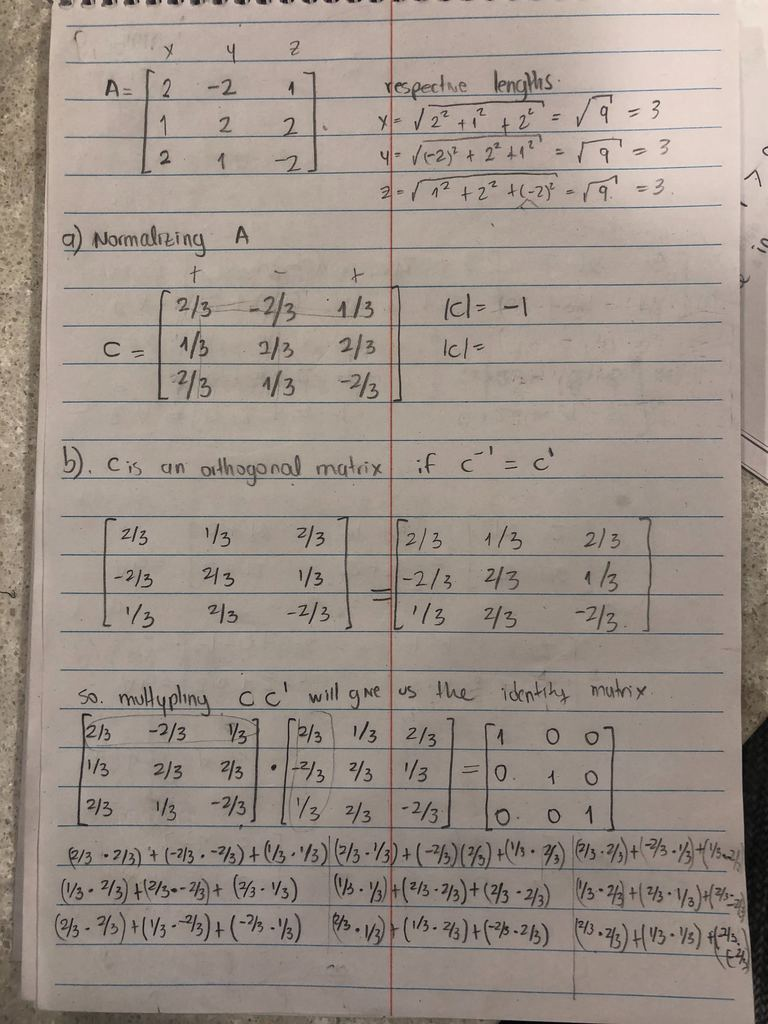
Midterm Exam

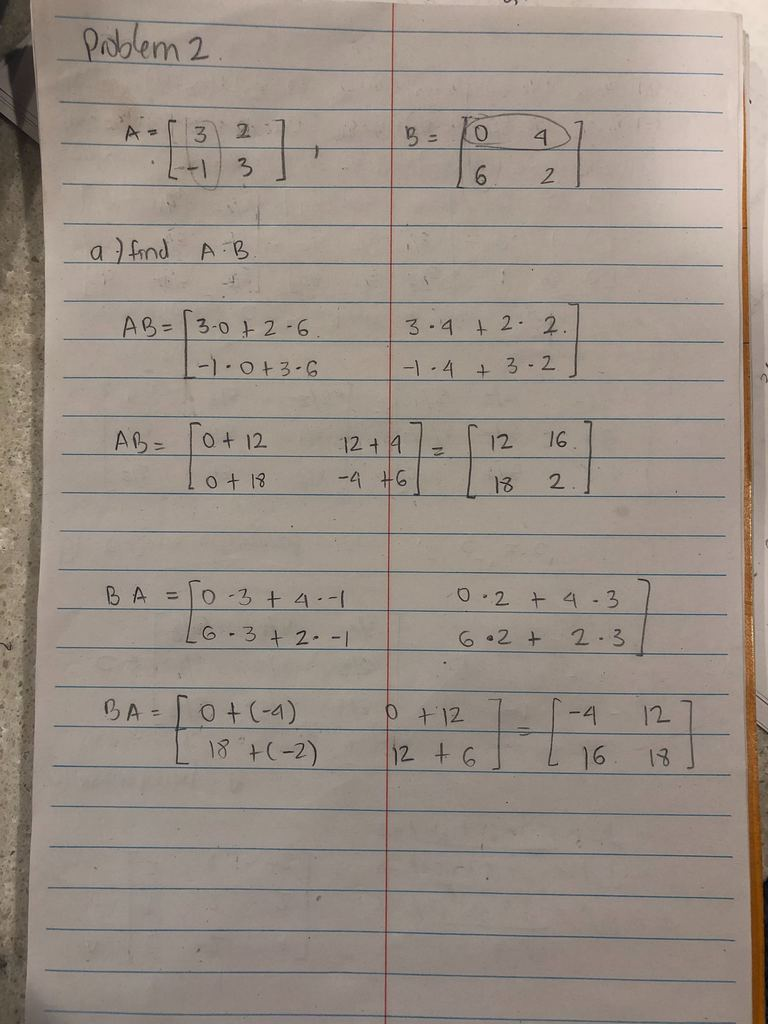
Erika Vargas

February 8, 2019

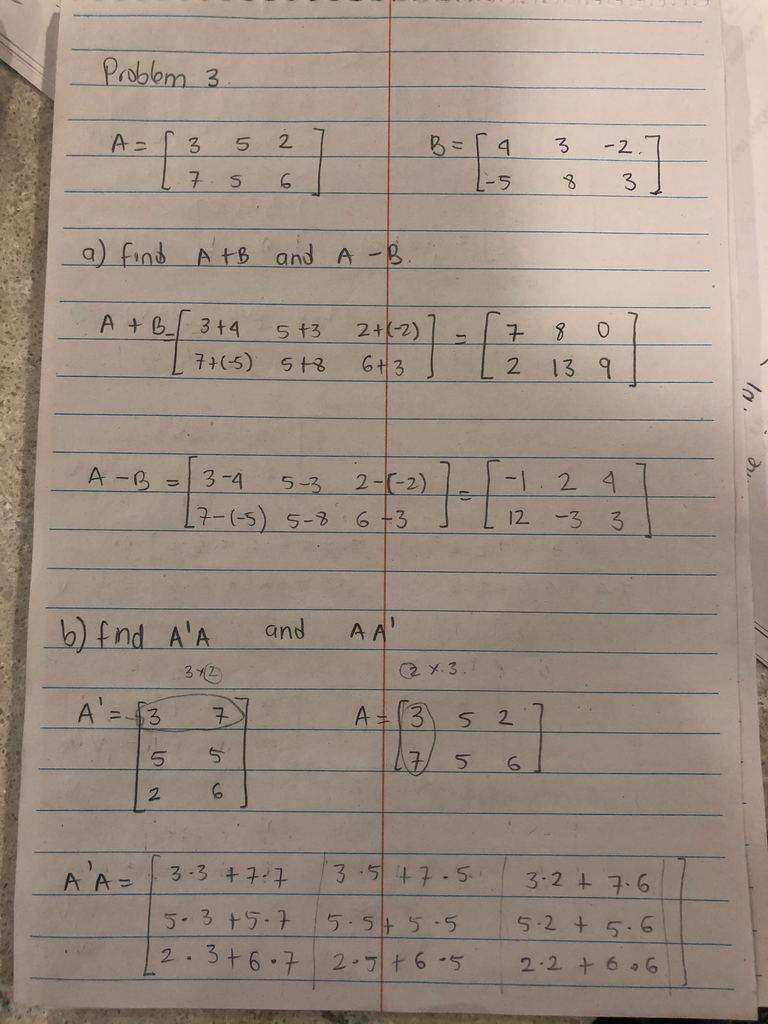
# Problem 1

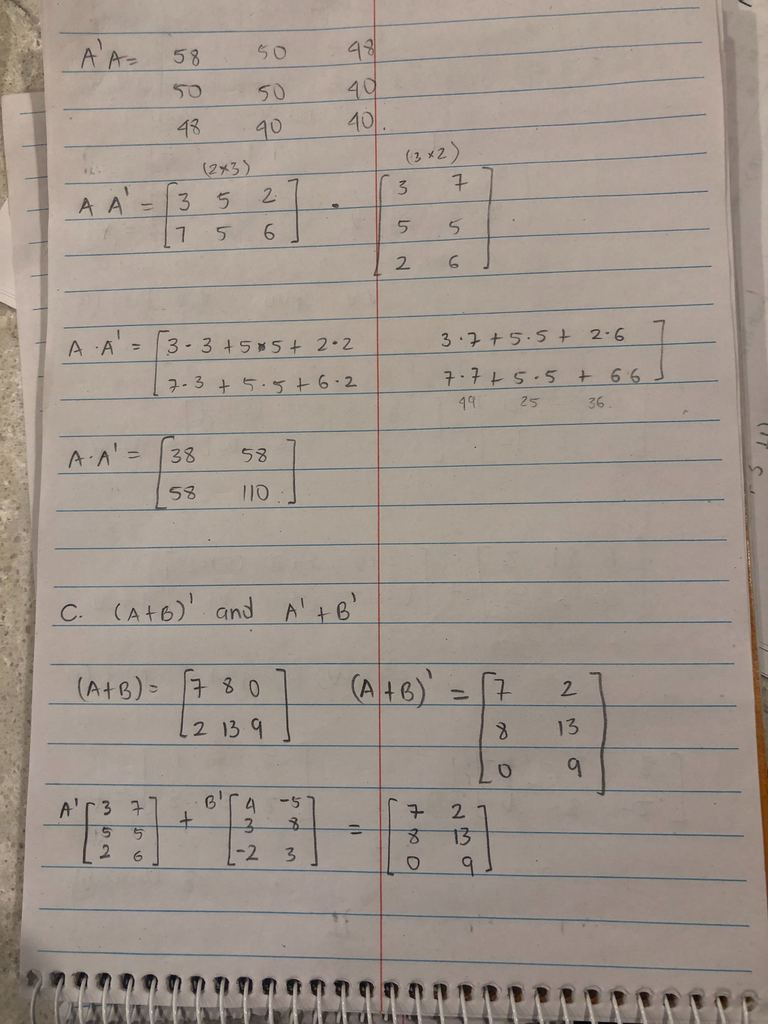


# Problem 2



# Problem 3





# Problem 4

#loading the data  
blood\_d <- read.table("~/Desktop/WINTER 2019/DA410-MULTIVARIATE-CHENG/multivariate\_analysis - 3rd Ed/multivariate\_analysis - 3rd Ed/REAGENT\_DATA.txt",   
 col.names = c('reagent', 'subject', 'y1', 'y2', 'y3'))  
blood\_d

## reagent subject y1 y2 y3  
## 1 1 1 8.0 3.96 12.5  
## 2 1 2 4.0 5.37 16.9  
## 3 1 3 6.3 5.47 17.1  
## 4 1 4 9.4 5.16 16.2  
## 5 1 5 8.2 5.16 17.0  
## 6 1 6 11.0 4.67 14.3  
## 7 1 7 6.8 5.20 16.2  
## 8 1 8 9.0 4.65 14.7  
## 9 1 9 6.1 5.22 16.3  
## 10 1 10 6.4 5.13 15.9  
## 11 1 11 5.6 4.47 13.3  
## 12 1 12 8.2 5.22 16.0  
## 13 1 13 5.7 5.10 14.9  
## 14 1 