Multivariate Analysis for the Behavioral Sciences, Second Edition (Chapman and Hall/CRC, 2019)

Examples of Chapter 17: Cluster Analysis

Kimmo Vehkalahti and Brian S. Everitt

18 December 2018

Contents

Examples

Table 12.1: Chest, Waist, and Hip Measurements of 20 Individuals

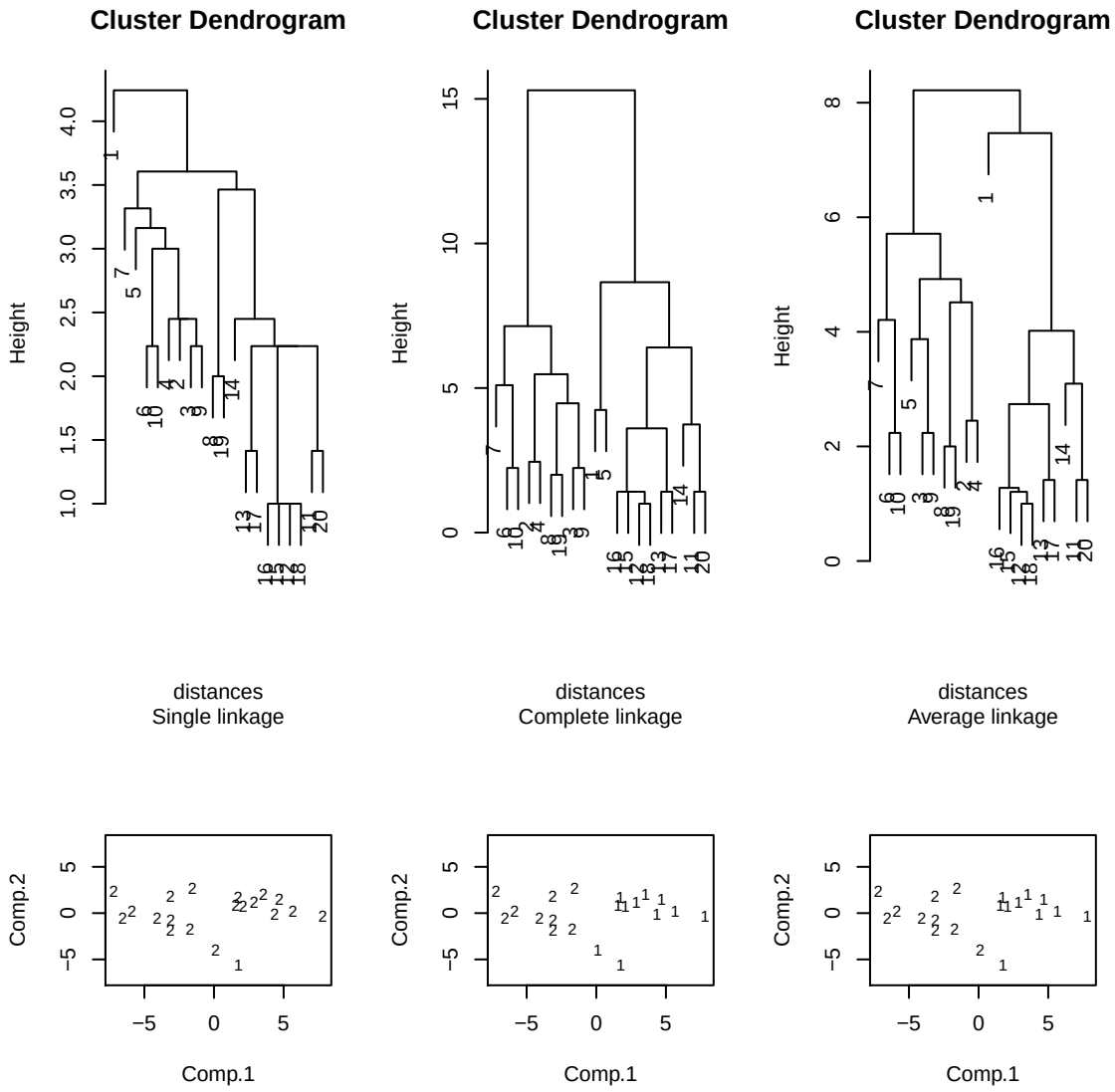
This data set was introduced in **Chapter 12** and it is briefly revisited here.

```
body <- structure(list(
  Chest = c(34, 37, 38, 36, 38, 43, 40, 38, 40, 41, 36, 36, 34, 33, 36, 37, 34, 36, 38, 35),
  Waist = c(30, 32, 30, 33, 29, 32, 33, 30, 30, 32, 24, 25, 24, 22, 26, 26, 25, 26, 28, 23),
  Hips = c(32, 37, 36, 39, 33, 38, 42, 40, 37, 39, 35, 37, 37, 34, 38, 37, 38, 37, 40, 35)),
  .Names = c("Chest", "Waist", "Hips"), row.names = c(NA, -20L), class = "data.frame")
body
```

```
##    Chest Waist Hips
## 1     34    30   32
## 2     37    32   37
## 3     38    30   36
## 4     36    33   39
## 5     38    29   33
## 6     43    32   38
## 7     40    33   42
## 8     38    30   40
## 9     40    30   37
## 10    41    32   39
## 11    36    24   35
## 12    36    25   37
## 13    34    24   37
## 14    33    22   34
## 15    36    26   38
## 16    37    26   37
## 17    34    25   38
## 18    36    26   37
## 19    38    28   40
## 20    35    23   35
```

