Homework 5 Data Mining

Max Machalek

April 29, 2019

***Revisited January 04, 2021 (5.2.c)*

```
library(ISLR)

## Warning: package 'ISLR' was built under R version 3.5.2

library(cluster)
flower <- read.csv("./flower.csv", sep = ",", header = TRUE)
colnames(flower) <-
c('winters','shadow','tubers','color','soil','preference','height','di
stance')
labs <- 1:18;

# only working with soil, preference, height, and distance #
flower$winters <- NULL; flower$shadow <- NULL; flower$tubers <- NULL;
flower$color <- NULL


n <- nrow(flower); p = ncol(flower)

flower.s <- scale(flower)
flower.d <- dist(flower.s)
```

H.5.1: Perform PCA with scale = TRUE

a) Give the 4 sets of PC loadings for all 4 variables. Interpret the first two PCs based upon the magnitude and sign of the loadings using a cutoff of 0.5.

```
##                     PC1        PC2        PC3          PC4
## soil       -0.07049919  0.8252355  0.5603175  0.007780557
## preference -0.44420330  0.4704739 -0.7506586  0.133601604
## height     -0.61447284 -0.2713673  0.3130334  0.671411191
## distance   -0.64817931 -0.1549211  0.1567353 -0.728901287
```

" Tabachnick and Fidell (2007, p 649) suggest stringent cut-offs 0.30 (poor), 0.45 (fair), 0.55 (good), 0.60 (very good), 0.70 (excellent)"

The first component (PC1) has loadings for two variables that are in the “poor” category, soil (-0.07049919) and preference (-0.44420330). The loadings for height and distance are both in the “very good” range (-0.61447284 and -0.64817931 respectively).

The second component (PC2) again has two variables in the “poor” category, except for this component these are height (-0.2713673) and distance (-0.1549211). For PC2, a single loading is “excellent” (soil, 0.8252355) and a single loading is “fair” (preference, 0.4704639).

Using a cutoff of 0.5, we observe that the first principal component decreases as height increases and decreases as distance increases. The loading for soil on the first component is below 0.5.
The second principal component increases as soil increases. It is worth noting that preference is very close to our 0.5 cutoff (0.47) and we would expect PC2 to increase as preference increases. Height and distance loadings are also below 0.5 in magnitude, but not close to it.

```
pc.fit <- prcomp(flower,scale=TRUE)
pc.sum <- summary(pc.fit)
p.var <- pc.sum$importance[2,]
c.var <- pc.sum$importance[3,]
pc.var <- (pc.sum$sdev)^2

pc.out <- rbind(pc.var,p.var,c.var)[,1:4]

loadings <- pc.fit$rotation

## show loadings ##
loadings
```

```
##                      PC1        PC2        PC3          PC4
## soil       -0.07049919  0.8252355  0.5603175  0.007780557
## preference -0.44420330  0.4704739 -0.7506586  0.133601604
## height     -0.61447284 -0.2713673  0.3130334  0.671411191
## distance   -0.64817931 -0.1549211  0.1567353 -0.728901287
```

b) <u>Give the PCA summaries (stdev, pve, cumulative pve). Explain how many components should be retained based upon the average eigenvalue and pve.</u>

The eigenvalue is greater than one for PC1 (2.111) and PC2 (1.194), but not PC3 (0.562) or PC4 (0.133). Based on eigenvalue, we should retain PC1 and PC2.

