实验报告

1. Clustering with sklearn

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers

1.1 K-Means

(1) 随机选择k个中心;

⑵ 遍历所有样本，把样本划分到距离最近的一个中心;

⑶ 划分之后就有K个簇，计算每个簇的平均值作为新的质心;

⑷ 重复步骤⑵，直到达到停止条件。

停止条件:

聚类中心不再发生变化；所有的距离最小；迭代次数达到设定值。

代价函数：误差平方和（SSE）

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} (C_i - x)^2$$

1.2 Affinity propagation

AP 聚类算法是基于数据点间的"信息传递"的一种聚类算法。与 k-均值算法或 k 中心点算法不同，AP 算法不需要在运行算法之前确定聚类的个数。AP 算法寻找的"examplars"即聚类中心点是数据集合中实际存在的点，作为每类的代表。

⑴ 计算初始的相似度矩阵，将各点之间的吸引度 r(i,k) 和归属度 a(i,k) 初始化为 0；

⑵ 更新各点之间的吸引度，随之更新各点之间的归属度

⑶ 确定当前样本 i 的代表样本(exemplar)点 k，k 就是使{a(i,k)+r(i,k)}取得最大值的那个 k；

（4）重复步骤 2 和步骤 3，直到所有的样本的所属都不再变化为止。

1.3 Mean-shift

Mean-shift（即：均值迁移）的基本思想：在数据集中选定一个点，然后以这个点为圆心，r 为半径，画一个圆(二维下是圆)，求出这个点到所有点的向量的平均值，而圆心与向量均值的和为新的圆心，然后迭代此过程，直到满足一点的条件结束。(Fukunage 在 1975 年提出)。

Mean-shift 算法函数：

a）核心函数：sklearn.cluster.MeanShift(核函数：RBF 核函数)

由上图可知，圆心(或种子)的确定和半径(或带宽)的选择，是影响算法效率的两个主要因素。所以在 sklearn.cluster.MeanShift

中重点说明了这两个参数的设定问题。

1.4 Spectral clustering

Spectral Clustering(SC,即谱聚类)，是一种基于图论的聚类方法,它能够识别任意形状的样本空间且收敛于全局最有解，其基本思想是利用样本数据的相似矩阵进行特征分解后得到的特征向量进行聚类.它与样本特征无关而只与样本个数有关。

基本思路：将样本看作顶点,样本间的相似度看作带权的边,从而将聚类问题转为图分割问题:找到一种图分割的方法使得连接不同组的边的权重尽可能低(这意味着组间相似度要尽可能低),组内的边的权重尽可能高(这意味着组内相似度要尽可能高)

1.5 Ward Hierarchical Clustering

Hierarchical Clustering(层次聚类)：就是按照某种方法进行层次分类，直到满足某种条件为止。

1.6 Agglomerative clustering

Hierarchical Clustering(层次聚类)：就是按照某种方法进行层次分类，直到满足某种条件为止。

算法步骤:

a）将每个对象归为一类，共得到 N 类，每类仅包含一个对象. 类与类之间的距离就是它们所包含的对象之间的距离.

b）找到最接近的两个类并合并成一类，于是总的类数少了一个.

c）重新计算新的类与所有旧类之间的距离.

d）重复第 2 步和第 3 步，直到最后合并成一个类为止(此类包含了 N 个对象).

1.7 DBSCAN

（1）DBSCAN（Density-Based Spatial Clustering of Application with Noise）基于密度的空间聚类算法。

（2）两个参数：

- Eps 邻域半径(epsilon,小量，小的值)
- MinPts(minimum number of points required to form a cluster 定义核心点时的阈值。

（3）DBSCAN 核心思想：从某个选定的核心点出发，不断向密度可达的区域扩张，从而得到一个包含核心点和边界点的最大化区域，区域中任意两点密度相连。

1.8 Gaussian Mixtures

GaussianMixtureModel(混合高斯模型，GMM)。

GMM 的基本思想就是：任意形状的概率分布都可以用多个高斯分布函数去近似，也就是说 GMM 就是有多个单高斯密度分布（Gaussian）组成的，每个 Gaussian 叫一个"Component"，这些"Component"线性加成在一起就组成了 GMM 的概率密度函数，也就是下面的函数。

2. NMI(Normalized Mutual Information)

Normalized Mutual Information(NMI)常用在聚类中，度量 2 个聚类结果的相近程度。

$$I(A,B)=H(A)+H(B)-H(A,B) \qquad (1)$$

若用 $P_A(a)$、$P_b(b)$ 表示 A、B 的概率分布，$P_{AB}(a,b)$ 表示 A 和 B 的联合概率分布[11]，则：

$$H(A)=-\sum_a P_A(a)\log P_A(a) \qquad (2)$$

$$H(B)=-\sum_a P_B(b)\log P_B(b) \qquad (3)$$

$$H(A,B)=-\sum_{a,b} P_{AB}(a,b)\log P_{AB}(a,b) \qquad (4)$$

3. 测试在 tweets 数据集上的聚类效果

3.1 k=89

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/
K-means的准确率: 0.777726787696039
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-pac
  FutureWarning)

