机器学习课程实验报告

k-means 聚类方法和混合高斯模型

学　　号　1180301007

姓　　名　赵锦涛

实验时间　2020 年 10 月

一、 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。

二、 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）。

应用：可以 UCI 上找一个简单问题数据，用你实现的 GMM 进行聚类。

三、 实验环境:

Python 3.8, Windows 10

四、 实验原理:

（一）k-means

k-means 算法是一种基于划分的聚类算法，它以 k 为参数，把 n 个数据对象分成 k 个簇，使簇内具有较高的相似度，而簇间的相似度较低。k-means 算法是根据给定的 n 个数据对象的数据集，构建 k 个划分聚类的方法，每个划分聚类即为一个簇。该方法将数据划分为 n 个簇，每个簇至少有一个数据对象，每个数据对象必须属于而且只能属于一个簇。同时要满足同一簇中的数据对象相似度高，不同簇中的数据对象相似度较小。聚类相似度是利用各簇中对象的均值来进行计算的。k-means 算法的处理流程如下。首先，随机地选择 k 个数据对象，每个数据对象代表一个簇中心，即选择 k 个初始中心；对剩余的每个对象，根据其与各簇中心的相似度（距离），将它赋给与其最相似的簇中心对应的簇；然后重新计算每个簇中所有对象的平均值，作为新的簇中心。不断重复以上这个过程，直到准则函数收敛，也就是簇中心不发生明显的变化。通常采用均方差作为准则函数，即最小化每个点到最近簇中心的距离的平方和。k-means 方法的复杂度为 $O(mnk)$ m n k

（二）混合高斯模型与 EM 算法

高斯混合模型假设数据中存在一定数量的高斯分布，并且每个分布代表一个簇，因此，高斯混合模型倾向于将属于同一分布的样本聚合为一类。高斯混合模型可以理解为一个将事物分解为若干的基于高斯概率密度函数（正态分布曲线）形成的模型。多个高斯分布函数的线性组合，理论上 GMM 可以拟合出任意类型的分布。

一种优雅的并且强大的寻找带有潜在变量的模型的最大似然解的方法被称为期望最大化算法，或者 EM 算法。给定一个高斯混合模型，目标是关于参数（均值、协方差、混合系数）最大化似然函数。算法流程如下：

- 初始化均值 μ_k、协　差 Σ_k 和混合系数 π_k，计算对数似然函数的初始值。
- E 步骤。使　当前参数值计算“责任”。

$$\gamma\left(z_{nk}\right)=\frac{\pi_k \mathcal{N}\left(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k,\ {}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j,\ {}_j\right)}$$

- M 步骤。使用当前的“责任”重新估计参数。

$$\begin{aligned}
\boldsymbol{\mu}_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) \mathbf{x}_n \\
\boldsymbol{\Sigma}_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right)\left(\mathbf{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)\left(\mathbf{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)^{\mathrm{T}} \\
\pi_k^{\text{new}} &= \frac{N_k}{N}
\end{aligned}$$

其中 $N_k=\sum_{n=1}^{N} \gamma(z_{nk})$

- 计算对数似然函数

$$\ln p(\boldsymbol{X} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi})=\sum_{n=1}^{N} \ln \left\{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)\right\}$$

检查参数或者对数似然函数的收敛性。如果没有满　收敛的准则，则返回第 2 步。

五、 代码实现

本次实验共有 7 个文件，其名称和作用分别为：

- *datagen.py*　生成训练数据

- *kmeans.py*　k-means 算法实现
- *GMM_EM.py*　混合高斯模型实现与 EM 算法
- *uci.py*　对 UCI 数据集进行预处理
- *uci_gmm.py*　对 UCI 数据集使用 GMM 模型 EM 算法进行聚类

在本次实验中，自主生成的数据样本共有两个维度，分为三个高斯分布，其均值分别为 $[-2, 2], [6.5, 8], [10, -2]$，协方差矩阵均为

$$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$$

UCI 数据集使用的是 Iris 数据集，具体信息请见Iris Data Set

六、 实验结果与分析

使用 k-means 算法对生成数据进行聚类，效果如图所示：

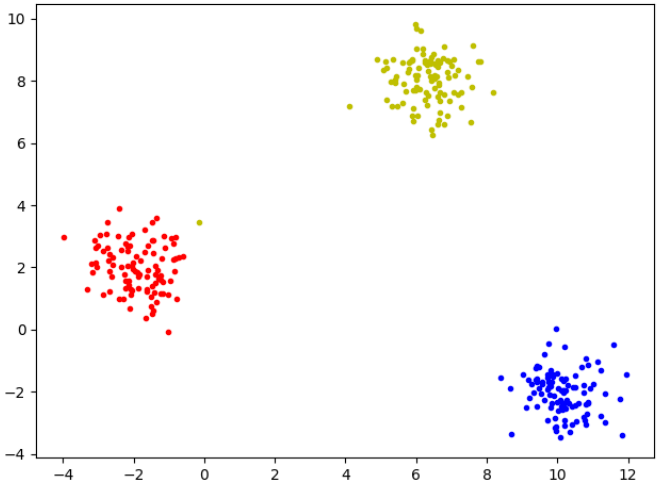

图 1: 加入正则化项

使用 GMM 模型 EM 算法对生成数据进行聚类，效果如图所示：

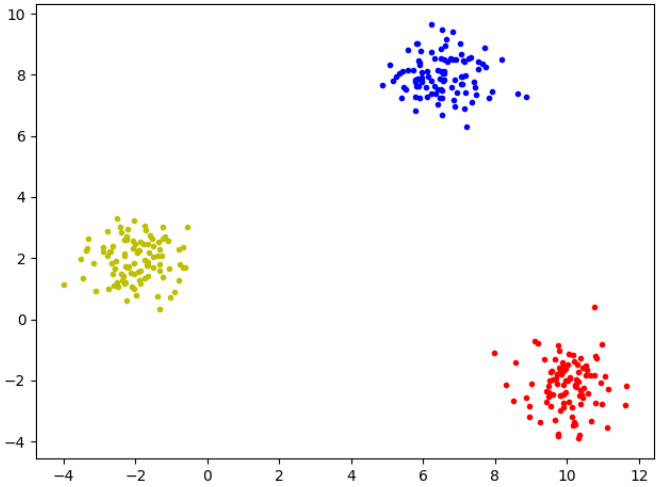

图 2: 加入正则化项

使用 EM 算法对 UCI 数据进行聚类，结果如图：

```
converged
Train result:
[53. 50. 47.]
Data result:
[50. 50. 50.]

>>>
```

图 3: 加入正则化项

从实验结果可以看出，k-means 算法和 EM 算法均能对数据进行较好的聚类。在实验中发现，GMM 模型对初始点的选择有要求，如果初始点选择地不好，则聚类结果可能会很差。一般的做法是使用 k-means 算法计算得到中心点，然后用该中心点来初始化 GMM 模型的均值。

参考文献

[1] Bishop C M. Pattern recognition and machine learning[M]. springer, 2006.