



任务5

聚类分析

层次聚类——trip

⑧ The part five

任务5：层次聚类——trip

层次聚类 —— gower距离

距离计算方法1: gower距离

- 由于我们的数据中二元变量和名义变量比较多，区间尺度变量比较少，所以按照老师上课所讲的方法：使用gower距离作为聚类的标准
 - 计算公式：

$$d(i,j) = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}}$$

用户ID	进站位置_东西	出站位置_东西	进站位置_环线	出站位置_环线	早晚高峰	换乘	乘坐时间 (s)	价格	优惠类别	
67	3101687888	浦东	浦西	外外	内	False	True	3253	5	True
68	3101687888	浦西	浦东	中	外外	False	True	3409	6	False
208	2902982888	浦西	浦西	中	内	False	False	846	3	True
209	2902982888	浦西	浦西	内	中	True	False	939	4	False
319	3104833888	浦西	浦西	外	外	True	False	37507	4	False

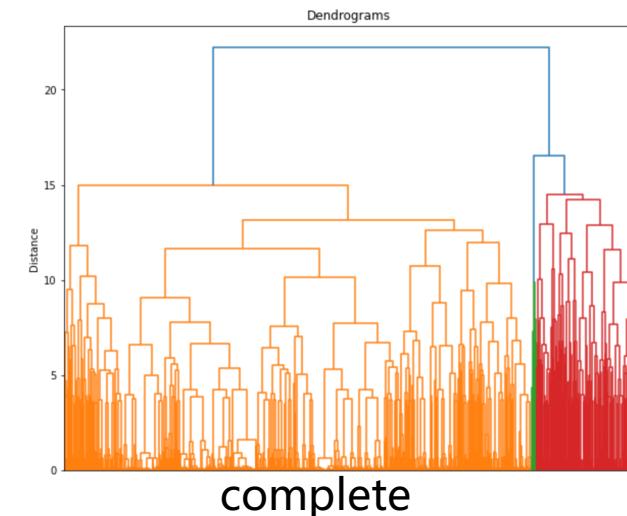
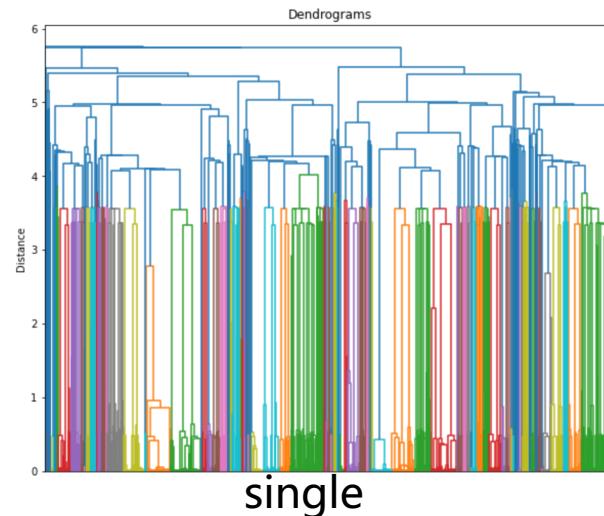
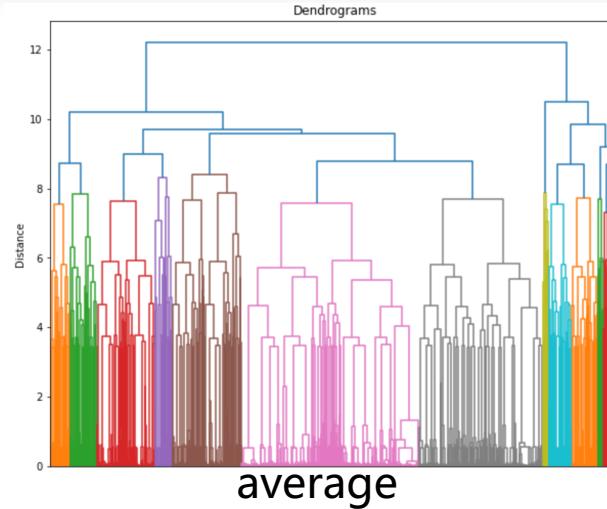
- 使用gower包中的gower_matrix函数
 - 只需输入原始未编码的数据，即可输出相似度矩阵如下：

任务5：层次聚类——trip

层次聚类 —— gower距离

距离计算方法1: gower距离

- 层次聚类共有四种聚类策略
- ward : 使得合并的类方差最小化
- average : 组间距离等于两组对象之间的平均距离 (计算量比较大)
- complete : 组间距离等于两组对象之间的最大距离 (两个不相似的组合数据点可能由于其中的极端值距离较远而无法组合在一起)
- single : 使用两组所有观测值之间的最小距离 (易受到极端值的影响。两个不相似的组合数据点可能由于其中的极端数据点而组合在一起)
- 由于是将计算好的相似度矩阵输入进行聚类，无法采用“ward”方法
- 比较剩下三种方法，使用 ‘average’ 法进行聚类
- 其余聚类策略的结果都很不平均