14 9.8 5.25 16.1  
## 15 1 15 5.9 5.28 15.8  
## 16 1 16 6.6 4.65 12.8  
## 17 1 17 5.7 4.42 14.5  
## 18 1 18 6.7 4.38 13.1  
## 19 1 19 6.8 4.67 15.6  
## 20 1 20 9.6 5.64 17.0  
## 21 2 1 8.0 3.93 12.7  
## 22 2 2 4.2 5.35 17.2  
## 23 2 3 6.3 5.39 17.5  
## 24 2 4 9.4 5.16 16.7  
## 25 2 5 8.0 5.13 17.5  
## 26 2 6 10.7 4.60 14.7  
## 27 2 7 6.8 5.16 16.7  
## 28 2 8 9.0 4.57 15.0  
## 29 2 9 6.0 5.16 16.9  
## 30 2 10 6.4 5.11 16.4  
## 31 2 11 5.5 4.45 13.6  
## 32 2 12 8.2 5.14 16.5  
## 33 2 13 5.6 5.05 15.3  
## 34 2 14 9.8 5.15 16.6  
## 35 2 15 5.8 5.25 16.4  
## 36 2 16 6.4 4.59 13.2  
## 37 2 17 5.5 4.31 14.9  
## 38 2 18 6.5 4.32 13.4  
## 39 2 19 6.6 4.57 15.8  
## 40 2 20 9.5 5.58 17.5  
## 41 3 1 7.9 3.86 13.0  
## 42 3 2 4.1 5.39 17.2  
## 43 3 3 6.0 5.39 17.2  
## 44 3 4 9.4 5.17 16.7  
## 45 3 5 8.1 5.10 17.4  
## 46 3 6 10.6 4.52 14.6  
## 47 3 7 6.9 5.13 