

1 混合模型

1.1 K 均值法

1.1.1 K 均值法的原理

假设我们需要把数据集 $\mathcal{D} : \{\mathbf{x}_n\}$ 中的数据点分到 K 个组. 直观地讲, 我们希望决定一个点分到某个组中时, 该点与该组中其他点的距离尽可能近, 而与其他组中点的距离尽可能远. 于是我们需要找到能衡量这一距离的指标.

聚类分析的一种思路是寻找一些原型点 (**Prototype points**) $\{\boldsymbol{\mu}_k\}$ 代表每个组. 计算任一数据点 \mathbf{x}_n 与第 k 个组的距离时, 只需计算 $\|\mathbf{x}_n - \boldsymbol{\mu}_k\|$ 即可. K 均值法就采用这样的思路, 用二乘误差衡量模型的误差, 就得到下面的畸变函数 (**Distortion function**):

$$J(\mathbf{R}, \{\boldsymbol{\mu}_k\}) = \sum_{n=1}^N \sum_{k=1}^K \mathbf{R}_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

其中 \mathbf{R}_{nk} 是第 n 个数据点分组结果的独热编码, 如果 \mathbf{x}_n 被分到第 k 组, 则 $\mathbf{R}_{nk} = 1$, 否则 $\mathbf{R}_{nk} = 0$. 于是, 聚类问题就转化为如下最小化问题:

$$\arg \min_{\mathbf{R}, \boldsymbol{\mu}_k} J(\mathbf{R}, \{\boldsymbol{\mu}_k\})$$

畸变函数 J 类似于监督学习中的误差函数.

求解上述最小化问题时, 我们可以采用交替优化 (**Alternating optimization**) 的方法, 具体步骤为:

1. 固定 $\{\boldsymbol{\mu}_k\}$, 求 \mathbf{R} 使得 $J(\mathbf{R}, \{\boldsymbol{\mu}_k\})$ 最小化. 不难看出, 只需对于每个数据点 \mathbf{x}_n 找到与其距离最近的原型点 $\boldsymbol{\mu}_k$, 并将 \mathbf{x}_n 分到对应的组中 (即令 $\mathbf{R}_{nk} = 1$) 即可.
2. 固定 \mathbf{R} , 求 $\{\boldsymbol{\mu}_k\}$ 使得 $J(\mathbf{R}, \{\boldsymbol{\mu}_k\})$ 最小化. 对每个原型点 $\boldsymbol{\mu}_k$, 不难看出当其取聚类中所有点的均值时, 畸变函数 J 取得最小值. 即:

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

这也是 K 均值法名称的由来.

3. 不断重复步骤 1 和 2, 直到畸变函数 J 收敛 (即不再发生变化). 可以证明 J 在此过程中单调下降, 但不一定收敛到全局最小值. 因此, 实际应用中通常需要多次运行 K 均值算法, 每次重新随机初始化, 并选择畸变函数 J 最小的结果作为最终结果.

对于在线学习的情形, 我们可以采用在线 K 均值算法 (**Online K-means algorithm**). 每次只处理一个数据点 \mathbf{x}_n , 并根据该点更新对应的原型点 $\boldsymbol{\mu}_k$. 具体地, 对于每个数据点 \mathbf{x}_n , 我们首先找到与其距离最近的原型点 $\boldsymbol{\mu}_k$, 然后根据如下公式更新该原型点:

$$\boldsymbol{\mu}_k \leftarrow \boldsymbol{\mu}_k + \eta_n (\mathbf{x}_n - \boldsymbol{\mu}_k)$$

其中 η_n 是学习率, 通常取 $\eta_n = \frac{1}{s_k}$, 其中 s_k 是到目前为止被分到第 k 组的数据点数量. 这种更新方式等价于将 μ_k 更新为当前所有被分到第 k 组的数据点的均值.

