Stat501: Final Exam

Kelby Kies

5/6/2021

Question 1: We consider the problem of classifying a multinomial observation vector X into one of two classes Mult(n,p1) or Mult(n, p2) with prior probabilities π and $(1-\pi)$, and equal misclassification costs. Show that the discriminant rule for this problem is linear: i.e. it can be reduced to classification based on whether a'x + c is positive or negative. [15 points]

My apologies, but I didn't have time to make it back to this question.

Question 2:

Part 1.) Is a multivariate normality assumption reasonable for the distribution of the attributes for each cultivar? [10 points; 1page]

```
wine <- read.table('~/Desktop/stat_501/wine.dat', sep = ',')

group1 <- dplyr::filter(wine, wine$V1 == 1)
group2 <- dplyr::filter(wine, wine$V1 == 2)
group3 <- dplyr::filter(wine, wine$V1 == 3)

source('~/Desktop/stat_501/testnormality.R')

# group1
print("Cultivar 1 q-value:")
```

```
## [1] "Cultivar 1 q-value:"
```

```
Cramer.test(group1[,-1])
```

```
## Loading required package: parallel
```

```
## [1] 0.7107487
```

```
print("Cultivar 2 q-value:")
```

```
## [1] "Cultivar 2 q-value:"
```

```
Cramer.test(group2[,-1])
```

```
## [1] 6.312644e-06
```

```
print("Cultivar 3 q-value:")
```

```
## [1] "Cultivar 3 q-value:"
```

```
Cramer.test(group3[,-1])
```

```
## [1] 0.9250732
```

If the q-value returned is less than 0.05, then normality is not good and is rejected. Thus we can not support Multivariate Normality Assumptions and the data should be transformed. It appears that we can assume multivaraite normality for the 1st and 3rd Cultivar, but not for the 2nd cultivar because the q-value of 6.58242e-06 is smaller than 0.05.

Part 2. Please perform a detailed factor analysis for wines in the third cultivar (i.e. cultivar given by 3). Use BIC to determine the number of factors. [15 points; 2 pages]

```
library(fad)
```

```
## Loading required package: RSpectra
```

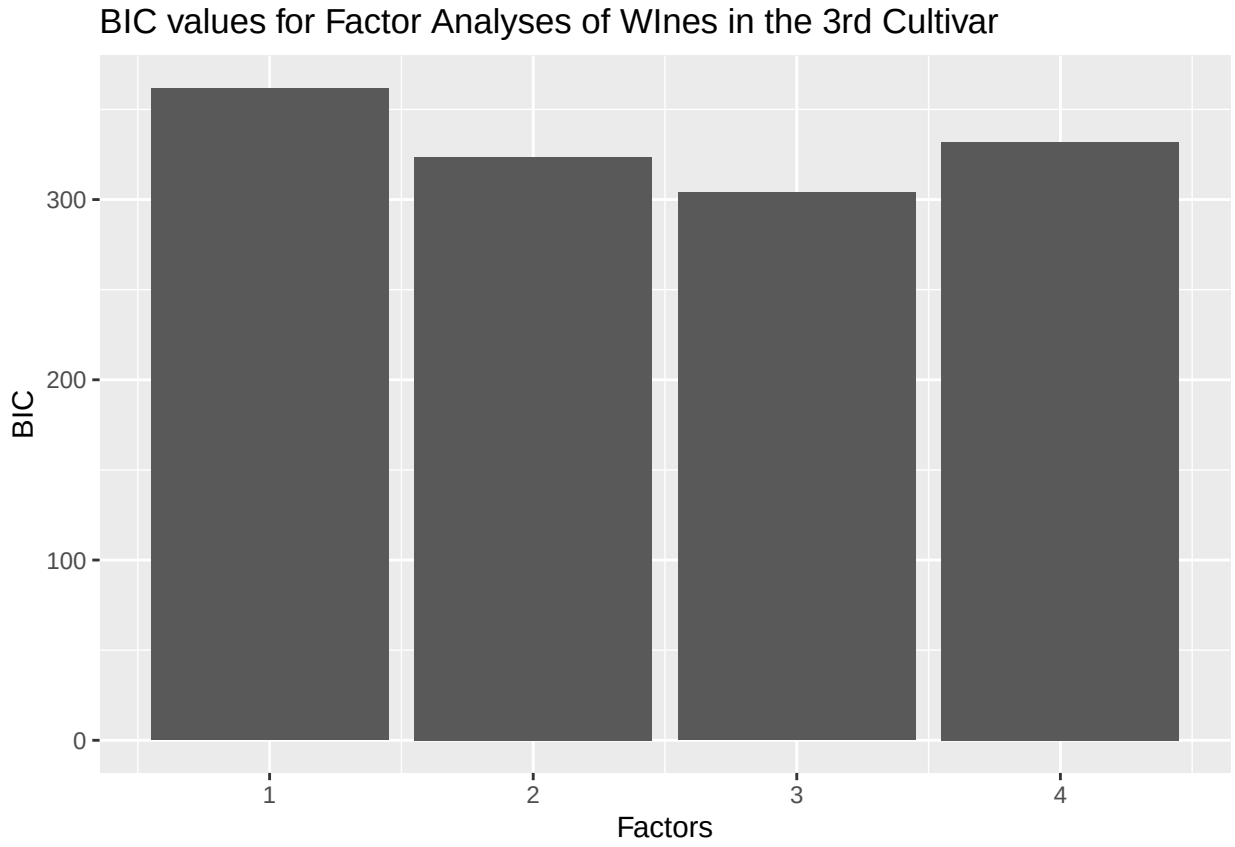
```
## Please cite the paper: Dai, F., Dutta, S., and, Maitra, R. (2020). A Matrix-Free Likelihood Method f
```

```
anlaysis_1 <- fad(x=as.matrix(group3[,-1]), factors=1, scores = "regression",rotation = "varimax", metho
```

```
## Warning in fad(x = as.matrix(group3[, -1]), factors = 1, scores =
## "regression", : Algorithm may not have converged. Try another starting value.
```

```
anlaysis_2 <- fad(x=as.matrix(group3[,-1]), factors=2, scores = "regression",rotation = "varimax", metho
anlaysis_3 <- fad(x=as.matrix(group3[,-1]), factors=3, scores = "regression",rotation = "varimax", metho
anlaysis_4 <- fad(x=as.matrix(group3[,-1]), factors=4, scores = "regression",rotation = "varimax", metho

df <- rbind(c(anlaysis_1$factors,anlaysis_1$BIC),c(anlaysis_2$factors,anlaysis_2$BIC),c(anlaysis_3$facto
df <- data.frame(Factors = c(anlaysis_1$factors,anlaysis_2$factors,anlaysis_3$factors,anlaysis_4$factors
                 BIC = c(anlaysis_1$BIC,anlaysis_2$BIC, anlaysis_3$BIC, anlaysis_4$BIC))
library(ggplot2)
ggplot(data = df, aes(x=Factors, y = BIC)) + geom_bar(stat="identity") + ggtitle("BIC values for Factor
```




```
print("Here is the full analysis:")
```

```
## [1] "Here is the full analysis:"
```

```
anlaysis_3
```

```
##
## Call:
## fad(x = as.matrix(group3[, -1]), factors = 3, scores = "regression",     rotation = "varimax", metho
##
## Uniquenesses:
##    V2    V3    V4    V5    V6    V7    V8    V9   V10   V11   V12   V13   V14
## 0.816 0.895 0.005 0.408 0.628 0.459 0.085 0.283 0.005 0.517 0.695 0.673 0.845
##
## Loadings:
##       [,1]   [,2]   [,3]
##  [1,]  0.337          0.253
##  [2,] -0.228  0.225
##  [3,]        -0.103  0.992
##  [4,]  0.121 -0.108  0.752
##  [5,]  0.106 -0.581  0.151
##  [6,]  0.545  0.105  0.483
##  [7,]  0.351 -0.870  0.187
##  [8,]  0.192  0.822
##  [9,]  0.979          0.187
```

```
## [10,]  0.675 -0.128  0.109
## [11,] -0.463  0.218  0.208
## [12,] -0.168  0.471  0.277
## [13,]  0.230  0.296 -0.119
##
##                  [,1]  [,2]  [,3]
## SS loadings     2.359 2.236 2.091
## Proportion Var 0.181 0.172 0.161
## Cumulative Var 0.181 0.353 0.514
##
## The BIC is:  303.8067
```

