多元统计第五次作业

蒋文馨

2020 年 6 月 22 日

目录

1 Agglomerative Hierarchical Clustering

Write a computer program to implement single-linkage, average-linkage, and complete-linkage agglomerative hierarchical clustering. Try it out on a data set of your choice.

Cluster the primate.scapulae data using single-linkage, average-linkage, and complete-linkage agglomerative hierarchical clustering methods. Find the five-cluster solutions for all three methods, which allows comparison with the true primate classifications. Find the misclassification rate for all three methods. Show that the lowest rate occurs for the complete-linkage method and the highest for the single-linkage method.

1.1 Agglomerative Hierarchical Clustering Function

分析：因为 3 种方法只是对类距离的度量不同，并没有本质的区别。所以尝试将 3 种方法封装在一个函数中，减少代码冗余。

- 函数输入:
  - dist: 距离矩阵，由 dist() 计算得到
  - n.cluster：聚类数目。默认为 1
  - ori.label：原始数据的标签
  - method：聚类方法
- 函数输出：返回结构体，其中 misclassifiction.rate 用于显示分类错误率，其他与 hclust 相同。

- 函数说明：
  - 使用 merge 记录每次合并的类
  - rcd 记录每个点所在类
  - 合并两个类时，将其中一个类对应距离矩阵的行列设置为 INF，另一类根据距离定义更新距离
  - h 和 iorder 用于画图，和作业无关
- 分类规则：聚类达到指定数目后，选该类所属标签的众数为整个类的标签

```
myclust = function(dist,
                   n.cluster = 1,
                   ori.label,
                   method = c("single", "complete", "average")) {
  d = as.matrix(dist)
  n = nrow(d)
  diag(d) = Inf
  cho.mthd = switch(
    match.arg(method),
    single   = min,
    complete = max,
    average  = weighted.mean
  )
  rcd = -(1:n) #record current group
  merge = matrix(NA, n - n.cluster, 2)
  h = rep(0, n - n.cluster)
  for (ii in 1:(n - n.cluster)) {
    # find min dist
    h[ii] = min(d)
    min.index = which(d == min(d), arr.ind = T)
    i = min.index[1, 1]
    j = min.index[1, 2]
    # merge i j and belongings to ii
    merge[ii,] = rcd[c(i, j)]

    if (method == "average") {
      prei = which(rcd == rcd[i])
      prej = which(rcd == rcd[j])
      ni = length(prei)
      nj = length(prej)
      pregroup = c(i, j, prei, prej)
      rcd[pregroup] = ii
      # recalculate dist
      d[i, ] = d[, i] = apply(d[c(i, j), ], 2,
                              cho.mthd, w = c(ni, nj) / (ni + nj))
    }
    else{
      pregroup = c(i, j, which(rcd == rcd[i]), which(rcd == rcd[j]))
      rcd[pregroup] = ii
      # recalculate dist
      d[i, ] = d[, i] = apply(d[c(i, j), ], 2, cho.mthd)
    }
    # rewrit [i,i] and delete j
    d[i, i] = d[, j] = d[j,] = Inf
  }

  # change type to 1:n.cluster
  typelist = unique(rcd)
  classify = vector(length = n)
  for (ii in 1:length(typelist)) {
    classify[which(rcd == typelist[ii])] = ii
  }

  # calculate misclassifiction rate
  clu.label = vector(length = length(classify))
  for (ii in 1:n.cluster) {
    temp = which(classify == ii)
    clu.label[temp] = mode(ori.label[temp])
  }
  misc.rate = sum(clu.label != ori.label) / n

  # build strct and return
  structure(
    list(
      merge = merge,
      height = h,
      order = iorder(merge),
      labels = ori.label,
      method = method,
      classify = classify,
      misclassifiction.rate = misc.rate,
      call = match.call(),
      dist.method = "euclidean"
    ),
    class = "hclust"
  )
}

# 求众数，用于判断聚类后的标签
mode = function(v) {
  uni = unique(v)
  uni[which.max(tabulate(match(v, uni)))]
}

# ref: https://github.com/bwlewis/hclust_in_R/blob/master/hc.R
# 用于画图，如果不画图可以不要这个函数
iorder = function(m)
{
  N = nrow(m) + 1
  iorder = rep(0, N)
  iorder[1] = m[N - 1, 1]
  iorder[2] = m[N - 1, 2]
  loc = 2
  for (i in seq(N - 2, 1))
  {
    for (j in seq(1, loc))
    {
      if (iorder[j] == i)
      {
        iorder[j] = m[i, 1]
        if (j == loc)
        {
          loc = loc + 1
          iorder[loc] = m[i, 2]
        } else
        {
          loc = loc + 1
          for (k in seq(loc, j + 2))
            iorder[k] = iorder[k - 1]
          iorder[j + 1] = m[i, 2]
        }
      }
    }
  }
  - iorder
}
```

