Pattern Recognition: Homework #8

Professor Wang Guijin

Qingyun Fang

2017311003

Use LaTeX in 5.23 2018

非监督聚类算法

问题描述

对这组数据进行 kmeans 聚类，令 $k = 2, 3, 4$。画出聚类结果及每类的中心点，观察聚类结果。记录使用不同初始点时的聚类结果，收敛迭代次数及误差平方和。

$k = 3$, 用给出如下几组初始点进行聚类:

初始点组 1： [-4.822 4.607;-0.7188 -2.493;4.377 4.864]

初始点组 2： [-3.594 2.857;-0.6595 3.111;3.998 2.519]

初始点组 3： [-0.7188 -2.493;0.8458 -3.59;1.149 3.345]

初始点组 4： [-3.276 1.577;3.275 2.958;4.377 4.864]

$k = 2$ 或 4 时, 自行给出初始点并聚类，观察聚类结果

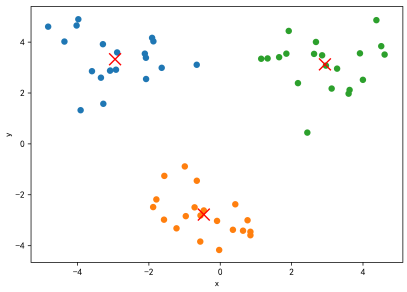
kmeans 算法

kmeans 算法是一种很常用的聚类算法，其基本思想是，通过迭代寻找 k 个聚类的一种划分方法，使用 K 个聚类的均值来代表相应的各类样本时所得到的总体误差最小。误差平方和公式如下：

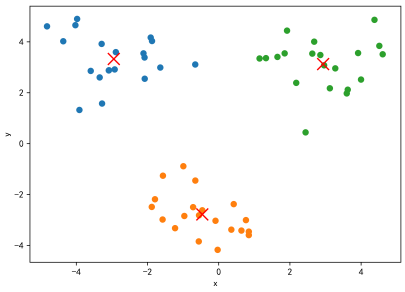
$$J_e = \sum_{i=1}^{c} \sum_{y \in \Gamma_i} \|\boldsymbol{y} - \boldsymbol{m}_i\|^2$$

其中 $\boldsymbol{m}_i = \frac{1}{N_i} \sum_{y \in \Gamma_i} \boldsymbol{y}$ 为第 i 类样本均值。