Here I performed a factor analysis using the varimax rotation for multiple factor values. What I found is that 3 factors is best to explain the data because it has the lowest BIC value.

Part 3.) Use a multivariate analysis of variance to investigate whether cultivars have an effect on the average hue,ash content, and color intensity of the wine. Set-up the model, and summarize the analysis and the results. In doing so, evaluate the necessary assumptions in your model. [15 points; 1 page]

```
library(car)
```

```
## Loading required package: carData
```

```
# Fit a linear Model
# we want to see if the cultivar has an effect on the avg.
# Hue(V4), ash content (V11) and color intensity( V12).
wine_lm <- lm(cbind(V4,V11, V12)~V1, data=wine)
# Run Manova
wine_manova <- Manova(wine_lm)
summary(wine_manova)
```

```
##
## Type II MANOVA Tests:
##
## Sum of squares and products for error:
##            V4       V11        V12
## V4  13.289009  30.62850  -1.168911
## V11 30.628501 884.13708 -33.558352
## V12 -1.168911 -33.55835   5.722776
##
## ----------------------------------------
##
## Term: V1
##
## Sum of squares and products for the hypothesis:
##             V4        V11         V12
## V4   0.0328310  -1.484685   0.3401699
## V11 -1.4846847  67.140462 -15.3831745
## V12  0.3401699 -15.383175   3.5245819
```

```
##
## Multivariate Tests: V1
##                   Df test stat approx F num Df den Df     Pr(>F)
## Pillai             1 0.3921984 37.42587      3    174 < 2.22e-16 ***
## Wilks              1 0.6078016 37.42587      3    174 < 2.22e-16 ***
## Hotelling-Lawley   1 0.6452736 37.42587      3    174 < 2.22e-16 ***
## Roy                1 0.6452736 37.42587      3    174 < 2.22e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I made a linear model with the formula: (Avg. Hue, Ash Content, Color Intensity) ~ Wine Cultivar to observe whether the wine cultivar has an effect on these other variables. After running the Manova on this linear model, I found that the Cultivar does in fact have a very significant effect on these variables. In order to make inferences about the difference between cultivars we must make several assumptions including: - Our sample should come from a multivariate normal distribution. This assumption is mostly true, as we can see from Part 1. THe 2nd cultivar was not found to support multivariate normality. - Each unit should respond independently of any other unit. In this case, I believe our 'unit' is the wine itself. Although these wines are coming from the same part of Italy and 3 different cultivars, we can assume that each sample was taken independently. - Covariance Matrices are Homogenous for all groups. For this we can use the Bartlett test.As we see below the covariance matrices are significantly different so this assumption is not valid.

```
## [1] 3
## ------------------------------------------------
##  MBox Chi-sqr. df P
## ------------------------------------------------
##   764.8065   684.2031         182       0.0000
## ------------------------------------------------
## Covariance matrices are significantly different.