1.2 Agglomerative Hierarchical Clustering with primate.scapulae data

数据 genus 列代表标签，Homo 的 gamma 列全部为 NA，故不能用于聚类，遂删去这两列。

```
# import data and calcu dist
library(readr)
primate_scapulae <- read_table2("primate.scapulae.txt")
# delete genus and gamma, than scale
data = primate_scapulae[, 2:8]
data = apply(data, 2, scale)
d = dist(data, method = "euclidean")

# cluster
n.clu = 5
mclst.si = myclust(d,
                   ori.label = primate_scapulae$class,
                   method = "single",
                   n.cluster = n.clu)
mclst.av = myclust(d,
                   ori.label = primate_scapulae$class,
                   method = "average",
                   n.cluster = n.clu)
mclst.cm = myclust(d,
                   ori.label = primate_scapulae$class,
                   method = "complete",
                   n.cluster = n.clu)
```

```
## [1] "Misclassification rate"
## [1] "single: 0.40952380952381"
```

```
## [1] "average: 0.266666666666667"
```

```
## [1] "complete: 0.180952380952381"
```

由聚类分析的结果可知，single 的错误率最高为 0.41，complete 的错误率最低为 0.18。绘图结果中，在同一个框表示分到同一类。

2 Appendix

作业花絮：

我在 GitHub 上用关键词 “cran Agnes” 和 “cran hclust” 搜索，尝试寻找这两个函数的源代码（主要是想试着复现。虽然根据我的经验，R 函数的实现代码我基本是看不懂的）。但是我只找到了其他用户自己写的代码：https://github.com/bwlewis/hclust_in_R/blob/master/hc.R（这人主页看起来很大佬的样子！）

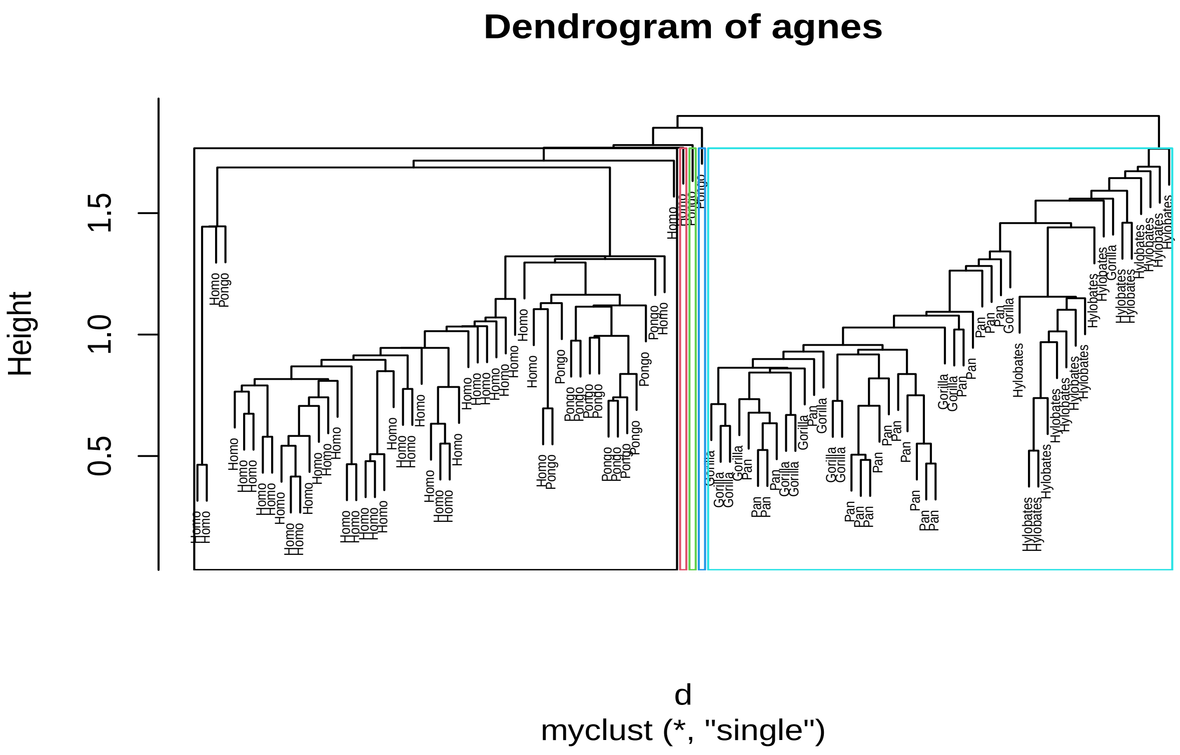
和作业相关的代码我都是自己写的，除了 “cho.mthd=switch(...)” 这步。因为之前没用过 switch 函数，担心出错才对着抄，顺手把人家的 “average = mean” 也直接抄了。写完后和 hclust 的结果一对比，发现 3 种方法就 average 不一样，并且 average 错误率比 complete 还小，与题目矛盾。在 “大佬不会错那就是我错了” 和 “我的代码没 bug” 之间徘徊了非常久，最后看公式发现取均值要带个类内点数的权重。然后给大佬提了个 issue。这是我 GitHub 第一个正式的 issue 想想还挺激动。