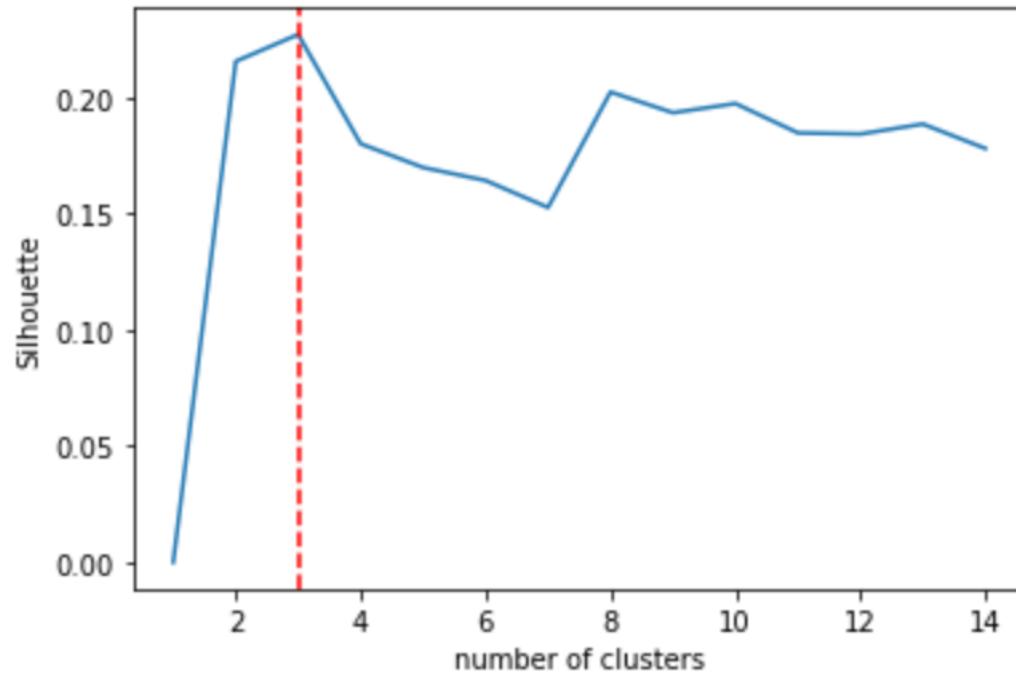


任务5：层次聚类——trip

层次聚类 —— gower距离

- 通过轮廓系数，寻找clusters的数量
- 由于层次聚类不是根据中心点的聚类，且由于没有进行编码，根据gower距离矩阵聚类的结果，无法使用平均值计算其虚拟中心点，故代替SSE，采用轮廓系数来帮助判断聚类的数量
- 轮廓系数取值[-1,1]，越大代表不同的类别之间的dissimilarity越好
- 根据图可以看到，分三类的时候轮廓系数最大，为0.22左右

n_cluster=3, Silhouette score= 0.22691934



任务5：层次聚类——trip

层次聚类 —— 欧式距离

- Gower距离的效果不是特别理想
- 将数据编码后，采用欧式距离以及ward策略进行聚类

1、对于区间标度变量进行标准化

- 区间标度变量：
- 乘坐时间 (s), 价格
- 标准化方法：
- 使用mean absolute deviation
- 计算z-score

2、对于二元变量和名义变量进行编码

- 二元变量：
进站位置_东西, 出站位置_东西, 早晚高峰, 换乘, 优惠类别
- 名义变量：
进站位置_环线, 出站位置_环线

进/出站位置_东西 {'浦东': 0, '浦西': 1}

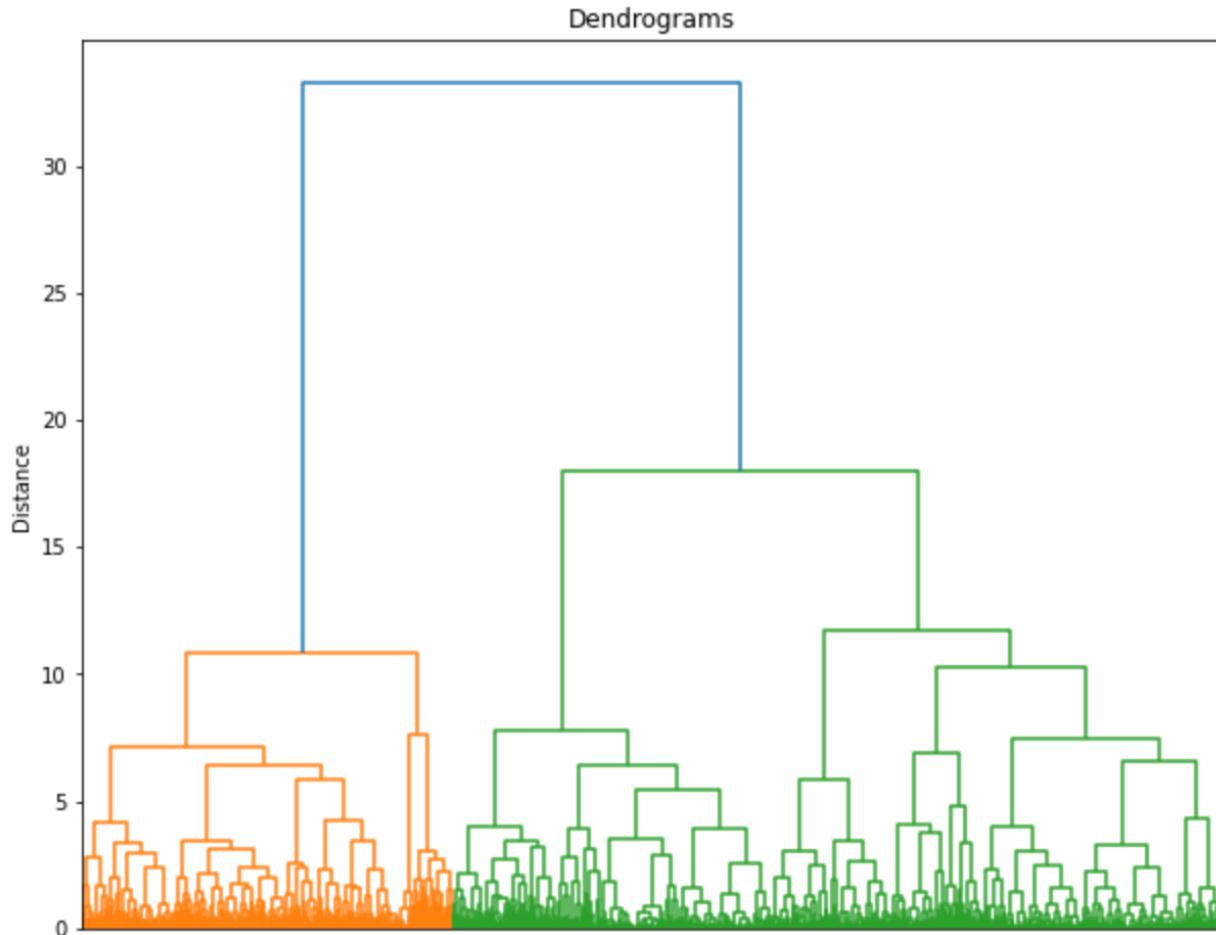
进/出站位置_环线 {'中': 0, '内': 1, '外': 2, '外外': 3}

true/false {False: 0, True: 1}

任务5：层次聚类——trip

数据处理

- Gower距离的效果不是特别理想
- 将数据编码后，采用欧式距离以及ward策略进行聚类
- Ward：使得合并的类方差最小化
- 在比较集中策略后，发现ward策略的各项指标都较优，同时分类也较为均匀
- 下面通过计算SSE和轮廓系数，来帮助寻找聚类的数量