## $MBox
##        1
## 764.8065
##
## $ChiSq
##        1
## 684.2031
##
## $df
## [1] 182
##
## $pValue
##            1
## 2.891851e-59
```

Part 4: Using set.seed() with seed given by the last four digits of your university ID,split the original dataset into a random training set of 128 observations and a test set of the remaining 50 observations. Call them wine.train and wine.test. Test classification rules obtained from the training set and tested on the test set using the following:

```r
# Using set.seed()
library(MASS)
set.seed(2641)
wine.train <- wine[sample(nrow(wine), 128), ]
wine.test <- wine[-sample(nrow(wine), 128), ]

# Quadratic Discriminant Analysis
wine.qda<-qda(V1 ~ ., data=wine.train)
wine.qda
```

```
## Call:
## qda(V1 ~ ., data = wine.train)
##
## Prior probabilities of groups:
##         1         2         3
## 0.3593750 0.3671875 0.2734375
##
## Group means:
##         V2       V3       V4       V5        V6       V7        V8        V9
## 1 13.69717 1.946957 2.455000 16.92391 106.41304 2.846957 3.0043478 0.2923913
## 2 12.23489 1.957660 2.247447 19.90000  94.40426 2.178511 2.0155319 0.3808511
## 3 13.14971 3.394000 2.457714 21.52857  98.17143 1.742857 0.7325714 0.4740000
##        V10      V11       V12      V13       V14
## 1 1.910000 5.609565 1.0600000 3.174565 1120.9130
## 2 1.603617 3.091277 1.0680851 2.761915  513.1064
## 3 1.178857 7.289714 0.6768571 1.734286  633.4286
```

```r
# Calculate the Error Rate using CV
wine.qda_2<-qda(V1 ~ ., data=wine.train, CV = T)
qda_error_rate <-mean(wine.qda_2$class!=wine.train$V1)
qda_error_rate
```

```
## [1] 0.0234375
```

Detailed Summary: Here I ran a quadratic discriminant analysis on the wine training data set with the formula.

$Cultivar = Alcohol + Malicacid + Ash + Alkalinityof Ash + Magnesium + TotalPhenols + Flavanoids + Nonfavanoidphenols$

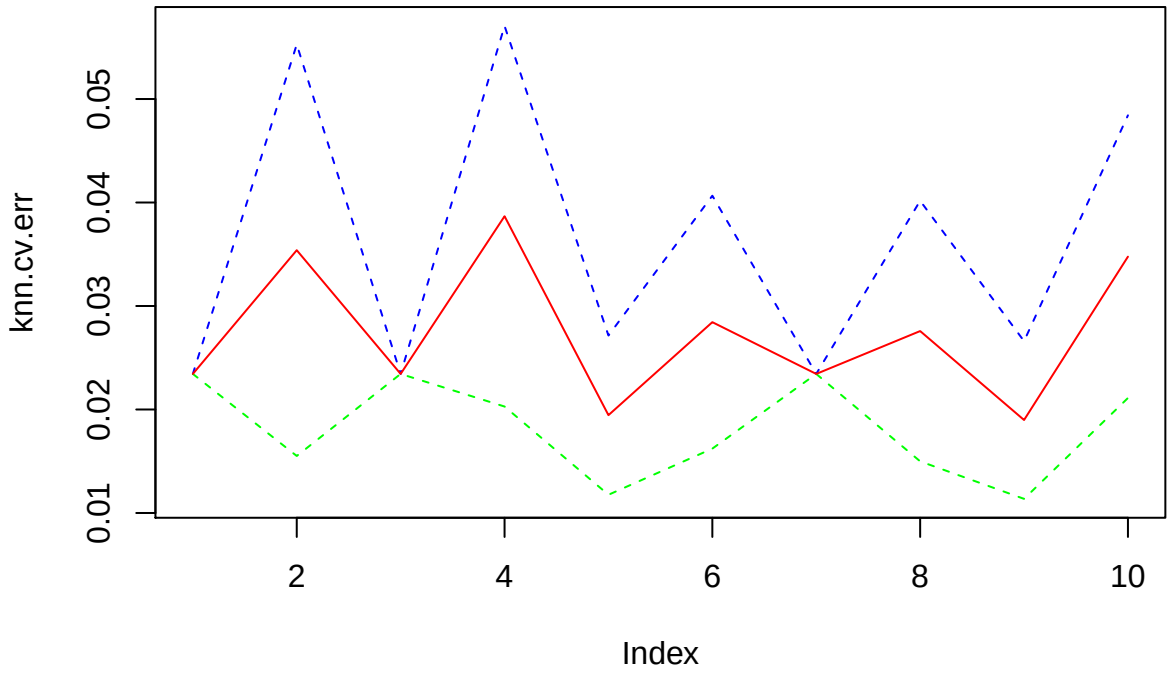
What I found is that the quadratic discriminant analysis had a relatively low error rate of 0.0078125.

```r
# K-Nearest neightbor classification with a cross-validated choice of k
library(class)
# Set up the cross-validated error rate of the K-NN method
wine.train.scaled <- scale(wine.train[,-1])
knn.cv.err<-NULL
knn.cv.sd<-NULL

for(i in 1:10) {
  temp<-NULL
  for(j in 1:100)
    temp <-c(temp,mean(knn.cv(wine.train.scaled, cl = wine.train$V1, k =i)!=wine.train$V1))
```

