哈尔滨工业大学计算机科学与技术学院

实验报告

课程名称：机器学习

课程类型：必修

实验题目：实现k-means聚类方法和混合高斯模型

学号：1190200523

姓名：石翔宇

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、实验要求及实验环境

实验要求

测试

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。

实验环境

Windows 11 + Python 3.7.8

三、设计思想

1. k-means算法

给定n个样本的集合$X=\{x_1,x_2,\dots,x_n\}$，$x_i\in\mathbb{R}^m$，k-means聚类的目标是将n个样本分到k个不同的类或簇中（假设$k<n$）。k个类$G_1,G_2,\dots,G_k$形成对集合X的划分，其中$G_i\cap G_j=\emptyset$，$\bigcup\limits_{i=1}^{k}G_i=X$。用$C$表示划分，一个划分对应着一个聚类结果。

k-means算法通过损失函数的最小化选取最优的划分C^*。

首先，我们将样本之间的距离$d(x_i,x_j)$定义为欧氏距离平方

$$
\begin{aligned}
d(x_i,x_j)=&\sum_{k=1}^{m}(x_{ik}-x_{jk})^2\\
=&\|x_i-x_j\|^2
\end{aligned}
\tag{1}
$$

我们定义损失函数$W(C)$为

$$
W(C)=\sum_{l=1}^{k}\sum_{C(i)=l}\|x_i-\bar{x}_l\|^2 \tag{2}
$$

其中，$\bar{x}_l=\frac{1}{n_l}\sum\limits_{C(i)=l}x_i$，$n_l=\sum\limits_{i=1}^{n}I(C(i)=l)$。

则k-means算法就是求解最优化问题

$$
\begin{aligned}
C^*=&arg\min_C W(C)\\
=&arg\min_C\sum_{l=1}^{k}\sum_{C(i)=l}\|x_i-\bar{x}_l\|^2
\end{aligned}
\tag{3}
$$

k-means算法是一个迭代的过程。首先，对于给定的中心值$(m_1, m_2, \ldots, m_k)$，将每个样本指派到与其最近的中心m_l的类G_l中，得到聚类结果，使得目标函数极小化

$$\min_{m_1,\ldots,m_k} = \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - m_l\|^2 \tag{4}$$

然后，对于每个包含n_l个样本的类G_l，更新其均值m_l

$$m_l = \frac{1}{n_l} \sum_{C(i)=l} x_i \tag{5}$$

其中，$l = 1, 2, \ldots, k$。

重复上述两个步骤，直到$W(C)$结果小于阈值。

2. 高斯混合模型（GMM）

高斯混合模型是指具有如下形式的概率分布模型：

$$P(y|\theta) = \sum_{k=1}^{K} \alpha_k \phi(y|\theta_k) \tag{6}$$

其中，$\alpha_k \geq 0$是系数，满足$\sum_{k=1}^{K} \alpha_k = 1$；$\phi(y|\theta_k)$是高斯分布密度，$\theta_k = (\mu_k, \sigma_k^2)$，$\phi(y|\theta_k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(y-\mu_k)^2}{2\sigma_k^2}\right)$。

现有观测数据$y = \{y_1, y_2, \ldots, y_N\}$由高斯混合模型生成，

$$P(y|\theta) = \sum_{k=1}^{K} \alpha_k \phi(y|\theta_k) \tag{7}$$

其中，$\theta = (\alpha_1, \alpha_2, \ldots, \alpha_K; \theta_1, \theta_2, \ldots, \theta_k)$。我们将用EM算法估计高斯混合概率模型的参数$\theta$。

我们定义隐变量0-1随机变量γ_{jk}为

$$\begin{aligned} &\gamma_{jk} = \begin{cases} 1, & \text{第 } j \text{ 个观测来自第 } k \text{ 个分模型} \\ 0, & \text{否则} \end{cases} \\ &j = 1, 2, \cdots, N; \quad k = 1, 2, \cdots, K \end{aligned} \tag{8}$$

那么完全数据为

$$(y_j, \gamma_{j1}, \gamma_{j2}, \ldots, \gamma_{jK}), \quad j = 1, 2, \ldots, N \tag{9}$$