任务5：层次聚类——trip

层次聚类 —— 欧式距离

- 定义计算SSE的函数

```
def sse(data):
    sse=0
    cen=data.groupby('Cluster_Labels').mean()
    # x=pd.DataFrame(gower.gower_matrix(dscaled,center))
    x = pd.DataFrame(distance_matrix(dscaled,cen))
    x['Cluster_Labels']=data['Cluster_Labels']
    for i in range(x.shape[0]):
        for j in range(4):
            if x['Cluster_Labels'][i]==j:
                sse=sse+(x.iloc[i,j])*(x.iloc[i,j])
    return(sse)
```

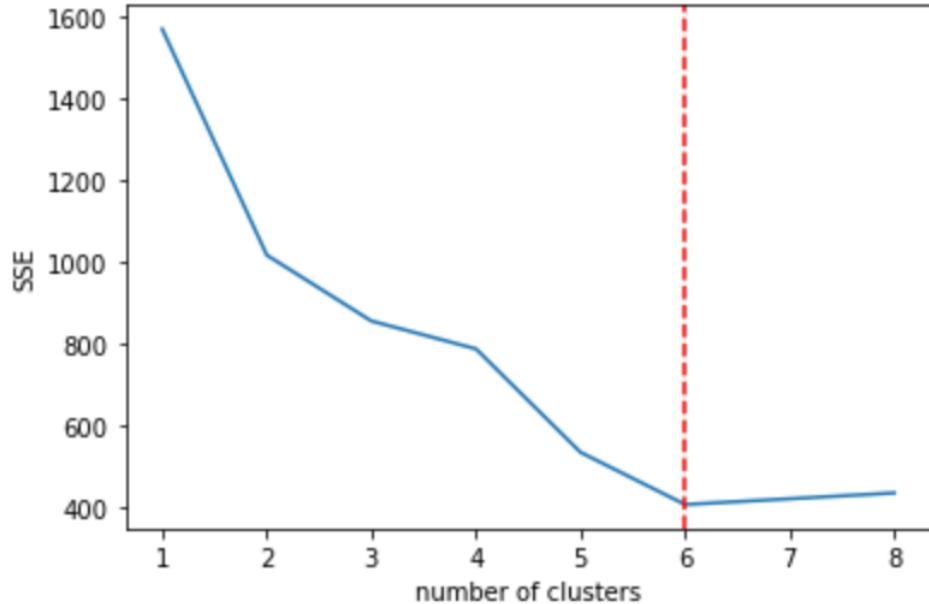
- 由于层次聚类没有指定中心点
- 所以选择每一类取均值，作为其虚拟中心点
- 再计算每个点到其中心点的平方和，再相加
- 最终得到SSE

任务5：层次聚类——trip

层次聚类

- 通过SSE曲线，寻找clusters的数量

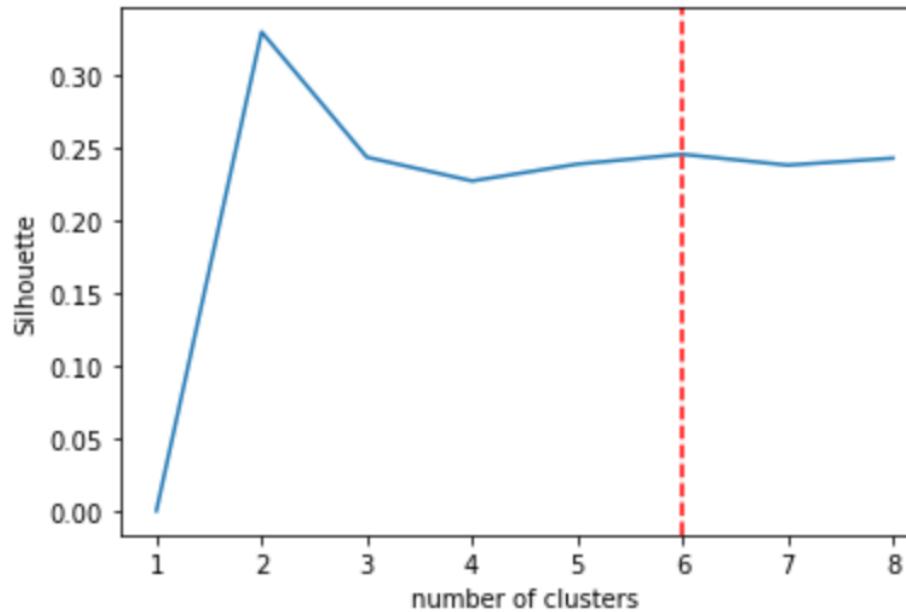
n_cluster=6, SSE= 405.39130261317274



- SSE曲线在n=6是出现拐点，证明此时不同类别间的similarity较高

- 通过轮廓系数，寻找clusters的数量

n_cluster=5, Silhouette score= 0.24590336815093097



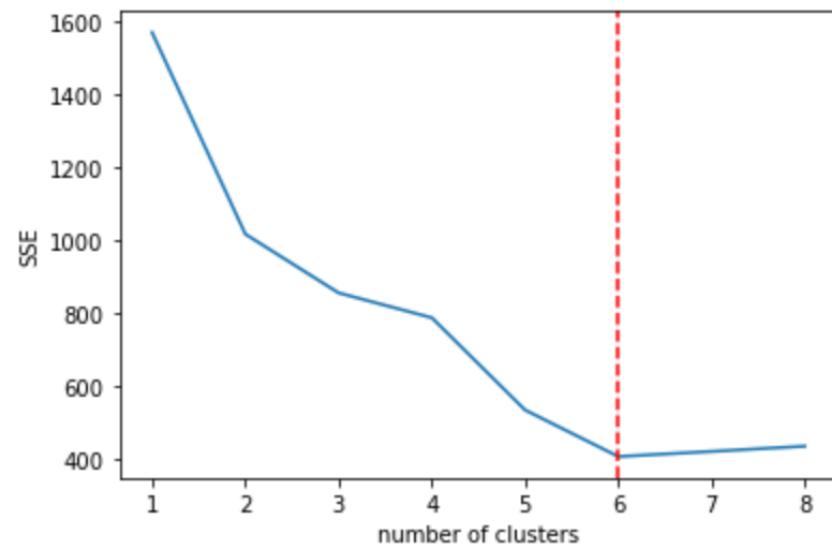
- 轮廓系数曲线虽然在n=2时取到最大值，但是综合SSE曲线，选择n=6作为最终聚类的数量

