哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

2020 年春季学期

计算学部《机器学习》课程

K-means 聚类方法和混合高斯模型

姓名	李卓君
学号	1180300210
班号	1803104
电子邮件	zhuojunlee724@gmail.com
手机号码	18845636557

目录

1 实验目的与实验要求

1.1 实验目的

实现一个 k-means 算法和混合高斯模型，并且用 EM 算法估计模型中的参数。

1.2 实验要求

用高斯分布产生 k 个高斯分布的数据（不同均值和方差）（其中参数自己设定）。

（1）用 k-means 聚类，测试效果；

（2）用混合高斯模型和你实现的 EM 算法估计参数，看看每次迭代后似然值变化情况，考察 EM 算法是否可以获得正确的结果（与你设定的结果比较）。

1.3 实验环境

Windows 10, Visual Studio Code, Python 3.8.5

2 实验背景与原理

2.1 实验背景

在之前的实验中我们进行的都是监督学习，即给定特征 x 和预测值 y，拟合一个模型来反映二者之间的映射关系。而本次实验我们进行的为非监督学习，即只给定特征 x，拟合模

型找到数据集 x 的内在结构，直白地说，就是给定数据集 X 和聚类数目 K，通过 K-means 和 GMM-EM 算法来将数据集分为 K 类，其中每个类内部聚合程度最高，类与类之间的聚合程度最低。

2.2 实验原理

为了描述方便，我们从较简单的二维点集的 2-Means 问题出发。假定有二维点集用矩阵 X 表示，利用算法将该点集中的所有点分为两类。接下来我们着重介绍 K-means 算法和 GMM-EM 算法。

2.2.1 K-means 算法

K-means 算法的步骤如下：

1. 首先随机地在数据集 $X = x^{(1)},x^{(2)},...,x^{(m)}$ 中选择两个点，称为聚类中心。

2. 分配类别：根据聚类中心将所有的样本分为两个组，分类标准为选择距离样本点最近的聚类中心所在组为我们为该样本点分配的组，距离以欧式距离计，即

$$\left\|x^{(i)} - \mu_k\right\| = \left\|\sqrt{\left(x_1^i - \mu_{1(k)}\right)^2 + \left(x_2^i - \mu_{2(k)}\right)^2 + ... + \left(x_n^i - \mu_{n(k)}\right)^2}\right\|$$

为方便计算，将等式两侧都进行平方处理。

3. 移动聚类中心：计算当前得到的两个组分别的中心 μ_k, $k = 1,2$,

$$\mu_k = \frac{1}{n}\left[x^{(k_1)} + x^{(k_2)} + ... + x^{(k_n)}\right]$$

然后将原有的聚类中心设置为新计算得到的中心。

4. 重复(2)(3)直到结果收敛。

这当中值得注意的是第一步初始化步骤，在之前的实验中，我们的初始化为随机的，一般将参数的各维度都置 0，通常情况下也不会出错，至多出现迭代次数过多的问题，但在该实验中不确定的元素太多，如果初值选择不合理可能出现聚类结果与设想出现较大偏差。

课堂上介绍了一种方法用来规避这种可能，多次随机化选择，对于每次选择出的 k 个聚类中心和用欧式距离计算得到的 k 个聚类，计算其欧式距离之和，选取得数最小的一组聚类中心作为 K-means 算法迭代的初始值。

2.2.2 GMM-EM 算法

首先给出 n 维高斯分布的密度函数如下：

$$p\left(x\middle|\mu,\Sigma\right) = \frac{1}{(2\pi)^{\frac{n}{2}}\left|\Sigma\right|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

再给出混合高斯分布的定义：

$$p = \sum_{i=1}^{k}\alpha_i p\left(x\middle|\mu_i,\Sigma_i\right), \Sigma_{i=1}^{k}\alpha_i = 1$$

可以看出，混合高斯分布由 k 个混合成分构成，每个混合成分对应一个高斯分布，每个高斯分布前都有一个混合系数α_i，从贝叶斯学派的角度，其意义是先验分布，即 $p(z_j = i) = \alpha_i$。现在假设对于样本集满足高斯混合分布，实际上在后续实验步骤中我们的样本集也是按照高斯分布产生的。根据贝叶斯定理，z_j的后验分布为

$$p(z_j = i|x_j) = \frac{p(z_j = i) * p(x_j|z_j = i)}{p(x_j)} = \frac{\alpha_i * p(x_j|\mu_i,\Sigma_i)}{\sum_{l=1}^{k} \alpha_l p(x_j|\mu_l,\Sigma_l)}$$

这样一来，对于每一个样本我们都可以给定一个标记λ_j:

$$\lambda_j = \arg\max_i p(z_j = i|x_j)$$

如果给定样本集 X 可以采用极大似然估计，再进行对数运算，我们有:

$$L(X) = \ln\left(\prod_{j=1}^{m} p(x_j)\right) = \sum_{j=1}^{m} ln\left(\sum_{i=1}^{k} \alpha_i * p(x_j|\mu_i,\Sigma_i)\right)$$