则似然函数为

$$\begin{aligned} P(y, \gamma|\theta) &= \prod_{j=1}^{N} P(y_j, \gamma_{j1}, \gamma_{j2}, \ldots, \gamma_{jK}|\theta) \\ &= \prod_{k=1}^{K} \prod_{j=1}^{N} [\alpha_k \phi(y_j|\theta_k)]^{\gamma_{jk}} \\ &= \prod_{k=1}^{K} \alpha_k^{n_k} \prod_{j=1}^{N} [\phi(y_j|\theta_k)]^{\gamma_{jk}} \\ &= \prod_{k=1}^{K} \alpha_k^{n_k} \prod_{j=1}^{N} [\frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(y-\mu_k)^2}{2\sigma_k^2}\right)]^{\gamma_{jk}} \end{aligned} \tag{10}$$

其中，$n_k=\sum\limits_{j=1}^{N}\gamma_{jk}$，$\sum\limits_{k=1}^{K}n_k=N$。

则对数似然函数为

$$\log P(y,\gamma|\theta)=\sum_{k=1}^{K}\{n_k\log\alpha_k+\sum_{j=1}^{N}\gamma_{jk}[\log\left(\frac{1}{\sqrt{2\pi}}\right)-\log\sigma_k-\frac{1}{2\sigma_k^2}(y-\mu_k)^2]\} \quad (11)$$

EM算法的E步要求我们确定Q函数

$$\begin{aligned}Q(\theta,\theta^{(i)})=&E[\log P(y,\gamma|\theta)]\\=&E\{\sum_{k=1}^{K}\{n_k\log\alpha_k+\sum_{j=1}^{N}\gamma_{jk}[\log\left(\frac{1}{\sqrt{2\pi}}\right)-\log\sigma_k-\frac{1}{2\sigma_k^2}(y-\mu_k)^2]\}\}\\=&\sum_{k=1}^{K}\{\sum_{j=1}^{N}(E\gamma_{jk})\log\alpha_k+\sum_{j=1}^{N}(E\gamma_{jk})[\log\left(\frac{1}{\sqrt{2\pi}}\right)-\log\sigma_k-\frac{1}{2\sigma_k^2}(y-\mu_k)^2]\}\end{aligned} \quad (12)$$

这里需要计算$E(\gamma_{jk}|y,\theta)$，记为$\hat{\gamma}_{jk}$：

$$\begin{aligned}\hat{\gamma}_{jk}=&E(\gamma_{jk}|y_j,\theta)=P(\gamma_{jk}=1|y_j,\theta)\\=&\frac{P(\gamma_{jk}=1,y_j|\theta)}{\sum\limits_{k=1}^{K}P(\gamma_{jk}=1,y_j|\theta)}\\=&\frac{P(y_j|\gamma_{jk}=1,\theta)P(\gamma_{jk}=1|\theta)}{\sum\limits_{k=1}^{K}P(y_j|\gamma_{jk}=1,\theta)P(\gamma_{jk}=1|\theta)}\\=&\frac{\alpha_k\phi(y_j|\theta_k)}{\sum\limits_{k=1}^{K}\alpha_k\phi(y_j|\theta_k)},\quad j=1,2,\ldots,N;\quad k=1,2,\ldots,K\end{aligned} \quad (13)$$

将$\hat{\gamma}_{jk}=E\gamma_{jk}$和$n_k=\sum\limits_{j=1}^{N}E\gamma_{jk}$代入式 13 得

$$Q(\theta,\theta^{(i)})=\sum_{k=1}^{K}\{n_k\log\alpha_k+\sum_{j=1}^{N}\hat{\gamma}_{jk}[\log\left(\frac{1}{\sqrt{2\pi}}\right)-\log\sigma_k-\frac{1}{2\sigma_k^2}(y-\mu_k)^2]\} \quad (14)$$