任务5：层次聚类——trip

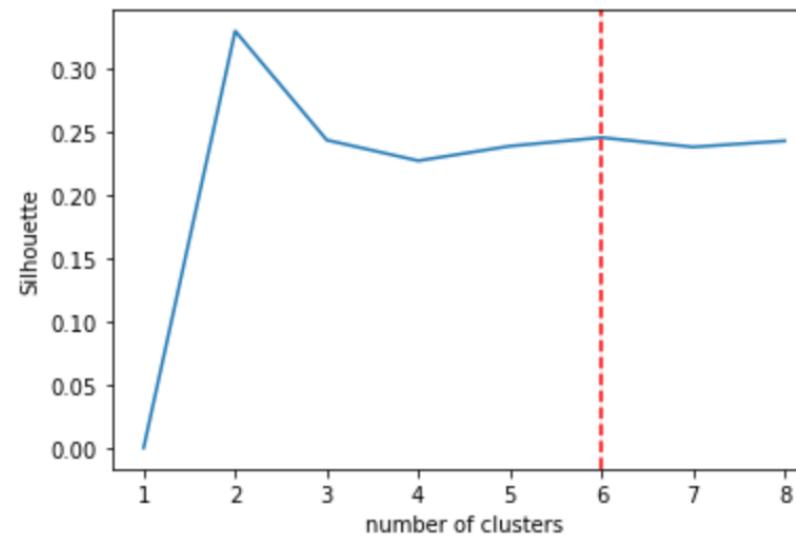
层次聚类 —— 欧式距离

- 最终选择n=6作为最终聚类的数量
- 此时的SSE为405左右，轮廓系数为0.24左右，比前面的gower聚类的表现稍好，所以选择欧式距离，n=6，作为对于trip的层次聚类最终的结果

n_cluster=6, SSE= 405.39130261317274



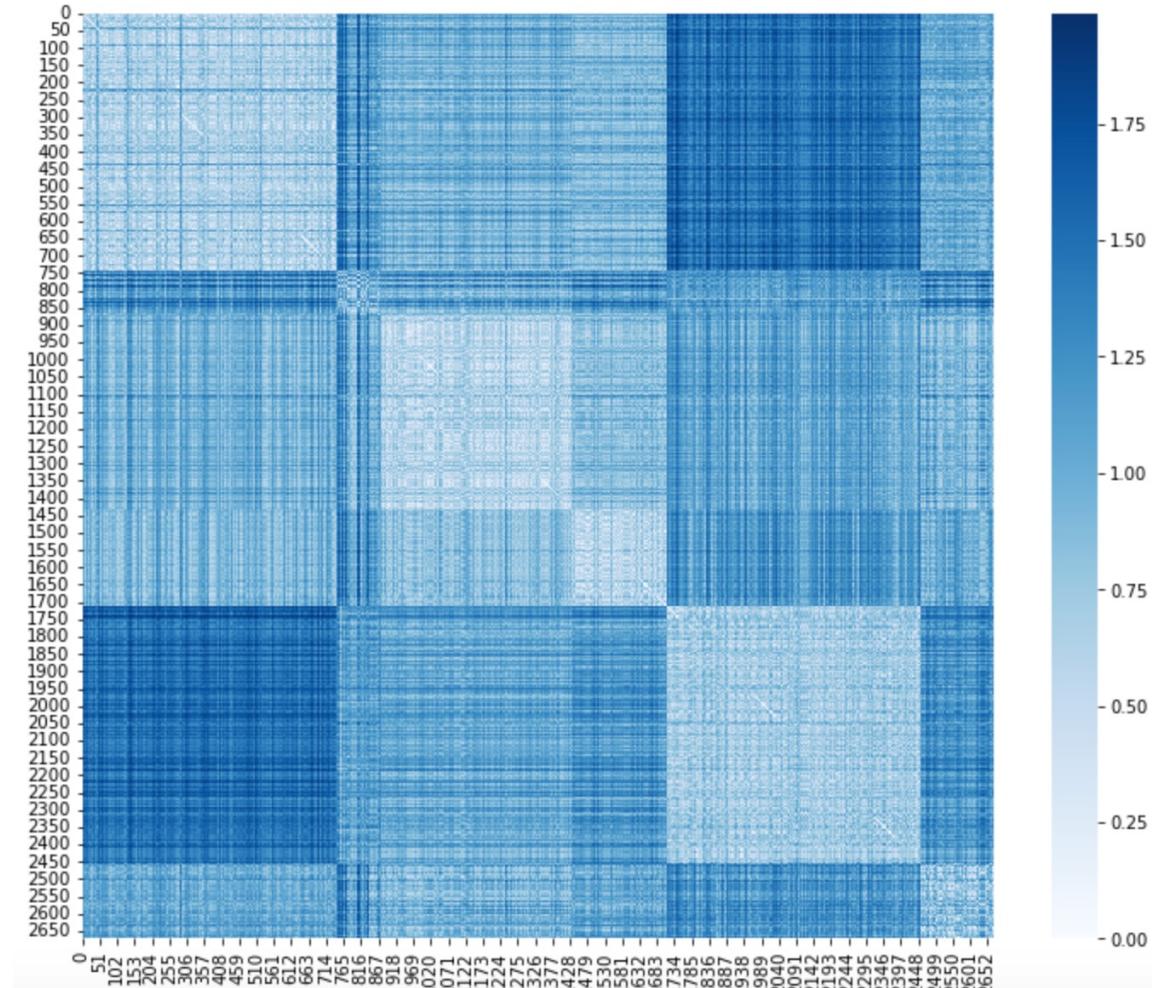
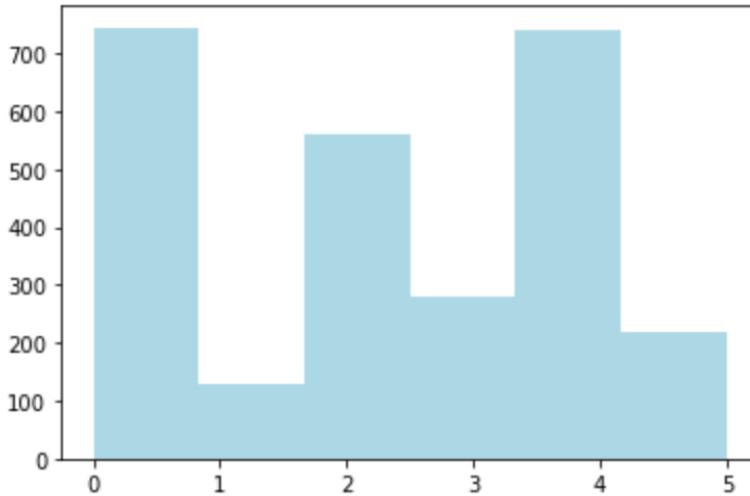
n_cluster=5, Silhouette score= 0.24590336815093097



