ds2_hw6

Siyan Chen

5/6/2019

```
library(ISLR)
library(factoextra)
```

```
## Loading required package: ggplot2
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

```
library(dendextend)
```

```
##
## ---------------------
## Welcome to dendextend version 1.10.0
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## Or contact: <tal.galili@gmail.com>
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## ---------------------
```

```
##
## Attaching package: 'dendextend'
```

```
## The following object is masked from 'package:stats':
##
##     cutree
```

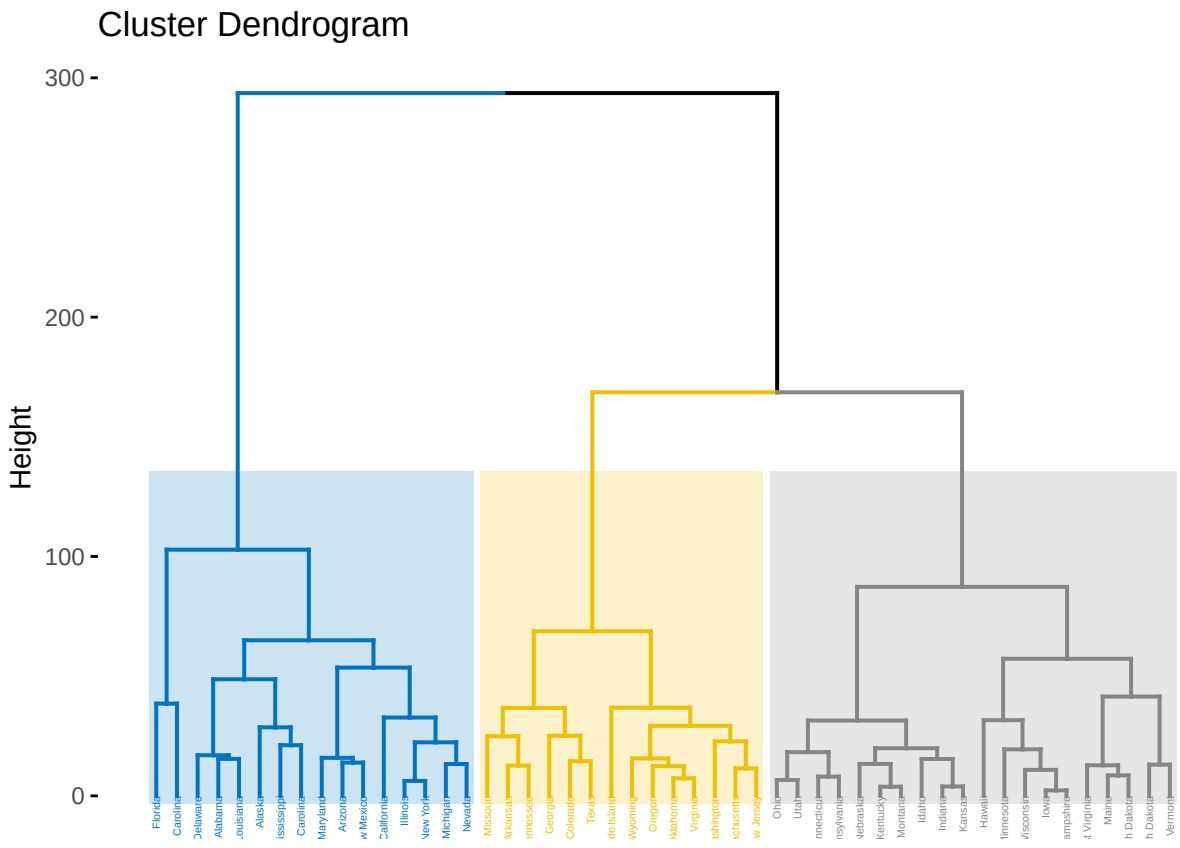
```
library(ggplot2)
```

```
data(USArrests)
head(USArrests)
```

```
##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
## Colorado      7.9     204       78 38.7
```

Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states.

```
hc.complete = hclust(dist(USArrests, method = "euclidean"), method = "complete")
fviz_dend(hc.complete, k = 3,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```



(b) Cut the dendrogram at a height that results in three distinct clusters. Which statesbelong to which clusters?

```
tree_cut = cutree(hc.complete,k = 3)
USArrests[tree_cut == 1,] %>% rownames()
```

```
##  [1] "Alabama"        "Alaska"         "Arizona"        "California"
##  [5] "Delaware"       "Florida"        "Illinois"       "Louisiana"
##  [9] "Maryland"       "Michigan"       "Mississippi"    "Nevada"
## [13] "New Mexico"     "New York"       "North Carolina" "South Carolina"
```

```
USArrests[tree_cut == 2,] %>% rownames()
```

