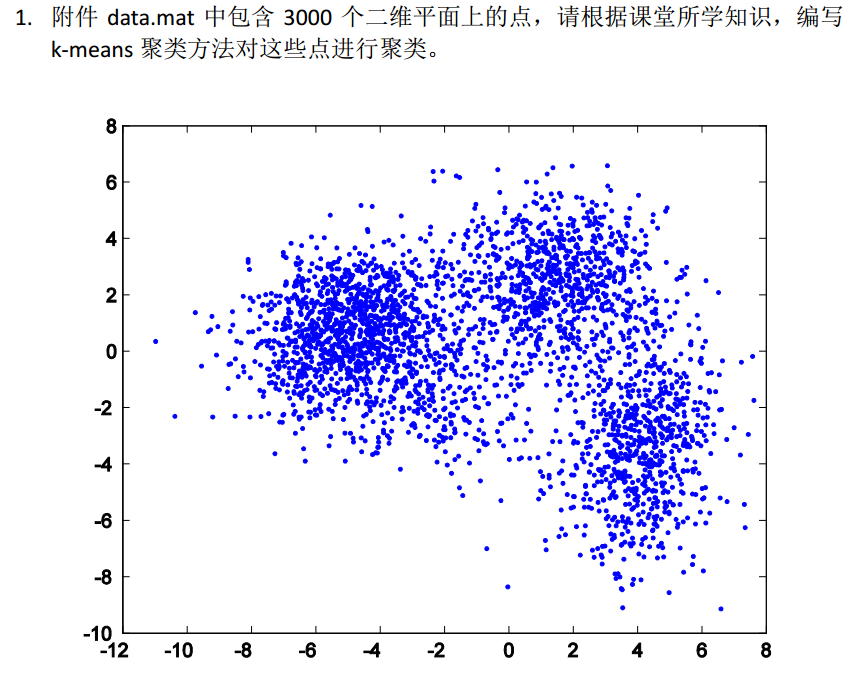
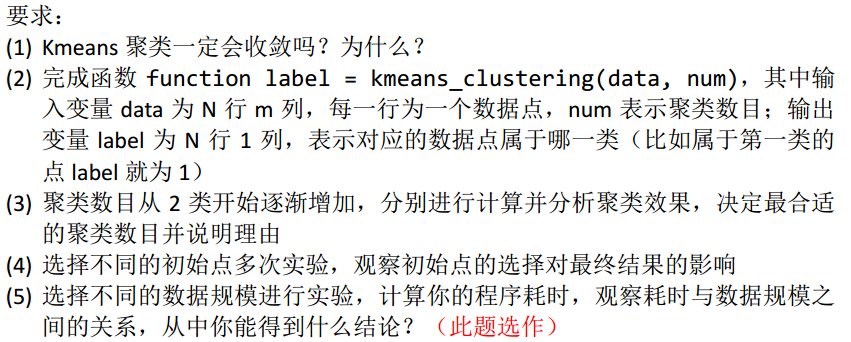
《系统工程导论》第七章作业

2013011570 自31 唐静娴





**1. 收敛性分析**

假设提取到的原始数据集合为，并且每一个都是N维向量，要将这些数据点划分为K类，那么Kmeans要做的就是最小化

其中。我们采取迭代的办法，先固定，选择最优的；而后固定，求取最优的。将J对求导，并令导数等于零，很容易得出为了得到最小的J，需要满足

亦即的值应当是所有第k类中的数据点的平均值。由于每一次迭代都是取到 J 的最小值，因此 J 只会不断地减小（或者不变），而不会增加。换言之，这是一个严格的坐标下降过程，保证了 Kmeans 最终会收敛到一个极小值。但因为J是一个非凸函数，因此不能保证找到全局最优解，只能确保局部最优解。

**2. 求解步骤及方法说明**

**2.1 初始中心点的选择**

由于Kmeans算法只能求解局部最优解，因此初始点的选择会影响分类结果和计算速度。本次作业中列举了三种初始中心点选取的方法，分别是：

A. 选取样本数据中前K个点；

B. 随机选取K个点；

C. 选取彼此距离尽可能远的K个点。即首先随机选择一个点作为第一个初始中心点， 然后选择距离该点最远的那个点作为第二个初始中心点，然后再选择距离前两个点的最近距离最大的点作为第三个初始中心点，以此类推，直至选出K个初始中心点。

**2.2 迭代求取最终中心点**

1. 分别计算剩下的元素到K个中心点的欧氏距离，将这些元素分别划归到距离最小的 中心点所在的类。

2. 根据聚类结果，重新计算K个类各自的中心点，计算方法是取簇中所有元素各自维 度的算术平均数。

3. 若新的K个中心点较之前的K个中心点不再变化（即变化范围在一个可容忍的误差 范围内，本次作业设置误差阈值为0.001），则输出当前结果，否则转至4 。

4.用新的中心点代替之前的中心点，转至1。

**2.3 聚类效果评估**

本次作业采用“轮廓系数”来描述聚类效果。

轮廓系数（Silhouette Coefficient）结合了聚类的凝聚度（Cohesion）和分离度（Separation），用于评估聚类的效果。该值处于-1~1之间，值越大，表示聚类效果越好。具体计算方法如下：