任务5：层次聚类——trip

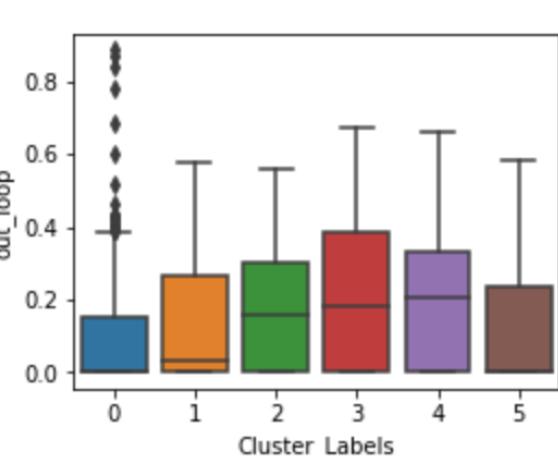
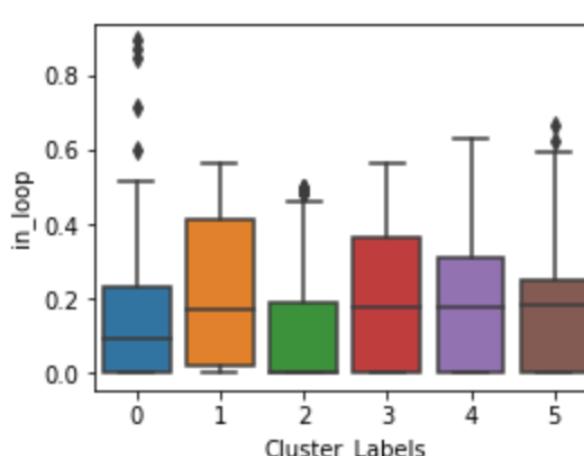
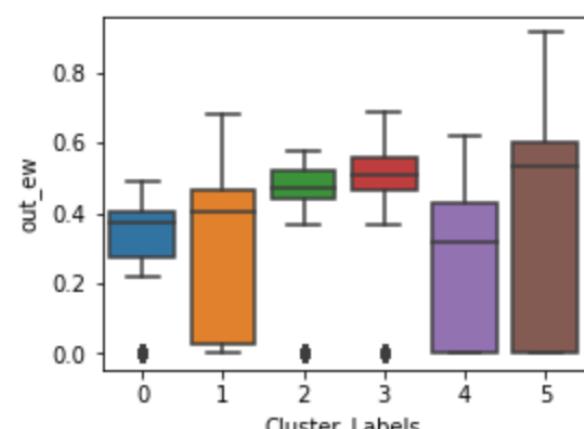
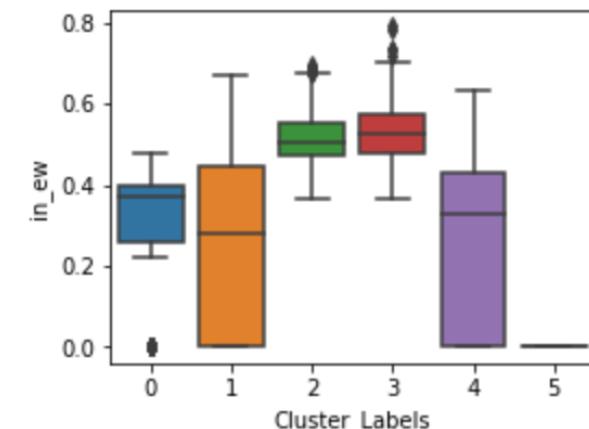
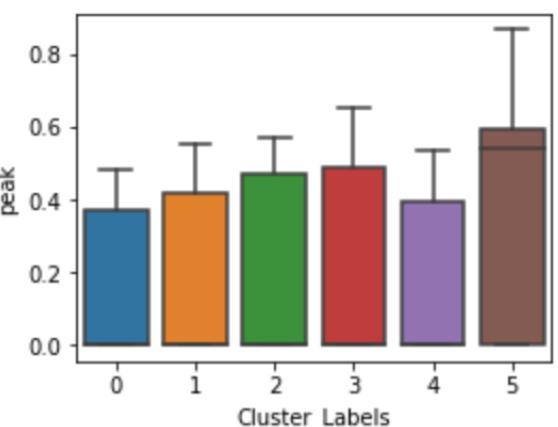
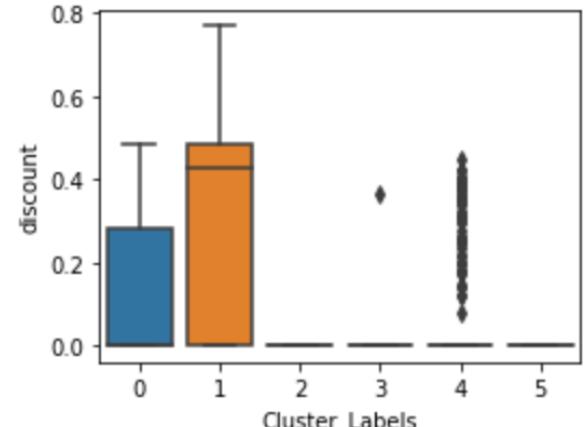
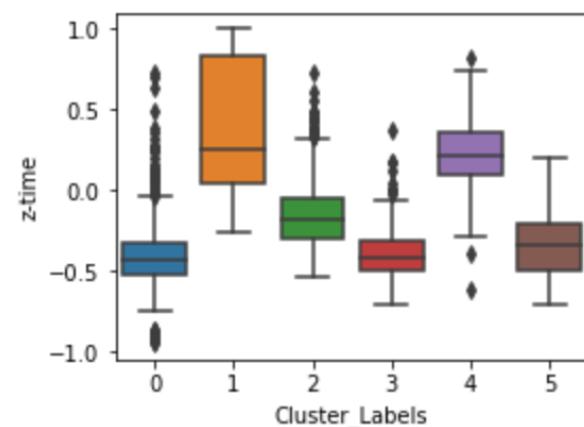
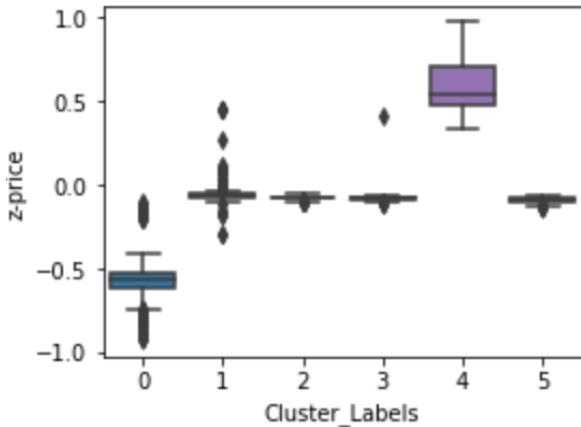
层次聚类 —— 欧式距离