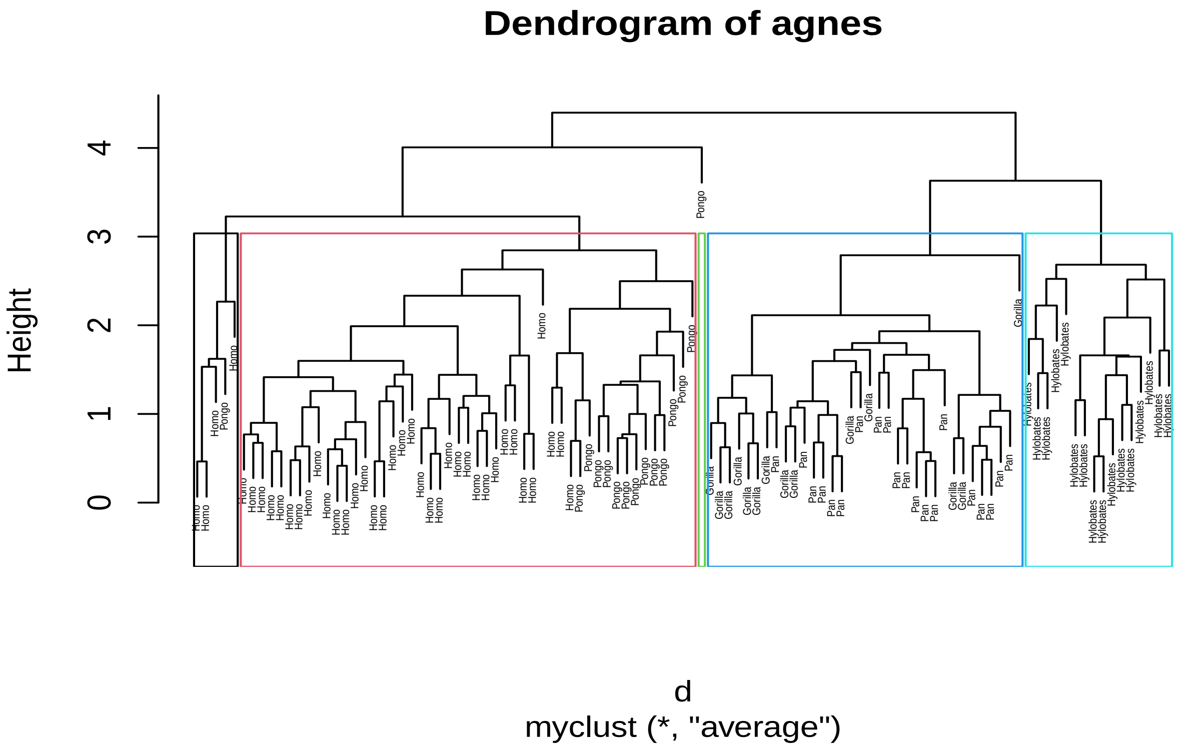
总之，包括 cho.mthd 在内所有与作业相关的代码就都是我自己写的了。如果我一开始就没去逛 GitHub，可能这会儿我都拿到这次作业的分数了。