Figure 17.5

```
attach(body)
distances <- dist(body)

body_sl3 <- cutree(hclust(distances, method = "single"), h=3.8)
body_cl2 <- cutree(hclust(distances, method = "complete"), h=10)
body_al2 <- cutree(hclust(distances, method = "average"), h=7.8)

layout(matrix(c(1,2,3,4,5,6), 2, 3, byrow=TRUE), c(1,1,1), c(2,1), TRUE)

plot(hclust(distances, method = "single"), ylab =" Height", sub = "Single linkage")
plot(hclust(distances, method = "complete"), ylab =" Height", sub = "Complete linkage")
plot(hclust(distances, method = "average"), ylab = "Height", sub = "Average linkage")

body_pc <- princomp(body)
xlim <- range(body_pc$scores[, 1])

plot(body_pc$scores[, 1:2], type = "n",xlim = xlim, ylim =xlim)
text(body_pc$scores[, 1:2], labels = body_sl3, cex=0.8)

plot(body_pc$scores[, 1:2], type = "n",xlim = xlim, ylim = xlim)
text(body_pc$scores[, 1:2], labels = body_cl2, cex=0.8)

plot(body_pc$scores[, 1:2], type = "n", xlim = xlim, ylim = xlim)
text(body_pc$scores[, 1:2], labels = body_al2, cex=0.8)
```


Cluster Dendrogram
Height
distances
Single linkage
Cluster Dendrogram
Height
distances
Complete linkage
Cluster Dendrogram
Height
distances
Average linkage
Comp.2
Comp.1
Comp.2
Comp.1
Comp.2
Comp.1

```
detach(body)
```

Table 17.1: Life Expectancies at Different Ages for Men in Seven Countries

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
W16M50 <- read.csv("data/w16m50.csv")
countries <- W16M50$country
row.names(W16M50) <- countries
W16M50 <- W16M50 %>% select(-country)

W16M50[c("Japan", "Italy", "Spain", "United Kingdom", "Finland", "Cuba", "United States"), ]
```

```
##                 birthM age25M age50M age75M age100M
## Japan             80.8   56.3   32.4   12.0     2.2
## Italy             80.3   55.9   31.9   11.6     1.8
## Spain             80.0   55.5   31.5   11.7     3.4
## United Kingdom    79.0   54.7   31.1   11.2     2.2
## Finland           78.5   54.1   30.5   11.1     1.7
## Cuba              76.5   52.5   29.2   11.0     2.0
## United States     76.4   52.6   29.8   11.2     2.1
```

```
var(W16M50)
```

```
##             birthM    age25M     age50M    age75M   age100M
## birthM  21.5768556 19.811813 15.4074752 6.0390495 0.6955556
## age25M  19.8118132 18.542383 14.4821057 5.6573956 0.7054940
## age50M  15.4074752 14.482106 11.5237102 4.6304612 0.6476694
## age75M   6.0390495  5.657396  4.6304612 2.1054734 0.3659306
## age100M  0.6955556  0.705494  0.6476694 0.3659306 0.4112204
```

Figure 17.6

```
distances <- dist(W16M50)
hclu <- hclust(distances, method = "complete")

#install.packages("ggdendro")
library(ggdendro)
library(ggplot2)

p1 <- ggdendrogram(hclu, rotate = TRUE, theme_dendro = FALSE)
p2 <- p1 + theme_bw()
p3 <- p2 + scale_y_continuous(name = "Height", breaks = seq(0, 25, 5)) # obs: rotation!
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which
## will replace the existing scale.
```