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-pa
  FutureWarning)
AffinityPropagation算法的准确率: 0.7839036435001395

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/i
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-
  FutureWarning)
meanshift算法的准确率: 0.7455412796815585

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/im
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-p
  warnings.warn("Graph is not fully connected, spectral embedd
SpectralClustering算法的准确率: 0.8284414737230406
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-p
  FutureWarning)

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/P
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-pack
  FutureWarning)
Ward hierarchical clustering算法的准确率: 0.783967749879778

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/i
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-
  FutureWarning)
AgglomerativeClustering算法的准确率: 0.8949194137166658

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Use
DBSCAN算法的准确率: 0.7073939018290382
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/s
  FutureWarning)

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python
/Users/ima/PycharmProjects/homework3/venv/lib/python3
  FutureWarning)
GaussianMixture算法的准确率: 0.779042358948001

Process finished with exit code 0
```

3.2 k=99

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/PycharmPro
K-means的准确率: 0.7970392343504124
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packages/skle
  FutureWarning)

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/Py
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packa
  FutureWarning)
AffinityPropagation算法的准确率: 0.7839036435001395

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/Pycharm
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packages/s
  FutureWarning)
meanshift算法的准确率: 0.7455412796815585

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packag
  FutureWarning)
SpectralClustering算法的准确率: 0.8268592824808249

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-pa
Ward hierarchical clustering算法的准确率: 0.780789944141587
  FutureWarning)

Process finished with exit code 0
```

```
AgglomerativeClustering算法的准确率: 0.896490584641139
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packages/sk
  FutureWarning)


Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima/Pyc
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packag
  FutureWarning)
DBSCAN算法的准确率: 0.7073939018290382

Process finished with exit code 0
```

```
GaussianMixture算法的准确率: 0.7865534477667807
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-packa
  FutureWarning)

Process finished with exit code 0
```

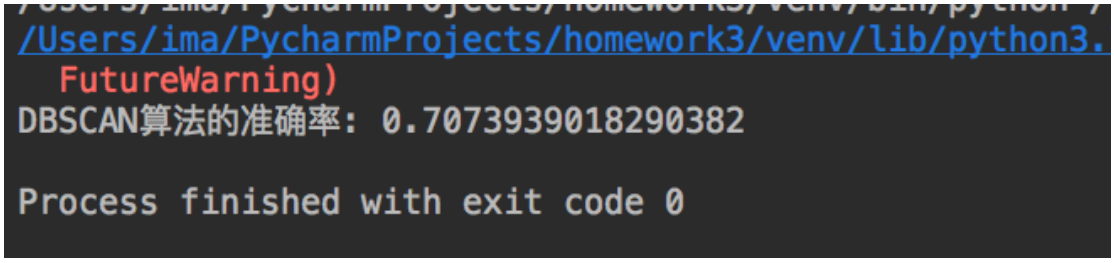
3.3 k=109

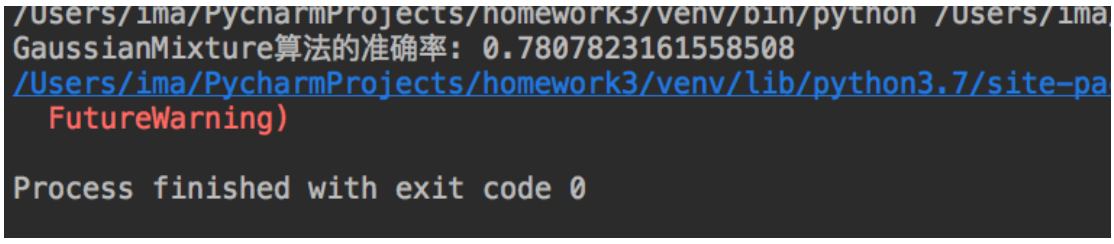
```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/ima
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site-pa
K-means的准确率: 0.7822427318048143
  FutureWarning)

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Use
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/s
  FutureWarning)
AffinityPropagation算法的准确率: 0.7839036435001395

Process finished with exit code 0
```

```
/Users/ima/PycharmProjects/homework3/venv/bin/python /Users/
/Users/ima/PycharmProjects/homework3/venv/lib/python3.7/site
  FutureWarning)
meanshift算法的准确率: 0.7455412796815585

Process finished with exit code 0
```

```
SpectralClustering算法的准确率: 0.8196069783685273

Process finished with exit code 0
```





4. 实验总结

在使用 sklearn 中的聚类算法的相关函数进行聚类时，十分的方便快捷，节省了大量的时间。

在测试了各类算法的聚类效果后发现，基于本人这个测试，相较于 K=89,K=109 来说，K=99 时整体都呈现出较好的效果，Agglomerative clustering 算法呈现出最好的聚类效果，其次是

Spectral clustering 算法效果较好，而 DBSCAN 呈现出了最差的聚类效果。整体这八类算法的聚类效果都还不错。