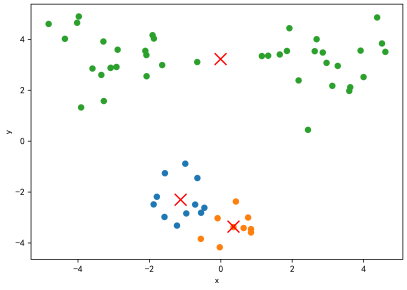
图 1 为当 $k = 3$ 时，初始点分别为 1,2,3,4 时的分类结果：



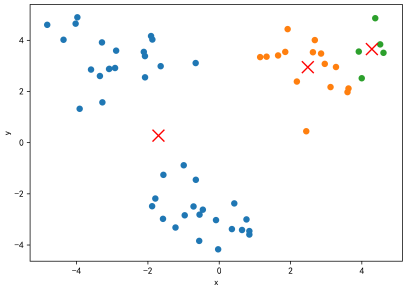
(a) cluster=3,init=1 分类结果



(b) cluster=3,init=2 分类结果



(c) cluster=3,init=3 分类结果



(d) cluster=3,init=4 分类结果

图 1: k=3, 不同初始点分类结果

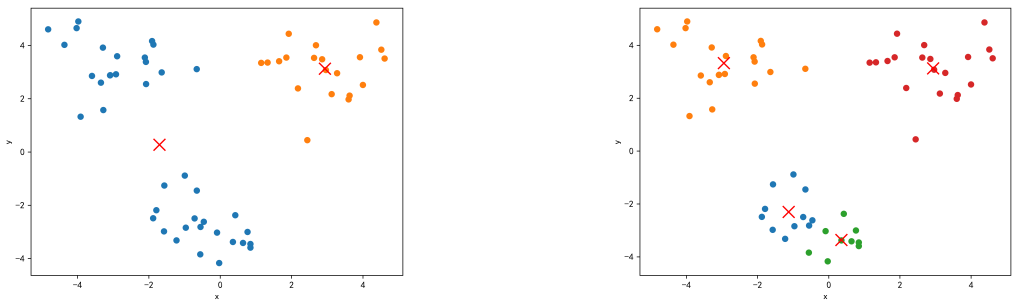
4 个初始的各类均值、迭代步骤和总体误差误差如下表：

初始组别	聚类中心	迭代次数	总体误差
init 1	$\begin{pmatrix} -2.94737575 & 3.3263781 \\ -0.45965615 & -2.7782156 \\ 2.93386365 & 3.12782785 \end{pmatrix}$	3	71.4169042011
init 2	$\begin{pmatrix} -2.94737575 & 3.3263781 \\ -0.45965615 & -2.7782156 \\ 2.93386365 & 3.12782785 \end{pmatrix}$	3	71.4169042011
init 3	$\begin{pmatrix} -1.12616164 & -2.30193564 \\ 0.35496167 & -3.36033556 \\ -0.00675605 & 3.22710297 \end{pmatrix}$	3	137.72017666
init 4	$\begin{pmatrix} -1.70351595 & 0.27408125 \\ 2.48449707 & 2.95091147 \\ 4.2819634 & 3.658577 \end{pmatrix}$	4	156.636629952

表 1: 4 个初值结果表

从上图和上表的结果来看，很容易得到初值的选择对于 kmeans 的最后的分类结果有着至关重要的作用。比如组 1 和组 2 初始的聚类中心本身就接近于最后的分类结果，因此这两者最后的结果相对比较好。组别 3 和组别 4 初始选择的不好，因此最后结果就不是很理想，即 kmeans 不一定会收敛到最优误差的解。

当 k=2,k=4 时，分别取初值为 $(0,0)(1,1)$ 和 $(-2,-2)(-2,2)(2,-2)(2,2)$ 其结果如下图 2：



(a) cluster=2 分类结果　　(b) cluster=4 分类结果

图 2: k=2,k=4 不同初始点分类结果

它们的均值、迭代步骤和总体误差误差如下表 2：

显然，亦可以观察到随着聚类个数的增加总体的误差是一直减少的，极限的情况就是每个点都是一个聚类，此时的误差和就是 0。

初始组别	聚类中心	迭代次数	总体误差
$k=2$	$\begin{pmatrix} -1.70351595 & 0.27408125 \\ 2.93386365 & 3.12782785 \end{pmatrix}$	5	161.625977357
$k=4$	$\begin{pmatrix} -1.12616164 & -2.30193564 \\ -2.94737575 & 3.3263781 \\ 0.35496167 & -3.36033556 \\ 2.93386365 & 3.12782785 \end{pmatrix}$	4	64.3168636185

表 2: $k=2,4$ 分类结果表

思考与改进

既然 Kmeans 算法的迭代步骤很少，说明其算法十分高效，另外算法对初值的依赖很大，初值选的好，结果就比较好。因此可以对算法进行适当的改进，一种朴素的思想就是随机选取几组初始点值，选取中间总的误差和最小的那组，作为分类的结果。实际上 Python sklearn 中的 kmeans 也采用了这种思想，调用库函数，其分类结果如下：

