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：选修

实验题目：实现 k-means 聚类方法和混合高斯模型

学号：1160300312
姓名：靳贺霖

一、实验目的

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

二、实验要求及实验环境

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）

（3） 可以从 UCI 上找一个简单的问题数据，用你实现的 GMM 进行聚类

实验环境：

操作系统：windows 7

编译环境：python3.7,

编译器：PyCharm

三、设计思想（本程序中的用到的主要算法及数据结构）

1．算法原理

之前我们实现均是带有“标签”的分类问题的解决，这次我们要实现的是对于没有“标签”的数据如何揭示其内在性质和规律，即聚类（cluster）问题。这里我们主要考虑两种模型，分别是 k-means 和混合高斯模型，然后用 EM 算法来估计这两种模型中的参数。

首先说明 k-means（k 均值）求解聚类问题的原理。对于给定的样本集D = $\{x_1, x_2, \dots x_m\}$以及对应的 k 个簇划分C = $\{C_1, C_2, \dots C_k\}$，我们需要最小化的误差函数为

$$\mathrm{E} = min_{\mu} min_{C} \sum_{i=1}^{k} \sum_{j:C(j)=i} ||x - \mu_i||_2^2$$

其中μ_i是每个簇C_i的均值向量。对于这个误差函数的最小化我们无法获得解析解，它一个 NP 难问题。对于这个模型，我们需要估计的就是每个簇中的向量元素以及他们的均值。我们可以通过 EM 算法来对模型中的参数进行估计。对于最小误差函数，我们可以先固定簇划分每个簇划分的均值μ，来优化簇 C 的划分，即使损失函数为

$$E_e = \sum_{j=1}^{m} min_{C(j)} ||\mu_{C(j)} - x_j||_2^2$$

然后最小化E_e。可以看出，其实最小化E_e就是将每一个样本点划分到它距离最近的簇均值向量中。其实，这就是 EM 算法的 E 步（Expectation step）。然后我们对于得到的结果，固定每一个簇划分 C，来对参数μ来进行优化估计，即使损失函数为

$$E_m = \sum_{i=1}^{k} min_{\mu_i} \sum_{j:C(j)=i} ||\mu_i - x_j||_2^2$$

然后最小化E_m。这个通过求导可以得到，每次只需要将每个簇的均值设置为该簇中所有向量的均值即可。其实，这就是 EM 算法中的 M 步（Maximization step）。这样，我们就可以

通过不断迭代优化E_e和E_m求得最小值时的参数，就可以对解析解进行逼近。当每一个簇中的元素不再改变的时候，我们可以认为已经得到一个优化解。但由于存在局部最优的可能，这个解不一定是全局最优解。

然后我们来说明混合高斯模型（Mixture of Gaussions）求解聚类问题的原理。和 k-mean 方法不同，混合高斯模型通过概率模型来表达聚类模型。混合高斯模型假定数据来源于 k 个有着相同参数的高斯分布。每个簇中的数据都是由一个均值向量为μ_i以及协方差矩阵为$\sigma^2 I$的高斯分布生成。而且，对于每个数据x_j，考虑它由簇 i 生成的概率$\mathrm{P}(y_j = \mathrm{i}) = \alpha_i$，这样，我们就能够得到高斯混合分布为

$$p(\mathrm{x}) = \sum_i p(x|y=i)p\,(y=i) = \sum_i \alpha_i * p(x|\mu_i, \Sigma_i)$$

其中，$p(x|\mu_i, \Sigma_i)$为高斯分布的概率密度函数，μ_i和Σ_i分别是第 i 个高斯混合成分的参数。这样，对于这个式子中的模型参数$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$，我们就可以通过最大似然法来进行估计，对于给定的样本集 D，似然函数为

$$\mathrm{LL(D)} = \sum_{j=1}^{k} \ln(\sum_i \alpha_i * p(x|\mu_i, \Sigma_i))$$

但是，和 k-means 相同，直接对这个似然函数进行估计比较困难，所以我们考虑使用 EM 算法。

首先我们可以很容易知道，对于样本x_j由第 i 个高斯混合成分生成的后验概率的分布为

$$p(y_j = \mathrm{i}|x_j) = \frac{p(y_j = \mathrm{i})p(x|y=i)}{p(\mathrm{x})} = \frac{\alpha_i * p(x_j|\mu_i, \Sigma_i)}{\sum_i \alpha_i * p(x|\mu_i, \Sigma_i)}$$