```
p4 <- p3 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p5 <- p4 + xlab("") # obs: rotation!
p6 <- p5 + theme(axis.text.y = element_text(color = "black", size = 7)) # obs: rotation!
p6
```

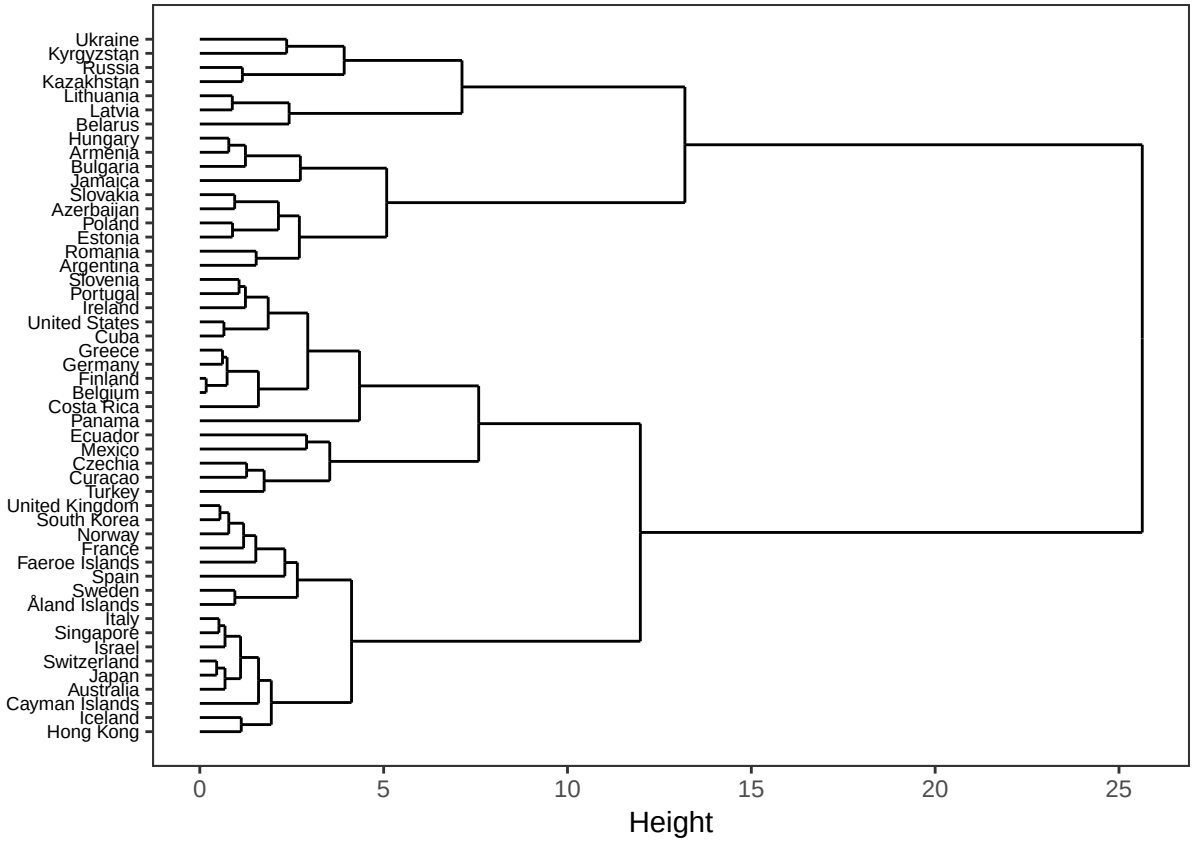


Figure 17.7

```
life_cl4 <- cutree(hclu, k=4)
pairs(W16M50, panel = function(x,y) text(x, y, labels = life_cl4, cex=1.0))
```

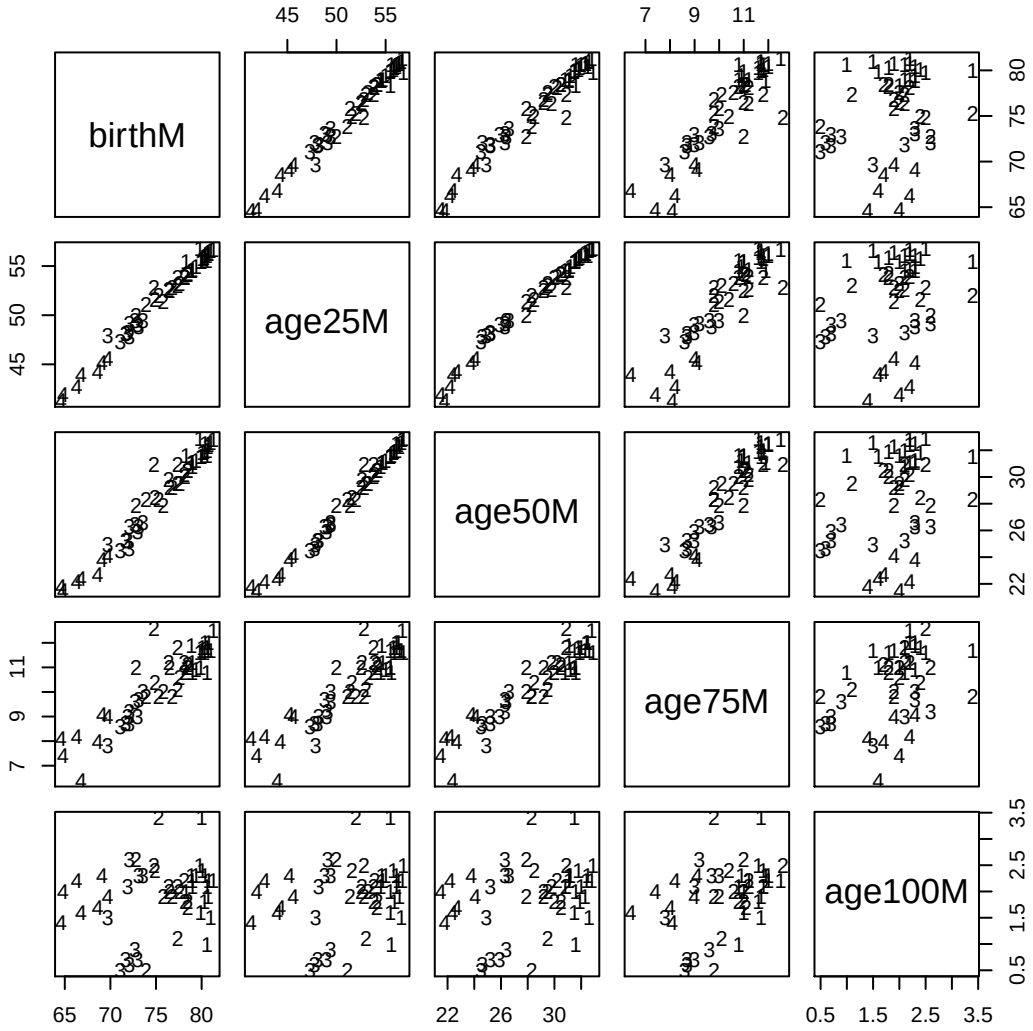


Figure 17.8

