STAT 223 HW 3

Galen Byrd

4/9/2018

1

A

```
library(readxl)
states <- read_excel("USStates.xlsx")
```

```
## Warning in strptime(x, format, tz = tz): unknown timezone 'zone/tz/2018c.
## 1.0/zoneinfo/America/New_York'
```

```
#Separating data into Obama / McCain voters
states<-states[,6:16]
states<-states[,-2:-6]
obama <- subset(states, ObamaMcCain=="O")[,-1]
mccain <- subset(states, ObamaMcCain=="M")[,-1]
#Calculating the basic statistics
p <- ncol(obama)
n1 <- nrow(obama)
n2 <- nrow(mccain)
#Calculating the mean vectors and covariance matrices
mean.obama <- colMeans(obama)
mean.mccain <- colMeans(mccain)
S.obama <- var(obama)
S.mccain <- var(mccain)
S.pl <- ((n1-1)*S.obama+(n2-1)*S.mccain)/(n1+n2-2)
#Calculating Hotelling's T2
T2 <- n1*n2/(n1+n2)*t(mean.obama-mean.mccain)%*%solve(S.pl)%*%(mean.obama-mean.mccain)
#Calculating the critical value
a <- p*(n1+n2-2)/(n1+n2-p-1)
crit.val <- a*qf(.95,p,n1+n2-p-1)
T2>crit.val
```

```
##      [,1]
## [1,] TRUE
```

```
p.val <- 1-pf(1/a*T2,p,n1+n2-p-1)
# Reject null. There is evidence (p-val=2.169745e-05) to suggest a difference among
# obama/mccain voters in at least one of the variables "Smokers, PhysicalActivity,
# Obese, College, NonWhite".
```

B

```
#Finding the E and H matrix using MANOVA
m1 <- manova(cbind(Smokers, PhysicalActivity, Obese, College, NonWhite)
             ~as.factor(ObamaMcCain),data=states)
```

```
H <- summary(m1)$SS[[1]]
E <- summary(m1)$SS[[2]]
#Calculating the eigenvalues and vectors for the discriminant analysis
e.vals <- Re(round(eigen(solve(E)%*%H)$values,digits=5))
e.vecs <- Re(round(eigen(solve(E)%*%H)$vectors,digits=5))
(a1 <- e.vecs[,1])
```

```
## [1]  0.65380  0.26596 -0.53546  0.45943  0.06338
```

```
# So the variables that contribute to the separation from most to least are:
# Smokers,Obese,College,PhysicalActivity,NonWhite
```

C

```
p.vals <- rep(1,5)
p.vals[1]<-t.test(obama$Smokers,mccain$Smokers)$p.value
p.vals[2]<-t.test(obama$Obese,mccain$Obese)$p.value
p.vals[3]<-t.test(obama$College,mccain$College)$p.value
p.vals[4]<-t.test(obama$PhysicalActivity,mccain$PhysicalActivity)$p.value
p.vals[5]<-t.test(obama$NonWhite,mccain$NonWhite)$p.value
p.vals
```

```
## [1] 6.531886e-02 5.657510e-05 1.738811e-06 1.407454e-02 3.868426e-01
```

```
# ranking from smallest p-value to largest: College,Obese,PhysicalActivity,Smokers,NonWhite
```

D

```
N <- nrow(states)
k <- 2
full.lambda <- summary(m1, test="Wilks")$stats[1,2]
# removing NonWhite
partial.lambda <- summary(manova(cbind(Smokers,PhysicalActivity,Obese,College)
                                 ~as.factor(ObamaMcCain),data=states),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 2.107215
```

```
# removing Smokers
partial.lambda <- summary(manova(cbind(NonWhite,PhysicalActivity,Obese,College)
                                 ~as.factor(ObamaMcCain),data=states),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 6.013514
```

```
# removing PhysicalActivity
partial.lambda <- summary(manova(cbind(NonWhite, Smokers, Obese, College)
                                 ~as.factor(ObamaMcCain),data=states),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 1.326244
```

```r
# removing Obese
partial.lambda <- summary(manova(cbind(NonWhite, Smokers, PhysicalActivity, College)
                                 ~as.factor(ObamaMcCain),data=states),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 3.434189
```

```r
# removing College
partial.lambda <- summary(manova(cbind(NonWhite, Smokers, PhysicalActivity, Obese)
                                 ~as.factor(ObamaMcCain),data=states),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 9.245268
```

```r
# Ranking which variable contributes to group separation most to least: College,Smokers,
# Obese,NonWhite,PhysicalActivity
```

