

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聚类和混合高斯模型

学号：1160300314

姓名：朱明彦

一、实验目的

实现一个k-means算法和混合高斯模型，并用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结果(与你的设定结果比较)。

应用

可以在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

- OS: Ubuntu 16.04.5 LTS
- python 3.7.0

三、设计思想(本程序中用到的主要算法及数据结构)

1.算法原理

1.1 K-Means算法原理

给定样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$ 和划分聚类的数量 k ，给出一个簇划分 $C = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_k\}$ ，使得该簇划分的平方误差 E 最小化，其中 E 如式(1)

$$E = \sum_{i=1}^k \sum_{\mathbf{x} \in \mathbf{C}_i} \|\mathbf{x} - \mu_i\|_2^2 \quad (1)$$

式(1)中， $\mu_i = \frac{1}{|\mathbf{C}_i|} \sum_{\mathbf{x} \in \mathbf{C}_i} \mathbf{x}$ 是簇 \mathbf{C}_i 的均值向量。 E 刻画了簇内样本的内聚的紧密程度，其值越小，则簇内样本的相似度越高。

K-Means的优化目标需要考察到样本集 D 的全部可能的划分，这是一个NP难的问题。因此K-Means采用贪心策略，通过迭代优化来近似求解。

迭代优化的策略如下：

1. 首先初始化一组均值向量
2. 根据初始化的均值向量给出样本集 D 的一个划分，样本距离那个簇的均值向量距离最近，则将该样本划归到哪个簇
3. 再根据这个划分来计算每个簇内真实的均值向量，如果真实的均值向量与假设的均值向量相同，假设正确；否则，将真实的均值向量作为新的假设均值向量，回到1.继续迭代求解。

1.2 GMM算法原理

首先给出 n 维样本空间中的随机变量 \mathbf{x} 服从高斯分布的密度函数：

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

其中 $\mu = \{\mu_1, \mu_2, \dots, \mu_n\}$ 为 n 维的均值向量， Σ 是 $n \times n$ 的协方差阵。

再给出混合高斯分布的定义：

$$p_{\mathcal{M}} = \sum_{i=1}^k \alpha_i p(\mathbf{x}|\mu_i, \Sigma_i) \quad (3)$$

这个分布由 k 个混合成分构成，每个混合成分对应一个高斯分布。其中 μ_i, Σ_i 是第 k 个高斯分布的均值和协方差矩阵， $\alpha_i > 0$ 为相应的混合系数，满足 $\sum_{i=1}^k \alpha_i = 1$ 。

我们假设对于样本集 D 由高斯混合分布给出：首先根据 $\alpha_1, \alpha_2, \dots, \alpha_k$ 定义的先验分布选择高斯分布混合成分，即 $p(z_j = i) = \alpha_i$ ，其中 $z_j \in \{1, 2, \dots, k\}$ ；然后，根据被选择的高斯混合成分的概率密度函数进行采样，从而生成相应的样本。那么根据贝叶斯定理， z_j 的后验分布对应于：

$$p_{\mathcal{M}}(z_j = i|\mathbf{x}_j) = \frac{p(z_j = i) \cdot p_{\mathcal{M}}(\mathbf{x}_j|z_j = i)}{p_{\mathcal{M}}(\mathbf{x}_j)} = \frac{\alpha_i \cdot p(\mathbf{x}_j|\mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l p(\mathbf{x}_j|\mu_l, \Sigma_l)} \quad (4)$$

$p_{\mathcal{M}}(z_j = i|\mathbf{x}_j)$ 给出了样本 \mathbf{x}_j 由第 i 个高斯混合分布生成的后验概率。

当式(3)已知时，混合高斯模型将样本集 D 划分成了 k 个簇 $C = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_k\}$ ，对于每一个样本 \mathbf{x}_j ，其簇标记为 λ_j ：

$$\lambda_j = \arg \max_i p_{\mathcal{M}}(z_j = i|\mathbf{x}_j) \quad (5)$$

关键在与参数 $\{\alpha_i, \mu_i, \Sigma_i | i \in \{1, 2, \dots, k\}\}$ 的求解，如果给定样本集 D 可以采用极大似然估计(最大化对数似然)：

$$LL(D) = \ln \left(\prod_{j=1}^m p_{\mathcal{M}}(\mathbf{x}_j) \right) = \sum_{j=1}^m \ln \left(\sum_{i=1}^k \alpha_i \cdot p(\mathbf{x}_j|\mu_i, \Sigma_i) \right) \quad (6)$$

