Final report

IBM ML course final project: the Battle of Neighbourhoods

Introduction

The problem we will solve is: which of the neighbourhoods of Toronto are more suitable to open there a restaurant?

To solve this problem we will find out by FourSquare API how many restaurants are within 1000 meters from each of Toronto Neighbourhoods.

Then we will divide all Toronto neighbourhoods into three groups by K-means clustering. The first group will contain neighbourhoods with a few of restaurants. And this group is the most suitable to be considered for launching there a new restaurant. The second group will contain neighbourhoods with a medium number of restaurants. And it is also not so bad choice to open there a new restaurant. And the third group is not so suitable to open there a new restaurant business because there are a lot of restaurants there.

Our analysis may be interesting for those people, who are planning to start (or extend, or improve) a restaurant business in Toronto, and looking for a suitable place for launching a new restaurant. Or may be someone has already had a restaurant in Toronto, and is not completely satisfied by the results, and is searching for the reasons why his/her restaurant business is not so successful.

Data

First we will scrape the following Wikipedia page,

https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M,

in order to obtain the list of all Toronto neighbourhoods, as well as their postal codes and boroughs. The 10 first rows of dataframe looks like

	Postcode	Borough	Neighbourhood
0	M1B	Scarborough	Rouge, Malvern
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union
2	M1E	Scarborough	Guildwood, Morningside, West Hill
3	M1G	Scarborough	Woburn
4	M1H	Scarborough	Cedarbrae
5	M1J	Scarborough	Scarborough Village
6	M1K	Scarborough	East Birchmount Park, Ionview, Kennedy Park
7	M1L	Scarborough	Clairlea, Golden Mile, Oakridge
8	M1M	Scarborough	Cliffcrest, Cliffside, Scarborough Village West
9	M1N	Scarborough	Birch Cliff, Cliffside West

This dataframe consists of 103 rows and 3 columns. Next for each row we add its latitude and longitude from the link: http://cocl.us/Geospatial_data
Thus the final dataframe df contains all Toronto postcodes and boroughs as well as Toronto

neighbourhoods grouped by their postcodes. Two last columns of the dataframe contain latitudes and longitudes of these neighbourhood groups. The dataframe df looks like

	Postcode	Borough	Neighbourhood	Latitude	Longitude
0	M1B	Scarborough	Rouge, Malvern	43.806686	-79.194353
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union	43.784535	-79.160497
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
3	M1G	Scarborough	Woburn	43.770992	-79.216917
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476

and contains 103 rows and 5 columns.

Now add to df one more column 'sum', where there will be the numbers of restaurants within 1000 meters from the neighbourhoods. Next for each row in df we take the values of latitude and longitude and form for them the FourSquare API query, where radius=1000 and search_query=restaurant. As a result we get url, which will be then transformed to json format (by the command “results = requests.get(url).json()”).

And then we put to the corresponding row of the column 'sum' the length of “results['response']['venues']”, which is exactly the number of restaurants within 1000 meters from the point we are interested in.

Here are the first 10 rows of the final dataframe

	Postcode	Borough	Neighbourhood	Latitude	Longitude	sum
0	M1B	Scarborough	Rouge, Malvern	43.806686	-79.194353	0
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union	43.784535	-79.160497	1
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711	5
3	M1G	Scarborough	Woburn	43.770992	-79.216917	3
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476	8
5	M1J	Scarborough	Scarborough Village	43.744734	-79.239476	7
6	M1K	Scarborough	East Birchmount Park, Ionview, Kennedy Park	43.727929	-79.262029	5
7	M1L	Scarborough	Clairlea, Golden Mile, Oakridge	43.711112	-79.284577	3
8	M1M	Scarborough	Cliffcrest, Cliffside, Scarborough Village West	43.716316	-79.239476	1
9	M1N	Scarborough	Birch Cliff, Cliffside West	43.692657	-79.264848	4

Methodology

First we import BeautifulSoup and use it to scrape the table from Wikipedia. The Python code looks like:

```
import requests
from bs4 import BeautifulSoup
```

```
res = requests.get("https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M")
```

```
soup = BeautifulSoup(res.content,'lxml')
```

```
table = soup.find_all('table')[0]
df = pd.read_html(str(table), header = None)[0]
```

Then we will refine and modify the dataframe df in the following way.

