Q3（DBSCAN）

姓名：王依睿

学号：1552651

a) DBSCAN

1. 引入`sklearn`和`DBSCAN`

```
import sklearn
from sklearn.cluster import DBSCAN
```

2. DBSCAN聚类并计算其silhouette系数

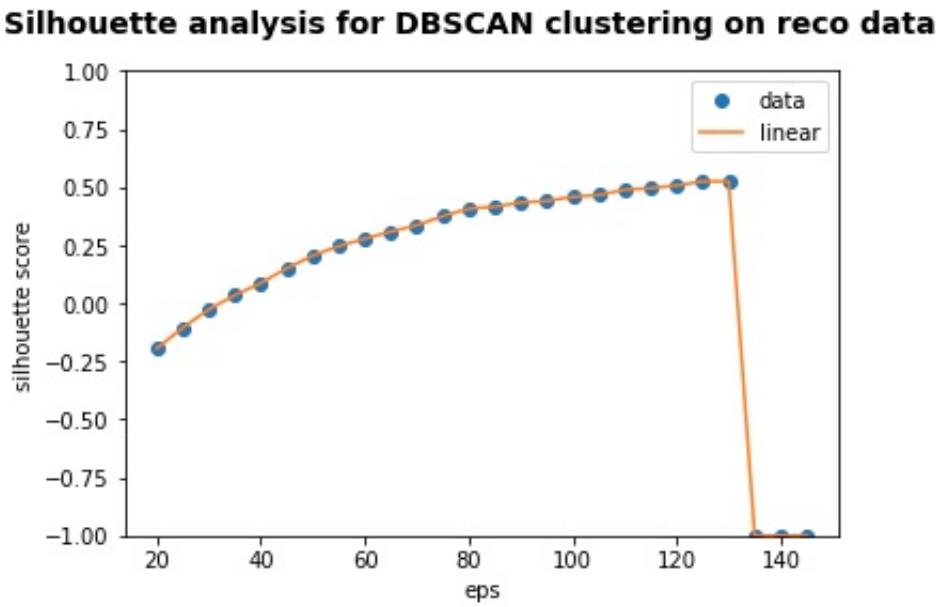
```
# DBSCAN聚类
def dbscan(eps):
    X = StandardScaler().fit_transform(data.T)
    db = DBSCAN(eps = eps).fit(X)
    labels = db.labels_
    if judge(labels):
        silhouette_score = metrics.silhouette_score(X, labels)
        return silhouette_score
    else:
        return -1
```

3. 选择不同的eps进行DBSCAN聚类并计算其silhouette系数

```
# 选择不同的eps聚类并计算其silhouette系数
def vip_dbscan():
    vip_dbscan_df = pd.DataFrame(columns = ['silhouette_score'], index
= np.arange(20, 150, 5))
    for eps in np.arange(20, 150, 5):
        silhouette_score = dbscan(eps)
        vip_dbscan_df['silhouette_score'][eps] = silhouette_score
    return vip_dbscan_df
```

结果

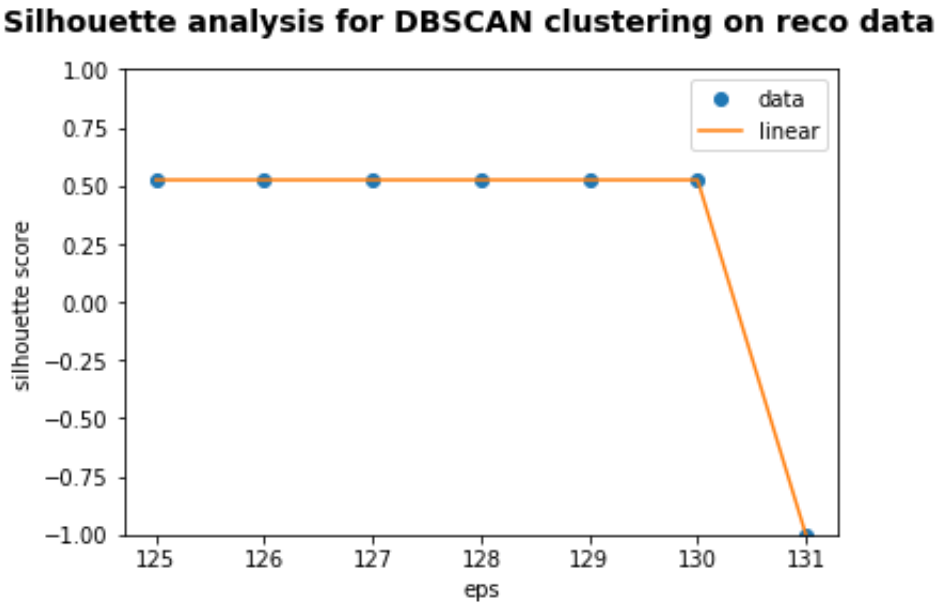
	silhouette_score
20	-0.194231
25	-0.10558
30	-0.0265472
35	0.034153
40	0.0870003
45	0.150627
50	0.204767
55	0.245794
60	0.27737
65	0.308034
70	0.336027
75	0.374431
80	0.406719
85	0.414988
90	0.433585
95	0.440235
100	0.457811
105	0.465831
110	0.488395
115	0.495219
120	0.506584
125	0.524478
130	0.524478
135	-1
140	-1
145	-1