- 相似性矩阵及可视化
- 从矩阵和直方图中可以看到，类别之间的数量不是特别的平均，编号为2，3，5的类别数量明显少于其他三类
- 编号为1的类别内similarity最大，编号为4的最小



任务5：层次聚类——trip

层次聚类 —— 欧式距离



任务5：层次聚类——trip

层次聚类 —— 聚类结果分析

类别	名称	特点
0	短途折扣	price低，time短，discount较多，不挤晚高峰
1	长途折扣	time长，discount最多，浦东出发
2	浦西逆行	无discount，浦西出发和到达，从市中心往外走
3	浦西短途	Time短，浦西出发，浦西到达
4	浦东长途	price高，time长，浦东出发，浦东达到
5	跨江去市中心的上班族	浦西出发，浦东到达，早高峰，无discount



任务5

聚类分析

层次聚类——passenger

👤 The part five

任务5：层次聚类——trip

层次聚类 —— gower距离

距离计算方法1: gower距离

- 同样的，由于我们的数据中二元变量和名义变量比较多，区间标度变量比较少，所以按照老师上课所讲的方法：**使用gower距离作为聚类的标准**
 - gower distance能在观测拥有不同的变量类型如双变量,无序变量,有序变量,数值变量等多种类型变量时,仍然有效

用户ID	上班族	换乘	乘坐时间 (s)	价格	跨江	乘坐次数
125	552888	False	False	3149	8	False
404	989888	True	False	37087	5	False
597	1337888	False	True	4050	7	True
659	1422888	True	False	2193	8	False
688	1444888	True	True	5201	7	False

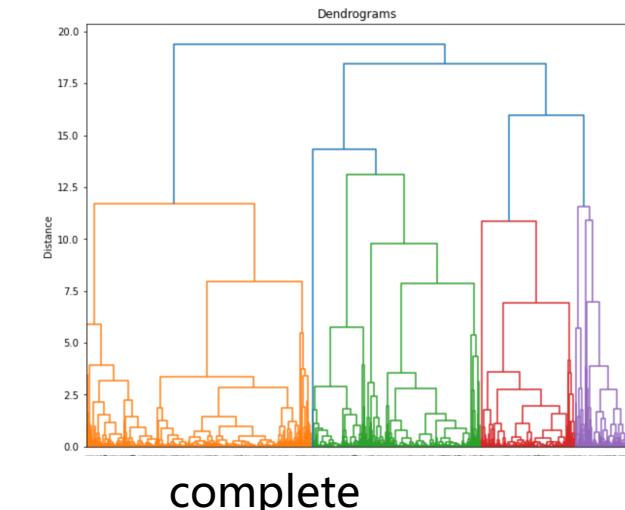
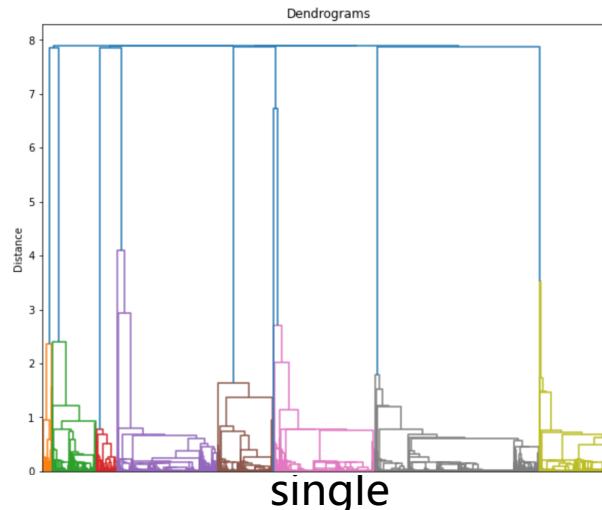
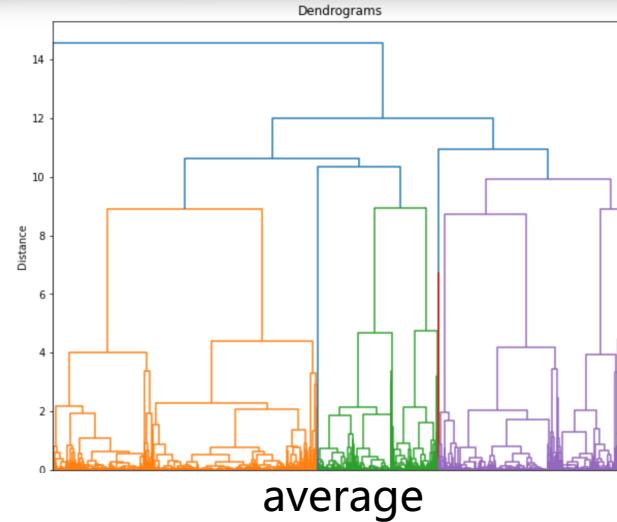
- 使用gower包中的gower_matrix函数
 - 只需输入原始未编码的数据，即可输出相似度矩阵如下：

