jupyter WR_CapstoneProject Last Checkpoint: 2 小时前 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Markdown

Rui Wu's Final Capstone Project -

Finding A Proper Location To Live In Shanghai, China

Table of Contents

Project Background (Introduction/Business Problem Section)

1. Problem
   Actually this problem is a real problem of myself. I got a job in Shanghai, China, and I came to this city last year. Because I am a consultant, so I often go on business trip to other cities.
   Living near metro station is important to me, which make easier for me to access to airport or train station. And besides living near metro station, of course, for living easy and convenient, I want to live in a place where there are restaurants, cafe, fitness centers, pharmacies, supermarkets or malls. Last year, I lived near Hailun road('海伦路' in Chinese) metro station and this place have all these venues. I like this place. However, I have to move to other places because of the end of house contract and some other reasons, so I want to find a similar place in Shanghai to live in.
   And my problem is, I want to find a similar place(as Hailun road) to live in, which can meet my following requirements:
   1) near metro station, within 500 meters;
   2) there are restaurants, cafe, fitness centers, pharmacies, supermarkets or malls around the station;
   3) this station has multi-lines across, which can be more convenient than just 1 line;
   4) stations near airport and train station are out of consideration because of the daily noise.

2. Who would be interested in this project?
   1) People who needs to rent apartments in Shanghai, China, like me. You can get some insights of these places to help you make desicion of where to live;
   2) Stakeholders of a renting apartment app, which can provide a recommendation of places to app users based on their conditions.

jupyter WR_CapstoneProject Last Checkpoint: 2 小时前 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Markdown

Before we get the data and start exploring it, let's download all the dependencies that we will need.

In [1]:

```
import numpy as np # library to handle data in a vectorized manner

import pandas as pd # library for data analsysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # library to handle JSON files

#!conda install -c conda-forge geopy --yes # uncomment this line if you haven't completed the Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude and longitude values

import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

# import k-means from clustering stage
from sklearn.cluster import KMeans

#!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't completed the Foursquare API lab
import folium # map rendering library

print('Libraries imported.')
```

```
Libraries imported.
```

1. Introduce Data Source And Clean Dataset

Data Source (Data Section)

I found geographical coordinates of almost all metro stations in Shanghai from https://download.csdn.net/download/sinat_29675423/10844042. The data is stored in sql file. I downloaded the file and copy these data directly into this notebook.
And I put these data into a dataframe. Each row represents a station and its coordinates. In column 'id', the 1st number is the line number and the 2nd, 3rd number represent the sequence number of stations of each line.
It also needs to be noticed that because some lines are under continual construction in these years, some new stations appear. And the data I use in this project was collected in 2015/07/28 and some was updated in 2018.

There are line 1 to 13, 16, 17 of Shanghai metro, so first I create a dataframe named sh_station.

jupyter WR_CapstoneProject Last Checkpoint: 2 小时前 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Markdown

There are line 1 to 13, 16, 17 of Shanghai metro, so first I create a dataframe named sh_station.

In [2]:

```
# subway1
subway1 = [(101, '莘庄', 121.385379, 31.111193),
(102, '外环路', 121.393020, 31.120899),
(103, '莲花路', 121.402943, 31.130986),
(104, '锦江乐园', 121.414107, 31.142217),
(105, '上海南站', 121.430041, 31.154579),
(106, '漕宝路', 121.434966, 31.168167),
(107, '上海体育馆', 121.437423, 31.182813),
(108, '徐家汇', 121.436837, 31.195338),
(109, '衡山路', 121.446424, 31.204528),
(110, '常熟路', 121.449141, 31.213524),
(111, '陕西南路', 121.459957, 31.217231),
(112, '黄陂南路', 121.473306, 31.222745),
(113, '人民广场', 121.475719, 31.232770),
(114, '新闸路', 121.468151, 31.238373),
(115, '汉中路', 121.458334, 31.241610),
(116, '上海火车站', 121.458216, 31.249690),
(117, '中山北路', 121.459204, 31.258891),
(118, '延长路', 121.455329, 31.271675),
(119, '上海马戏城', 121.452023, 31.279895),
(120, '汶水路', 121.450251, 31.292556),
(121, '彭浦新村', 121.448642, 31.306604),
(122, '共康路', 121.447063, 31.318936),
(123, '通河新村', 121.441546, 31.331130),
(124, '呼兰路', 121.437711, 31.339703),
(125, '共富新村', 121.434063, 31.355082),
(126, '宝安公路', 121.430914, 31.369555),
(127, '友谊西路', 121.427953, 31.381296),
(128, '富锦路', 121.424661, 31.392260)]

# subway2
subway2 = [(201,'浦东国际机场',121.799283,31.150459),
(202, '海天三路', 121.796878, 31.168459),
(203, '远东大道', 121.755296, 31.199360),
(204, '凌空路', 121.723791, 31.192826),
(205, '川沙', 121.698210, 31.186741),
(206, '华夏东路', 121.681098, 31.196553),
(207, '创新中路', 121.673713, 31.213871),
(208, '唐镇', 121.656547, 31.213340),
(209, '广兰路', 121.621072, 31.211050),
(210, '金科路', 121.601989, 31.204213),
(211, '张江高科', 121.587687, 31.201832),
(212, '龙阳路', 121.557634, 31.203575),
(213, '世纪公园', 121.550909, 31.209421),
(214, '上海科技馆', 121.544313, 31.218771),
(215, '世纪大道', 121.527221, 31.228764),
(216, '东昌路', 121.515556, 31.233270),
```

