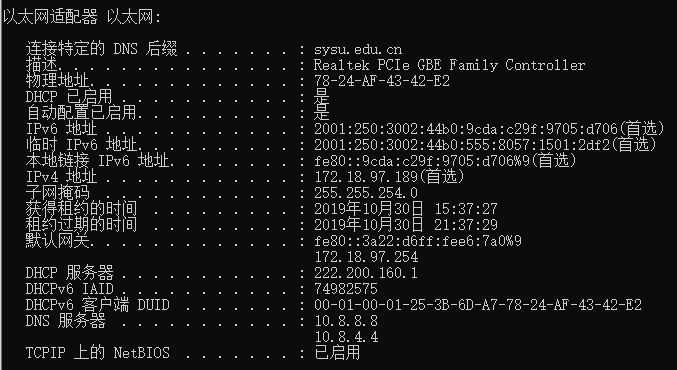
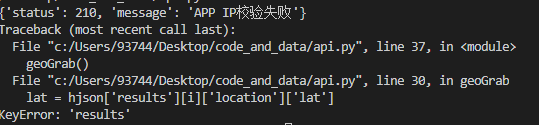
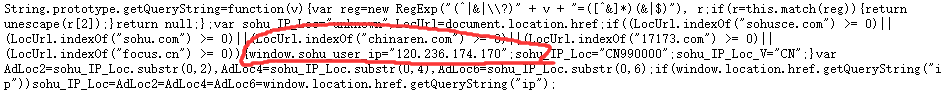
**Lab 7** 实验题目

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | **陈政培** | 学院 | 专业 |
| 姓名 | **17363011** | **智能工程学院** | **智能科学与技术** |

1. 实验目的： **k-means应用实践**
2. 实验环境：vs code、python3.6.8、百度Web API
3. 实验步骤：
4. 获取地图数据

在百度地图开放平台中创建应用时填写访问方的公网IP，一开始在cmd中输入ipconfig /all 获取本机ip地址

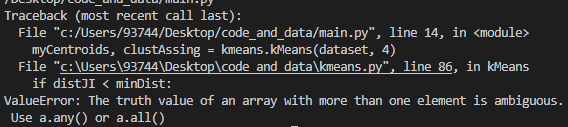
误以为在IP白名单处填写对应的网关IP就可以，结果出现报错

发现APP IP 校验失败，后来发现原来此处cmd获取的ip仍然是对于学校内网的ip。为了获取公网ip，我打开了<http://txt.go.sohu.com/ip/soip>以检查我的实际访问公网ip

原来我的实际公网ip是中山大学的120.236.174.170

1. 将应用AK放到python代码中，成功获取了Restaurant\_Data\_Beijing.txt
2. 测试loadDataSet()、randCent()和distEclud()函数
3. kMeans算法实现

在未修改kmeans.py文件前，可以直接显示测试结果，但是当把语句“from numpy import \*”修改成“import numpy as np”后代码出现了报错



本以为是语句

if distJI < minDist:

是数据结构之间的比较出现了问题，后来才发现python中除了numpy库中有一个sum函数，python本身也有一个sum函数。但是两者实现的功能和输出结果并不相同，所以出现了数据结构的错误

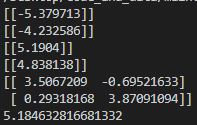
    return np.sqrt(np.sum(np.power(vecA - vecB, 2)))

需要将kmeans.py文件19行的代码中sum改成np.sum即可正常运行

1. 用kMeans算法对地图上的点聚类

原理和之前使用testSet一样

1. 实验结果与分析：
2. 从文本文件构建矩阵并建立辅助函数



输出结果分别为

print(min(dataset[:,0]))

print(min(dataset[:,1]))