将其简记为γ_{ji}，这样，对于样本x_j要划分入的簇的标记即为$argmax_i \gamma_{ji}$。可以看出，簇的划分也是由模型参数$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$确定的。其实，计算后验概率的这一步即为 EM 算法中的 Expectation step。然后对于每个参数的最大似然估计，为 EM 算法中的 Maximization step。

首先，若$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$能使 LL(D)最大化，则有$\frac{\partial \mathrm{LL(D)}}{\partial \mu_i} = 0$，经过推导计算，得到

$$\mu_i = \frac{\sum_{j=1}^{m} \gamma_{ji} x_j}{\sum_{j=1}^{m} \gamma_{ji}}$$

而且，还要有$\frac{\partial \mathrm{LL(D)}}{\partial \Sigma_i} = 0$，推导得到

$$\Sigma_i = \frac{\sum_{j=1}^{m} \gamma_{ji}(x_j - \mu_i)(x_j - \mu_i)^T}{\sum_{j=1}^{m} \gamma_{ji}}$$

对于高斯混合系数α_i而言，因为其还要满足归一化，即$\sum_i \alpha_i = 0$，所以还要在似然函数的基础上加上拉格朗日乘子，即要最大化的函数为

$$\sum_{j=1}^{k} \ln(\sum_i \alpha_i * p(x|\mu_i, \Sigma_i)) + \lambda \sum_i \alpha_i$$

将其对α_i的偏导为 0，得到

$$\alpha_i = \frac{1}{m} \sum_{j=1}^{m} \gamma_{ji}$$

这样，我们就已经得到了模型参数$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$的估计。所以整体的 EM 算法就是，使用参数$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$来求得后验概率$\gamma_{ji}$，然后由后验概率$\gamma_{ji}$来再次估计模型参数$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$，多次迭代就能够对解进行优化。

2. 算法的实现

首先要对实验数据进行生成。为了能够在图中表示，选择生成有着二维特征的数据。然后，生成三个簇的数据，每个数据的生成满足一个二维均值向量和对角线元素相同的协方差阵的高斯分布。然后对于不同的三个簇的数据，为其设定不同的标签用于判定聚类的结果是否为正确的。

首先介绍 k-means 模型通过 EM 算法实现聚类的方法。首先初始化均值为三个随机的样本点，三个簇为空。然后计算所有的样本点要三个均值点的欧式距离，距离哪个均值点最近，就将其归到那个均值对应的簇中。之后对于每个簇，计算所有向量的均值来获得每个簇新的均值。然后迭代重复这两步，直到在一次迭代中簇中的向量元素不再变化时，就停止迭代，得到三个簇的结果和每个簇对应的均值向量。

然后介绍混合高斯模型通过 EM 算法实现聚类的方法。首先初始化参数模型$\langle(\alpha_i, \mu_i, \Sigma_i)\rangle$，其中$\alpha_i$初始化$\frac{1}{cluster_num}$，其中$cluster_num$为簇的数量；$\mu_i$初始化为随机的三个样本点；$\Sigma_i$初始化为所有样本的方差。然后，开始进行迭代。在 E 步中迭代计算所有的$\gamma_{ji} = \frac{\alpha_i * p(x_j|\mu_i,\Sigma_i)}{\sum_i \alpha_i * p(x|\mu_i,\Sigma_i)}$，在 M 步中分别更新模型参数$\mu_i = \frac{\sum_{j=1}^{m} \gamma_{ji} x_j}{\sum_{j=1}^{m} \gamma_{ji}}$、$\Sigma_i = \frac{\sum_{j=1}^{m} \gamma_{ji}(x_j-\mu_i)(x_j-\mu_i)^T}{\sum_{j=1}^{m} \gamma_{ji}}$、$\alpha_i = \frac{1}{m}\sum_{j=1}^{m} \gamma_{ji}$，迭代一定的步数之后就可以停止。然后就可以通过$argmax_i \gamma_{ji}$来获得每个样本应该被划分到的簇，最终就得到结果。