In order to get over a 0.80 threshold for cumulative proportion of variance, PC1 and PC2 are the necessary minimum of components to retain. Their cumulative proportion of variance is 0.8262.

```
pc.out
```

```
##             PC1      PC2       PC3       PC4
## pc.var 2.111077 1.193757 0.5622918 0.1328743
## p.var  0.527770 0.298440 0.1405700 0.0332200
## c.var  0.527770 0.826210 0.9667800 1.0000000
```

```r
summary(pc.fit)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4
## Standard deviation     1.4530 1.0926 0.7499 0.36452
## Proportion of Variance 0.5278 0.2984 0.1406 0.03322
## Cumulative Proportion  0.5278 0.8262 0.9668 1.00000
```

c) <u>Plot the first two PC scores with the labels. Explain whether or not there are PC1 or PC2 outliers.</u>

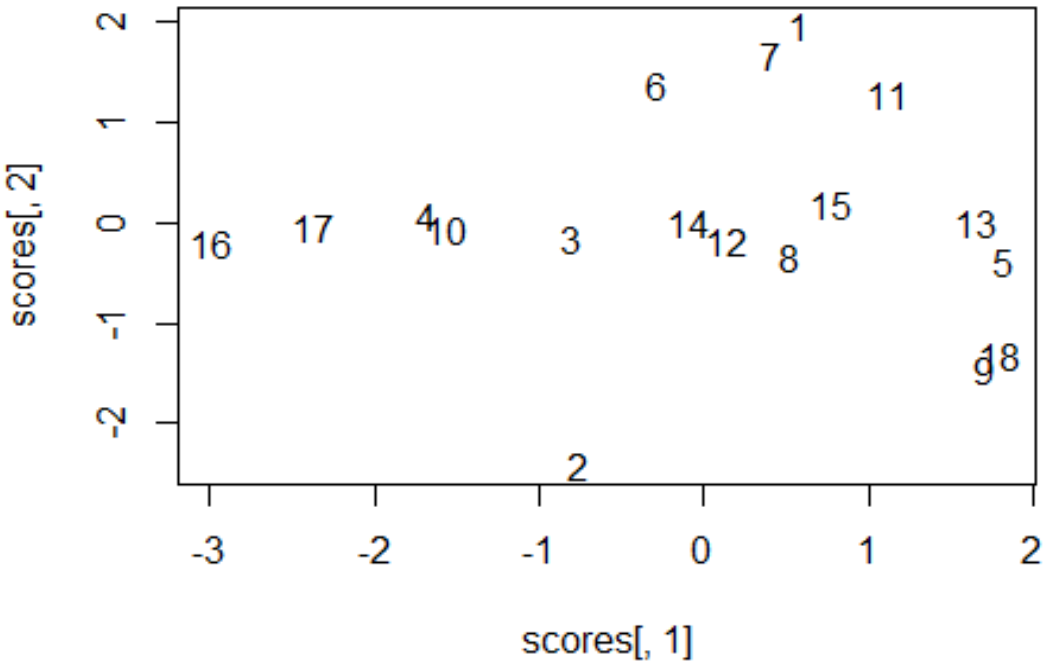
See Fig 01. Flower 2 looks like a potential outlier, but did not meet the criteria. No outliers were found. The max value of |dij| is 2.216829, which is not higher than 3 so does not qualify as an outlier.

```r
scores <- pc.fit$x

cols = function(vec) {
  cols = rainbow(length(unique(vec)))
  return(cols[as.numeric(as.factor(vec))])
}

plot(scores[,1], scores[,2], type = "n", main = "Fig 01 - Plot of PC1 vs PC2")
text(scores[,1],scores[,2], labs)
```

Fig 01 - Plot of PC1 vs PC2




```
m <- 2 ## num principal components (2: pc1 and pc2)
dd <- scores / (matrix(1,n,1)%*%sqrt(t(pc.var)))
ii.d <- which(abs(dd[,1:m]) > 3, arr.in = TRUE)

pcj.outlier <- cbind(ii.d,dd[ii.d])

tt <- apply(dd[,-c(1:m)]^2, 1, sum)
test <- sum(dd[,-c(1:m)]^2)
cc <- qchisq(1-.05/n,n-m)
ii.resid <- which(tt>cc)

pcresid.outlier <- cbind(ii.resid,tt[ii.resid])

max(abs(dd[,1:m]))

## [1] 2.216829
```

H.5.2

a) Fit Heirarchial CA using method = 'average'. Give the dendrogram with the gap criterion line along with the number of observations in each cluster. Explain whether any flowers may be outliers.