16.8  
## 48 3 8 8.9 4.58 15.0  
## 49 3 9 6.1 5.14 16.9  
## 50 3 10 6.4 5.11 16.4  
## 51 3 11 5.3 4.46 13.6  
## 52 3 12 8.0 5.14 16.5  
## 53 3 13 5.5 5.02 15.4  
## 54 3 14 8.1 5.10 13.8  
## 55 3 15 5.7 5.26 16.4  
## 56 3 16 6.3 4.58 13.1  
## 57 3 17 5.5 4.30 14.9  
## 58 3 18 6.5 4.32 13.6  
## 59 3 19 6.5 4.55 16.0  
## 60 3 20 9.3 5.50 17.4  
## 61 4 1 7.9 3.87 13.2  
## 62 4 2 4.0 5.35 17.3  
## 63 4 3 6.1 5.41 17.4  
## 64 4 4 9.1 5.16 16.7  
## 65 4 5 7.8 5.12 17.5  
## 66 4 6 10.5 4.58 14.7  
## 67 4 7 6.7 5.19 16.8  
## 68 4 8 8.6 4.55 15.1  
## 69 4 9 6.0 5.21 16.9  
## 70 4 10 6.3 5.07 16.3  
## 71 4 11 5.3 4.44 13.7  
## 72 4 12 7.8 5.16 16.5  
## 73 4 13 5.4 5.05 15.5  
## 74 4 14 9.4 5.16 16.6  
## 75 4 15 5.6 5.29 16.2  
## 76 4 16 6.4 4.57 13.2  
## 77 4 17 5.4 4.32 14.8  
## 78 4 18 6.5 4.31 13.5  
## 79 4 19 6.5 4.56 15.9  
## 80 4 20 9.2 5.46 17.5