jupyter WR_CapstoneProject Last Checkpoint: 2 小时前 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Markdown

```
396
397 # subway16
398 subway16 = [(1601, '淘水湖', 121.929583, 30.907245),
399 (1602, '临港大道', 121.910851, 30.923519),
400 (1603, '书院', 121.850520, 30.959264),
401 (1604, '惠南东', 121.793800, 31.026448),
402 (1605, '惠南', 121.761677, 31.053828),
403 (1606, '野生动物园', 121.699218, 31.050325),
404 (1607, '新场', 106.842061, 28.292216),
405 (1608, '航头东', 121.617494, 31.054919),
406 (1609, '鹤沙航城', 121.611239, 31.077797),
407 (1610, '周浦东', 121.606946, 31.110090),
408 (1611, '罗山路', 121.593152, 31.153259),
409 (1612, '华夏中路', 121.583109, 31.175759),
410 (1613, '龙阳路', 121.557634, 31.203575)]
411
412 # subway17
413 subway17 = [(1701, '虹桥火车站', 121.321550, 31.193950),
414 (1702, '中国博览会北', 121.561771, 31.211005),
415 (1703, '蟠龙路', 121.277679, 31.187017),
416 (1704, '徐盈路', 121.257950, 31.179514),
417 (1705, '徐泾北城', 121.237140, 31.182759),
418 (1706, '嘉松中路', 121.219276, 31.174879),
419 (1707, '赵巷', 121.195457, 31.147995),
420 (1708, '汇金路', 121.151802, 31.163389),
421 (1709, '外青松公路', 121.124428, 31.137487),
422 (1710, '漕盈路', 121.093460, 31.160048),
423 (1711, '淀山湖大道', 121.094757, 31.142006),
424 (1712, '朱家角', 121.060234, 31.108959),
425 (1713, '东方绿舟', 121.007406, 31.099011)]
426
427 lines = [subway1,subway2,subway3,subway4,subway5,subway6,subway7,subway8,subway9,subway10,subway11,subway12,subway13,subway
428 stations = [station for line in lines for station in line]
429
430 sh_station = pd.DataFrame(stations)
431 sh_station.columns = ['id', 'st_name', 'lng', 'lat']
432 sh_station.head()
```

Out[2]:

	id	st_name	lng	lat
0	101	华庄	121.385379	31.111193
1	102	外环路	121.393020	31.120899
2	103	莲花路	121.402943	31.130986
3	104	锦江乐园	121.414107	31.142217
4	105	上海南站	121.430041	31.154579

jupyter WR_CapstoneProject Last Checkpoint: 2 小时前 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Markdown

In [4]:
```
sh_station.shape

```

Out[4]: (396, 4)

Here we can see there are 396 stations. However, some stations are the same. So let's check the stations which has multi-lines across there.

In [5]:
```
freq = sh_station.groupby('st_name').size()
freq_over1 = freq[freq>1]
freq_over1
```

Out[5]:
```
st_name
上海体育馆      2
上海南站        2
上海火车站      3
世纪大道        4
东安路          2
东方体育中心    3
东明路          2
中山公园        3
中潭路          2
交通大学        2
人民广场        3
华夏中路        2
南京东路        2
南京西路        3
四平路          2
大木桥路        2
大连路          2
天潼路          2
宜山路          3
宝山路          2
巨峰路          2
常熟路          2
延安西路        2
徐家汇          3
新天地          2
曲阜路          2
曹杨路          3
汉中路          3
江苏路          2
浦电路          2
海伦路          2
漕宝路          2
罗山路          2
耀华路          2
老西门          2
```

```
莘庄          2
蓝村路         2
虹口足球场       2
虹桥2号航站楼      2
虹桥火车站        3
虹桥路          3
西藏南路         2
金沙江路         3
金海路          2
镇坪路          3
长寿路          2
长清路          2
陕西南路         3
隆德路          2
静安寺          2
马当路          2
高科西路         2
龙华           2
龙华中路         2
龙漕路          2
龙阳路          3
dtype: int64
```

From the data shown above, Hailun Road('海伦路') station has 2 line across. So if I want to find a better place to live in, I had better find a place where the station has over 2 lines across. But 2 lines is also good for me.

In [6]:
```
# drop stations near airport and train station
freq_over1 = freq_over1.drop(index=['上海南站','上海火车站','虹桥2号航站楼', '虹桥火车站'])
freq_over1 = freq_over1.sort_values(ascending=False)
freq_over1
```

Out[6]:
```
st_name
世纪大道       4
龙阳路        3
人民广场       3
汉中路        3
徐家汇        3
宜山路        3
南京西路       3
虹桥路        3
金沙江路       3
镇坪路        3
曹杨路        3
陕西南路       3
东方体育中心     3
中山公园       3
交通大学       2
东明路        2
东安路        2
```



```
华夏中路       2
大连路        2
大木桥路       2
四平路        2
中潭路        2
新天地        2
南京东路       2
江苏路        2
曲阜路        2
西藏南路       2
龙华中路       2
龙华           2
高科西路       2
马当路        2
静安寺        2
隆德路        2
长清路        2
长寿路        2
金海路        2
虹口足球场      2
龙漕路        2
蓝村路        2
莘庄           2
肇嘉浜路       2
老西门        2
耀华路        2
罗山路        2
漕宝路        2
海伦路        2
浦电路        2
上海体育馆      2
dtype: int64
```

In [7]:
```
freq_over1.shape
```

Out[7]:
```
(53,)
```

So I will find the proper place to live in, which is one of the freq_over1 and is similar to Hailun road.

Clean Dataset

In [8]:
```
# delete the replicated stations(rows) and just keep the first one
sh_station = sh_station[~sh_station.duplicated(subset='st_name',keep='first')]
```