EM算法的M步是要求得函数$Q(\theta,\theta^{(i)})$对θ的极大值，即

$$\theta^{(i+1)}=\arg\max_{\theta}Q(\theta,\theta^{(i)}) \quad (15)$$

将式 14 分别对μ_k和σ_k^2求偏导并令其为0，可得

$$\begin{aligned}\frac{\partial Q}{\mu_k}=&\sum_{j=1}^{N}[\hat{\gamma}_{jk}\frac{1}{\sigma_k^2}(y-\mu_k)]=0\\\sum_{j=1}^{N}(\hat{\gamma}_{jk}y)=&\sum_{j=1}^{N}(\hat{\gamma}_{jk}\mu_k)\\\sum_{j=1}^{N}(\hat{\gamma}_{jk}y)=&\mu_k\sum_{j=1}^{N}\hat{\gamma}_{jk}\\\mu_k=&\frac{\sum_{j=1}^{N}(\hat{\gamma}_{jk}y)}{\sum_{j=1}^{N}\hat{\gamma}_{jk}},\quad k=1,2,\ldots,K\end{aligned} \quad (16)$$

$$\frac{\partial Q}{\ldots}=\sum^{N}[\ldots\frac{1}{\ldots}\frac{1}{\ldots}$$

$$
\begin{aligned}
\frac{\partial Q}{\partial \sigma_k^2} &= \sum_{j=1}^{N}[\hat{\gamma}_{jk}(-\frac{1}{2\sigma_k^2} + \frac{1}{2\sigma_k^4}(y-\mu_k)^2)] = 0 \\
\sum_{j=1}^{N}[\hat{\gamma}_{jk}\frac{1}{\sigma_k^2}(y-\mu_k)^2] &= \sum_{j=1}^{N}\hat{\gamma}_{jk} \\
\frac{1}{\sigma_k^2}\sum_{j=1}^{N}[\hat{\gamma}_{jk}(y-\mu_k)^2] &= \sum_{j=1}^{N}\hat{\gamma}_{jk} \\
\sigma_k^2 &= \frac{\sum_{j=1}^{N}[\hat{\gamma}_{jk}(y-\mu_k)^2]}{\sum_{j=1}^{N}\hat{\gamma}_{jk}}, \quad k=1,2,\ldots,K
\end{aligned} \tag{17}
$$

对于$\hat{\alpha}_k$，需满足$\sum_{k=1}^{K}\alpha_k = 1$，则构造拉格朗日多项式：

$$
Q' = \sum_{k=1}^{K}\{n_k \log \alpha_k + \sum_{j=1}^{N}\hat{\gamma}_{jk}[\log\left(\frac{1}{\sqrt{2\pi}}\right) - \log\sigma_k - \frac{1}{2\sigma_k^2}(y-\mu_k)^2]\} + \lambda(\sum_{k=1}^{K}\alpha_k - 1) \tag{18}
$$

将式 18 对α_k求导并令导数为0得

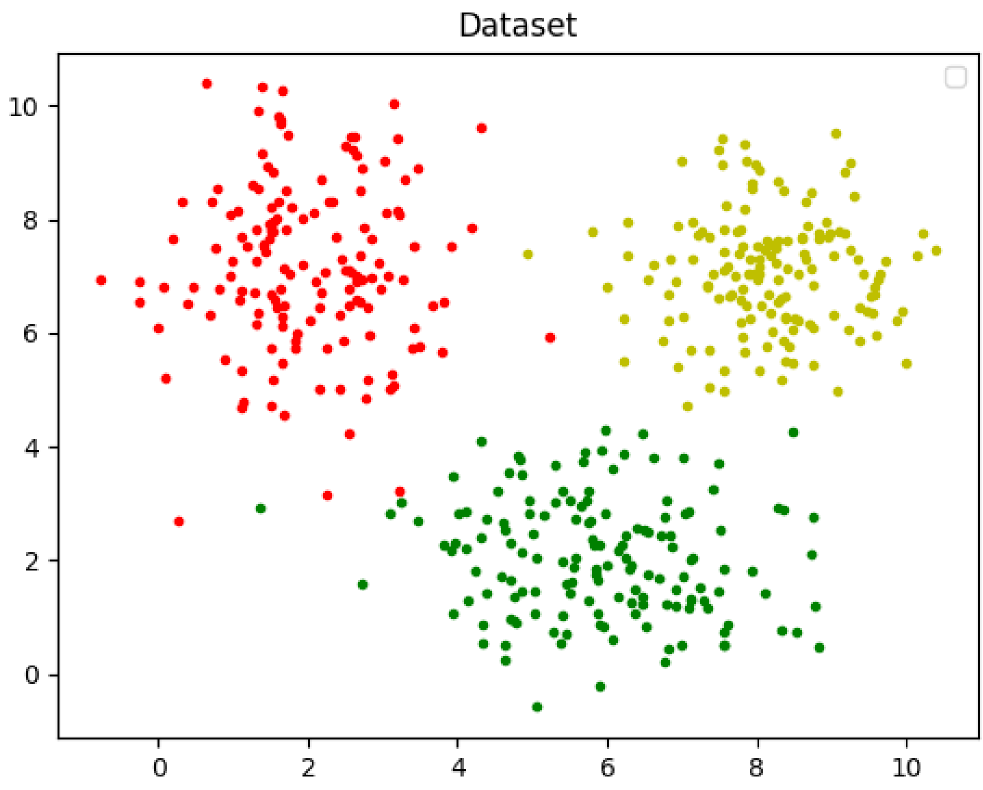
$$
\begin{aligned}
\frac{\partial Q'}{\partial \alpha_k} &= \frac{n_k}{\alpha_k} + \lambda = 0 \\
n_k + \lambda\alpha_k &= 0 \\
\sum_{k=1}^{K} n_k + \sum_{k=1}^{K}\lambda\alpha_k &= 0 \\
N + \lambda &= 0 \\
\lambda &= -N \\
\alpha_k &= \frac{n_k}{N} = \frac{\sum_{j=1}^{N}\hat{\gamma}_{jk}}{N}, \quad k=1,2,\ldots,K
\end{aligned} \tag{19}
$$