E

```r
# NonWhite is last in two and second to last in one so that contributes least to group
# separation. College was first twice and third once so it contributes most. All of the
# testing methods disagree in some way
```

F

```r
zO <- as.matrix(obama)%*%a1
zM <- as.matrix(mccain)%*%a1
t.test(zO,zM)
```

```
##
##  Welch Two Sample t-test
##
## data:  zO and zM
## t = 6.72, df = 47.802, p-value = 2.014e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.198568 5.930211
## sample estimates:
## mean of x mean of y
##  38.88697  34.32258
```

```r
# This p-value is smaller than all of the p-values in C. This means the discriminant
# function worked and increased the distance between the variables
```

2

A

```
library(MASS)
LDA <- lda(ObamaMcCain~Smokers+PhysicalActivity+Obese+College+NonWhite,
           data=states,prior=rep(1,k)/k)
predLDA <- predict(LDA)$class
(error <- mean(states$ObamaMcCain != predLDA) )
```

```
## [1] 0.2
```

```
#Apparent error rate = .2
```

B

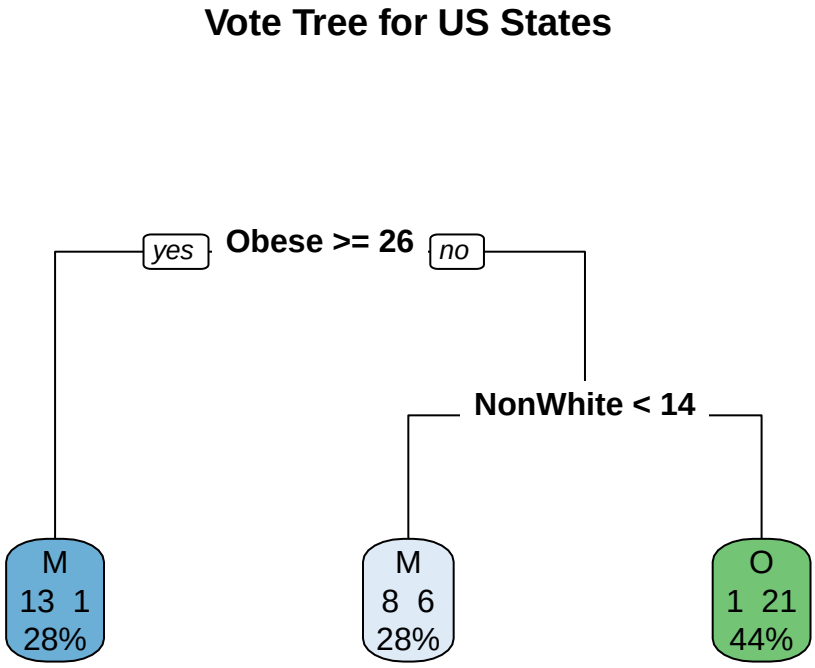
```
library(class)
m3 <- knn(train=states[,-1], test=states[,-1], cl = states$ObamaMcCain, k=5)
tab.knn <- table(Winner = states$ObamaMcCain, Predicted = m3)
# Use k=5 because sqrt(mean(22+28))
(sum(diag(tab.knn))/sum(tab.knn))
```

```
## [1] 0.8
```

```
#Apparent error rate = .2
```

C

```
library(rpart)
library(rpart.plot)
m5 <- rpart(ObamaMcCain~Smokers+PhysicalActivity+Obese+College+NonWhite,
            data=states,method="class")
rpart.plot(m5, main= "Vote Tree for US States", type=0,extra=101)
```




```
p5 <- predict(m5, states, type="class")
tab.ct <- table(Winner = states$ObamaMcCain, Predicted = p5)
(sum(diag(tab.ct))/sum(tab.ct))
```

```
## [1] 0.84
```

```
#Apparent error rate = .16
```

