基于高斯混合模型的二分类

一、K-Means 聚类算法

算法思想：以空间中 K 个点为中心进行聚类，对最靠近他们的对象进行归类。通过多次迭代，逐次更新各聚类中心的值，直到达到最好的聚类结果。

目标：最小化每个簇中样本与聚类中心的距离

K：K 个簇

Means：簇中心为簇所含的值的均值

K-Means 算法流程：

Algorithm 1 Algorithm of K-Means.

Input: dataset $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_m\}$; cluster number k.

Initialization: select k sample points from D as the initialized central points $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k\}$;
repeat
  let $C_i = \varnothing (1 \leqslant i \leqslant k)$;
  for $j = 1, 2, \dots, m$ **do**
    calculate the distance between $\boldsymbol{x}_j$ and cluster central points $\mu_i (1 \leqslant i \leqslant k)$: $d_{ji} = \|\boldsymbol{x}_i - \boldsymbol{\mu}_i\|_2$;
    determine the cluster index of $\boldsymbol{x}_j$ according to the minimum distance: $\lambda_j = \arg\min_{i \in \{1,2,\dots,k\}} d_{ji}$;
    divide the sample point $\boldsymbol{x}_j$ to the corresponding cluster: $C_{\lambda_j} = C_{\lambda_j} \cup \{\boldsymbol{x}_j\}$;
  end for
  for $i = 1, 2, \dots, k$ **do**
    calculate the new cluster central points: $\boldsymbol{\mu}'_i = \frac{1}{|C_i|} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$;
    if $\boldsymbol{\mu}'_i \neq \boldsymbol{\mu}_i$ **then**
      update the current cluster central points $\boldsymbol{\mu}_i$ to $\boldsymbol{\mu}'_i$;
    else
      keep the current cluster central points;
    end if
  end for
until the max iteration time or the cluster central points not updated.

Output: the divided clusters $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$.

可选项：

1、初始化：

（1）随机选取 K 个点作为初始的类簇中心点

（2）选择批次距离尽可能远的 K 个点

（3）先对数据用层次聚类算法或 Canopy 算法进行聚类，再从得到的 K 个簇中选择簇中心点

2、迭代条件：达到最大迭代次数；簇中心点不再更新

3、距离度量：欧式距离；曼哈顿距离；余弦距离…

K-means 算法的缺陷：

1. K 值需要预先指定
2. 聚类效果对初始选取的聚类中心敏感

优化算法：bisecting K-Means，K-Means++

库调用：

```
class sklearn.cluster.KMeans(
    n_clusters=8,
    init='k-means++',
    n_init=10,
    max_iter=300)
```

属性：

cluster_centers_：簇中心点

labels_：每个样本点的分类

inertia_：每个点到其簇的质心的距离之和

功能：fit，predict，score

二、混合高斯

使用复杂模型来改进实验三中的简单高斯分类器，即高斯混合模型（GMM）来对每个类建模。同样地，建设每个高斯函数都有对角协方差矩阵，使用 K-Means 方法初始化 GMM，然后基于 EM 算法对 GMM 模型进行迭代改进。测试含有 2、4、8 个混个高斯函数的 GMM 模型效果。

GMM-EM 算法推导（参考：Pattern Recognition and Machine Learning）：

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

$$\mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)=\frac{1}{(2\pi)^{D/2}}\frac{1}{\left|\boldsymbol{\Sigma}_k\right|^{1/2}}\exp\left\{-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_k\right)^{\mathrm{T}}\boldsymbol{\Sigma}_k^{-1}\left(\mathbf{x}-\boldsymbol{\mu}_k\right)\right\}$$