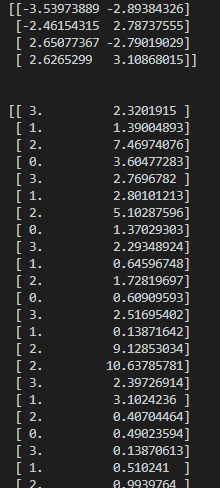
print(max(dataset[:,1]))

print(max(dataset[:,0]))

print(kmeans.randCent(dataset, 2))

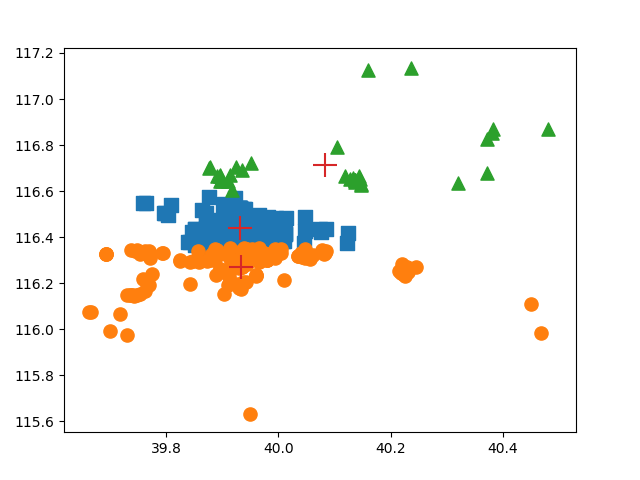
print(kmeans.distEclud(dataset[0], dataset[1]))