1. 对于第i个元素，计算与其同一个类里的所有其他元素距离的平均值，记作，用于量化该类内部的凝聚度；

2. 选取外的一个类b，计算与b中所有点的平均距离，遍历所有其他类，找到最近的这个平均距离,记作，用于量化各类之间的分离度；

3. 对于元素，轮廓系数；

4. 计算所有的轮廓系数，求出平均值sil即为当前聚类的整体轮廓系数。

从上面的公式不难发现，若，说明与本类其他元素的平均距离小于最近的其他类，聚类效果不好。如果趋于0，或者足够大，那么更接近1，说明聚类效果比较好。

**3. 运行结果示例**

采用第3种选取初始点的方法，聚类数目K=2,3,4,5,6时，得到的聚类结果列举如下：

|  |  |
| --- | --- |
| k=2 |  |
| k=3 |  |
| k=4 |  |
| k=5 |  |
| k=6 |  |

**4. 结果分析**

**4.1 聚类数目对聚类效果的影响（）**

采用随机选取初始中心点的方法。为了避免偶然出现的收敛到某个局部最优解的情况，在评估当前聚类数目K对应的聚类效果时，进行5次重复Kmeans聚类计算和效果评估，将得到的5个轮廓系数取均值，作为当前K的最终轮廓系数。表格列举如下：

|  |  |
| --- | --- |
| 聚类数目K | 轮廓系数 |
| 2 | 0.492366 |
| 3 | 0.529661 |
| 4 | 0.448249 |
| 5 | 0.400856 |
| 6 | 0.385029 |
| 7 | 0.351692 |
| 8 | 0.338520 |
| 9 | 0.338567 |

为了直观起见，将二者关系描述为柱状图，

可以看到，轮廓系数在K=3处有最大值，而当K>7后，轮廓系数的大小便几乎不再改变。因此可以认为最合适的聚类数目是3类。

**4.2 初始点选取对最终结果的影响**

**4.2.1三种方法比较（K = 4）**

为了避免偶然性，仍然采取同一种条件下计算5次，并将平均值作为最终评价的做法。为了验证选取方法的稳定性，将5次计算中耗时的最大值与最小值之差也列举在表格中。

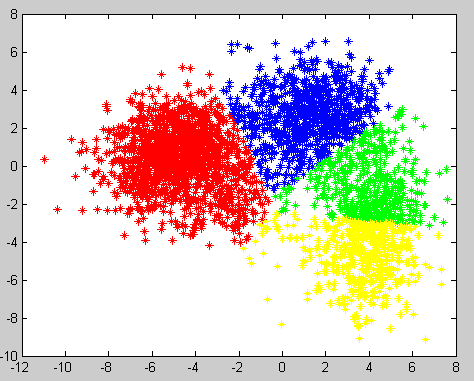
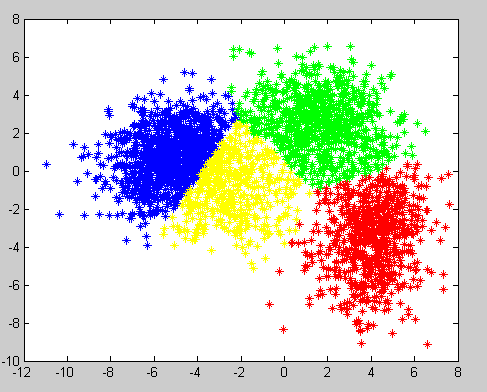
|  |  |  |  |
| --- | --- | --- | --- |
| 选取方法 | 迭代次数 | 计算耗时（s） | 最大耗时-最小耗时 |
| ① | 30 | 0.7113 | 0.01 |
| ② | 24 | 0.5663 | 0.313 |
| ③ | 23 | 0.5468 | 0.190 |

① 选取样本数据中前K个点;② 随机选取K个点;③ 选彼此距离尽可能远的K个点

可以看到，方法二及方法三的迭代次数和计算耗时均优于方法一，但方法三的稳定性更好，随机选取初始中心点的方法有可能会选到坏的初始点，导致算法收敛速度慢。

**4.2.2 不同初始点对聚类结果的影响**

当采用不同的初始中心点时，聚类结果会有不同，因为在某些初始中心点的情况下，Kmeans算法并没有收敛到全局最小。当K=4时，产生的两种不同的聚类结果：

前者的轮廓系数为0.4615，而后者的轮廓系数为0.4412，聚类效果不如前者，Kmeans算法收敛到了局部最优。

**4.3 数据规模对聚类耗时的影响**

在K=3，顺序选取初始中心点的条件下，评估不同数据规模下的聚类计算时间。为了防止偶然因素，这里取10次计算耗时的平均值。

|  |  |
| --- | --- |
| 数据规模 | 计算时间/s |
| 100 | 0.008 |
| 500 | 0.037 |
| 1000 | 0.058 |
| 2000 | 0.122 |
| 3000 | 0.201 |

为了直观起见，做出二者之间的关系曲线：

不难看出，计算时间与数据规模近似于线性关系，且二者正相关。