CompBio Homework2

陈昭熹 2017011552

2020 年 3 月 21 日

目录

1 层次聚类

数据经过预处理后 (主要是将行列互换使得两个文件中的表头对齐), 可以绘制出如下层次聚类结果。

1.1 树状图

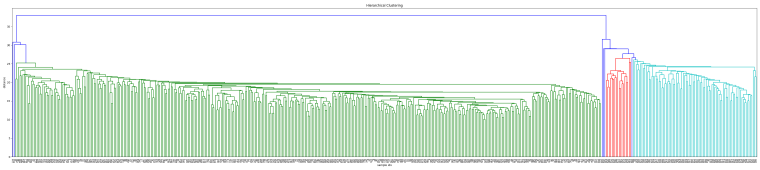

图 1: Hierarchy Cluster

利用该树进行二分类, 并使用 $ER\_Status\_nature2012$ 作为分类标准, 评估分类准确率为:

$$Accuracy = 93.68\% \tag{1}$$

1.2 Heatmap

利用 seaborn 库中的聚类方法，以 Gene 为横轴，Patients 为纵轴绘制出 Heatmap 如下，可以看出，病人被明显的分为了两类，这为后面的分析提供了便利。

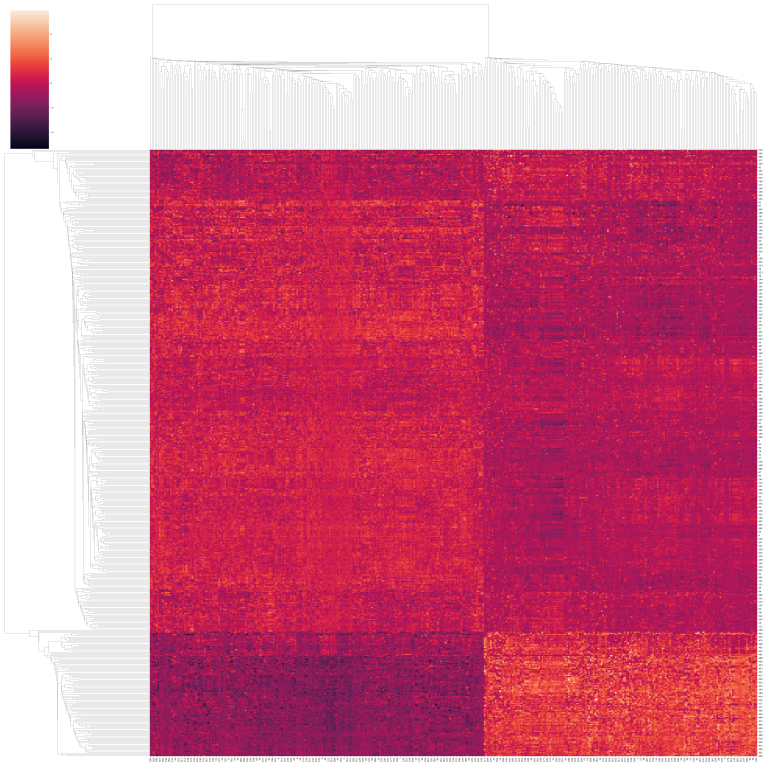

图 2: HeatMap

2 PCA 降维

2.1 实现思路

主要是依据 PCA 的数学原理进行实现，首先计算协方差阵，后计算协方差阵特征值和特征向量，并按照特征值从大到小排序，根据特征值大小选择主成分，从而确定变换矩阵。利用变换矩阵完成原始数据的降维。

2.2 主成分选取依据

在本次作业的数据中，使用上述方法后，输出前 5 个主成分所对应的特征值如下:

$$[190.72474391, 11.58899724, 8.35373783, 6.27501677, 5.33837007] \quad (2)$$

可以看出第一个主成分远远超出后面的成分，代表着以第一个主成分为特征时就可以将数据较好的区分开，因此选取第一个主成分表示原数据特征，将原始数据降至一维。

2.3 降维后聚类效果

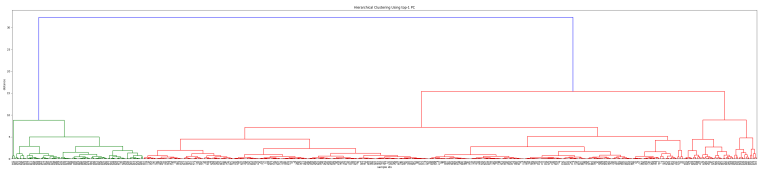

图 3: Hierarchy Cluster with TOP1 Principle Component

可以直观的看到，降维后层次聚类明显迭代次数更少了。

利用该树进行二分类，评估分类准确率较比使用全部特征略有降低，但仍具有高置信度表现:

$$Accuracy_{pca} = 90.80\% \tag{3}$$

3 PCA 原理

下面推导 PCA 数学原理，从两个角度进行推导。首先进行一些定义。

PCA 的主要目的是在低维空间中寻找一组正交基 $[u_1, \ldots, u_M]$，使得数据在这组正交基形成的超平面上的投影方差最大。我们定义 $U = [u_1, \ldots, u_M]$，同时数据协方差定义为:

$$S = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})(x_n - \bar{x})^T$$

3.1 Maximum Variance

在最大化方差思路中，我们目标是最大化:

$$J = \frac{1}{N}||U^T x_n - U^T \bar{x}||^2 \tag{4}$$

将其中的正交基展开可以有如下形式:

$$J = \frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}||u_m^T x_n - u_m^T \bar{x}||^2 = \sum_{m=1}^{M}\frac{1}{N}\sum_{n=1}^{N}||u_m^T x_n - u_m^T \bar{x}||^2 \tag{5}$$

$$\Rightarrow J = \sum_{m=1}^{M} u_m^T S u_m \tag{6}$$

下面求解 J_m = 最大时对应的 u_m 满足什么条件，即转化为如下带约束的优化问题:

$$\begin{aligned} &argmin\ J_i = u_i^T S u_i \\ &s.t. \quad u_i^T u_j = \delta_{ij} \end{aligned} \tag{7}$$

利用拉格朗日乘子法，构造函数:

$$L = \sum_{m=1}^{M} u_m^T S u_m + \lambda_i(1 - u_i^T u_i) \tag{8}$$

求解 $\frac{\partial L}{\partial u_i} = 0$，有:

$$S u_i = \lambda_i u_i \tag{9}$$

将上式结果带回目标中得到:

$$J = \sum_{m=1}^{M} \lambda_m \tag{10}$$

因此若想要目标 J 取得最大值，则需要选取协方差阵 S 前 m 个最大的特征值，故相应的转换矩阵中对应着前 m 个最大的特征值的特征向量。

3.2 Minimun Error

假设一组正交基构成 M 维空间，$\{u_i\}, i \in D$，用 x_k 表示原 D 维空间中的 k 个点，z_k 表示其在空间 M 上的投影，则

$$z_k = \sum_{i=1}^{M}(u_i^T x_k)u_i \tag{11}$$

而为了最小化误差，我们的优化目标变成了;

$$\begin{aligned} &argmin \sum_{i=1}^{k} ||x_k - z_k||^2 \\ &s.t. \quad u_i^T u_j = \delta_{ij} \end{aligned} \tag{12}$$

类似的利用拉格朗日乘子法，我们可以得到：

$$argmin \sum_{i=M+1}^{D} u_i^T S u_i \tag{13}$$

而上式的约束恰恰与上一节中的推导相吻合，此处要求最小化剩余的维数，那么就应当使得前 m 维对应的特征值最大。

4 EM & GMM

EM 算法 (最大期望) 是一种寻找概率模型中寻找参数最大似然估计或最大后验估计的算法。主要有两个步骤:

E 步骤

$$Q(\Theta, \Theta^{(i-1)}) = E(logp(X, Y|\Theta)|X, \Theta^{(i-1)})$$

M 步骤

$$\Theta^{(i)} = argmaxQ(\Theta, \Theta^{(i-1)})$$

4.1 EM of GMM

首先该算法是一个迭代算法，因此初始值可以是一个估计，也可以是随机值 Θ，并开始迭代直至收敛。混合高斯模型中需要有三个参数进行估计:

$$p(x|\pi, \mu, \epsilon) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \epsilon_k) \tag{14}$$

则利用 EM 算法估计这三个参数的步骤如下：

E 步骤

$$Q(\Theta, \Theta^{i-1}) = \sum_{k=1}^{M}\sum_{i=1}^{N} log(\pi_k)p(k|x_i, \Theta^{i-1}) + \sum_{k=1}^{M}\sum_{i=1}^{N} log(p_k(x_i|\theta_k))p(k|x_i, \Theta^{i-1})$$

利用后验概率 $P_i = p(k|x_i, \Theta^{i-1})$ 估计新的参数值:

M 步骤

$$\pi_k^i = \frac{1}{N}\sum_{i=1}^{N} P_i$$

$$\mu_k^i = \frac{\sum_{i=1}^N x_i P_i}{\sum_{i=1}^N P_i}$$

$$\epsilon_k^i = \frac{\sum_{i=1}^N P_i (x_i - \mu_k^i)(x_i - \mu_k^i)^T}{\sum_{i=1}^N P_i}$$

4.2 K-means 与 GMM 联系

两种算法都是聚类迭代算法，但在每一次迭代中，K-means 给出一个严格的分类，而 GMM 则给出模糊的分类（更倾向于是类别的概率）。从模型上，可以说 K-means 是一种特殊情况下的 GMM 模型，其中额外约束是其混合组分的协方差形如 ϵI，其中 $\epsilon \to 0$