Because nodes 2 and 3 have a large height and each belong to a very small cluster, there is potential that these are outliers. See Fig 02. Cluster 3 only contains flower 3, and cluster 2 only contains flower 2.

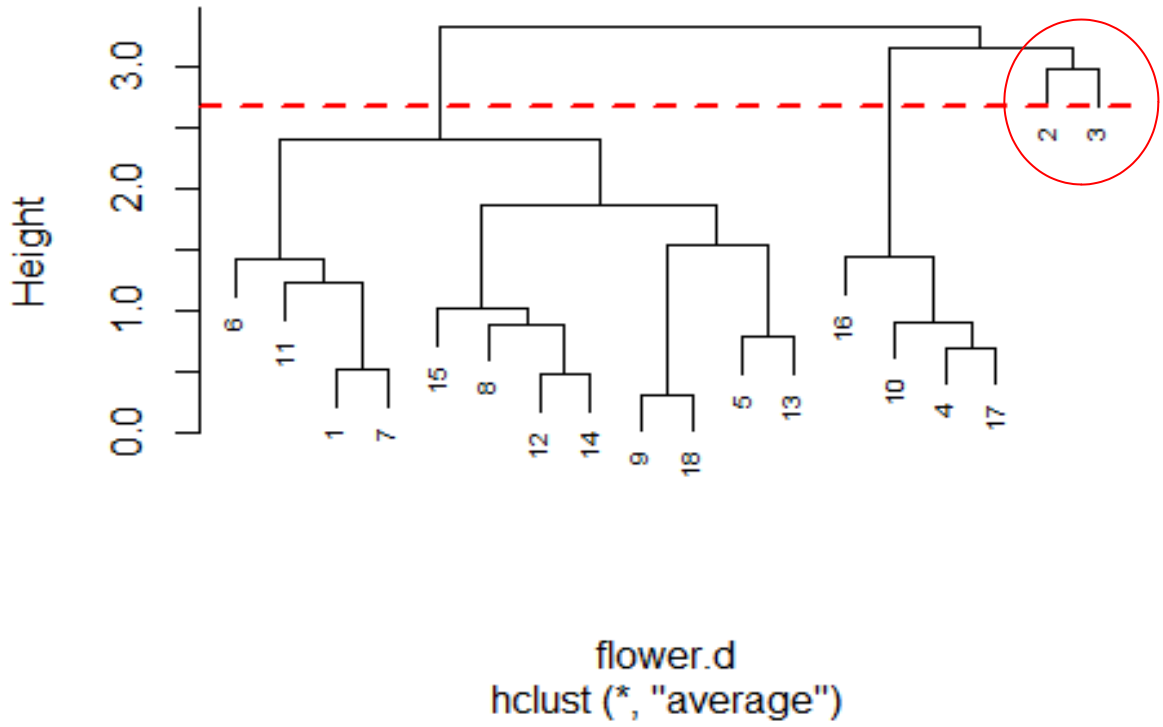
```
table(hc.clusters,labs)
```

```
##             labs
## hc.clusters 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
##           1 1 0 0 0 1 1 1 1 1  0  1  1  1  1  1  0  0  1
##           2 0 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0
##           3 0 0 1 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0
##           4 0 0 0 1 0 0 0 0 0  1  0  0  0  0  0  1  1  0
```

```
hc.fit <- hclust(flower.d, method = 'average')

hh <- hc.fit$height
k <-  1.25; hh.crit <-  mean(hh) + k * sd(hh)
plot(hc.fit, labels = labs, cex = 0.65, ylim = c(0,5), main = "Fig 02
- Cluster Dendrogram")
abline(h = hh.crit, col = "red", lty = 2, lwd = 2)
```