重复以上计算，直到对数似然值不再有明显的变化为止。

四、实验结果分析

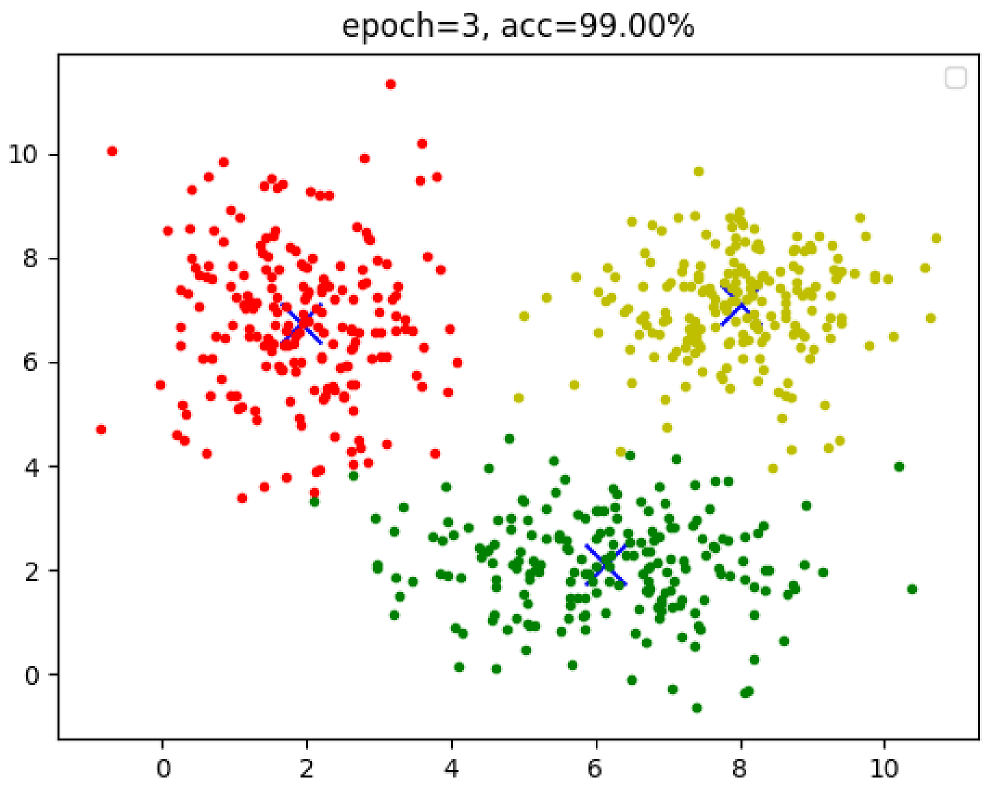
1. 生成数据

我们将类别数k设置为3，均值分别为$(2,7)$、$(6,2)$和$(8,7)$，协方差矩阵分别为$[[1,0],[0,2]]$、$[[2,0],[0,1]]$和$[[1,0],[0,1]]$，每个类别的数目设置为150，生成数据，结果如下图所示。