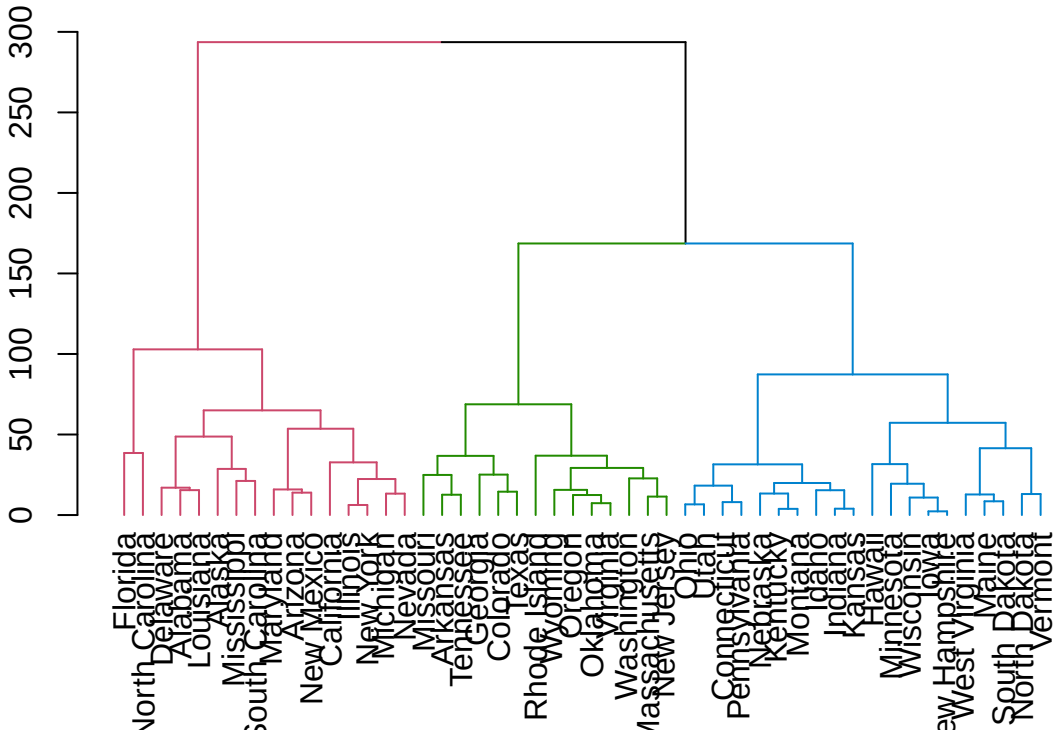
```
##  [1] "Arkansas"      "Colorado"      "Georgia"       "Massachusetts"
##  [5] "Missouri"      "New Jersey"    "Oklahoma"      "Oregon"
##  [9] "Rhode Island"  "Tennessee"     "Texas"         "Virginia"
```

```
## [13] "Washington"    "Wyoming"
```

```
USArrests[tree_cut == 3,] %>% rownames()
```