任务5：层次聚类——trip

层次聚类 —— gower距离

距离计算方法1: gower距离

- 层次聚类共有四种聚类策略
- ward : 使得合并的类方差最小化
- average : 组间距离等于两组对象之间的平均距离 (计算量比较大)
- complete : 组间距离等于两组对象之间的最大距离 (两个不相似的组合数据点可能由于其中的极端值距离较远而无法组合在一起)
- single : 使用两组所有观测值之间的最小距离 (易受到极端值的影响。两个不相似的组合数据点可能由于其中的极端数据点而组合在一起)
- 由于是将计算好的相似度矩阵输入进行聚类，无法采用“ward”方法
- 比较剩下三种方法，使用 ‘complete’ 法进行聚类
- 其余聚类策略的结果都很不平均，会出现一类只有1-2个item的情况

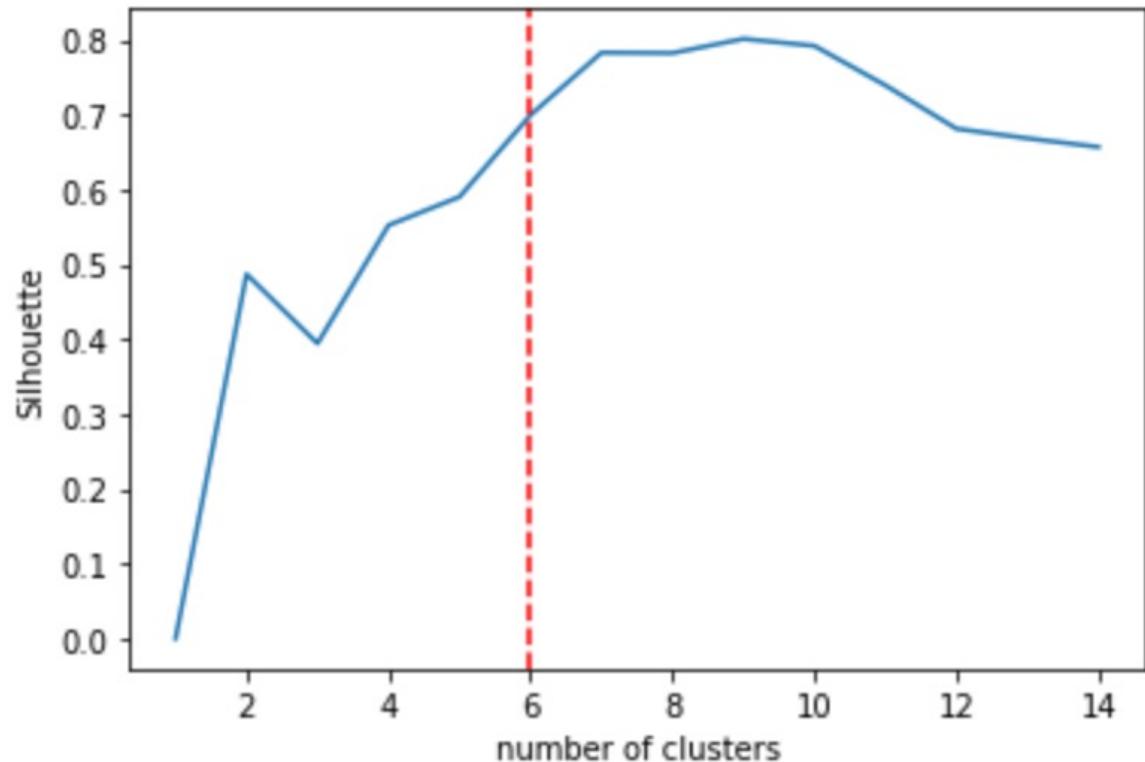


任务5：层次聚类——trip

层次聚类 —— gower距离

- 通过轮廓系数，寻找clusters的数量
- 轮廓系数取值[-1,1]，越大代表不同的类别之间的dissimilarity越好
- 根据图可以看到，分9类的时候轮廓系数最大
- 但是由于变量只有6个，考虑到解释性的关系，选择n = 6作为聚类的数量，此时的轮廓系数为0.78左右，已经不错
- Gower距离在passenger数据集上的表现较好