```
life_pc <- princomp(W16M50, cor = TRUE)
pc_scores <- as.data.frame(life_pc$scores)
pc_scores <- cbind(pc_scores, life_cl4)
pc_scores$life_cl4 <- as.factor(pc_scores$life_cl4)

pcvar <- round(100 * life_pc$sdev / sum(life_pc$sdev), 2)
xlabel <- paste("PC 1 (", pcvar[1], " %)", sep = "")
ylabel <- paste("PC 2 (", pcvar[2], " %)", sep = "")

p1 <- ggplot(pc_scores, aes(x = Comp.1, y = Comp.2, colour = life_cl4))
# p1 <- ggplot(pc_scores, aes(x = Comp.1, y = Comp.2)) # b&w points (book)
p2 <- p1 + geom_text(aes(label = life_cl4), size=3)
p3 <- p2 + geom_text(aes(label = countries), position = position_nudge(y = -0.15), size=1.5)
p4 <- p3 + scale_x_continuous(name = xlabel,
                              breaks = seq(-4, 3, 1)) + scale_y_continuous(name = ylabel)
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                 legend.position = "none")
p7 <- p6 + coord_fixed(ratio = 1)
p7
```

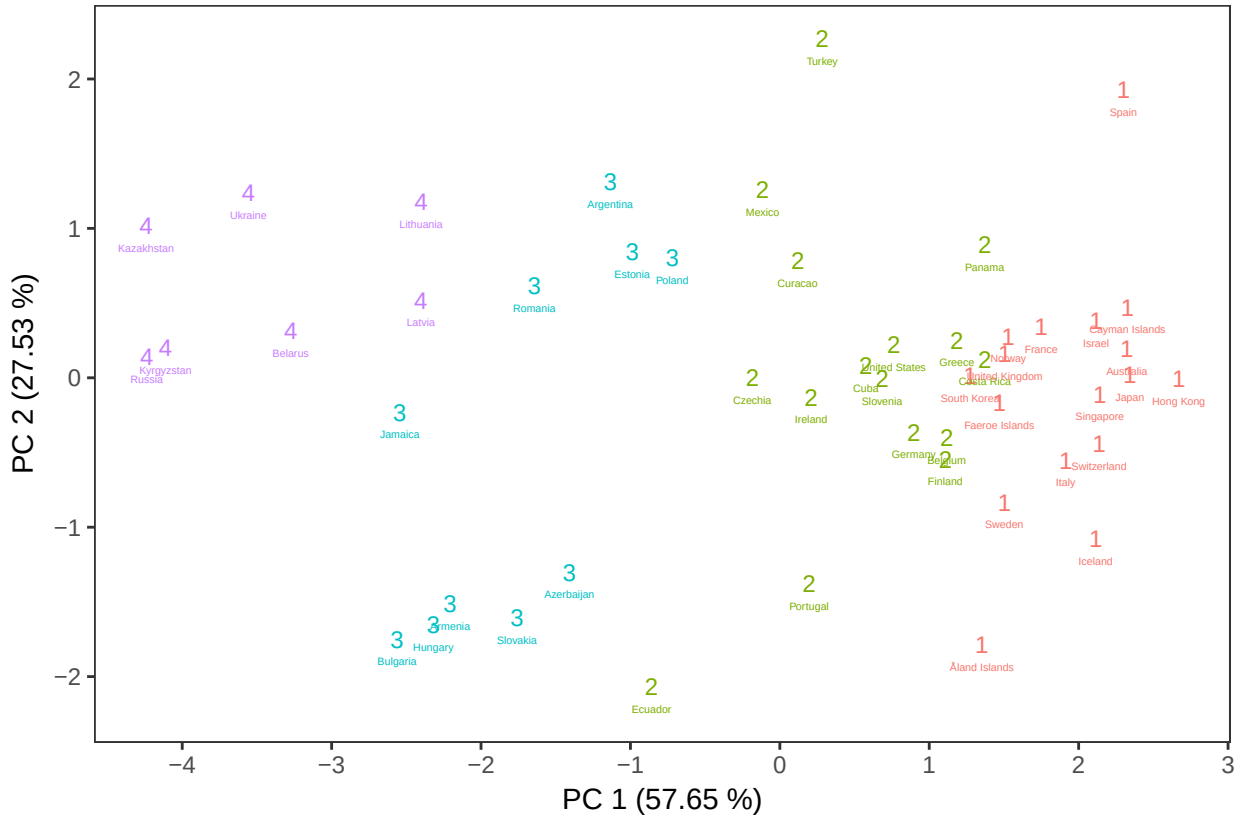


Table 17.2