1.1.2 超参数 K 的选取

不难看出 K 均值法只有一个超参数 K . 聚类作为非监督学习不能通过已知的标签辅助选择 K . 除去通过实际问题的情况选择 K 外, 可以使用一些辅助指标选择 K . 一种常用的方法是肘部法则 (**Elbow method**). 具体地, 我们可以计算不同 K 值下畸变函数 J 的值, 并绘制 J 随 K 变化的曲线. 通常情况下, 随着 K 的增加, 畸变函数 J 会减小, 但减小的幅度会逐渐变小. 当曲线出现明显的“肘部”时, 对应的 K 值就是一个较好的选择.

1.2 高斯混合模型

1.2.1 高斯混合模型的原理

K 均值法的优势在于算法简单易行, 执行高效, 但它的主要缺点是它对于聚类中心平均值的使用太单一, 倾向于认为组内数据的分布呈球形. 这可能在某些数据分布复杂的情况下表现不佳. 为了解决这个问题, 我们可以使用更复杂的模型来描述数据的分布, 例如高斯混合模型 (**Gaussian Mixture Model, GMM**).

高斯混合模型假设 \mathbf{x} 的分布是多个高斯分布的加权和. 具体而言假设有 K 个高斯分布, 每个高斯分布有其均值 μ_k 和协方差矩阵 Σ_k , 以及对应的混合系数 π_k , 满足 $\sum_{k=1}^K \pi_k = 1$, 则高斯混合模型的概率密度函数为:

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)$$

每个高斯分布代表聚类的一个组别. 引入隐藏变量 \mathbf{z} , 其第 k 个分量 z_k 取 1 时表示 \mathbf{x} 属于第 k 个组别 (对于给定的数据点而言就是独热编码). 于是可知

$$p(z_k = 1) = \pi_k, \quad p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)$$

于是根据 Bayes 公式可知:

$$p(\mathbf{x}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$$

现在我们考虑如何求解高斯混合模型的参数 $\{\pi_k, \mu_k, \Sigma_k\}$. 如果 \mathbf{z} 是已知的, 那么可以很容易地根据多元高斯分布的性质求解 π_k, μ_k, Σ_k 的值. 然而实际上 \mathbf{z} 是未知的 (这正是聚类的结果), 因此我们需要使用期望最大化 (**Expectation-Maximization, EM**) 算法来估计参数.

简单而言, 首先对于每个 \mathbf{x}_n 估计 \mathbf{z}_n 的值, 然后基于 $\mathbf{x}_n, \mathbf{z}_n$ 求解模型参数; 然后用求出的模型参数重新推断 \mathbf{z}_n , 如此迭代直到收敛为止.

1. 初始化参数 π_k, μ_k, Σ_k .

2. E 步骤: 根据猜测的 \mathbf{z}_n , 记 $\gamma_k(\mathbf{z}_n)$ 为数据点 \mathbf{x}_n 属于第 k 个聚类的概率, 则属于第 k 个聚类的数据数目为

$$N_k = \sum_{n=1}^N \gamma_k(\mathbf{z}_n)$$

基于多元高斯分布的性质可得

$$\begin{aligned} \boldsymbol{\mu}_k &= \frac{\sum_{n=1}^N \gamma_k(\mathbf{z}_n) \mathbf{x}_k}{\sum_{n=1}^N \gamma_k(\mathbf{z}_n)} = \frac{1}{N_k} \sum_{n=1}^N \gamma_k(\mathbf{z}_n) \mathbf{x}_n \\ \boldsymbol{\Sigma}_k &= \frac{\sum_{n=1}^N \gamma_k(\mathbf{z}_n) (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^t}{\sum_{n=1}^N \gamma_k(\mathbf{z}_n)} = \frac{1}{N_k} \sum_{n=1}^N \gamma_k(\mathbf{z}_n) (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^t \\ \pi_k &= \frac{N_k}{N} \end{aligned}$$

3. M 步骤: 基于更新后的参数, 重新计算 $\gamma_k(\mathbf{z}_n)$. 根据 Bayes 公式可得

$$\gamma_k(\mathbf{z}_n) = p(\mathbf{z}_k = 1 | \mathbf{x}_n) = \frac{p(\mathbf{x}_n | \mathbf{z}_k = 1) p(\mathbf{z}_k = 1)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

4. 重复 E 步骤和 M 步骤, 直到参数收敛.