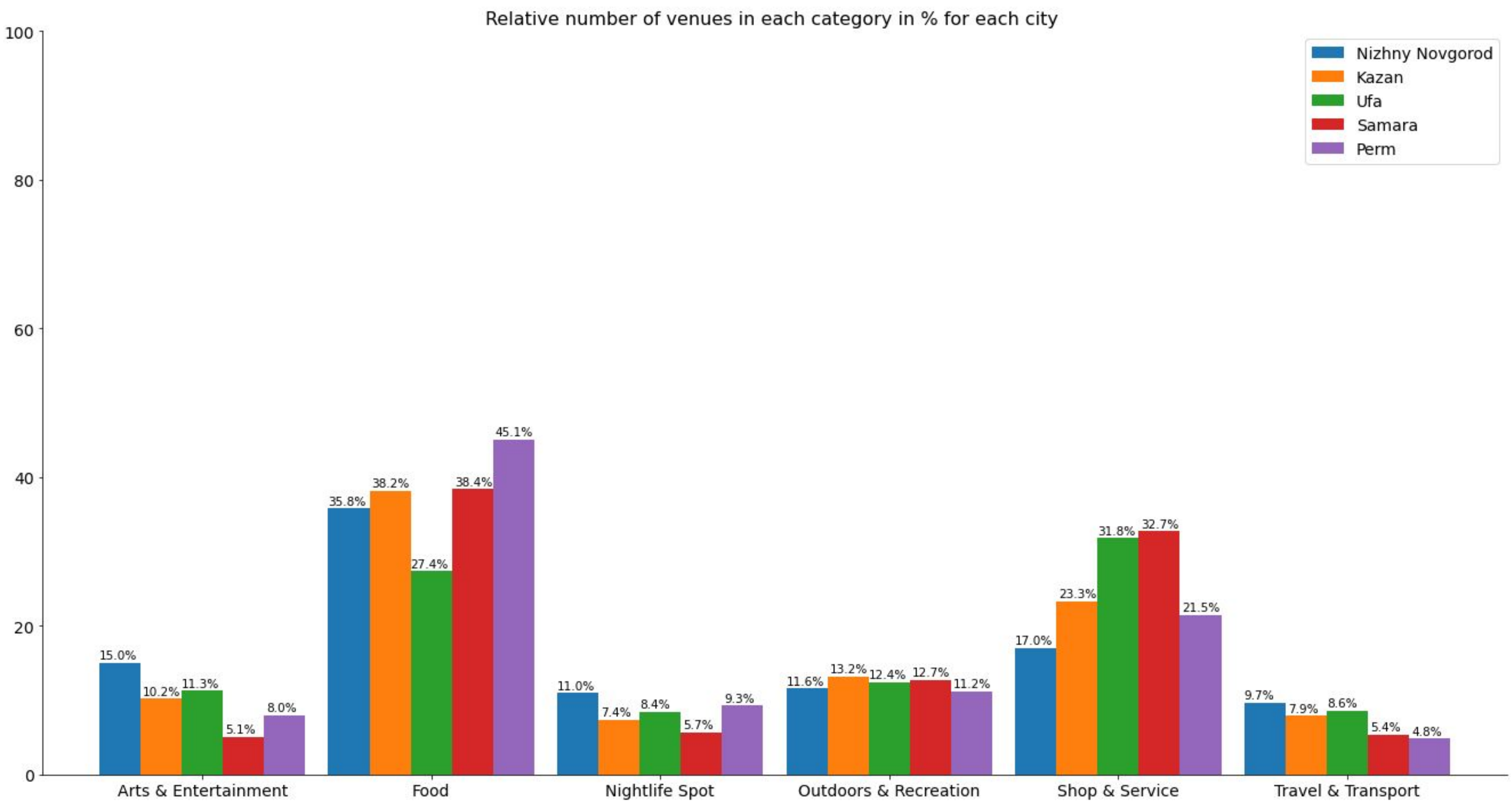
2.2 Data Cleaning

Let us count the number of all venues for each category in each city:

Category\City	Nizhny Novgorod	Kazan	Ufa	Samara	Perm
Arts & Entertainment	689	287	341	241	188
College & University	29	3	4	3	1
Event	0	0	0	0	0
Food	1643	1077	824	1827	1061
Nightlife Spot	503	208	252	270	220
Outdoors & Recreation	530	372	373	605	264
Professional & Other Places	9	36	20	12	18
Residence	1	0	3	0	3
Shop & Service	778	657	957	1554	507
Travel & Transport	444	222	260	256	114
Total	4626	2862	3034	4768	2376

We can see that there is almost no data about the categories ‘Event’ and ‘Residence’ for all the cities. Moreover, even if there is some data about ‘College & University’ in Nizhny Novgorod, it is almost unavailable for the other cities, so this category cannot be used to estimate similarity of the cities. For the same reason, the category ‘Professional & Other Places’ cannot be used to estimate similarity between these cities, because the data about it is almost unavailable for Nizhny Novgorod. The data about the other categories is widely available for all the cities, so we will use these 6 categories to estimate similarity between the cities: Arts & Entertainment (id 4d4b7104d754a06370d81259), Food (id 4d4b7105d754a06374d81259), Nightlife Spot (id 4d4b7105d754a06376d81259), Outdoors & Recreation (id 4d4b7105d754a06377d81259), Shop & Service (id 4d4b7105d754a06378d81259), Travel & Transport (id 4d4b7105d754a06379d81259)

In order to visualize better the distribution of all venues between categories for each city, let us calculate the relative number of venues in each category dividing the number of venues from the previous table by the total number of venues for each city (the categories excluded previously are not considered here). The results are visualized in the following barplot for each city and each venue category.



We can also see from the previous table that the number of venues in each category is different in the cities. For this reason, let us calculate the normalized number of venues for each neighborhood in each city, dividing the number of nearby venues for each neighborhood in each category by the total number of venues in each category: e.g., for Nizhny Novgorod, the following data is assigned for the first 5 neighborhoods:

	Neighborhood	Arts & Entertainment	Food	Nightlife Spot	Outdoors & Recreation	Shop & Service	Travel & Transport
0	603000	0.010160	0.026172	0.017893	0.016981	0.024422	0.024775
1	603001	0.023222	0.021302	0.029821	0.018868	0.007712	0.018018
2	603002	0.005806	0.004869	0.000000	0.005660	0.017995	0.013514
3	603003	0.004354	0.006086	0.001988	0.009434	0.007712	0.002252
4	603004	0.002903	0.004869	0.000000	0.009434	0.006427	0.006757

The obtained data will be used for clustering the cities and determining the closest city to Nizhny Novgorod in terms of available venue categories. After that, the number of venues in each micro category for each neighborhood obtained previously will be used to determine the best places to locate the restaurants in the chosen city.