D

```
LDA.CV <- lda(ObamaMcCain~Smokers+PhysicalActivity+Obese+College+NonWhite,
              data=states,prior=rep(1,k)/k, CV=T)
(error <- mean(states$ObamaMcCain != LDA.CV$class))
```

```
## [1] 0.28
```

```
#Apparent error rate using cross validation = .28

m4 <- knn.cv(train=states[,-1], cl = states$ObamaMcCain, k=5)
tab.knncv <- table(Winner=states$ObamaMcCain, Predicted = m4)
(sum(diag(tab.knncv))/sum(tab.knncv))
```

```
## [1] 0.62
```

```
#Apparent error rate using cross validation = .38

pred.ct.cv <- rep(0,N)
#Looping through each state, removing it,
#then predicting which region it would be classified in
for (i in 1:N){
  m.ct.cv <- rpart(ObamaMcCain~Smokers+PhysicalActivity+Obese+College+NonWhite,
                   data=states[-i,], method="class")
  pred.ct.cv[i] <- predict(m.ct.cv, states[i,-1], type="class")
}
```

```
pred.ct.cv <- factor(pred.ct.cv,labels=c("M","O"))
tab.ct.cv <- table(Actual = states$ObamaMcCain, Predicted = pred.ct.cv)
(sum(diag(tab.ct.cv))/sum(tab.ct.cv))
```

```
## [1] 0.8
```

```
#Apparent error rate using cross validation = .2
```

E

```
states <- read_excel("USStates.xlsx")
totElecVotes<-0
for (i in 1:N){
  if (LDA.CV$class[i]=="O"){
    totElecVotes<-totElecVotes+states[i,18]
  }
}
(totElecVotes)
```

```
##   ElectoralVotes
## 1            247
```

```
# The predicted number of electoral votes for obama is 247, which is far less than 365
```

3

A

```
# 2 Discriminant functions can be calculated for this data from min(k-1,p)
```

B

```
iris <- read.table("iris.txt", header=T)
N <- nrow(iris)
p <- ncol(iris[,-1])
k <- 3
m1 <- manova(cbind(Sepal.Length,Sepal.Width,Petal.Length,Petal.Width)
             ~as.factor(Species),data=iris)
H <- summary(m1)$SS[[1]]
E <- summary(m1)$SS[[2]]
(e.vals <- Re(round(eigen(solve(E)%*%H)$values,digits=5)))
```

```
## [1] 32.19193  0.28539  0.00000  0.00000
```

```
(e.vecs <- Re(round(eigen(solve(E)%*%H)$vectors,digits=5)))
```

```
##          [,1]     [,2]     [,3]     [,4]
## [1,]  0.20874 -0.00653  0.65786 -0.77854
## [2,]  0.38620 -0.58661  0.00881  0.41628
## [3,] -0.55401  0.25256  0.07274  0.42978
```

```
## [4,] -0.70735 -0.76945 -0.74957 -0.18941
```

C

```
(e.vals[1]/sum(e.vals))
```

```
## [1] 0.9912126
```

```
# Just one discriminant function is needed as it contributes .991 to the separation of the groups
```

