使用R进行凝聚聚类分析

Alex / 2024-03-24 / free_learner@163.com / AlexBrain.cn

本文介绍在R中进行凝聚聚类分析（Agglomerative Clustering Analysis）的基本方法。

一、凝聚聚类的基本原理

聚类算法的目标一般是为了将样本划分为不同的类别（Cluster），使得同一类别内的样本尽可能相似，不同类别间的样本尽可能不同。凝聚聚类分析是一种自下而上的层次聚类（Hierarchical Clustering）分析方法：首先每个样本当做一个类别，然后不同类别根据相似性进行合并形成新的类别，直到只剩下一个类别，从而形成一个类别的层次结构（hierarchy）或者树状图（dendrogram）。具体的算法步骤包括：

1. 计算任意两个样本之间的不相似性（dis-similarity）或者距离（distance），常用的距离指标包括欧式距离（Euclidean）和曼哈顿距离（Manhattan）。
2. 计算类别间的相似性，将最相似的两个类别合并成一个新的类别。在最开始每个样本就是一个类别。重复这个合并类别的过程直到所有样本都属于一个类别。两个类别间的相似性通过连接函数（linkage function）来衡量，常用的连接函数包括：
    - Complete linkage (Maximum linkage):（两个类别的距离定义为）两个类别中不同样本间最大的距离。
    - Single linkage (Minimum linkage): 两个类别中不同样本间最小的距离。
    - Mean linkage: 两个类别中不同样本间距离的平均值。
    - Centroid linkage: 两个类别中心的距离，中心就是每个类别中样本的均值。
    - Ward linkage: 合并前后平方和（Sum of Squares）的差异。

二、使用R进行凝聚聚类分析

这里我们使用R自带的`iris`数据集，该数据集包含5个变量，前4个变量分别表示花萼长度、花萼宽度、花瓣长度、花瓣宽度，第5个变量表示花的种类（共三类）。

```
> data("iris")
> str(iris)
'data.frame':    150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1
1 ...
```

为了让结果看起来更简洁，这里在每种类别中随机抽取10个样本，作为聚类分析的测试数据。

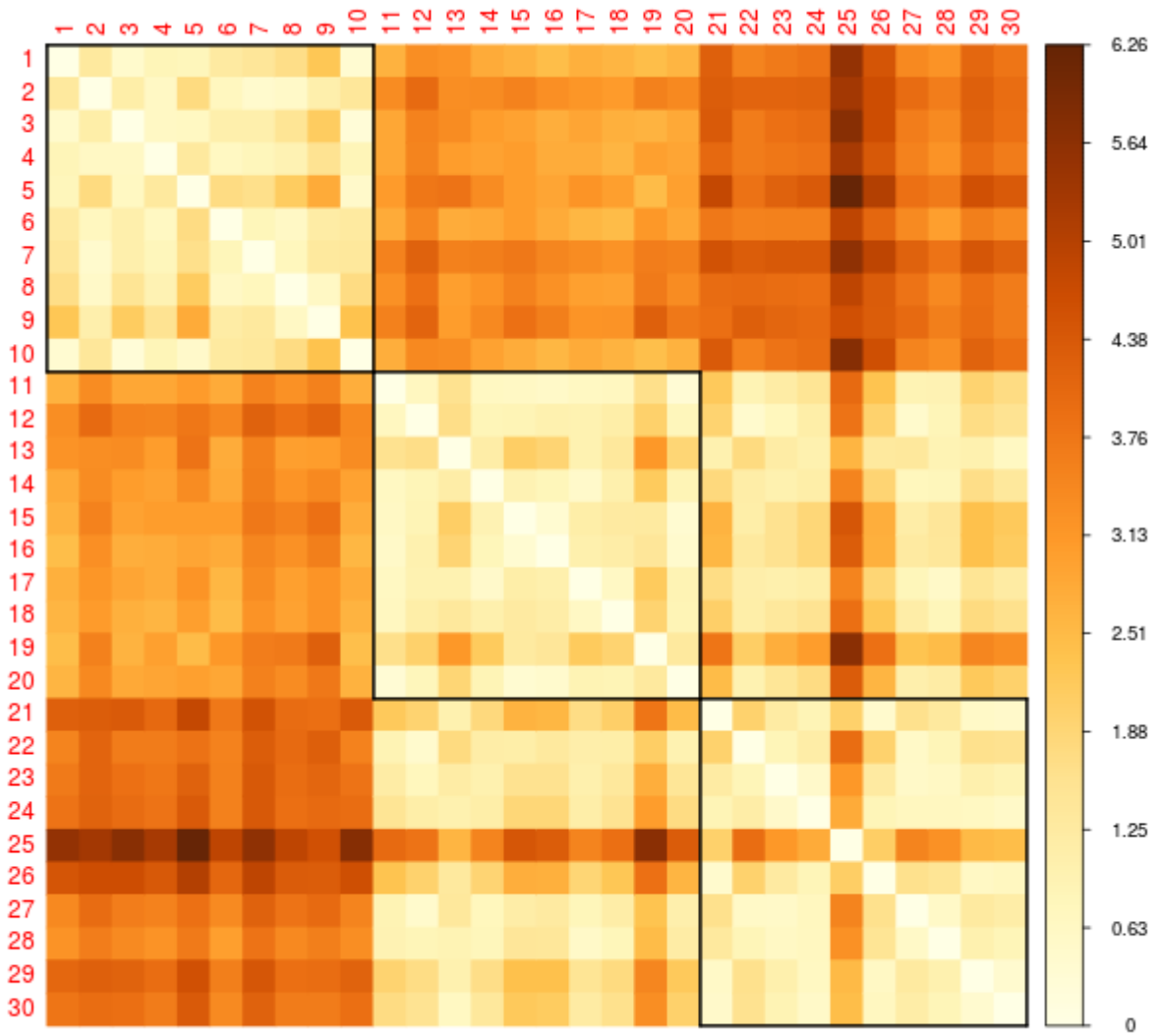
```
> ## Select 10 samples from each species
> set.seed(100)
> test_dat <- NULL
> for (curr_grp in levels(iris$Species)){
+   curr_dat <- iris[iris$Species == curr_grp, ]
+   in_idx <- sample(1:nrow(curr_dat), 10)
+   test_dat <- rbind(test_dat, curr_dat[in_idx, ])
+ }
> row.names(test_dat) <- c(1:nrow(test_dat))
```