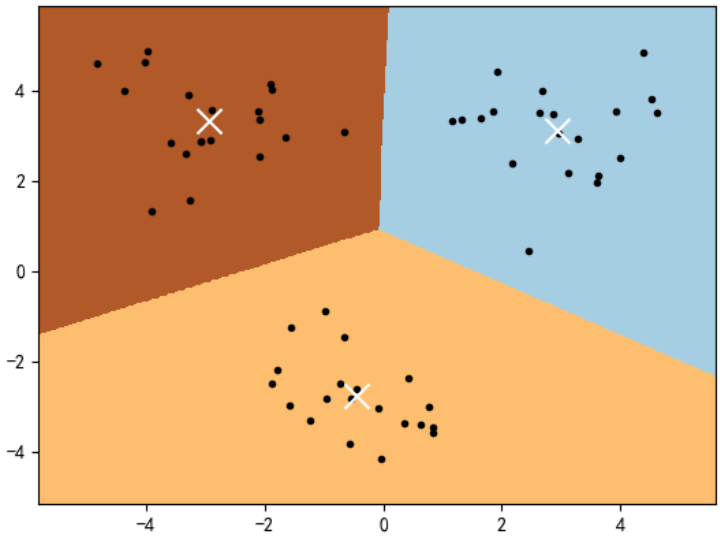


图 3: python sklearning kmeans 分类结果

显然采用这种随机初始值得方法能得到一种较好的结果。

Spectral Clustering 谱聚类

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，比起传统的 K-Means 算法，谱聚类对数据分布的适应性更强，聚类效果也很优

秀，同时聚类的计算量也小很多，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).

谱聚类过程主要有两步，第一步是构图，将采样点数据构造成一张网图，表示为 G(V,E)，V 表示图中的点，E 表示点与点之间的边，如下图：

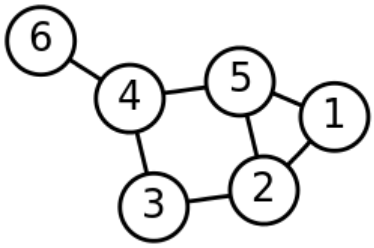


图 4: 无向图

第二步是切图，即将第一步构造出来的按照一定的切边准则，切分成不同的图，而不同的子图，即我们对应的聚类结果，举例如下：

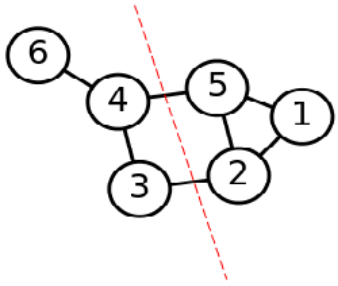
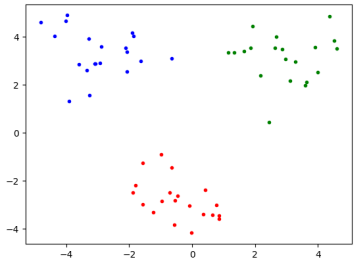
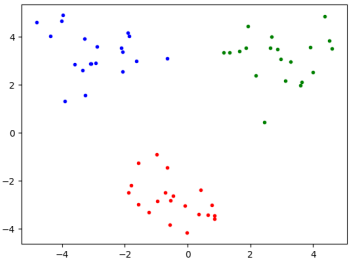


图 5: 切图

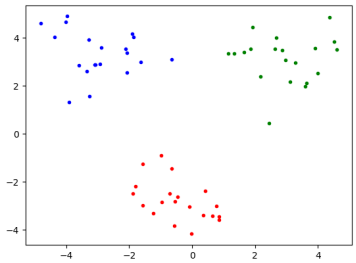
同样使用 $k = 3$ 的 4 个初值，用 Spectral Clustering 分类结果如下图 6。相比于 kmean 对初值的敏感，Spectral Clustering 的分类效果可以说是比较理想了，在 4 个初值不同情况下，都能收敛到一个很好的结果。



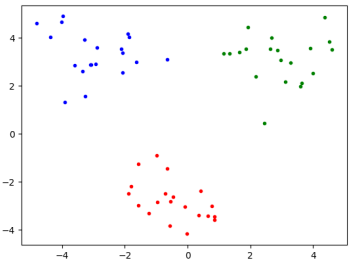
(a) cluster=3,init=1 分类结果



(b) cluster=3,init=2 分类结果



(c) cluster=3,init=3 分类结果



(d) cluster=3,init=4 分类结果

图 6: k=3,Spectral Clustering 分类结果