D

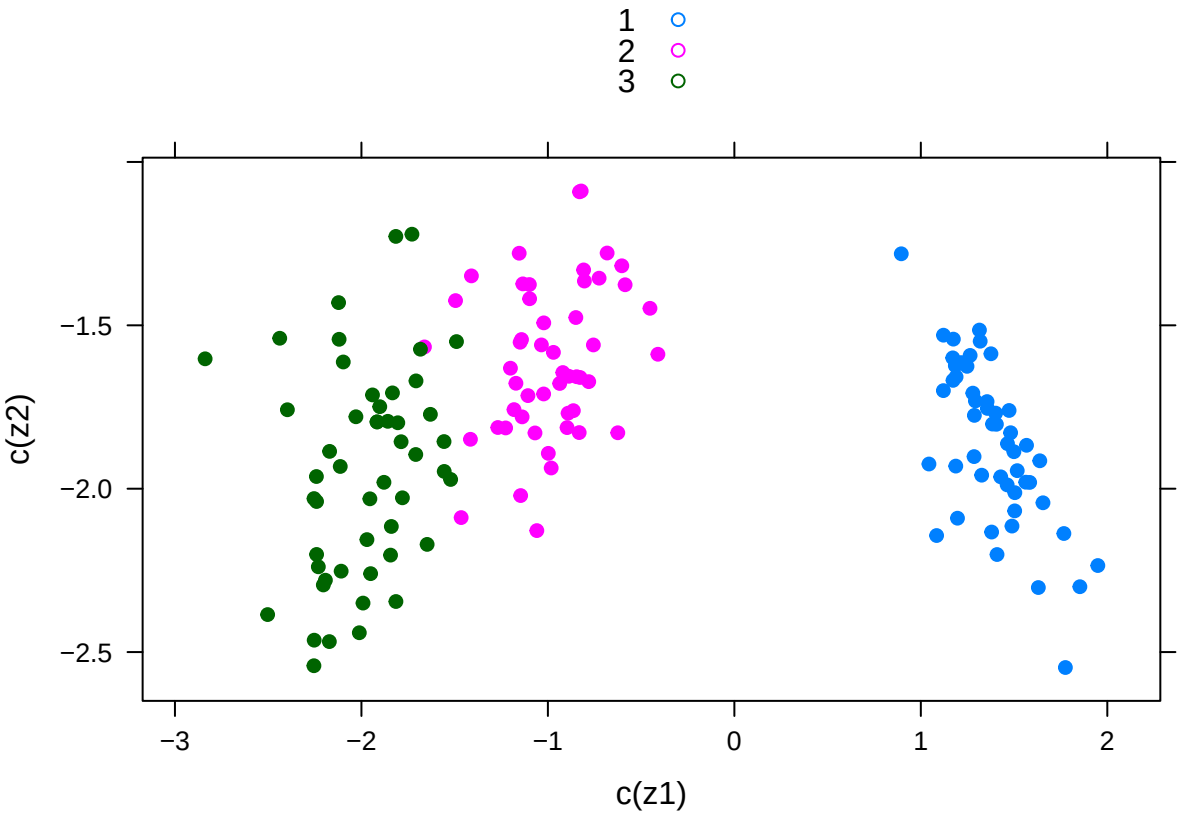
```
(a1 <- e.vecs[,1])
```

```
## [1]  0.20874  0.38620 -0.55401 -0.70735
```

```
a2 <- e.vecs[,2]
# From most to least contribution: -.707 Petal.Width, -.554 Petal.Length,
# .386 Sepal.Width, .209 Sepal.Length
```

E

```
z1 <- as.matrix(iris[,-5])%*%a1
z2 <- as.matrix(iris[,-5])%*%a2
library(lattice)
xyplot(c(z2)~c(z1), group=c(iris$Species), pch=19, auto.key = T)
```




```
# The first discriminant funtion separates the variables most, this agrees with part C
```

F

```
full.lambda <- summary(m1, test="Wilks")$stats[1,2]
# removing Petal.Width
partial.lambda <- summary(manova(cbind(Sepal.Length, Sepal.Width, Petal.Length)
                                 ~as.factor(Species),data=iris),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 24.90433
```

```
# removing Petal.Length
partial.lambda <- summary(manova(cbind(Sepal.Length, Sepal.Width, Petal.Width)
                                 ~as.factor(Species),data=iris),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 35.59017
```

```
# removing Sepal.Width
partial.lambda <- summary(manova(cbind(Sepal.Length, Petal.Length, Petal.Width)
                                 ~as.factor(Species),data=iris),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 21.93593
```

```r
# removing Sepal.Length
partial.lambda <- summary(manova(cbind(Sepal.Width, Petal.Length, Petal.Width)
                                 ~as.factor(Species),data=iris),test="Wilks")$stats[1,2]
lambda.ratio <- full.lambda/partial.lambda
(partial.F <- (N-k-p+1)/(k-1)*(1-lambda.ratio)/lambda.ratio)
```

```
## [1] 4.721152
```

```r
# Ranking which variable contributes to group separation most to least:
# Petal.Length, Petal.Width, Sepal.Width, Sepal.Length
# This ranking does agree with our standardized coefficients
```

G

```r
(qf(.95,k-1,N-k-p-1))
```

```
## [1] 3.059831
```

```r
# he critical value for part f is 3.06. The important variables in separating the
# subspecies are all of them
```