```
  knn.cv.err<-c(knn.cv.err,mean(temp))
  knn.cv.sd<-c(knn.cv.sd,sd(temp))}

plot(knn.cv.err, xlim = c(1, 10),
     ylim=c(min(knn.cv.err - 1.96 * knn.cv.sd),
       max(knn.cv.err + 1.96 * knn.cv.sd)), type = "n")
lines(knn.cv.err + 1.96 * knn.cv.sd, lty = 2, col = "blue")
lines(knn.cv.err - 1.96 * knn.cv.sd, lty = 2, col = "green")
lines(knn.cv.err, col = "red")
```

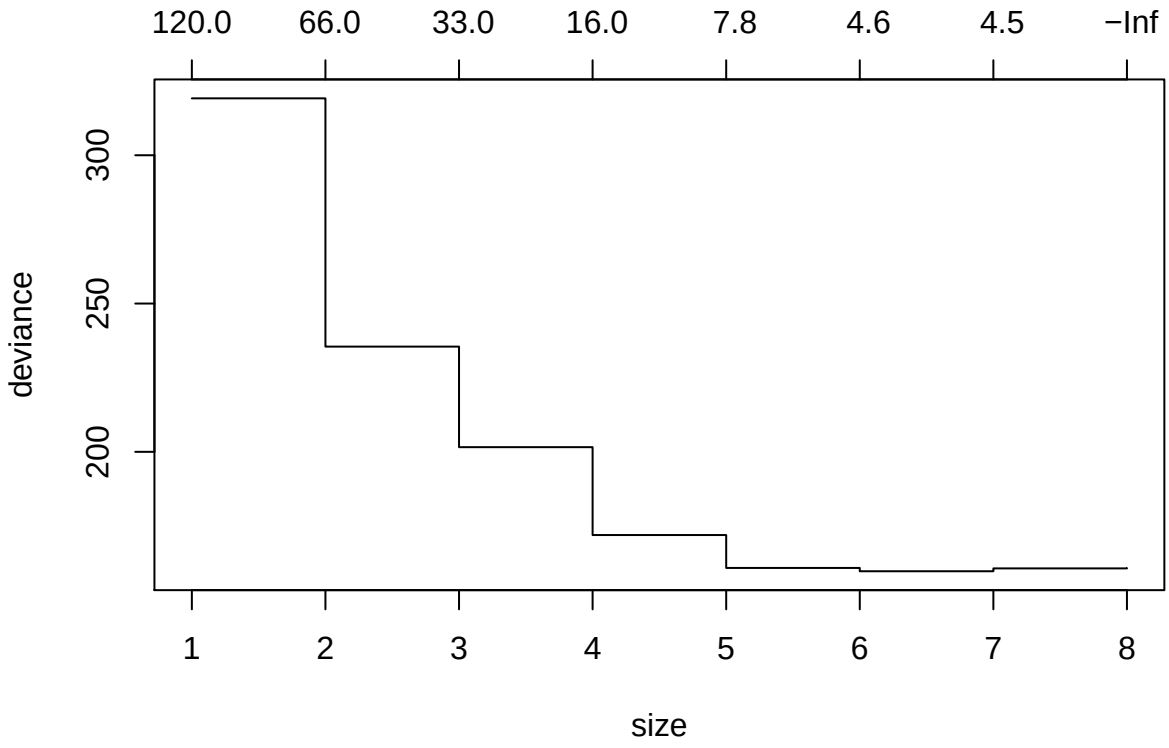



```
# Misclassification Rate
mean(knn(train = wine.train.scaled, cl = wine.train$V1, test = wine.train.scaled, k = 5)!=wine.train$V1)
```

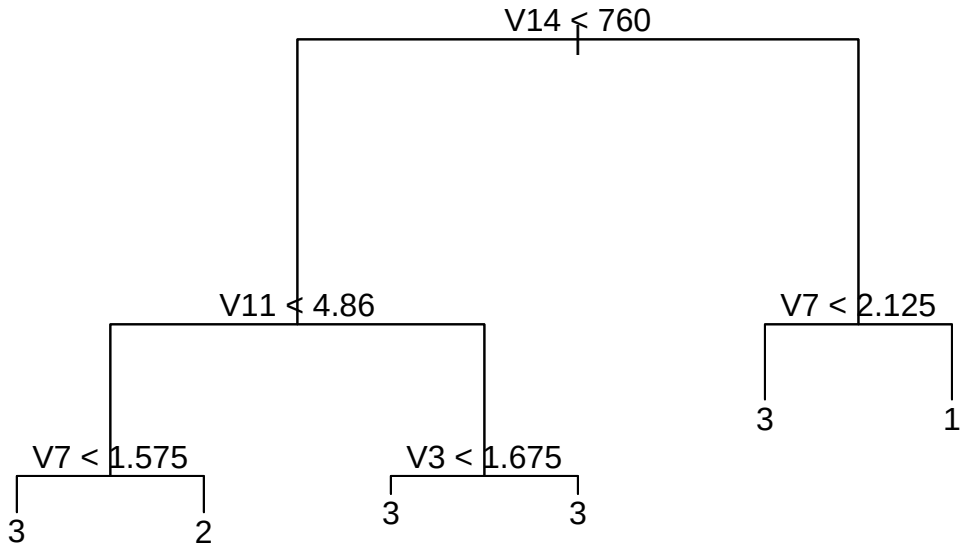
```
## [1] 0.015625
```

Detailed Summary: We can see here that the optimal k for the wine training dataset is 5 because that is the lowest point on the graph where the red line reaches. The resulting error rate is 0.0234375.

```
# Best cross-validated classification tree
library(tree)
wine.tree <-tree(formula =factor(V1)~., data = wine.train)
wine.tree.cv <-cv.tree(wine.tree, K =nrow(wine.train))
plot(wine.tree.cv)
```




```
wine.prune.tree <-prune.tree(wine.tree, best = 6)
plot(wine.prune.tree)
text(wine.prune.tree)
```




```
wine.tree.pred <-apply(predict(wine.prune.tree),1,which.max)
mean(wine.tree.pred!=wine.train$V1)
```

