模式识别与机器学习 Assignment-3 报告

生成数据

见 source.py 中的 generate_data() 函数。
生成了由三个正态分布组成的数据。

构建GMM模型

GMM模型进行聚类的思路类似k-mean算法。

GMM假定大小为n的数据集是由K个正态分布构成的，则GMM的概率密度函数为：

$$p(x)=\sum_{k=1}^{K}p(k)p(x|k)=\sum_{k=1}^{K}\pi_kN(x|\mu_k,\sigma_k)$$

每一次循环我们求出对于当前的GMM模型，每个数据x_i由正态分布k生成的概率$\gamma(i,k)$, 由贝叶斯公式可以得到：

$$\gamma(i,k)=\frac{\pi_kN(x|\mu_k,\sigma_k)}{\sum_{j=1}^{K}\pi_jN(x|\mu_j,\sigma_j)}$$

根据γ我们可以求出新的π,μ,σ. 令$n_k=\sum_{i=1}^{n}\gamma(i,k)$, 不难得出$\pi_k=n_k/n$, 同时由定义可得：

$$\mu_k=\frac{\sum_{i=1}^{n}\gamma(i,k)x_i}{n_k}$$

$$\sigma_k=\frac{\sum_{i-1}^{n}\gamma(i,k)(x_i-\mu_i)(x_i-\mu_i)^{\mathrm{T}}}{n_k}$$

重复上述迭代直到其收敛，在实际实现中我们指定迭代次数为200次。

效果

当参数选择很好（即数据由三个正态分布构成，同时将K设置为3）时效果很好，迭代两百次以后可以基本完美聚类。可是当$K \neq 3$时效果非常差，这就是DMM模型的局限性。

运行

```
python source.py
```