使式(6)最大化，对 μ_i 求导令导数为0有：

$$\sum_{j=1}^m \frac{\alpha_i \cdot p(\mathbf{x}_j|\mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l \cdot p(\mathbf{x}_j|\mu_l, \Sigma_l)} \Sigma_i^{-1} (\mathbf{x}_j - \mu_i) = 0 \quad (7)$$

两边同乘 Σ_i 进行化简有：

$$\mu_i = \frac{\sum_{j=1}^m p_{\mathcal{M}}(z_j = i | \mathbf{x}_j) \cdot \mathbf{x}_j}{\sum_{j=1}^m p_{\mathcal{M}}(z_j = i | \mathbf{x}_j)} \quad (8)$$

即各个混合成分的均值可以通过样本加权平均来估计，权重样本式每个样本属于该成分的后验概率。

同理式(6)对 Σ_i 求导令导数为0有：

$$\Sigma_i = \frac{\sum_{j=1}^m p_{\mathcal{M}}(z_j = i | \mathbf{x}_j) \cdot (\mathbf{x}_j - \mu_i)(\mathbf{x}_j - \mu_i)^T}{\sum_{j=1}^m p_{\mathcal{M}}(z_j = i | \mathbf{x}_j)} \quad (9)$$

对于混合系数 α_i ，由于其还需要满足 $\alpha_i \geq 0$, $\sum_{i=1}^k \alpha_i = 1$ ，所以在式(6)的基础上增加拉格朗日项：

$$LL(D) + \lambda \left(\sum_{i=1}^k \alpha_i - 1 \right) \quad (10)$$

其中 λ 为拉格朗日乘子，由式(10)对 α_i 求导并令导数为0有：

$$\sum_{j=1}^m \frac{p(\mathbf{x}_j | \mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l \cdot p(\mathbf{x}_j | \mu_l, \Sigma_l)} + \lambda = 0 \quad (11)$$

式(11)两边同乘 α_i 并将 $i \in \{1, 2, \dots, k\}$ 代入相加得：

$$\sum_{i=1}^k \left(\alpha_i \cdot \sum_{j=1}^m \frac{p(\mathbf{x}_j | \mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l \cdot p(\mathbf{x}_j | \mu_l, \Sigma_l)} \right) + \lambda \sum_{i=1}^k \alpha_i = 0 \quad (12)$$

整理一下，由于 $\sum_{i=1}^k \alpha_i = 1$ ：

$$\sum_{j=1}^m \left(\frac{\sum_{i=1}^k \alpha_i \cdot p(\mathbf{x}_j | \mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l \cdot p(\mathbf{x}_j | \mu_l, \Sigma_l)} \right) + \lambda = m + \lambda = 0 \quad (13)$$

从而有 $\lambda = -m$ ，结合式(11)有：

$$\alpha_i = \frac{1}{m} \sum_{j=1}^m \frac{p(\mathbf{x}_j | \mu_i, \Sigma_i)}{\sum_{l=1}^k \alpha_l \cdot p(\mathbf{x}_j | \mu_l, \Sigma_l)} \quad (14)$$

即每个高斯成分的混合系数由样本属于该成分的平均后验概率确定。

2. 算法的实现

2.1 K-Means算法实现

2.1.1 随机选择样本作为初始均值向量

1. 从样本集 D 中随机选择 k 个样本作为初始化的假设均值向量 $\{\mu_1, \mu_2, \dots, \mu_k\}$
2. 重复迭代直到算法收敛：
 1. 初始化 $\mathbf{C}_i = \emptyset, i = 1, 2, \dots, k$

2. 对 $\mathbf{x}_j, j = 1, 2, \dots, m$ 标记为 λ_j , 使得 $\lambda_j = \arg \min_i \|\mathbf{x}_j - \mu_i\|$, 即使得每个 \mathbf{x}_j 都是属于距离其最近的均值向量所在的簇
3. 将样本 \mathbf{x}_j 划分到相应的簇 $\mathbf{C}_{\lambda_j} = \mathbf{C}_{\lambda_j} \cup \{\mathbf{x}_j\}$
4. 重新计算每个簇的均值向量 $\hat{\mu}_i = \frac{1}{|\mathbf{C}_i|} \sum_{\mathbf{x} \in \mathbf{C}_i} \mathbf{x}$
5. 如果对于所有的 $i \in 1, 2, \dots, k$, 均有 $\hat{\mu}_i = \mu_i$, 则终止迭代; 否则将重新赋值 $\mu_i = \hat{\mu}_i$ 进行迭代