I）$\boldsymbol{\mu}_k$ 的推导

$$\begin{aligned}\frac{\partial}{\partial\boldsymbol{\mu}_k}\ln p(\mathbf{X}|\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\pi})&=\sum_{n=1}^{N}\frac{\pi_k}{\sum_{j=1}^{K}\pi_j\mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)}\frac{\partial\mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\partial\boldsymbol{\mu}_k}\\&=-\sum_{n=1}^{N}\frac{\pi_k\mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K}\pi_j\mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)}\boldsymbol{\Sigma}_k^{-1}\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)\\&=-\sum_{n=1}^{N}\gamma_{nk}\boldsymbol{\Sigma}_k^{-1}\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)\\&=0\end{aligned}$$

$$\sum_{n=1}^{N}\gamma_{nk}\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)=\sum_{n=1}^{N}\gamma_{nk}\mathbf{x}_n-N_k\boldsymbol{\mu}_k=0$$

$$\boldsymbol{\mu}_k=\frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}\mathbf{x}_n$$

$$\text{where：}\quad N_k=\sum_{n=1}^{N}\gamma_{nk}$$

II） $\boldsymbol{\Sigma}_k$ 的推导

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\Sigma}_k}\ln p(\mathbf{X}|\ \boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\pi}) &= \sum_{n=1}^{N}\frac{\pi_k}{\sum_{j=1}^{K}\pi_j\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)}\frac{\partial\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\partial\boldsymbol{\Sigma}_k} \\
&= \sum_{n=1}^{N}\frac{\pi_k\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K}\pi_j\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)}\frac{\partial\ln\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\partial\boldsymbol{\Sigma}_k} \\
&= \sum_{n=1}^{N}\gamma_{nk}\left(\frac{1}{2}\boldsymbol{\Sigma}_k-\frac{1}{2}\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)^{\mathrm{T}}\right) \\
&= 0
\end{aligned}$$

$$\boldsymbol{\Sigma}_k=\frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)\left(\mathbf{x}_n-\boldsymbol{\mu}_k\right)^{\mathrm{T}}$$

III） π_k 的推导

拉格朗日函数：

$$\ln p(\mathbf{X}|\ \boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\pi})+\lambda\left(\sum_{k=1}^{K}\pi_k-1\right)$$

$$s.t.\quad \sum_{k=1}^{K}\pi_k=1$$

求偏导：

$$\sum_{n=1}^{N}\frac{\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K}\pi_j\mathcal{N}\left(\mathbf{x}_n|\ \boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)}+\lambda=0$$

两边同乘 π_k 再对 k 求和，得

$$\lambda=-N$$

代入拉格朗日函数：

$$\sum_{n=1}^{N}\frac{\gamma_{nk}}{\pi_k}-N=\frac{N_k}{\pi_k}-N=0$$

$$\hat{\pi}_k=\frac{N_k}{N}$$

EM 算法流程：

Algorithm 2 Algorithm of GMM.

Input: dataset $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$; Gaussian models number k

Initialization: use K-Means algorithm to initialize GMM model parameters: means $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k\}$; covariances $\{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \ldots, \boldsymbol{\Sigma}_k\}$; mixing coefficients $\{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_k\}$, where $\pi_k = \frac{N_k}{N}, N_k = \sum_{n=1}^{N} \gamma_{nk}$;

repeat

  for $n = 1, 2, \ldots, N$ **do**

    for $j = 1, 2, \ldots, K$ **do**

      E-step: calculate the posterior probability of the k-th gaussian model for observed data $\boldsymbol{x}_n$ according to the current GMM model parameters: $\gamma_{nk} = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$;

    end for

  end for

  for $k = 1, 2, \ldots, K$ **do**

    M-step: update GMM model parameters:

    $\hat{\boldsymbol{\mu}}_k$: $\hat{\boldsymbol{\mu}}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} \boldsymbol{x}_n$;

    $\hat{\boldsymbol{\Sigma}}_k$: $\hat{\boldsymbol{\Sigma}}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$;

    $\hat{\pi}_k$: $\hat{\pi}_k = \frac{N_k}{N}$;

  end for

until the max iteration time or the likelihood function restrained.

Output: the established Gaussian Mixture Model parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \pi$.

输出模型参数：

各高斯函数混合系数π_k([k])，均值$\boldsymbol{\mu}_k$([k，3])，协方差$\boldsymbol{\Sigma}_k$([k，3，3])。