```
country.mean <- lapply(1:4, function(nc) apply(W16M50[life_cl4 == nc, ], 2, mean))
country.clus <- lapply(1:4, function(nc) countries[life_cl4 == nc])

country.mean
```

```
## [[1]]
##    birthM    age25M    age50M    age75M   age100M
## 80.005883 55.741176 31.858824 11.529412  2.105882
##
## [[2]]
##   birthM   age25M   age50M   age75M  age100M
## 76.43125 52.67500 29.44375 10.76250  2.00000
##
## [[3]]
##  birthM  age25M  age50M  age75M age100M
##   72.10   48.50   25.59    9.03    1.42
##
## [[4]]
##    birthM    age25M    age50M    age75M   age100M
## 67.128572 43.571429 22.642857  8.028572  1.871429
```

```
country.clus
```

```
## [[1]]
##  [1] Cayman Islands Hong Kong      Israel         Japan
##  [5] South Korea    Singapore      Åland Islands  Faeroe Islands
##  [9] France         Iceland        Italy          Norway
## [13] Spain          Sweden         Switzerland    United Kingdom
## [17] Australia
## 50 Levels: Argentina Armenia Australia Azerbaijan Belarus ... Åland Islands
##
## [[2]]
##  [1] Costa Rica    Cuba          Curacao       Mexico        Panama
##  [6] United States Ecuador       Turkey        Belgium       Czechia
## [11] Finland       Germany       Greece        Ireland       Portugal
## [16] Slovenia
## 50 Levels: Argentina Armenia Australia Azerbaijan Belarus ... Åland Islands
##
## [[3]]
##  [1] Jamaica    Argentina  Armenia    Azerbaijan Bulgaria   Estonia
##  [7] Hungary    Poland     Romania    Slovakia
## 50 Levels: Argentina Armenia Australia Azerbaijan Belarus ... Åland Islands
##
## [[4]]
## [1] Kazakhstan Kyrgyzstan Belarus    Latvia     Lithuania  Russia
## [7] Ukraine
## 50 Levels: Argentina Armenia Australia Azerbaijan Belarus ... Åland Islands
```

Figure 17.9: Plot of within-groups sum of squares against number of clusters

See **Chapter 13**, where this data set was used for the first time.

```
crime <- read.table("data/crime.txt", sep = '\t')
rlabs <- row.names(crime)

head(crime)
```

```
##    Murder Rape Robbery Assault Burglary Theft Vehicle
## ME    2.0 14.8      28     102      803  2347     164
## NH    2.2 21.5      24      92      755  2208     228
## VT    2.0 21.8      22     103      949  2697     181
## MA    3.6 29.7     193     331     1071  2189     906
## RI    3.5 21.4     119     192     1294  2568     705
## CT    4.6 23.8     192     205     1198  2758     447
```

```
# DC (outlier, see Chapter 13):
crime[24, ]
```

```
##    Murder Rape Robbery Assault Burglary Theft Vehicle
## DC     31 52.4     754     668     1728  4131     975
```

```
# remove DC:
crime <- crime[-24, ]

# variances:
apply(crime, 2, var)
```

```
##       Murder         Rape      Robbery      Assault     Burglary
##     11.93492    209.76335  11889.56122  19373.53510 175895.00449
##        Theft      Vehicle
## 565276.55878  43997.35878
```

```
# standardize by range
rge <- apply(crime, 2, max) - apply(crime, 2, min)
crime_std <- sweep(crime, 2, rge, FUN = "/")

# variances of the std data:
apply(crime_std, 2, var)
```

```
##     Murder       Rape    Robbery    Assault   Burglary      Theft
## 0.07638350 0.05618847 0.04625407 0.05900647 0.05218049 0.06218510
##    Vehicle
## 0.06755843
```