2.1.2 利用最大化初始均值向量之间距离方式进行选择

在下面的实验结果分析中, 我们可以看到2.1.1 K-Means算法的聚类结果严重依赖于初始化的簇中心, 所以当初始化的簇中心“不好”的时候, 会导致整个的聚类结果不好, 所以下面采用了一种最大化簇中心距离的方法, 选择均值向量。

仅对于2.1.1节中2.1进行优化:

- 首先随机选择一个样本作为均值向量
- 进行迭代, 直到选择到 k 个均值向量:
 - 假设当前已经选择到 i 个均值向量 $\{\mu_1, \mu_2, \dots, \mu_i\}$, 则在 D $\{\mu_1, \mu_2, \dots, \mu_i\}$ 选择距离已选出的 i 个均值向量距离最远的样本
 - 将其加入初始均值向量, 得到 $\{\mu_1, \mu_2, \dots, \mu_i, \mu_{i+1}\}$ 通过这种初始化均值向量的方式, 能够有效降低初始簇中心的“集中程度”, 在一定程度上避免结果陷入局部最优解。

2.2 GMM算法实现

GMM常采用EM算法进行迭代优化求解, 其中每次迭代中, 先根据当前参数来计算每个样本属于每个高斯成分的后验概率, 所谓“E步”; 再根据式(8)(9)(14)更新参数, 所谓“M步”。

给定样本集 D 和高斯混合成分数目 k 。

1. 随机初始化参数 $\{\alpha_i, \mu_i, \Sigma_i | i \in \{1, 2, \dots, k\}\}$ 以及 $\mathbf{C}_i = \emptyset$
2. 开始迭代至达到迭代次数或者是参数值不再发生变化:
 1. E步, 根据式(4)计算每个样本由各个混合高斯成分生成的后验概率
 2. M步, 根据式(8)(9)(14)更新参数 $\{\alpha_i, \mu_i, \Sigma_i | i \in \{1, 2, \dots, k\}\}$
3. 根据式(5)确定每个样本的簇标记 λ_j , 并将其加入相应的簇 $\mathbf{C}_{\lambda_j} = \mathbf{C}_{\lambda_j} \cup \{\mathbf{x}_j\}$
4. 输出簇划分 $C = \{\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_k\}$