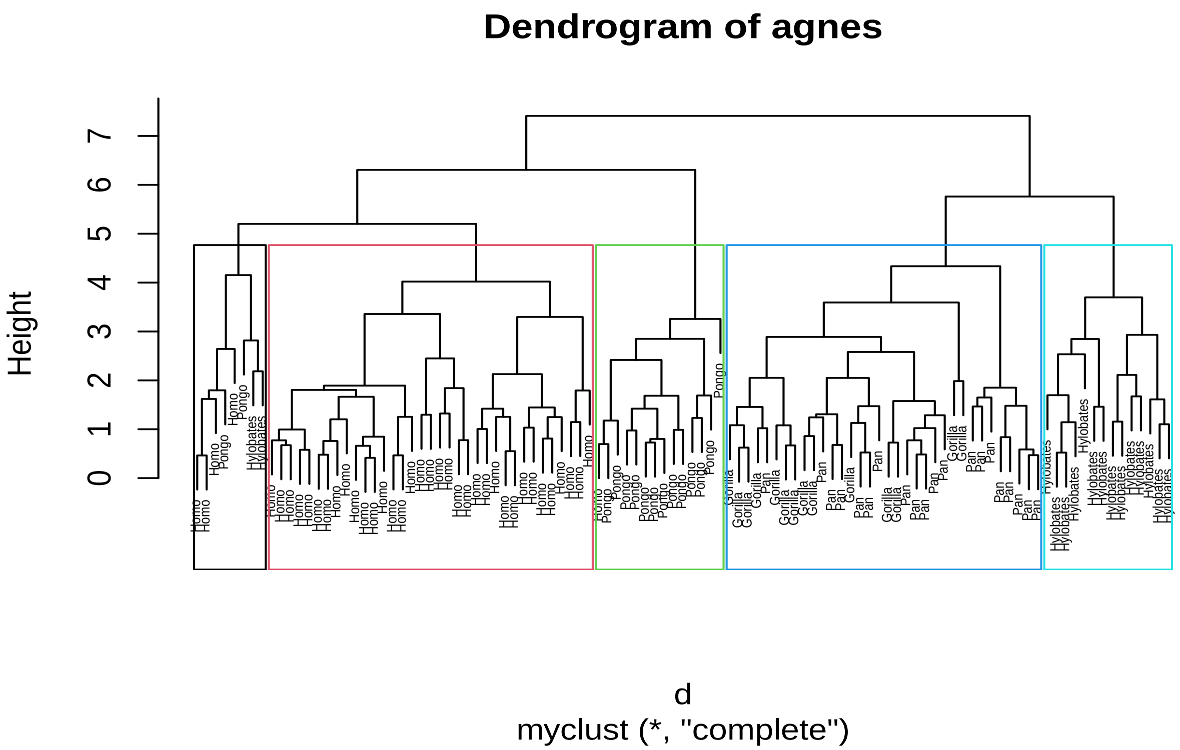
突然想起去年这时候，我在数据结构作业里吐槽 CSDN 的代码（同样是为了让自己的作业更好看，引用别人的代码，但是课程相关的核心部分是自己写的），历史何其相似。

总之，感谢大佬给了我 iorder 函数让我可以画个图，以及告诉我：不仅 CSDN 代码不能抄，GitHub 也不一定可以。

感谢老师看到这里，吐槽完感觉自己不那么郁闷了。**下一页还有图。**


Dendrogram of agnes
Height
0.5
1.0
1.5
Homo
Pongo
Gorilla
Pan
Hylobates
d
myclust (*, "single")


Dendrogram of agnes
Height
0
1
2
3
4
Homo
Pongo
Gorilla
Pan
Hylobates
d
myclust (*, "average")


Dendrogram of agnes
Height
0
1
2
3
4
5
6
7
Homo
Homo
Homo
Pongo
Homo
Pongo
Hylobates
Hylobates
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Homo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Pongo
Gorilla
Gorilla
Gorilla
Gorilla
Pan
Gorilla
Gorilla
Gorilla
Gorilla
Gorilla
Pan
Pan
Pan
Gorilla
Pan
Pan
Pan
Gorilla
Gorilla
Pan
Pan
Pan
Pan
Pan
Gorilla
Gorilla
Pan
Pan
Pan
Pan
Pan
Pan
Pan
Pan
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
Hylobates
d
myclust (*, "complete")