1. Rename the columns as 'Postcode', 'Borough', and 'Neighbourhood'
2. Drop the rows, where Borough is "Not assigned"
3. If Neighbourhood is "Not assigned", then assign its value the same as for Borough
4. Neighbourhoods with the same Postcode are joined and separated by comma
5. Download the coordinates for each Neighbourhood group from by the link: http://cocl.us/Geospatial_data and add them to two new columns of df “latitude” and “longitude”

After these steps the dataframe df contains all Toronto postcodes and boroughs as well as Toronto neighbourhoods grouped by their postcodes. Two last columns of the dataframe contain latitudes and longitudes of these neighbourhood groups.

Now we will use FourSquare API to complete preparing df for machine learning analysis. For each group of neighbourhoods in df we will be interested in number of restaurants within 1000 meters from the geographical position of that group (which is given by two last columns of df in the corresponding row). So, first put radius = 1000 and search_query='restaurant'.

First add to df an (empty) column 'sum', where there will be the numbers of restaurants within 1000 meters from the neighbourhoods. Next for each row in df we take the values of latitude and longitude and form for them the FourSquare API query, where radius and search_query are defined above. As a result we get url, which will be then transformed to json format (by the command results = requests.get(url).json()). And then we put to the corresponding row of the column 'sum' the length of “results['response']['venues']”, which is exactly the number of restaurants within 1000 meters from the point we are interested in. The python code looks like:

```
for k, row in df.iterrows():
    (lat,lng)= (row["Latitude"], row["Longitude"])
    url = 'https://api.foursquare.com/v2/venues/search?client_id={}&client_secret={}&ll={},\
    {}&v={}&query={}&radius={}&limit={}'\
    .format(CLIENT_ID, CLIENT_SECRET, lat, lng, VERSION, search_query, radius, LIMIT)
    results = requests.get(url).json()
    df.set_value(index=k,col='sum',value=len(results['response']['venues']))
```

Now df is the final dataframe that we will use in the next step for machine learning analysis.

We chose the K-means clustering method with the number of clusters is equal to 3. And we will clustering the list of 103 groups of Toronto neighbourhoods by three parameters: latitude, longitude and sum (which equals to the number of restaurants within 1000 meters from the neighbourhood). Our hypothesis is that 3 clusters may be interpreted as:

- Neighbourhoods with a few restaurants. These neighbourhoods will be the best choice to open there a new restaurant.
- Neighbourhoods with a medium number of restaurants. These neighbourhoods will be also not so bad choice to open there a new restaurant.
- Neighbourhoods with a lot of restaurants. They are not recommended to open there a new restaurant.

And this hypothesis will be approved (see the section “Results”).

The python code for our machine learning analysis looks like:

```
from sklearn.cluster import KMeans
```

```
X = df[['sum','Longitude','Latitude']]
```

```
# set number of clusters
N = 3

# run k-means clustering
kmeans = KMeans(n_clusters=N, random_state=0).fit(X)
```

Results

The clusters are:

```
array([0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 2, 1, 2, 2, 0, 0, 0, 2, 0, 0, 2,
       1, 0, 0, 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0, 2, 0, 2, 1, 1, 2, 2,
       0, 1, 2, 1, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 0, 2, 1,
       1, 1, 0, 1, 1, 0, 0, 2, 2, 1, 2, 1, 2, 0, 0, 0, 0, 2, 2, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0], dtype=int32)
```

Cluster “0”: 0-6 restaurants
Cluster “2”: 7-16 restaurants
Cluster “1”: 17-30 restaurants

So we recommend to open a new restaurant in the Neighbourhoods of the cluster “0”, we recommend to consider the neighbourhoods of the cluster “2” as the possible variant to open there a new restaurant, and we do not recommend to open a new restaurant in the neighbourhoods of the cluster “1”.

Discussion

Sure the number of restaurants within 1000 meters from a neighbourhood is not the unique factor to decide is it a good idea to open there a new restaurant or not. And our clustering may be considered just as the first step for more detailed analysis based on a lot of other factors. But this more detailed analysis is not the subject of our consideration now.

Conclusion

In this investigation we have clustered all Toronto neighbourhoods by three groups. Which are interpreted as:

- Strongly recommended for launching there a new restaurant (cluster “0”)
- Recommended for launching there a new restaurant (cluster “2”)
- Not recommended for launching there a new restaurant (cluster “1”)