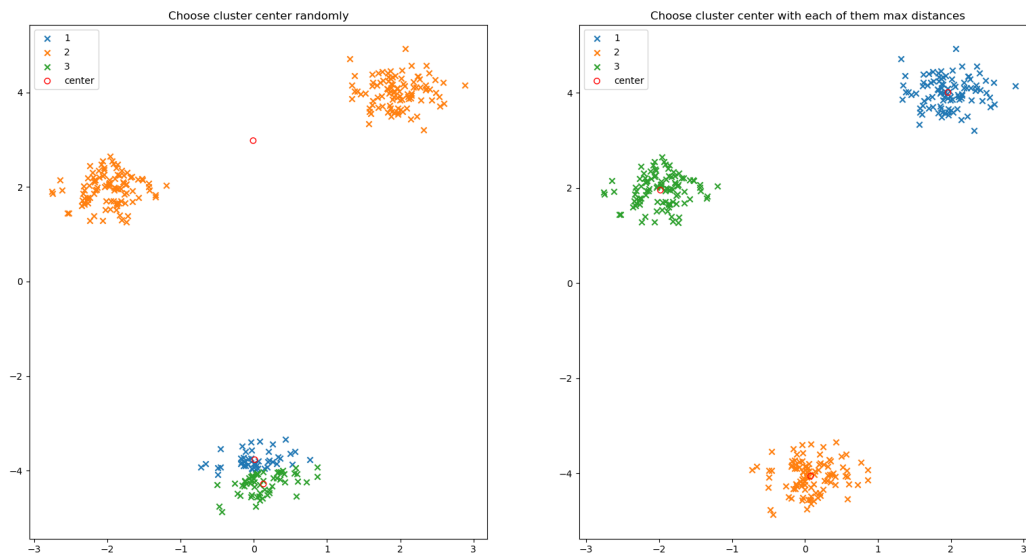
四、实验结果分析

1. 生成数据的测试

在生成数据时, 使用的是二维空间上的数据, 便于数据可视化; 利用二维高斯分布, 按照给定的均值和样本数量要求生成数据。

1.1 K-Means两种不同初始值方法结果对比

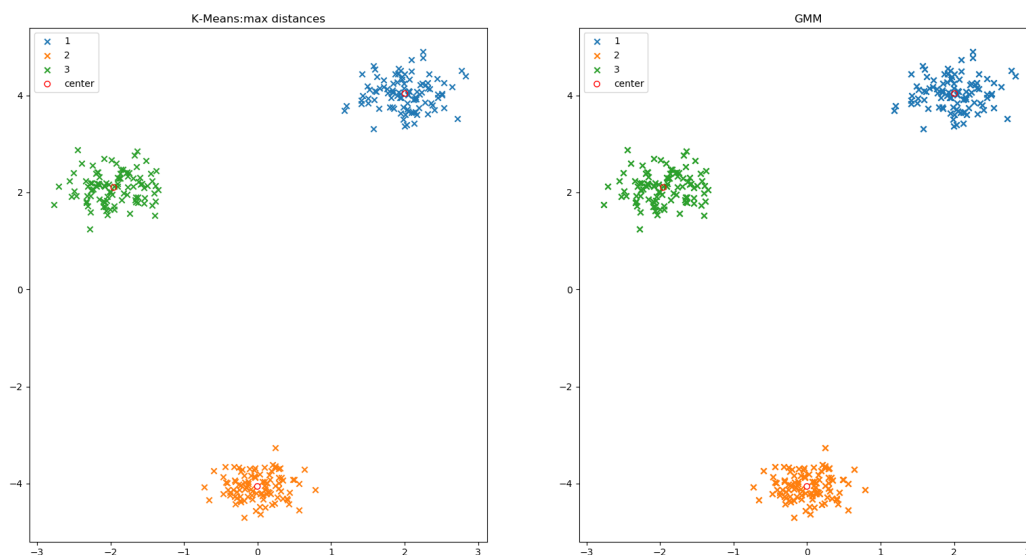
在三.2.1中提到了两种初始化的方式，在这里我们对比一下他们聚类的结果。



可以看到，在左侧，使用的随机选择初始簇中心的方式，在这次的运行中，就由于初始簇中心的问题，导致陷入局部最优解，没能有效地将样本分为三类；而在右侧则是使用选择距离最远的 k 个初始簇中心，将彼此之间的距离增大有效地划分了三类。

1.2 K-Means与GMM对比

同样使用生成数据，对比K-Means和GMM的结果如下：



可以看到，两种方法在生成数据上的表现类似，都可以实现聚类。

2. UCI数据测试

使用的UCI的数据是Iris(鸢尾花)数据集，根据其4个属性：

- 花萼长度
- 花萼宽度

- 花瓣长度
- 花瓣宽度 来预测鸢尾花属于(Setosa, Versicolour, Virginica)三类中的哪一类。

由于k-means和GMM输出的结果中，类别的编号可能是不同的，所以将所有可能的序号排列均进行测试，与测试样本中给出的**label**进行对比，得到的最优的结果作为最终的结果。此外还需要标注每个样本属于哪个类别。最终的测试结果如下,上面为GMM结果，下面为K-means的准确度：

```
0.7133333333333334
0.8866666666666667
```

3. 关于GMM算法迭代中变化

在实际执行中，将GMM初始参数的初始化方式与K-Means类似，均是选择 k 个距离最远的均值 μ_i ，协方差阵初始化为 $n \times n$ 的对角阵，对角元素均为0.1，混合系数取 $\frac{1}{k}$ 。在执行过程中，查看似然值的变化，如下：

```
GMM
0
-inf
1
-274.58376891176636
2
-255.01870933124792
3
-238.1119235634578
4
-225.23891479668256
5
-219.22643139995486
6
-216.5986611795422
7
-215.25848291145974
8
-214.4466394600144
9
-213.81931561860236
10
-213.25697700180416
```

可以看到，似然值始终在增大，与预期相符。

五、结论

- K-Means实际上假设数据呈球状分布，与之相比GMM使用更加一般的数据表示即高斯分布
- K-Means假设使用的欧式距离来衡量样本与各个簇中心的相似度(假设数据的各个维度对于相似度计算的作用是相同的)

- K-Means的簇中心初始化对于最终的结果有很大的影响，如果选择不好初始的簇中心值容易使之陷入局部最优解
- GMM使用EM算法进行迭代优化，因为其涉及到隐变量的问题，没有之前的完全数据，而是在不完全数据上进行。

六、参考文献

- [Christopher Bishop. Pattern Recognition and Machine Learning.](#)
- [周志华 著. 机器学习, 北京: 清华大学出版社, 2016.1](#)
- [UCI Iris](#)
- [AI Note](#)

七、附录:源代码(带注释)

- 主程序见lab3.py
- K-means聚类算法见k_means.py
- 混合高斯模型见gaussian_mixture_model.py
- 从Iris数据集中读取数据见iris_read.py