Fig 02 - Cluster Dendrogram




```
coph <- cor(flower.d, cophenetic(hc.fit))
sil_cl <- silhouette(cutree(hc.fit,h=hh.crit),flower.d)
values <- sil_cl[1:n,]
mean_sil <- summary(sil_cl)[[1]][4]

hc.clusters <- cutree(hc.fit,h=hh.crit)
table(hc.clusters,labs)

##             labs
## hc.clusters 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
##           1 1 0 0 0 1 1 1 1 1  0  1  1  1  1  1  0  0  1
##           2 0 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0
##           3 0 0 1 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0
##           4 0 0 0 1 0 0 0 0 0  1  0  0  0  0  0  1  1  0
```

b) Perform k-means CA with k = 3 using set.seed(2) and nstart = 20. Give a table of the cluster sizes and the cluster means. Interpret what the clusters represent based upon the cluster means.

Examples of the first cluster have a mean (standardized) soil value of -.5338725, while the second and third clusters have examples with means of -0.1642685 and 1.3141477 respectively. This means that cluster 1 is expected to have lower soil values than cluster 2, and cluster 2 is expected to have lower soil values than cluster 3.

This same trend is followed for preference, with clusters 1, 2, and 3 having examples with standardized means of -0.5151222, 0.3746343, and 0.4682929.

For height, examples from cluster 3 have the lowest standardized mean (-0.7072614). The highest standardized mean belongs to cluster 2 (1.1771204), with cluster 1 (-0.5292096) between the two (but closer to cluster 3).

Finally, cluster 1 has the lowest distance standardized mean (-0.6919305) and cluster 2 has the highest (1.2541241). Cluster 3 (-0.4973251) is in between 1 and 2, however closer to cluster 1 than to cluster 2.

So, plants with very low soil, preference, and distance values as well as moderate height are likely to be associated with cluster 1.

Plants with moderate soil and preference values but high height and distance are likely to be associated with cluster 2.

Plants with high soil and preference values, but low height and moderate distance are likely to be associated with cluster 3.

("Moderate" is used relative to the other clusters).

```
set.seed(2)
km.out <- kmeans(flower.s, 3, nstart = 20)
km.clusters <- km.out$cluster
km.centers <- km.out$centers
km.centers
```

```
##         soil preference     height   distance
## 1 -0.5338725 -0.5151222 -0.5292096 -0.6919305
## 2 -0.1642685  0.3746343  1.1771204  1.2541241
## 3  1.3141477  0.4682929 -0.7072614 -0.4973251
```

(table on next page)

```
table(km.clusters, hc.clusters)
```

```
##             hc.clusters
## km.clusters 1 2 3 4
##           1 8 0 0 0
##           2 4 0 0 0
##           3 0 1 1 4
```

c) Fit LDA using the k-Means clusters in (b). Give the confusion matrix and error rates for each of the 3 classes. Explain which cluster is not predicted as well.

*****edit* (Jan 04, 2021): X = scores[,1:2] changed to X = flower.s[,1:2], previously falsely perfect predictions are now fixed.***

Prediction Errors:

Class 1: 1/8 misclassified (12.5%)

Class 2: 0/4 misclassified (0%)

Class 3: 2/6 misclassified (33.3%)

Observations belonging to class 3 are misclassified most frequently; it is not predicted as well as classes 1 or 2.

```
library(MASS)
err.f <- function(y,yh) { ct = table(yh,y); err = 1-
sum(y==yh)/length(y);
rates = prop.table(ct,2);
list(predictions=ct,overall_error=err,prediction_rates=rates) }
y = as.factor(km.clusters)
X = flower.s[,1:2]
#X = scores[,1:2] **
lda.fit <- lda(X,y,cv=TRUE)
lda.pred <- predict(lda.fit)$class
err.f(y,lda.pred)
```

```
## $table
##        y
## yh   1 2 3
##    1 7 0 1
##    2 0 4 1
##    3 1 0 4
## $err
## [1] 0.1666667
## $prediction_rates
##                   y
## yh          1         2         3
##   1 0.8750000 0.0000000 0.1666667
##   2 0.0000000 1.0000000 0.1666667
##   3 0.1250000 0.0000000 0.6666667
```