```r
# plot of wgss against number of clusters:
n <- length(crime_std[, 1])
wss1 <- (n-1) * sum(apply(crime_std, 2, var))
wss <- numeric(0)
for(i in 2:6) {
    W <- sum(kmeans(crime_std, i)$withinss)
    wss <- c(wss, W)
}
wss <- c(wss1, wss)
plot(1:6, wss, type = "l",
     xlab = "Number of groups", ylab = "Within groups sum of squares", lwd=2)
```

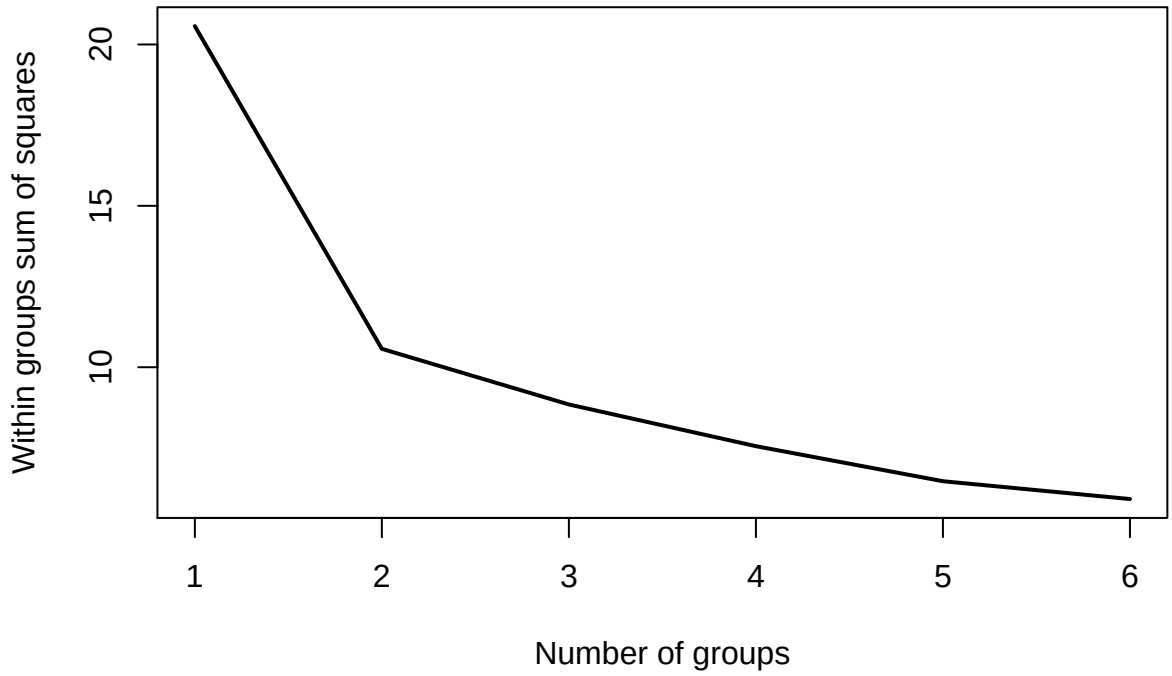


Table 17.3

```
# get two-group solution from k-means and group means and membership
crime_kmean2 <- kmeans(crime_std, 2)
lapply(1:2, function(nc) apply(crime[crime_kmean2$cluster == nc, ], 2, mean))
```

```
## [[1]]
##      Murder        Rape     Robbery     Assault    Burglary       Theft
##    9.368182   45.372727  229.000000  394.772727 1543.409091 3368.045455
##     Vehicle
##  554.272727
##
## [[2]]
##      Murder        Rape     Robbery     Assault    Burglary       Theft
##    4.739286   24.803571   73.821429  182.071429  924.214286 2564.714286
##     Vehicle
##  247.035714
```

```
lapply(1:2, function(nc) rlabs[crime_kmean2$cluster == nc])
```

```
## [[1]]
##  [1] "MA" "NY" "NJ" "IL" "MI" "MO" "MD" "NC" "SC" "GA" "KY" "AR" "LA" "OK"
## [15] "WY" "CO" "NM" "UT" "NV" "WA" "OR" "CA"
##
## [[2]]
##  [1] "ME" "NH" "VT" "RI" "CT" "PA" "OH" "IN" "WI" "MN" "IA" "ND" "SD" "NE"
## [15] "KS" "DE" "DC" "VA" "WV" "FL" "TN" "AL" "MS" "TX" "MT" "ID" "AZ" "AK"
## [29] "HI"
```

Figure 17.10