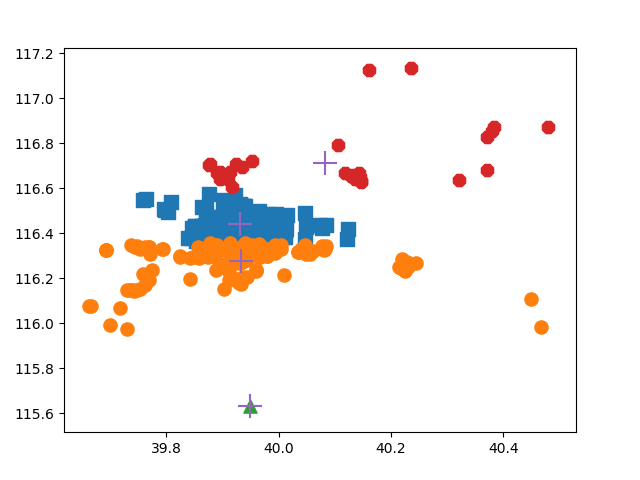
1. kMeans算法实现

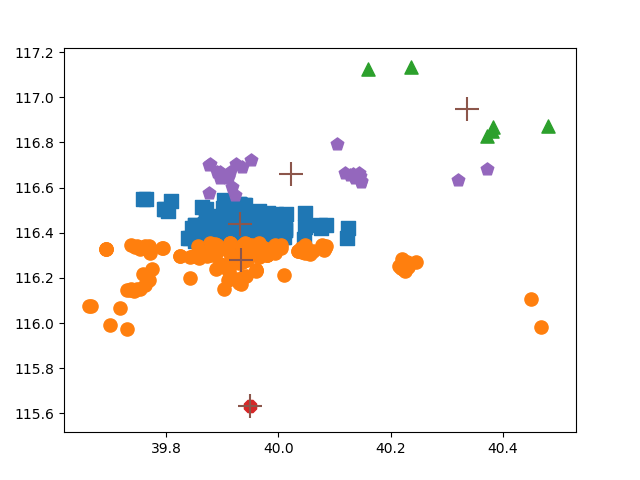


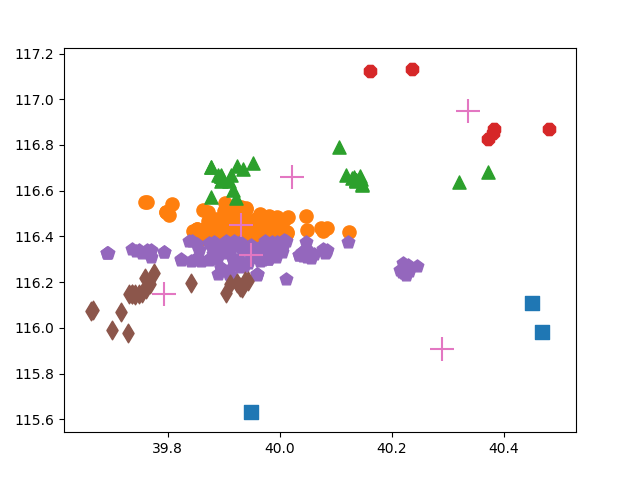
得到了4个质心的坐标和各个簇的分配结果（结果截图了一小部分）

1. 用kMeans算法对地图上的点聚类

Kmeans.clusterPlaces(3)

Kmeans.clusterPlaces(4)

Kmeans.clusterPlaces(5)

Kmeans.clusterPlaces(6)

1. 作业：
2. 对聚类结果可视化（包含样本点和簇中心，用不同颜色、记号标记）

4结果展示中已经实现了

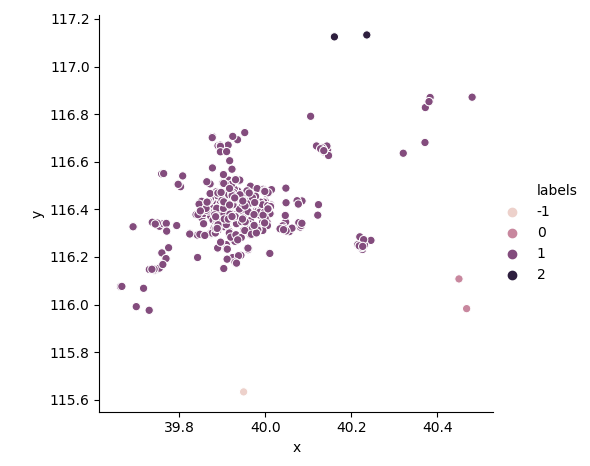
1. Kmeans.py中的语句代替并修改代码使其正常执行

修改结果见代码，遇到的问题和bug修正过程详见3.④中描述

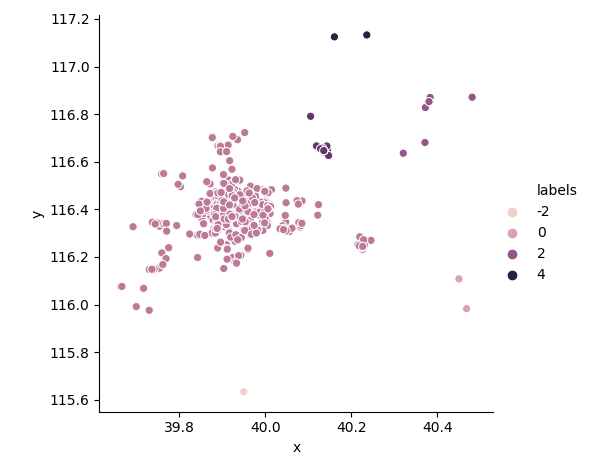
1. 尝试用DBSCAN聚类该数据集

使用了sklearn库中的cluster里的函数KMeans和DBSCAN进行聚类

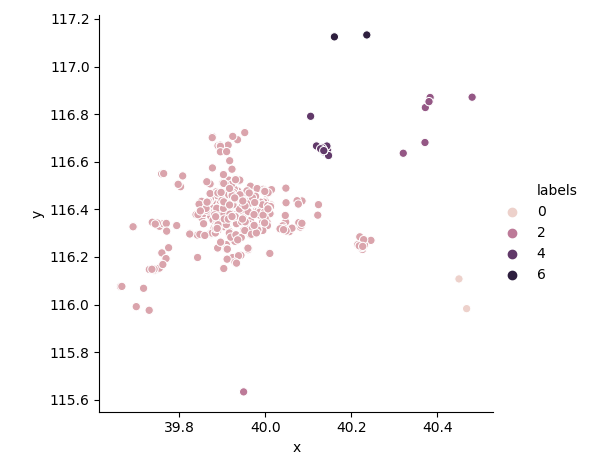
* 因为DBSCAN聚类函数是基于密度的聚类函数，所以不需要预先指定簇的个数，最终的簇的个数不确定。分类的依据是聚类的半径eps和邻域内最小点数目min\_samples
* 想要获得和KMeans相似的聚类效果则需要调试eps和min\_samples的大小，以得到合适的簇数目
* Eps = 0.2 ， min\_samples = 2 时获得3个簇