```
## [1] 0.046875
```

Detailed Summary: When using a cross validate classification tree, I chose 6 as the optimal number of decisions to put the tree. The resulting error rate is 0.0234375.

Out of all the methods, we see that the quadratic discriminant analysis is best to classify the data.

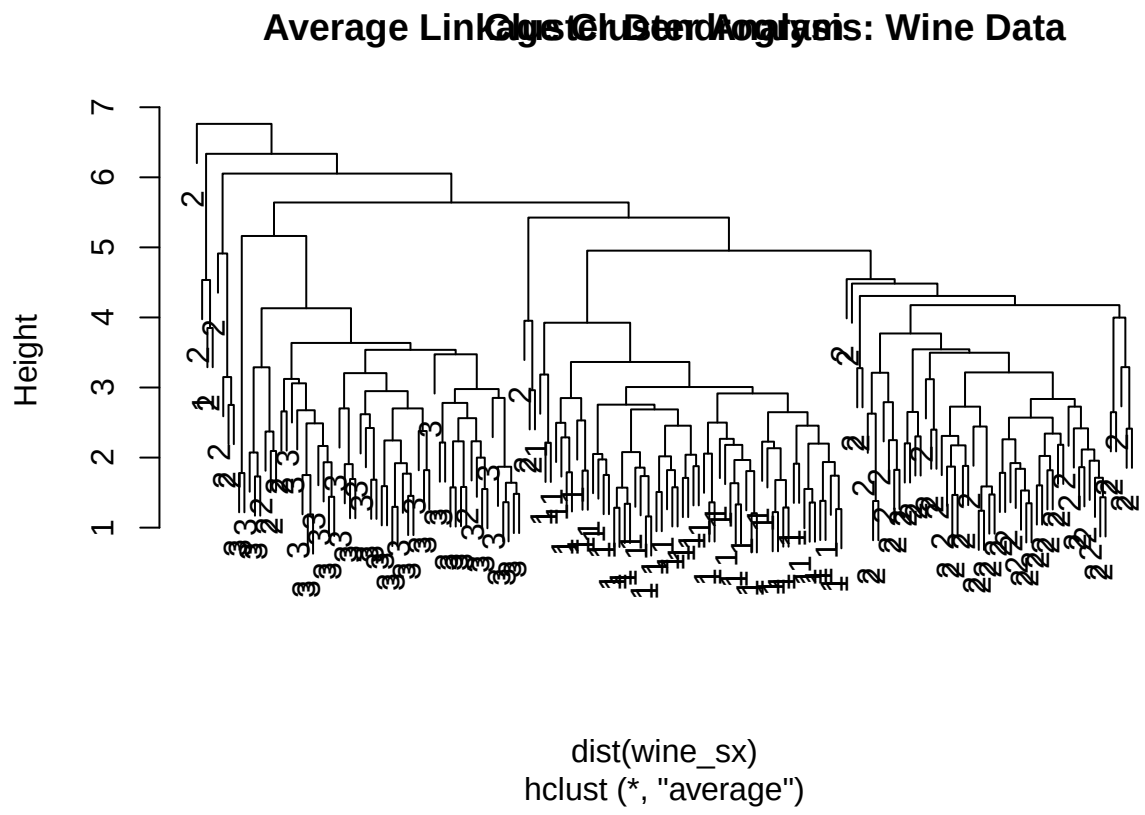
Part 5: Ignoring the cultivar information,group the constituent attributes using:

a.) Hierarchical clustering with average linkage, and K = 3 groups.

```
wine_cultivar<-wine[ ,1]
wine_data<-wine[ ,-1]

#  Standardize the data
wine_mean<-apply(wine_data,2,mean)
wine_std<-sqrt(apply(wine_data,2,var))
wine_sx<-sweep(wine_data,2,wine_mean,FUN="-")
wine_sx<-sweep(wine_sx,2,wine_std,FUN="/")
```

```r
hc <- hclust(dist(wine_sx),method="average")
plot(hc,label=wine_cultivar)
title("Average Linkage Cluster Analysis: Wine Data")
```



b.) K-means clustering with appropriate initialization and with K = 3.

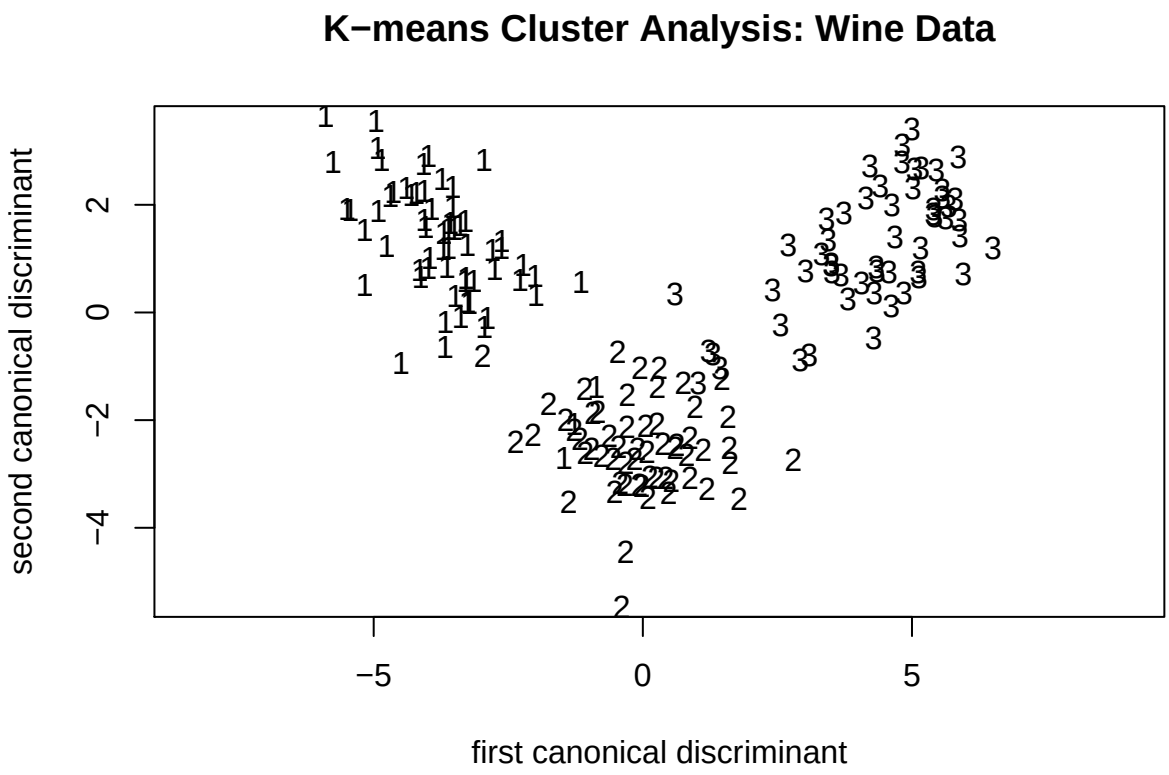
```r
library(MASS)