通常我们需要对原始变量进行标准化（均值为0，标准差为1），因为不同变量的尺度可能差别很大。

```
> ## Scaling
> test_dat[,c(1:4)] <- scale(test_dat[,c(1:4)], center = TRUE, scale = TRUE)
```

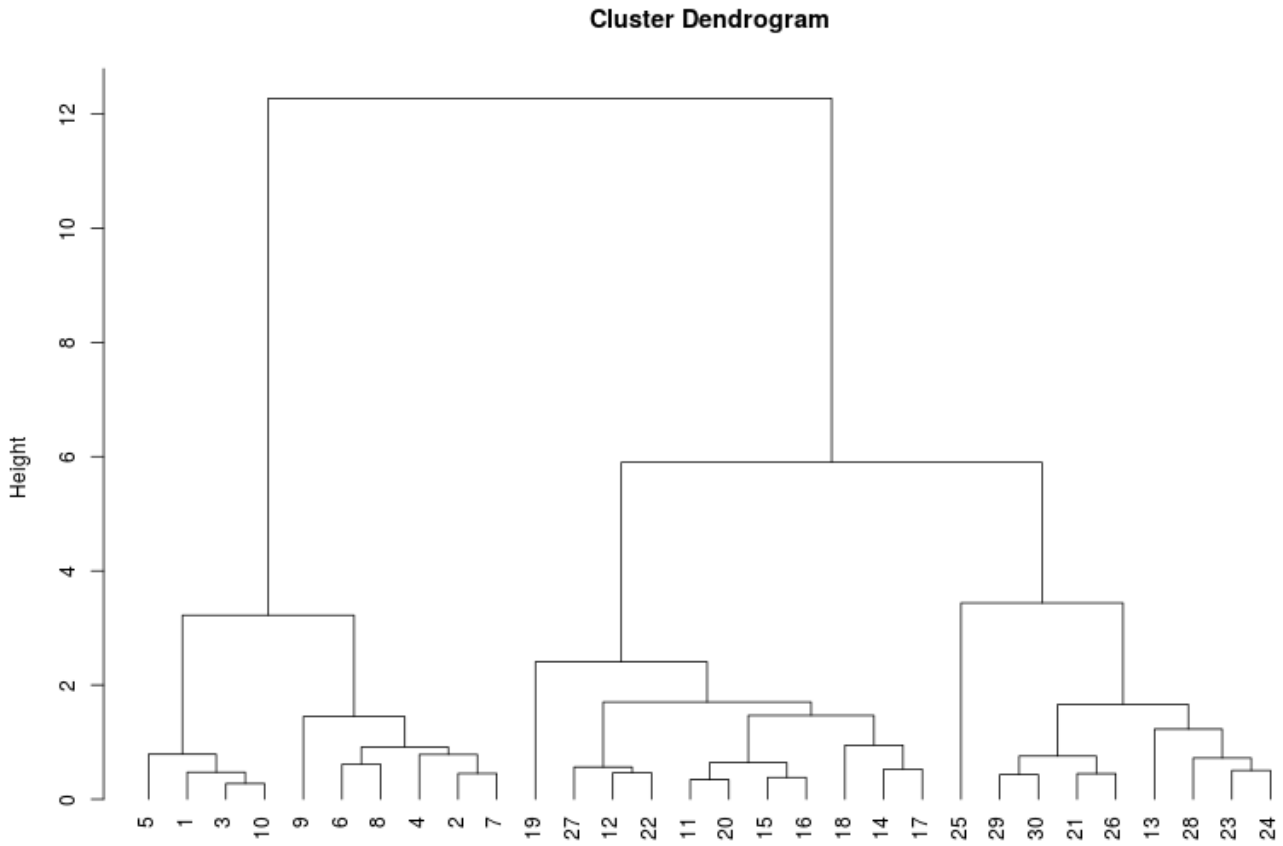
使用`dist`函数计算样本间的距离矩阵，并使用`corrplot`包对距离矩阵进行可视化。

```
> ## Calculate distance matrix
> dist_mat <- dist(test_dat[,c(1:4)], method = 'euclidean')
> corrRect(corrplot(as.matrix(dist_mat), is.corr=FALSE, method = 'color'), c(1,
11, 21,30))
```



使用`hclust`函数进行凝聚聚类分析，并进行可视化。这里连接函数使用的是`ward.D2`，使用不同连接函数，对结果影响很大。

```
> ## Agglomerative Clustering
> clust_dat <- hclust(dist_mat, method = 'ward.D2')
> plot(clust_dat, xlab="", sub="", hang=-1)
```



由于已知这个测试数据包含三种类别的花，所以我们根据聚类结果将样本分别三类，然后比较聚类分析得到的类别和真实的类别有多大区别。 从下面的结果可以看到，目前的分析对于versicolor和virginica不能进行完美的区分。

```
> ## Check the correspondence between clusters and original species
> test_dat$Cluster <- cutree(clust_dat, k = 3)
> table(test_dat$Cluster, test_dat$Species)

    setosa versicolor virginica
  1     10          0         0
  2      0          9         2
  3      0          1         8
```