为使其最大化，分别对μ_i，Σ_i求导为 0 有:

$$\mu_i = \frac{\sum_{j=1}^{m} p(z_j = i|x_j) * x_j}{\sum_{j=1}^{m} p(z_j = i|x_j)}$$

$$\Sigma_i = \frac{\sum_{j=1}^{m} p(z_j = i|x_j) * (x_j - \mu_i)(x_j - \mu_i)^T}{\sum_{j=1}^{m} p(z_j = i|x_j)}$$

对于混合系数α_i，由于其有限制条件$\alpha_i \geqslant 0, \sum_i^m \alpha_i = 1$，所以使用拉格朗日乘数法，有

$$L(X) + \lambda\left(\sum_{i-1}^{k} \alpha_i - 1\right)$$

对α_i求导并令导数为 0 有:

$$\sum_{j=1}^{m} \frac{p(x_j|\mu_i,\Sigma_i)}{\sum_{l=1}^{k} \alpha_l * p(x_j|\mu_l,\Sigma_l)} + \lambda = 0$$

等式两边分别乘上$\alpha_1,\alpha_2,\ldots,\alpha_k$并进行加和，得到:

$$\sum_{i=1}^{k}\left(\alpha_i * \sum_{j=1}^{m} \frac{p(x_j|\mu_i,\Sigma_i)}{\sum_{l=1}^{k} \alpha_l * p(x_j|\mu_i,\Sigma_i)}\right) + \lambda \sum_{i=1}^{k} \alpha_i = 0$$

由于$\sum_{l=1}^{k} \alpha_l = \sum_{i=1}^{k} \alpha_i = 1$，所以有

$$m + \lambda = 0$$

则有

$$\alpha_i = \frac{1}{m} \sum_{j=1}^{m} \frac{p(x_j|\mu_i,\Sigma_i)}{\sum_{l=1}^{k} \alpha_l * p(x_j|\mu_l,\Sigma_l)}$$

以上就是高斯混合模型(GMM)的原理分析，在 GMM 问题的求解过程中常用的方法是 EM 算法：首先根据当前参数计算每个样本属于每个混合高斯成分的后验概率，再根据之前对参数求导得到的公式对参数进行更新。

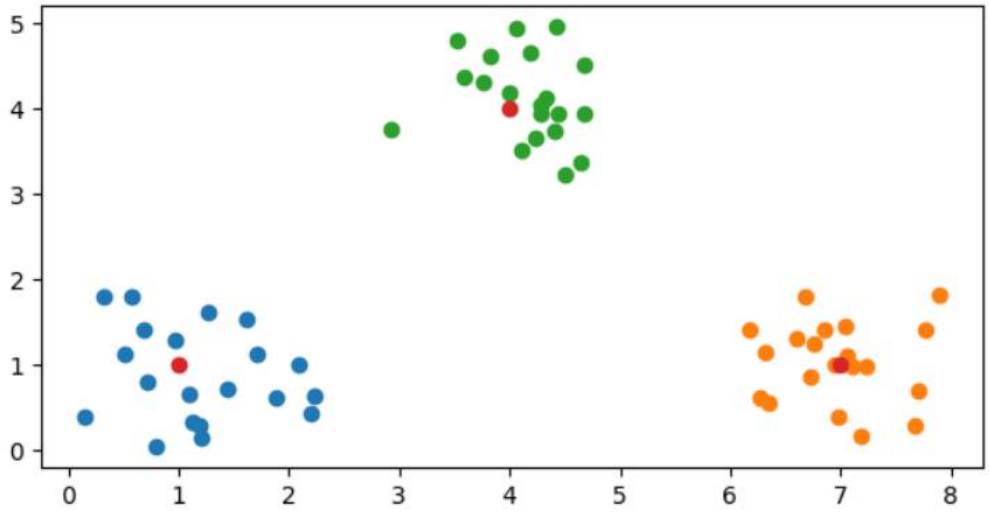
3 K-Means 算法

3.1 生成数据

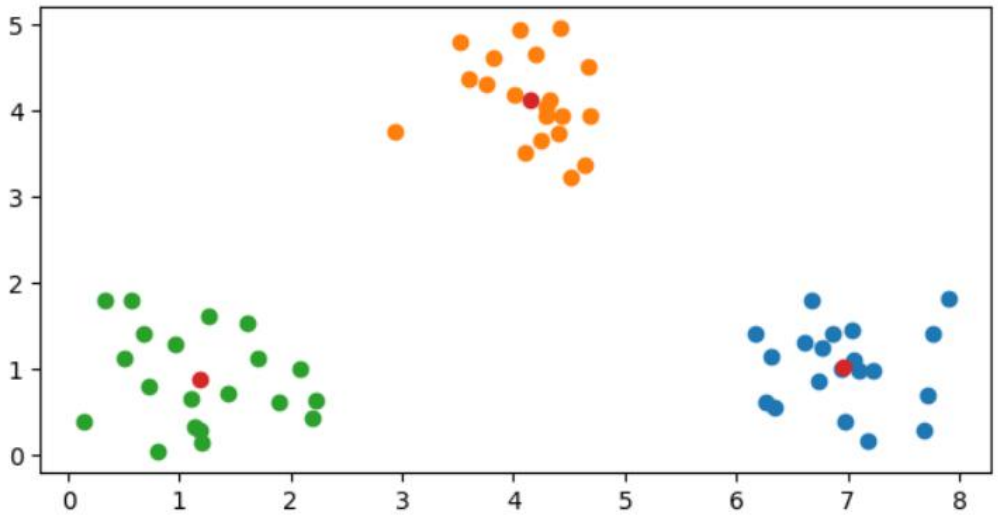
数据选取二维点，便于作图表现，一共设置三种类型的高斯分布，每一聚类的均值分别是[1,1],[7,1],[4,4]，协方差矩阵均为$\begin{bmatrix} 0.3 & 0 \\ 0 & 0.3 \end{bmatrix}$，每一类的数目均为 20 个。