# K-Means
kmnsinithcl <- function(x.data, nclus, ncut = nclus, hcl.tree)
{
    x.hcl <- hcl.tree
    x.cl <- cutree(x.hcl, k = ncut)
    data.x <- data.frame(x.data, cl = x.cl)
    means <- aggregate(. ~ cl, data = data.x, FUN = mean)
    return(kmeans(x.data,centers= means[, -1]))
}

hc_2 <- hclust(dist(wine_sx),method="ward.D2")
km <- kmnsinithcl(wine_sx, nclus = 3, ncut = 3, hcl.tree = hc_2)


a <- lda(wine_sx, km$cluster)
```

```
scores <- as.matrix(wine_sx) %*% a$scaling[ ,1:2]
eqscplot(scores, type="n", xlab="first canonical discriminant",
         ylab="second canonical discriminant")
text(x = scores[ ,1], y = scores[ ,2], cutree(hc_2, 3))
title("K-means Cluster Analysis: Wine Data")
```



Here we can see that the K-means cluster analysis resulted in 3 pretty distinctive clusters. There is a small overlap betwee cluster 2 and 3, but otherwise it is pretty separated.

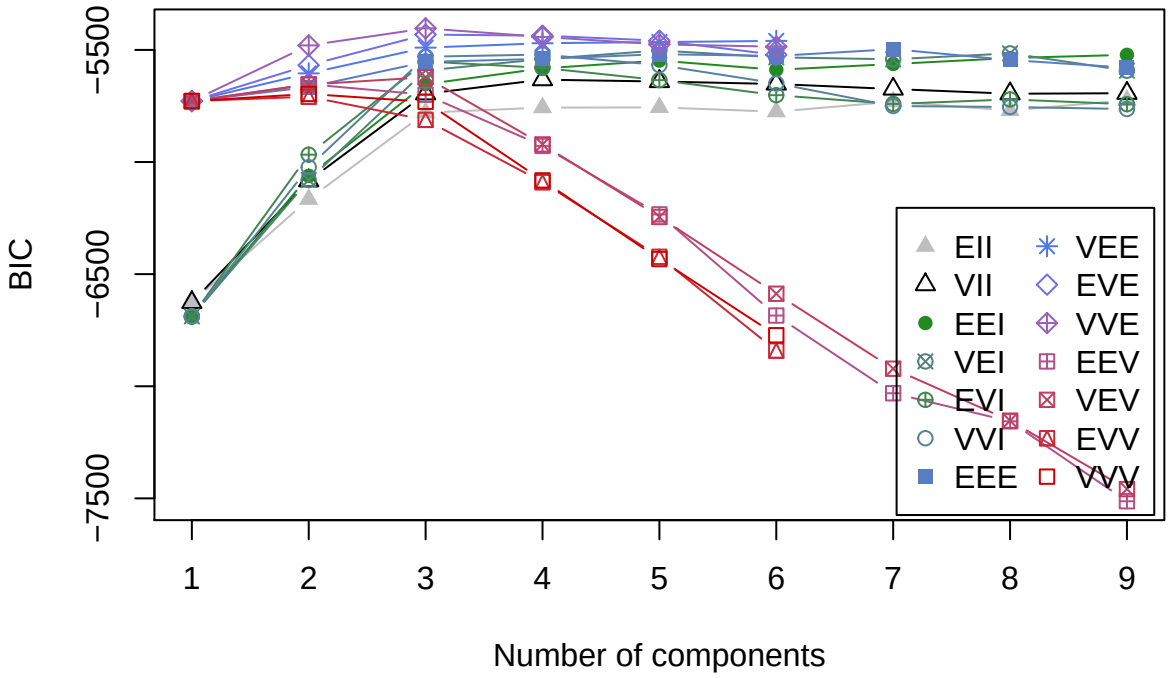
c.) Model-based clustering,with BIC to determine the number of groups and the variance-covariance structure.