2. k-means算法

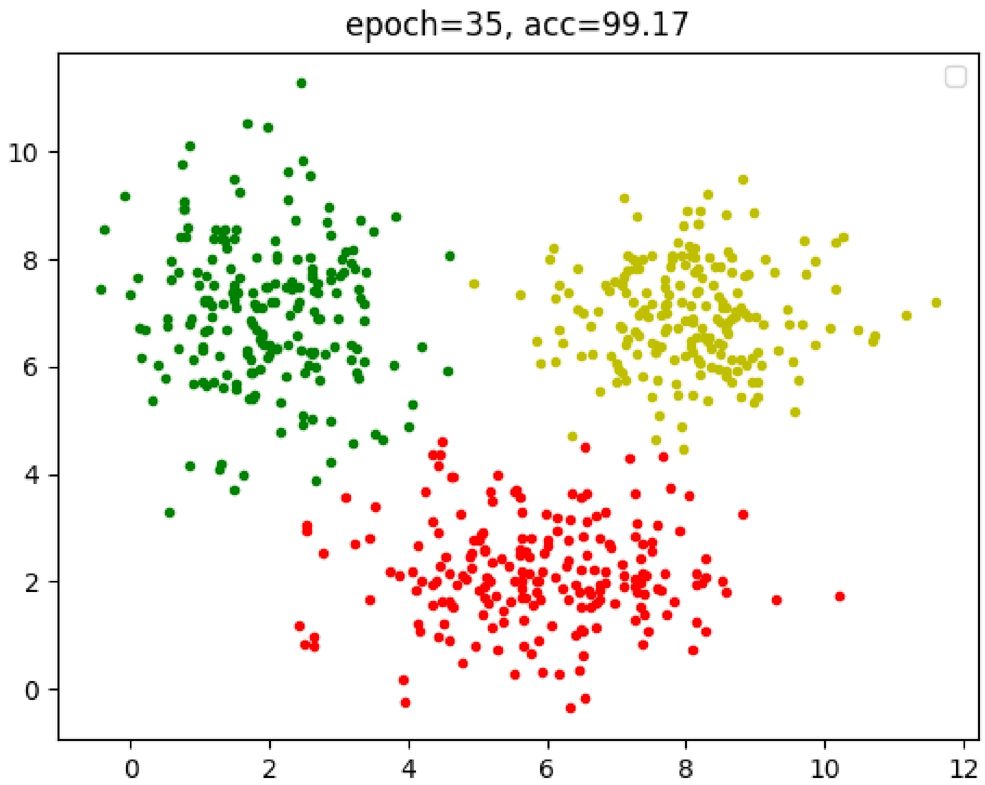
我们数据的每个类别的数目设置为200，其他参数不变，生成数据。我们将k-means算法的最大轮数设置为100轮，停止策略的系数设置为10^{-7}，实验结果如下图所示。



可以看到，在完成3轮迭代之后k-means算法收敛，得到最佳的中心点，预测准确率为99.00%。k-means算法收敛速度较快，准确率也比较高。

3. GMM

我们数据的每个类别的数目设置为200，其他参数不变，生成数据。我们将GMM算法的最大轮数设置为100轮，停止策略的系数设置为10^{-7}，实验结果如下图所示。



可以看到，在完成35轮迭代之后GMM算法收敛，得到最佳的中心点，预测准确率为99.17%。GMM算法收敛速度较快，准确率也比较高。

4. UCI数据集

我们选用UCI数据集Iris，该数据集共有150个数据，类别为4，分别选用k-means和KMM算法进行实验，实验结果如下：

> k-means：epoch=6, acc=89.33%
>
> GMM：epoch=61, acc=96.67%

我们可以看到，在UCI数据集上，GMM算法比k-means算法效果更好，但收敛速度较慢。

五、结论

k-means算法较易理解与实现，在简单数据集上效果很好并且收敛较快；GMM的实现复杂，推导繁琐，在各种数据集上都能取得良好的效果，收敛速度较k-means缓慢。

六、参考文献

七、附录:源代码(带注释)