3.2 具体过程

首先选取初始的聚类中心，随机选择 100 次，每次利用欧几里得距离计算花费，最终选取最小的一组聚类中心作为最佳的初始点对。接下来通过迭代找出聚类中心，并根据最终的聚类中心划分聚类，实验结果以及控制台输出如下：



1 根据高斯分布生成的数据和聚类中心



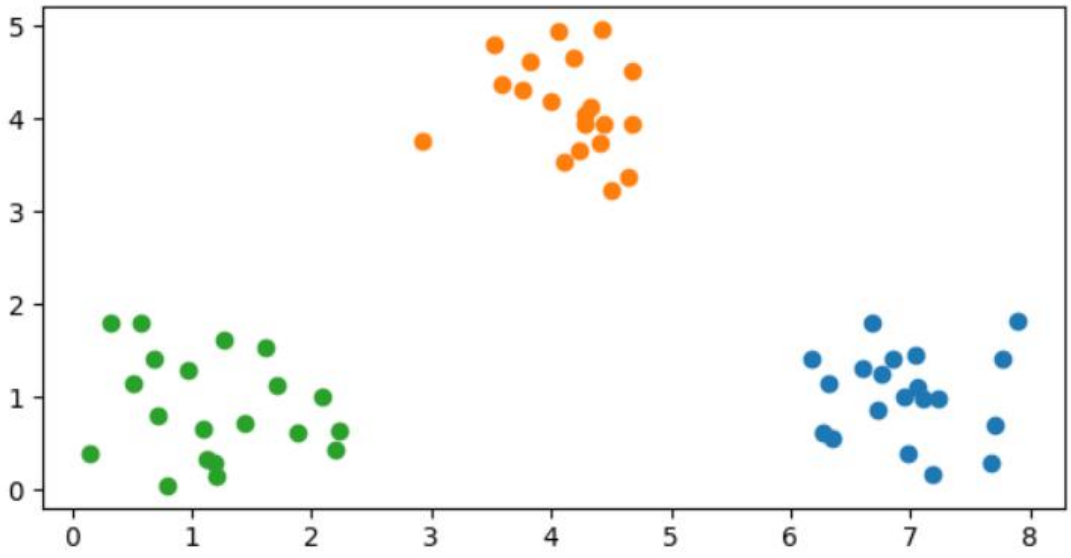
2 根据 K-Means 算法计算的聚类中心及划分聚类

```
迭代了1次
K-means生成的聚类中心为
 [[6.966468   1.03419459]
 [4.14564036 4.13456689]
 [1.18787515 0.88954136]]
```

3 控制台输出迭代次数以及聚类中心

3.3 GMM-EM 算法

因为 GMM-EM 算法不需要聚类中心，所以结果以及控制台输出如下：



4.利用 GMM-EM 算法划分的聚类

```
迭代了2次
```

5.控制台输出如图

3.4 UCI 数据集

选用 UCI 上较为流行的 iris 数据集，聚类点的特征有 4 个，聚类类型 K 共 3 种，每种有 50 个聚类点共 150 种，设置初始协方差矩阵为 $\begin{bmatrix} 0.3 & 0 & 0 & 0 \\ 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0.3 & 0 \\ 0 & 0 & 0 & 0.3 \end{bmatrix}$ 实验结果如下：

```
K-Means迭代了4次
K-means生成的聚类中心为
 [[5.88360656 2.74098361 4.38852459 1.43442623]
 [5.006      3.418      1.464      0.244     ]
 [6.85384615 3.07692308 5.71538462 2.05384615]]
K-Means准确率为 0.8866666666666667
GMM-EM迭代了33次
GMM-EM准确率为 0.9666666666666667
```

6.**控制台输出如图**

可以看出，对于该数据集，GMM-EM 的算法效果较好。

4 分析以及结论

由于 K-Means 算法使用了平方差公式，实际上在进行聚类时对于每一类是使用一个圆形范围圈定每一类，具有一定的局限性，而 GMM-EM 算法利用贝叶斯后验分布公式，直观上使用椭圆来圈定每一类，应用性更加广泛，也可以解释在 iris 数据集的拟合效果较好。