最后介绍实验结果的分析。因为我们实现的聚类是无监督的，所以得到的分类是没有标签的，如果要计算正确率就需要知道划分出来的簇对应的标签的值。这里选择将得到的簇结果与对应的标签进行全排列对于，对于每个全排列都计算一个正确率，然后认为正确率最大的那个即为簇和标签的对应。

四、实验结果与分析

1. k-means 和混合高斯模型的实现和对比

首先，自己生成的数据点共有 150 个，分三个簇，每个簇中有 50 个样本点。样本点有二维特征，由二维高斯分布得到。各个簇的均值向量分别为[1, 3]、[4, 1]、[3, 5]，协方差阵均为[[0.7, 0], [0, 0.7]]，散点图如下所示：

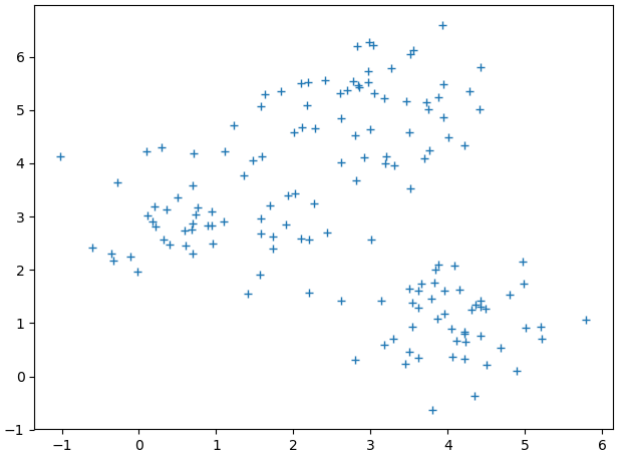

figure1:生成的散点图

对于 k-means 和混合高斯模型的结果，大多数情况下都是好的，正确率如下表

	test 1	test 2	test 3	average
k-means	0.95	0.53	0.97	0.82
混合高斯模型	0.95	0.96	0.96	0.96