2. 在小范围内查看silhouette score以选出最佳

```
def detail_vip_dbscan():
    vip_dbscan_df = pd.DataFrame(columns = ['silhouette_score'], index
= np.arange(125, 132, 1))
    for eps in np.arange(125, 132, 1):
        silhouette_score = dbscan(eps)
        vip_dbscan_df['silhouette_score'][eps] = silhouette_score
    return vip_dbscan_df
# 在小范围内查看silhouette score以选出最佳的
detail_vip_dbscan_df = detail_vip_dbscan()
```

结果



当eps在125到130之间时，Silhouette系数稳定在0.524478，且当eps大于130时，只能得到1类。

b) 验证lsh的knn结果

1. 选择聚类效果较好的 eps = 130时，查看聚类结果

```
# 选择聚类效果最好的 eps = 130 时，查看聚类结果
X = StandardScaler().fit_transform(data.T)
db = DBSCAN(130).fit(X)
dbscan_labels = db.labels_
```

```
array([ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0])
```

2. 查看knn结果

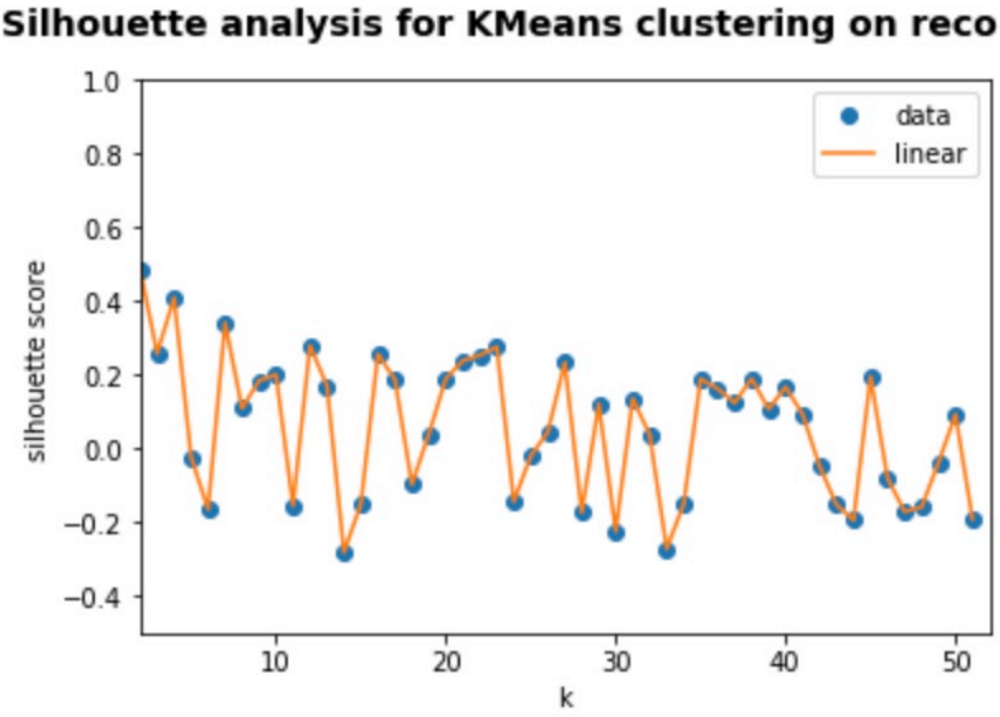
```
hash_size:   3
k:   5
   distance          vipno
0     11056  2900000549289
1     11176  1593140967467
2     11466  1595142205462
3     11766  1590141216914
4     11790  1592140611301
```