```
library(mclust)
```

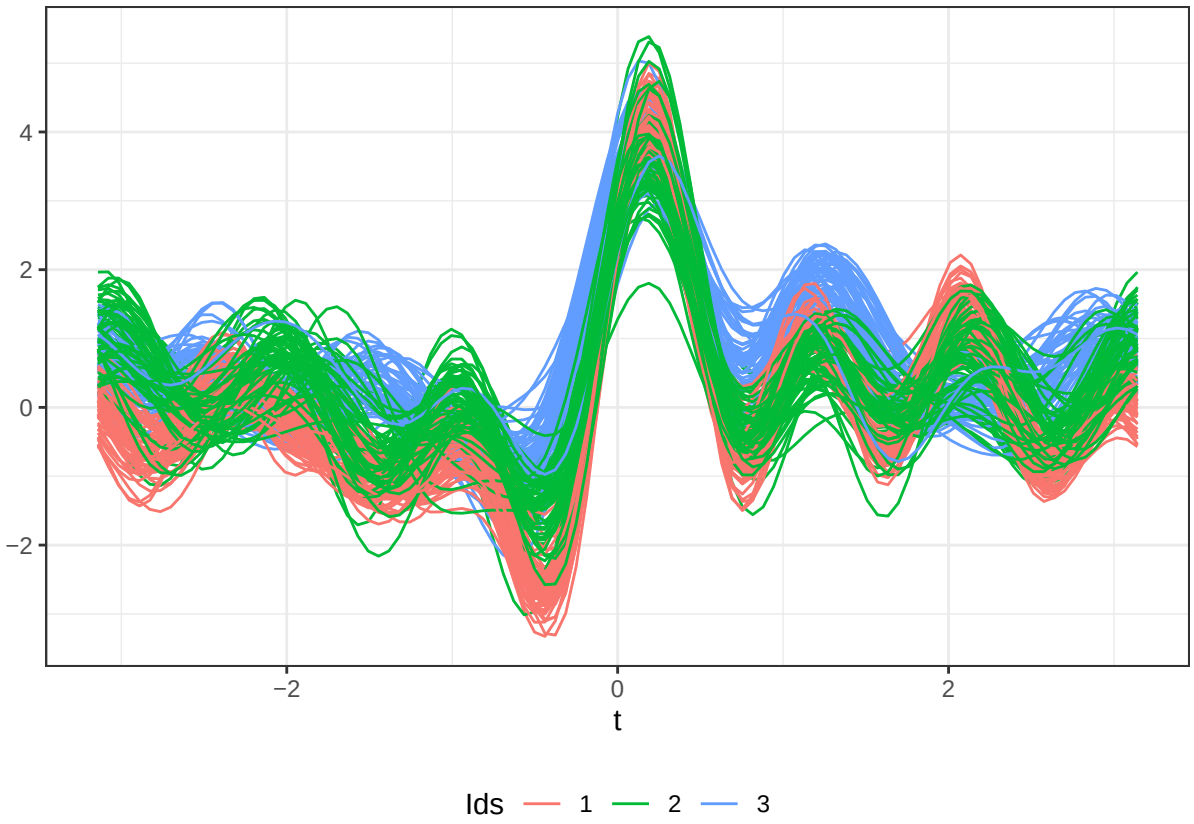
```
## Package 'mclust' version 5.4.7
## Type 'citation("mclust")' for citing this R package in publications.
```

```
source('~/Desktop/stat_501/ggandrews.R')
hc_3 <- Mclust(wine_sx)

plot(hc_3$BIC)
title("Model Based CLuster Analaysis BIC Values: Wine Data")
```




```
df <- data.frame(hc_3$classification, wine[,-1])
ggandrews(df = df, clr = 1, return_value=F)
```



THe model based cluster resulted in a 3 cluster solution for clustering the wines.

Part 6.) Perform an appropriate principal components analysis to reduce the dimensionality of the variables in the dataset. How many principal components are enough to account for at least 80% of the variation? Provide an interpretation for the first few principal components, if possible. [10 points; 2 pages]

```
pca <- prcomp(wine[,-1], scale = T)
print("PCA SUMMARY:")
```

```
## [1] "PCA SUMMARY:"
```

```
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6     PC7
## Standard deviation     2.169 1.5802 1.2025 0.95863 0.92370 0.80103 0.74231
## Proportion of Variance 0.362 0.1921 0.1112 0.07069 0.06563 0.04936 0.04239
## Cumulative Proportion  0.362 0.5541 0.6653 0.73599 0.80162 0.85098 0.89337
##                           PC8     PC9    PC10    PC11    PC12    PC13
## Standard deviation     0.59034 0.53748 0.5009 0.47517 0.41082 0.32152
## Proportion of Variance 0.02681 0.02222 0.0193 0.01737 0.01298 0.00795
## Cumulative Proportion  0.92018 0.94240 0.9617 0.97907 0.99205 1.00000
```