```
crime_pc <- princomp(crime_std, cor = TRUE)
xlim <- range(crime_pc$scores[, 1])
plot(crime_pc$scores[, 1:2], type = "n", xlim = xlim, ylim = xlim)
text(crime_pc$scores[, 1:2], labels = crime_kmean2$cluster, cex=0.8)
```

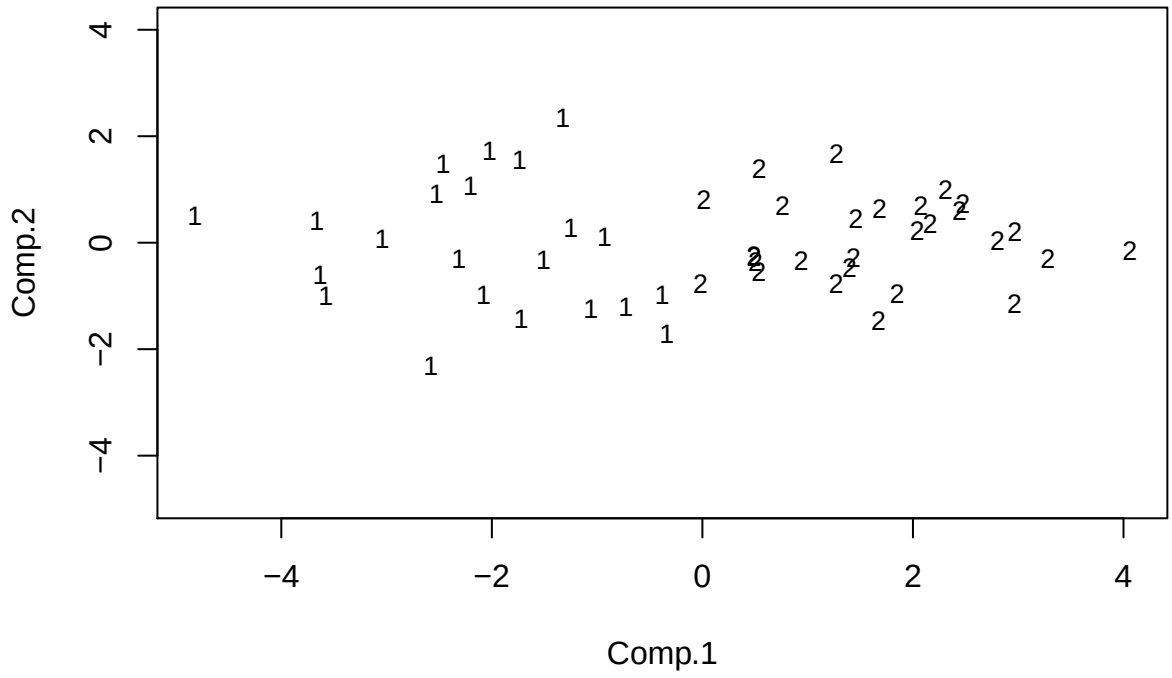


Table 17.4: Proportion of Respondents Answering Yes to Each of the Questions in the Survey of Gastroenterologists