n_cluster=6, Silhouette score= 0.7832318



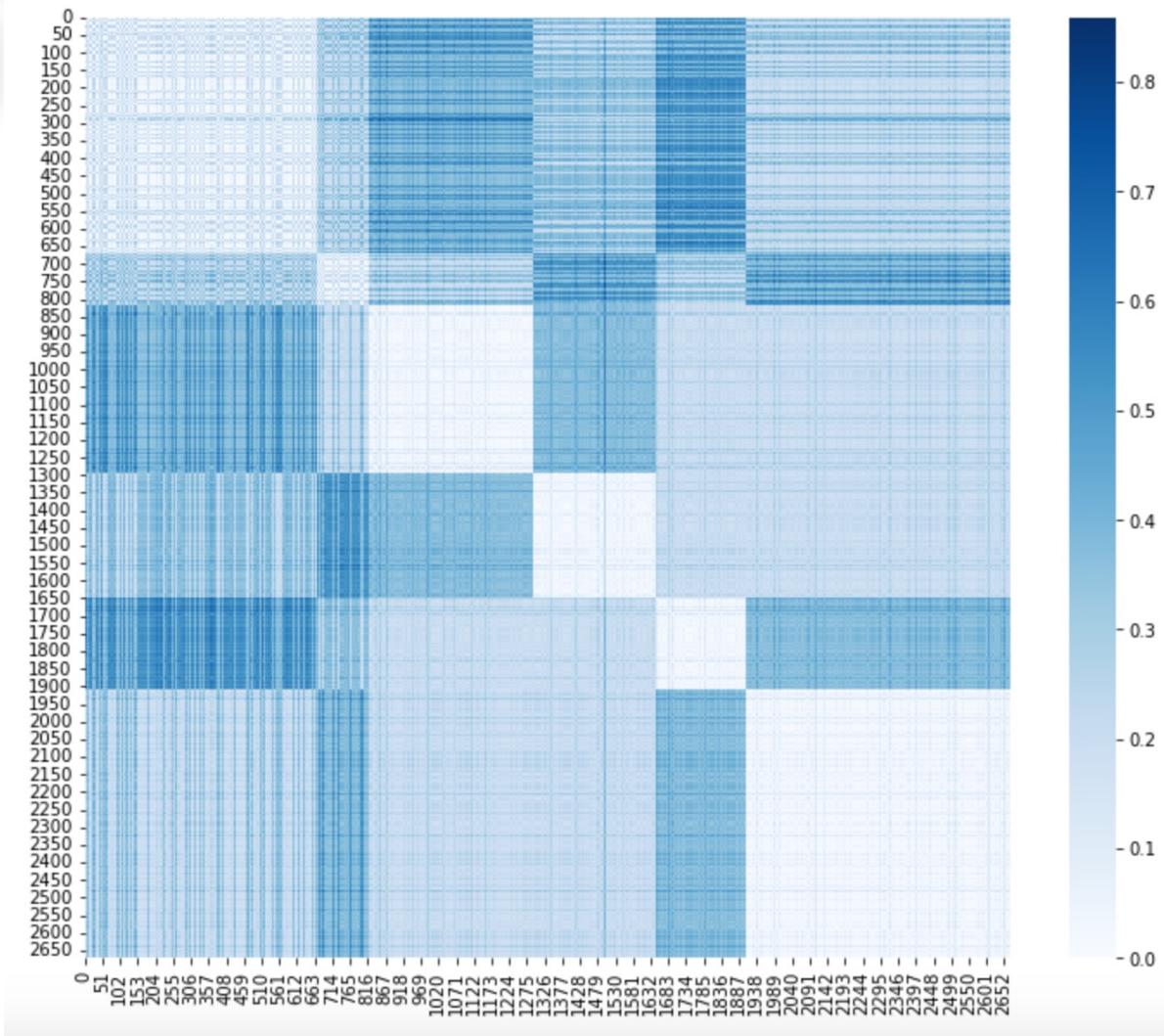
任务5：层次聚类——trip

层次聚类 —— 欧式距离

- 相似性矩阵及可视化

	0	1	2	3	4	5	6
0	0.000000	0.014439	0.039572	0.010393	0.064673	0.030926	0.032668
1	0.014439	0.000000	0.025133	0.022064	0.050234	0.016486	0.018228
2	0.039572	0.025133	0.000000	0.047197	0.031527	0.008646	0.006904
3	0.010393	0.022064	0.047197	0.000000	0.072298	0.038551	0.040292
4	0.064673	0.050234	0.031527	0.072298	0.000000	0.033747	0.033632
...
2667	0.191886	0.177447	0.181019	0.199511	0.206121	0.172373	0.174115
2668	0.194972	0.180533	0.177933	0.202597	0.204945	0.173055	0.171313
2669	0.219059	0.204620	0.201465	0.226684	0.229032	0.197143	0.195401
2670	0.180251	0.167522	0.192655	0.187875	0.217756	0.184009	0.185750
2671	0.192401	0.177962	0.180504	0.200026	0.205605	0.171858	0.173600

2672 rows × 2672 columns

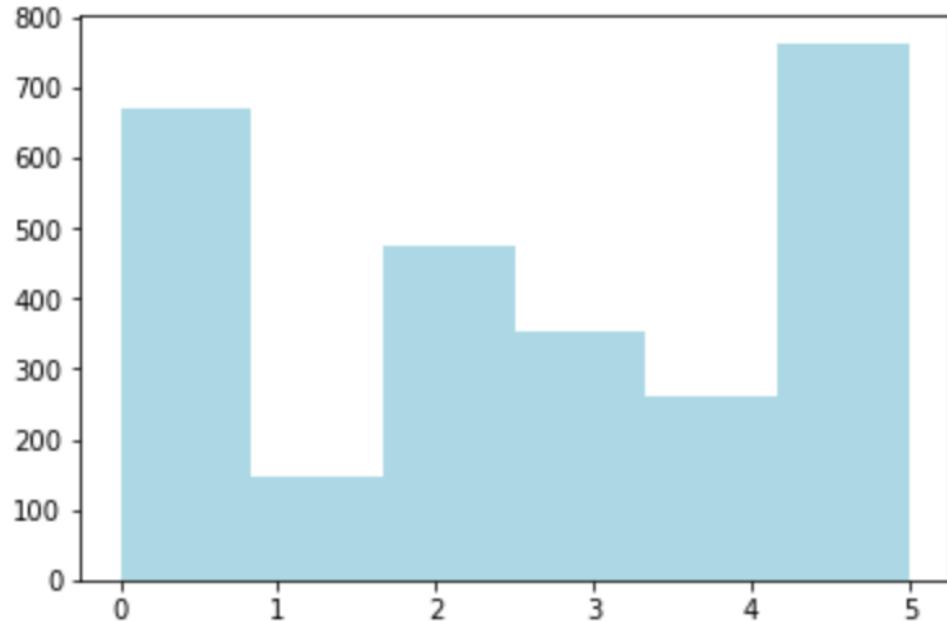


任务5：层次聚类——trip

层次聚类 —— 欧式距离

- 聚类结果中心点

	上班族	换乘	乘坐时间 (s)	价格	跨江	乘坐次数
Cluster_Labels						
0	0.693452	1.0	5390.394345	8.961310	1.0	1.995536
1	0.673469	0.0	3507.619048	7.183673	1.0	1.727891
2	1.000000	0.0	4877.031646	6.569620	0.0	1.812236
3	0.000000	1.0	4260.276056	7.078873	0.0	1.687324
4	0.000000	0.0	3526.765385	5.265385	0.0	1.484615
5	1.000000	1.0	4705.087696	8.367801	0.0	2.020942



- 可以看出第二类人数最少，最后一类人数最多，与 kprototype方法的聚类结果一致

任务5：层次聚类——trip

层次聚类 —— 欧式距离

类别	名称	特点
0	常来的长途跨江者	换乘，时间长，价格高，跨江，次数较多
1	短途跨江直达	不换乘，时间短，跨江
2	不跨江的长途上班族	上班族，不换乘，时间较长，不跨江，价格低
3	偶尔来长途的非上班族	非上班族，换乘，不跨江，乘坐次数较少
4	偶尔来短途的非上班族	非上班族，不换乘，时间短，不跨江，乘坐次数少
5	最常来的长途不跨江上班族	上班族，换乘，时间较长，价格较高，不跨江，次数最多

		类的数量	SSE	轮廓系数
trip	欧式距离	6	405.39	0.2459
	Gower距离	3	\	0.2269
passenger	Gower距离	6	\	0.7832