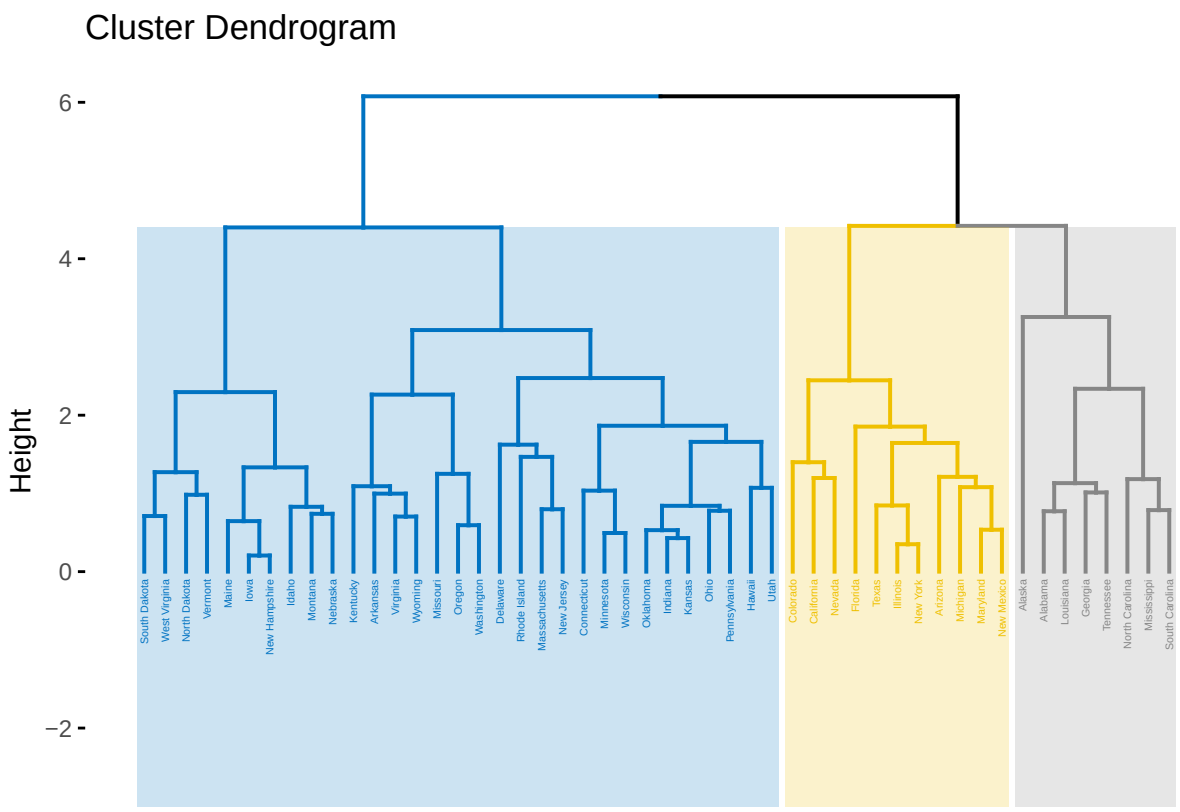
```
##  [1] "Connecticut"   "Hawaii"        "Idaho"         "Indiana"
##  [5] "Iowa"          "Kansas"        "Kentucky"      "Maine"
##  [9] "Minnesota"     "Montana"       "Nebraska"      "New Hampshire"
## [13] "North Dakota"  "Ohio"          "Pennsylvania"  "South Dakota"
## [17] "Utah"          "Vermont"       "West Virginia" "Wisconsin"
```

```
#### show the three cluster by plot
dend_players = as.dendrogram(hc.complete)
dend_colored = color_branches(dend_players, k = 3)
plot(dend_colored)
```



###(c) Hierarchically cluster the states using complete linkage and Euclidean distance, afterscaling the variables to have standard deviation one.

```
df = scale(USArrests)
hc_complete_scaled = hclust(dist(df, method = "euclidean"), method = "complete")
fviz_dend(hc_complete_scaled, k = 3,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```



(d) What effect does scaling the variables have on the hierarchical clustering obtained? Inyour opinion, should the variables be scaled before the inter-observation dissimilarities arecomputed?

The classification of state by cluster altered and the distance between the three clusters become much lower than that of previous hierarchical clustering.

I think varaibles should be scaled before the the inter-observation dissimilarities arecomputed, because all four variables should be equaly considered for the effect on the cllassification. Without scaling, the *Assault* will have greater effect.

Problem 2 PCA

PCA can be used for image compression. In this question, we use thejpegpackage toread and write the .jpeg files. We use a image of cat for illustration, and the sample codesare given in "image.R". Read the image usingimg <- readJPEG('example.jpg'). Theimage will be represented as three matrices as an array with each matrix correspondingto the RGB color value scheme and each element in a matrix corresponding to one pixel.Extract the individual color value matrices to perform PCA on each of them. Reconstructthe original image using the projections of the data with the first 20 PCs.Now use your own .jpg image to perform image compression via PCA with different numbersof PCs (e.g., 50, 100, 200, . . .).

```
library(jpeg)
image1 = readJPEG("/Users/siyanchen/Desktop/bm2_hw6/data/happy.jpeg")
dim(image1)
```

```
## [1] 1920 1079    3
```

```
r1 <- image1[,,1]
g1 <- image1 [,,2]
b1 <- image1 [,,3]
img.r.pca1 <- prcomp(r1, center = FALSE)
img.g.pca1 <- prcomp(g1, center = FALSE)
img.b.pca1 <- prcomp(b1, center = FALSE)

rgb.pca1 <- list(img.r.pca1, img.g.pca1, img.b.pca1)

# Approximate X with XV_kV_k^T
compress <- function(pr, k)
{
  compressed.img <- pr$x[,1:k] %*% t(pr$rotation[,1:k])
  compressed.img
}


# Using 100 PCs
pg100_new <- sapply(rgb.pca1, compress, k = 100, simplify = "array")
writeJPEG(pg100_new , "pca100_new.jpeg")
knitr::include_graphics("./pca100_new.jpeg")
```

```r
# Using 150 PCs
pg150_new <- sapply(rgb.pca1, compress, k = 150, simplify = "array")
writeJPEG(pg150_new , "pca150_new.jpeg")
knitr::include_graphics("./pca150_new.jpeg")
```

```r
# USING 200

pg200_new <- sapply(rgb.pca1, compress, k = 200, simplify = "array")
writeJPEG(pg200_new , "pca200_new.jpeg")
knitr::include_graphics("./pca200_new.jpeg")
```

200 pc is good enough