可以看出，大多数情况向下聚类的效果都是好的，对于好的分类效果如 figure2 所示，可以看出 k-means 和混合高斯模型都能得到比较好的结果

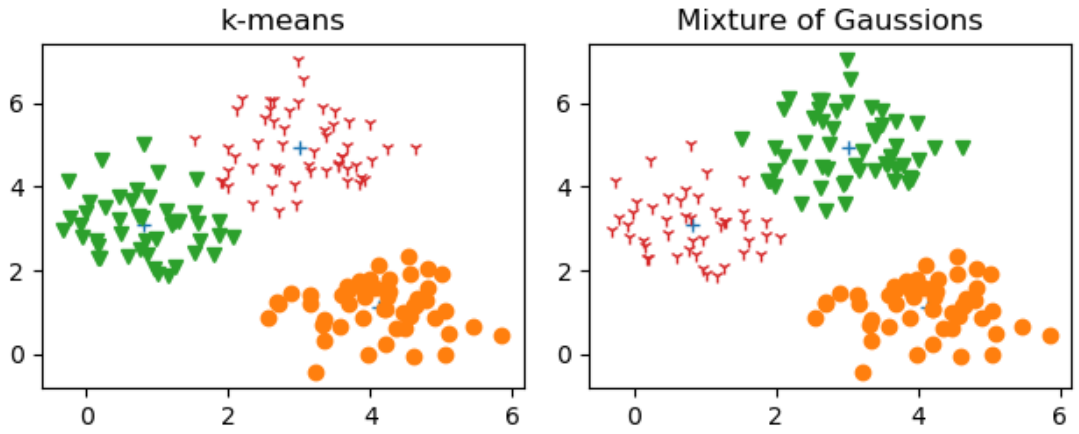

figure2:k-means 和混合高斯模型的结果

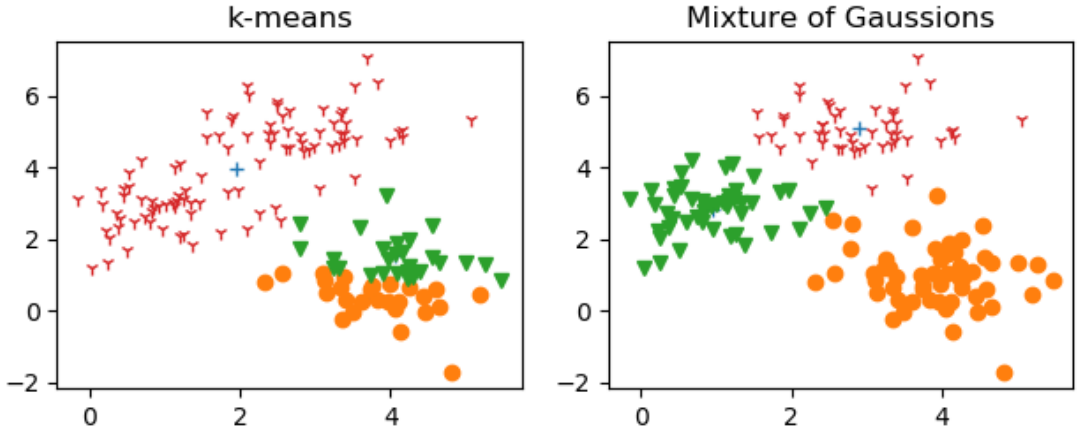

figure3:k-means 出现局部最优的一种情况

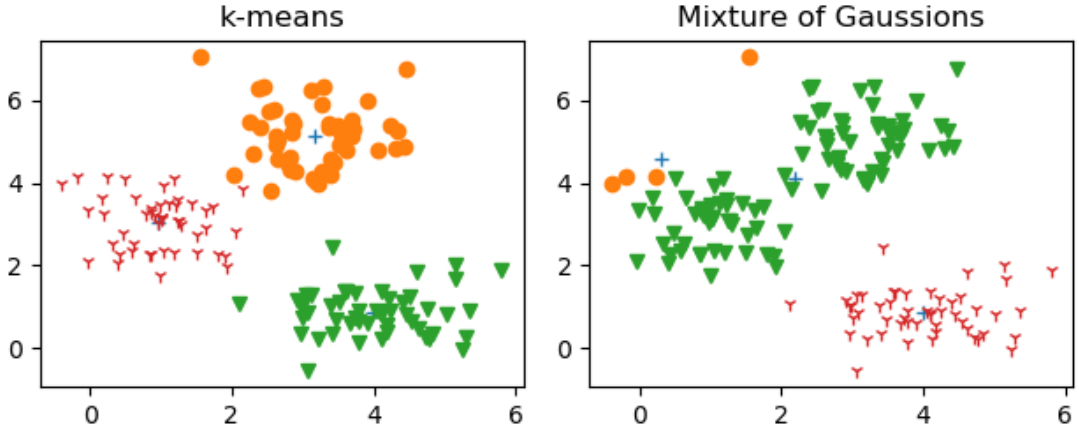

figure4:混合高斯模型出现局部最优的一种情况

然而对于 k-means 和混合高斯模型，因为初始均值点随机，所以就有可能出现局部最优的情况。如 test2 中的 k-means 得到的正确率，以及 figure3 和 figure4 中展示 k-means 和混合高斯模型出现的局部最优解，可以看出正确率只有大约 50%。这是算法的局限性导致的。其实这种情况可以通过优化初始点的方法来解决。