* Eps = 0.15 ， min\_samples = 1.5 时获5个簇



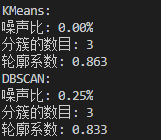
* Eps = 0.15 ， min\_samples = 1 时获得6个簇



1. 利用轮廓系数评估kMeans和DBSCAN对该数据集的聚类效果

通过sklearn中metrics里的函数silhouette\_score来对两种聚类算法的结果进行轮廓系数评估

* DBSCAN函数在实际计算时把一部分餐厅作为噪声屏蔽了
* 簇为3时



* 簇为5时

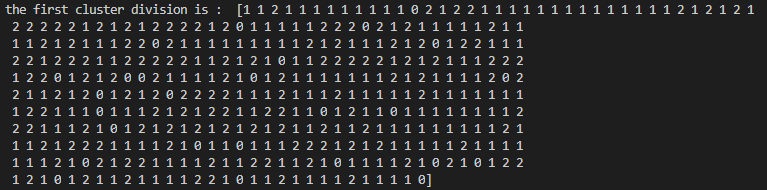


* 簇为6时



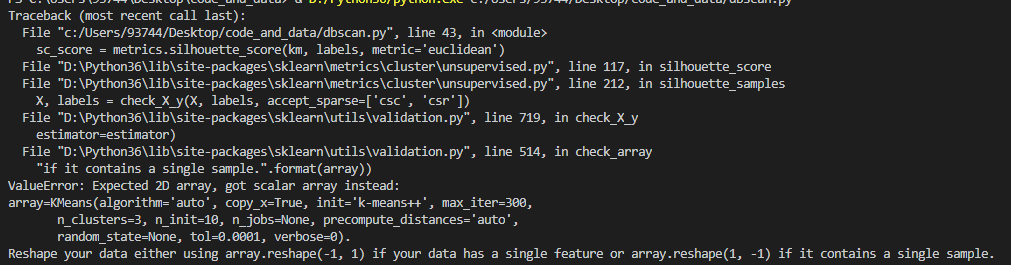
* 整体结论为在轮廓系数上DBSCAN相对更优，但是有一定噪声损失

1. 尝试在Tensorflow环境下实现kMeans

详见代码

可以实现聚类结果，但是由于时间原因可视化未进行颜色分类

1. 实验总结：

* APP IP 校验失败，后来发现原来此处cmd获取的ip仍然是对于学校内网的ip。为了获取公网ip，我打开了<http://txt.go.sohu.com/ip/soip>以检查我的实际访问公网ip
* python中除了numpy库中有一个sum函数，python本身也有一个sum函数。但是两者实现的功能和输出结果并不相同，所以出现了数据结构的错误
* 调试DBSCAN代码时报错

原因是silhouette\_score函数接受的参数中需要的list类型是（1，-1）的，而我们传进去的数据shape是错误的，对相应数据进行reshape即可

* 在网上找到的教程使用的tensorflow版本是1.0，所以很多tf函数都不能直接使用，必须通过tf.compat.v1.来访问老版本的函数

1. 参考文献：
   1. <https://www.jianshu.com/p/82db047eea13>
   2. <https://blog.csdn.net/u012967763/article/details/79149703>
   3. <https://github.com/aliceleee/KMeans_Tensorflow/tree/master/code>