4

A

```r
LDA <- lda(Species~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
           data=iris,prior=rep(1,k)/k)
predLDA <- predict(LDA)$class
(error <- mean(iris$Species != predLDA))
```

```
## [1] 0.02
```

```r
#Apparent error rate = .02
```

B

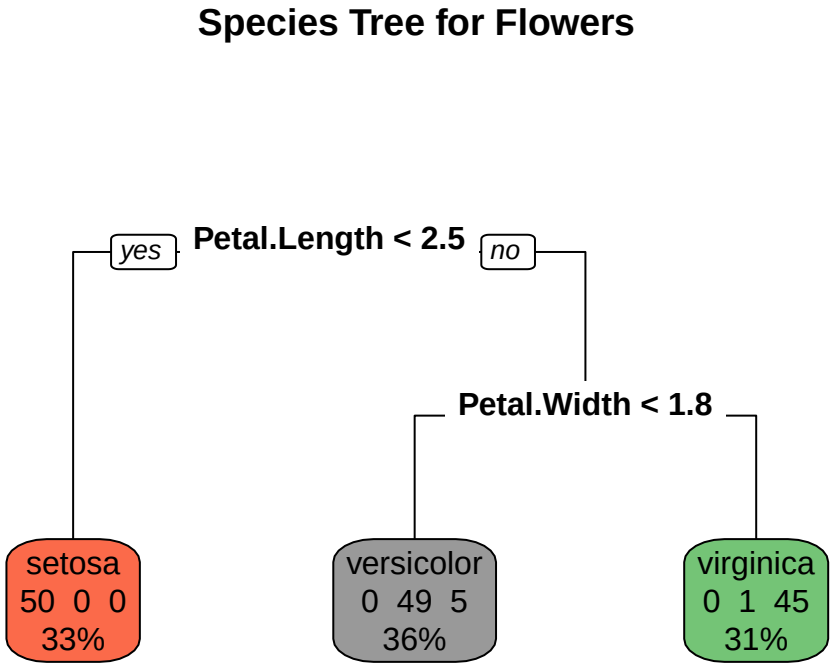
```r
# Use k=7 cuz sqrt(50)
m3 <- knn(train=iris[,-5], test=iris[,-5], cl = iris$Species, k=7)
tab.knn <- table(Winner = iris$Species, Predicted = m3)
(sum(diag(tab.knn))/sum(tab.knn))
```

```
## [1] 0.9733333
```

```r
#Apparent error rate = .02666667
```

C

```
m5 <- rpart(Species~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
            data=iris, method="class")
rpart.plot(m5, main= "Species Tree for Flowers", type=0,extra=101)
```




```
p5 <- predict(m5, iris, type="class")
tab.ct <- table(Winner = iris$Species, Predicted = p5)
(sum(diag(tab.ct))/sum(tab.ct))
```

```
## [1] 0.96
```

```
#Apparent error rate = .04
```

D

```
LDA.CV <- lda(Species~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
              data=iris,prior=rep(1,k)/k, CV=T)
(error <- mean(iris$Species != LDA.CV$class))
```

```
## [1] 0.02
```

```
#Apparent error rate using cross validation = .02

m4 <- knn.cv(train=iris[,-5], cl = iris$Species, k=7)
tab.knncv <- table(Winner=iris$Species, Predicted = m4)
(sum(diag(tab.knncv))/sum(tab.knncv))
```

```
## [1] 0.9733333
```

```
#Apparent error rate using cross validation = .02666667

pred.ct.cv <- rep(0,N)
for (i in 1:N){
  m.ct.cv <- rpart(Species~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
```

```
                  data=iris[-i,], method="class")
  pred.ct.cv[i] <- predict(m.ct.cv, iris[i,-5], type="class")
}
pred.ct.cv <- factor(pred.ct.cv,labels=c("etasoa","versicolor","virginica"))
#Creating the cross-validated confusion matrix and corresponding error
tab.ct.cv <- table(Actual = iris$Species, Predicted = pred.ct.cv)
(sum(diag(tab.ct.cv))/sum(tab.ct.cv))
```

```
## [1] 0.9333333
```

```
#Apparent error rate using cross validation = .066666666667
```