```
prop <- read.table("data/prop.txt", sep = '\t')
options(digits=3)
prop
```

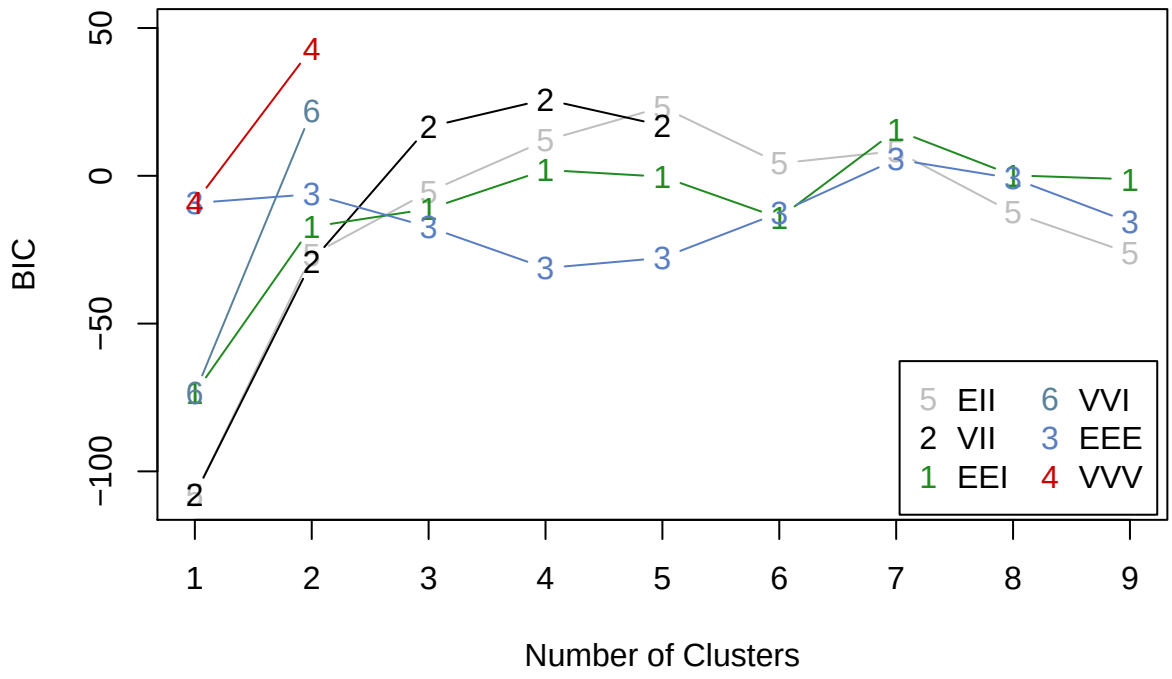
```
##                    Q.1   Q.2    Q.3   Q.4   Q.5   Q.6
## Iceland         1.0000 1.000 1.0000 1.000 1.000 1.000
## Norway          0.8571 0.833 1.0000 1.000 1.000 0.800
## Sweden          1.0000 0.636 1.0000 1.000 0.500 0.667
## Finland         1.0000 0.667 1.0000 1.000 0.833 0.667
## Denmark         0.9231 0.692 1.0000 0.750 0.364 0.538
## UK              0.6316 0.889 1.0000 0.950 0.526 1.000
## Ireland         1.0000 0.667 1.0000 0.000 0.000 1.000
## Germany         1.0000 1.000 1.0000 0.857 0.154 0.929
## Netherlands     1.0000 1.000 1.0000 0.875 0.714 0.875
## Belgium         0.0000 1.000 1.0000 0.500 0.000 1.000
## Switzerland     1.0000 1.000 1.0000 0.500 0.000 1.000
## France          0.3000 0.875 0.6250 0.200 0.000 0.875
## Spain           0.0833 1.000 0.8000 0.545 0.000 1.000
## Portugal        0.1667 1.000 0.6667 0.500 0.000 1.000
## Italy           0.4667 1.000 0.9286 0.400 0.133 1.000
## Greece          0.1250 1.000 0.6250 0.125 0.000 1.000
## Yugoslavia      0.2667 1.000 0.5333 0.267 0.000 1.000
## Albania         0.4000 0.600 0.4000 0.400 0.600 0.600
## Bulgaria        0.0000 1.000 0.3333 0.000 0.000 1.000
## Romania         0.0000 1.000 0.1429 0.143 0.143 1.000
## Hungary         0.2000 1.000 0.8000 0.000 0.000 1.000
## Czechoslovakia 0.0606 0.971 0.0882 0.000 0.000 0.571
## Poland          0.0000 1.000 0.2632 0.105 0.000 0.947
## Russia          0.0000 0.857 0.2857 0.000 0.000 0.857
## Lithuania       0.0000 1.000 0.0000 0.000 0.000 1.000
## Latvia          0.0000 1.000 0.0000 0.000 0.000 1.000
## Estonia         0.6667 1.000 1.0000 0.000 0.000 1.000
```

Figure 17.11

```
#install.packages("mclust")
library(mclust)
```

```
## Package 'mclust' version 5.4.2
## Type 'citation("mclust")' for citing this R package in publications.
```

```
# obs: seeking a correspondence with the 1st ed. (2009) version that was done
# using mclust ver.2 (whereas the results below were achieved with mclust ver.5):
prop_mclust <- Mclust(prop, modelNames = c("EII", "VII", "EEI", "VVI", "EEE", "VVV"))
plot(prop_mclust, what = "BIC", symbols = c("5","2","1","6","3","4"),
     xlab = "Number of Clusters", ylim = c(-110,50))
```




```
# Note:
# "EII" = spherical, equal volume
# "VII" = spherical, unequal volume
# "EEI" = diagonal, equal volume and shape
# "VVI" = diagonal, varying volume and shape
# "EEE" = ellipsoidal, equal volume, shape, and orientation
# "VVV" = ellipsoidal, varying volume, shape, and orientation
# (there are several more options in mclust ver.5)
```

Table 17.5

```
# obs: the optimal model (4: "VVV") with mclust ver.5 has only two clusters
prop_mclust$parameters$mean[, 1]
```

```
##   Q.1   Q.2   Q.3   Q.4   Q.5   Q.6
## 0.822 0.830 0.874 0.694 0.474 0.804
```

```
prop_mclust$parameters$mean[, 2]
```

```
##    Q.1    Q.2    Q.3    Q.4    Q.5    Q.6
## 0.1517 0.9821 0.5336 0.1857 0.0184 0.9786
```

```
row.names(prop)[prop_mclust$classification == 1]
```

```
##  [1] "Iceland"        "Norway"         "Sweden"         "Finland"
##  [5] "Denmark"        "UK"             "Ireland"        "Germany"
##  [9] "Netherlands"    "Switzerland"    "Albania"        "Czechoslovakia"
```

```
row.names(prop)[prop_mclust$classification == 2]
```

```
##  [1] "Belgium"    "France"     "Spain"      "Portugal"   "Italy"
##  [6] "Greece"     "Yugoslavia" "Bulgaria"   "Romania"    "Hungary"
## [11] "Poland"     "Russia"     "Lithuania"  "Latvia"     "Estonia"
```