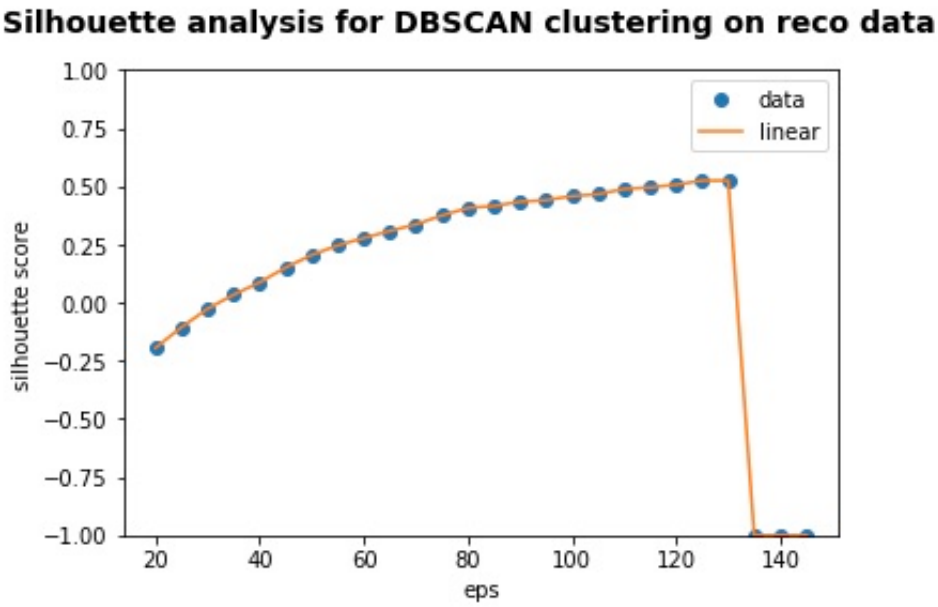
讨论

1. 查看多个聚类结果，发现DBSCAN将数据点分为两类，其中一类是值为-1的异常点
2. 若将Silhouette系数作为评价DBSCAN聚类质量的标准，则DBSCAN聚类倾向于将极大多数点分为一簇（非异常点），其现实意义可能是由于大多数人的购物习惯都较为相似，数据量较少也是其中一个原因。
3. 随着eps的增大，异常点类中的数目减小，Silhouette系数随之增大，knn查询结果与输入vipno所在同一个簇的概率也随之增大。
4. 当eps = 125～130时Silhouette系数取到最大点，聚类结果显示只有一个点被标记为异常点。

c）比较DBScan和kmeans最优聚类

1. 聚类结果





1. 从Silhouette系数来看，当k-means算法k取2时，取到最大值，为0.48722（标准化后）；当DBSCAN算法eps取130时，取到最大值，为0.524478（标准化后）。两者Silhouette系数相差不大，DBSCAN稍高一些。
2. k-means算法和DBSCAN算法都倾向于将数据点分为两类，并将极大多数点分为一类
3. k-means需要人为输入要聚的类数k，而DBSCAN可以发现任意形状的数据集且不用输入类别数K
4. DBSCAN可区分核心对象、边界点和噪音点，因此对噪声的过滤效果好

DBSCAN原理

基于密度的定义，可将所有样本点分为三类：

- 稠密区域内部的点（核心点）：在半径Eps内含有超过MinPts数目的点。
- 稠密区域边缘上的点（边界点）：在半径Eps内点的数量小于MinPts（即不是核心点），但是其邻域内包含至少一个核心点。
- 稀疏区域中的点（噪声或背景点）：任何不是核心点或边界点的点。

用图形表示如下图：