对于混合高斯模型，可以计算出每次迭代的似然值变化来判断是否 EM 算法再往正确的结果靠拢对给出的测试样例，这里迭代 20 次来进行观察，得到的折线图如 figure5 所示

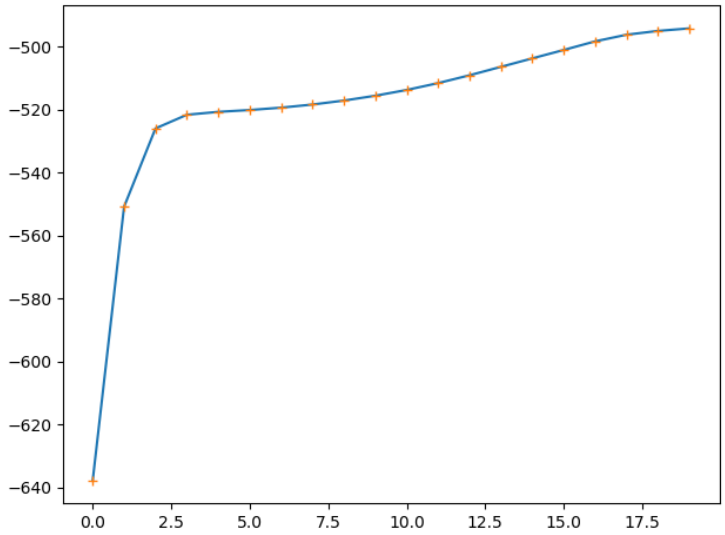

figure5:混合高斯模型 20 次迭代得到的似然值折线图

可以看出，似然值是在不断增大而且趋向于一个平稳的值的，所以可以认为混合高斯模型的 EM 算法估计得到了一个优化解。

2. UCI 上数据的测试

考虑在 UCI 上选择一组数据来进行测试。这里选择的是一个具有 210 个样本的数据（文件夹中的 experiment3_data），数据有 7 个特征值，每个特征值对应的属性如下图所示：

Attribute Information:

To construct the data, seven geometric parameters of wheat kernels were measured:
1. area A,
2. perimeter P,
3. compactness C = 4*pi*A/P^2,
4. length of kernel,
5. width of kernel,
6. asymmetry coefficient
7. length of kernel groove.
All of these parameters were real-valued continuous.

数据期望被分成三类。对数据进行处理，将期望结果去掉，然后分别用 k-means 和混合高斯模型读取这些数据进行聚类，其中由于混合高斯模型迭代多次后会出现负溢出，所以只迭代十次，得到的正确率结果如下表所示

	test 1	test 2	test 3	average
k-means	0.88	0.90	0.90	0.89
混合高斯模型	0.64	0.90	0.60	0.71

可以看出，k-means 得出的结果正确而稳定，而混合高斯模型因为初始点选取随机而且能

迭代次数较少，所以会出现正确率忽高忽低的情况，但是说明还是能够得到正确的结果的。

五、结论

1. k-means 算法和混合高斯模型通过 EM 算法估计参数来聚类能够获得比较好的结果；
2. k-means 算法和混合高斯模型聚类有时会陷入局部最优解，而且出现这种解的原因往往和初始点的选择有关；
3. 实际数据中 k-means 和混合高斯模型也能获得比较好的结果，但一些数据情况下混合高斯模型的协方差阵会特别小影响迭代；
4. 混合高斯模型 EM 迭代过程中似然函数的值是在不断增大的，但是会趋向于一个平稳的值，说明混合高斯模型得到的优化解是是收敛的，但是不一定是全局最优解。

六、参考文献

[1] http://archive.ics.uci.edu/ml/datasets/seeds#

[2] https://en.wikipedia.org/wiki/K-means_clustering

[3] https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm

[4] https://en.wikipedia.org/wiki/Mixture_model

七、附录：源代码（带注释）

已在附录中提交