```r
pca$rotation[,1:3]
```

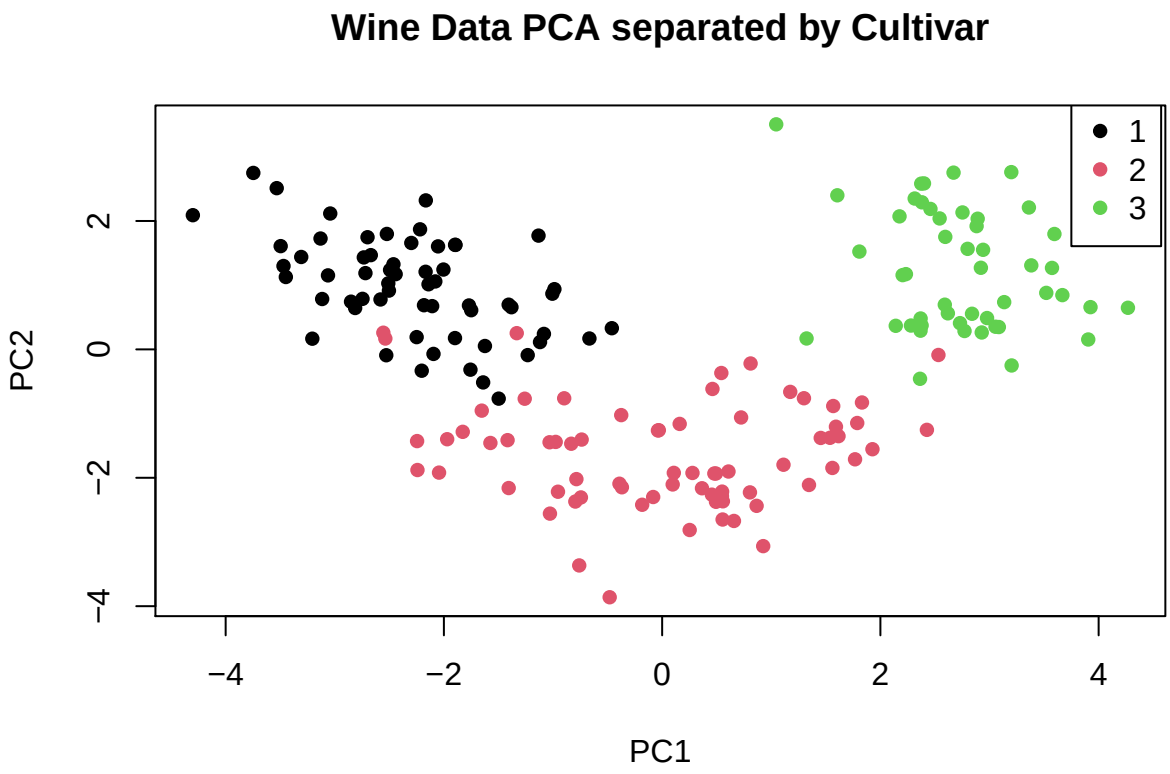
```
##              PC1          PC2         PC3
## V2  -0.144329395  0.483651548 -0.20738262
## V3   0.245187580  0.224930935  0.08901289
## V4   0.002051061  0.316068814  0.62622390
## V5   0.239320405 -0.010590502  0.61208035
## V6  -0.141992042  0.299634003  0.13075693
## V7  -0.394660845  0.065039512  0.14617896
## V8  -0.422934297 -0.003359812  0.15068190
## V9   0.298533103  0.028779488  0.17036816
## V10 -0.313429488  0.039301722  0.14945431
## V11  0.088616705  0.529995672 -0.13730621
## V12 -0.296714564 -0.279235148  0.08522192
## V13 -0.376167411 -0.164496193  0.16600459
## V14 -0.286752227  0.364902832 -0.12674592
```

Five principal components are enough to account for at least 80.16% of the variation of the data.

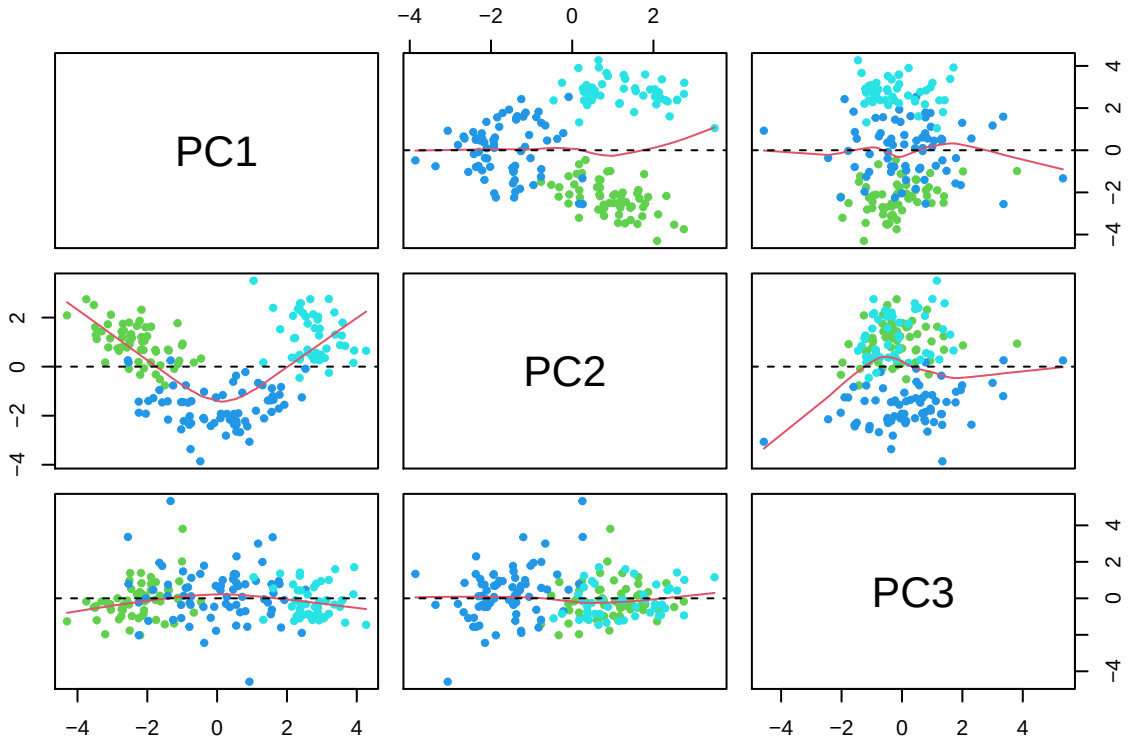
When looking at the first 3 principal components wee see that there is a contrast in the variable means (V2,V6,V7,V8,V10,V12,V13,V14) and (V3,V4,V5,V9, V11). This first principal component accounts for 36.2 of the total variation of the data. When looking at the second principal component there is a contrast between the variable means for (V5, V8, V12, V13) and the rest of the variables. With the second component, 55.4% of the total variation in the data is accounted for. For the third PC, we mainly see a contrast in the variable means (V2, V11, V14) and the rest of the variables.

Part 7.) Carrying on with the "first few" principal components analysis, does reducing dimensionality of the dataset using principal components in this way improve the distinctiveness of the cultivars as explained by these new projections of the dataset? Answer this question descriptively (visually). [5 points; 2 pages]

```r
plot(pca$x[,1:2], pch=16, col=as.numeric(wine[,1]), main="Wine Data PCA separated by Cultivar")
legend(3.75, 4, pch=16, col=unique(as.numeric(wine[,1])),legend=unique(wine[,1]))
```




```
pairs(pca$x[,1:3],
      panel=function(x,y){panel.smooth(x,y, col = as.numeric(wine[,1])+2,
        pch = 20, cex = 1)
        abline(lsfit(x,y),lty=2) })
```



The first plot is only showing the first 2 principal components. We can see that for the most part the three cultivars are separated on this plot.

I included the second plot so we could look at more than 2 PCs. Again we see good separation between the 3 cultivars using the first 2 PC's. PC1 and PC3 also show good separation between the 3 cultivars, but when we look at PC2 and PC3 there isn't as much distinctiveness between the 3 cultivars.