blood\_reagent <- as.factor(blood\_d$reagent)  
  
str(blood\_d) # the data contains 5 variables and 80 observations

## 'data.frame': 80 obs. of 5 variables:  
## $ reagent: int 1 1 1 1 1 1 1 1 1 1 ...  
## $ subject: int 1 2 3 4 5 6 7 8 9 10 ...  
## $ y1 : num 8 4 6.3 9.4 8.2 11 6.8 9 6.1 6.4 ...  
## $ y2 : num 3.96 5.37 5.47 5.16 5.16 4.67 5.2 4.65 5.22 5.13 ...  
## $ y3 : num 12.5 16.9 17.1 16.2 17 14.3 16.2 14.7 16.3 15.9 ...

blood\_reagent # different types of regeant

## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [36] 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4  
## [71] 4 4 4 4 4 4 4 4 4 4  
## Levels: 1 2 3 4

**Assumptions**

The data from all groups have **common variance**-covariance matrix Σ.

For this I used Bartlett test

Bartlett test of homogeneity of variances

data: y1 by blood\_reagent

Bartlett's K-squared = 0.11422, df = 3, p-value = 0.9901

Since p-value is not less than alpha, we conclude that all groups have common covariance matrix.

**Independence:** The subjects are independently sampled. We assumed the data is randomly sampled

**Normality:** The data are multivariate normally distributed.

**HYPOTHESIS**

H0: all reagent group mean vectors are all equal to one another.

HA: all reagent group mean vectors are different from one another.

*This says that the null hypothesis is false if at least one pair of treatments is different on at least one variable. The mean vectors represent 4 points in 3-dimensional space. We can compare the mean vectors for significant differences using wilks, Pillai, Roy, Hotelling.*

#conducting manova test to see if there are significant differences in the group mean vectors  
Reagent.manova <- manova(cbind(blood\_d$y1, blood\_d$y2, blood\_d$y3) ~ blood\_reagent,   
 data = blood\_d)  
summary(Reagent.manova)

## Df Pillai approx F num Df den Df Pr(>F)  
## blood\_reagent 3 0.12763 1.1257 9 228 0.3454  
## Residuals 76

# comparison of the 4 tests  
**#pillai**  
summary(Reagent.manova, test='Pillai')

## Df Pillai approx F num Df den Df Pr(>F)  
## blood\_reagent 3 **0.12763** 1.1257 9 228 **0.3454**  
## Residuals 76

**#roy**  
summary(Reagent.manova, test='Roy')

## Df Roy approx F num Df den Df Pr(>F)   
## blood\_reagent 3 0.14336 3.6319 3 76 0.01657 \*  
## Residuals 76   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#Roy output conversion -> (Roy output)/(1+Roy output)  
roy <-0.1433/(1+0.1433)  
roy

## [1] **0.1253389**

**#wilks**  
summary(Reagent.manova, test='Wilks')

## Df Wilks approx F num Df den Df Pr(>F)  
## blood\_reagent 3 **0.87265** 1.1529 9 180.25 **0.3281**  
## Residuals 76

**#Hotelling-Lawley**  
summary(Reagent.manova, test='Hotelling-Lawley')

## Df Hotelling-Lawley approx F num Df den Df Pr(>F)  
## blood\_reagent 3 **0.14561** 1.1757 9 218 **0.3118**  
## Residuals 76

**Since H0 is true, that means that all reagent mean vectors are at the same point. All of them have the same probability of rejection, hence, all 4 MANOVA test have the same type 1 error rate alpha.**

At alpha= 0.05, and the p- values of Pillai, Wilks, and Hotelling tests (0.3454 , 0.3281 , 0.3118) we do not reject H0. Hence, we conclude that the reagent groups mean vectors are the same.

The significance values for all effects are greater than 0.05, indicating they do not contribute to the model, with the exception of the test for Roy. However, this particular test provides a lower bound for the significance value, so we can conclude that the reported value of 0.016 is an overly optimistic estimate.

##The test statistics rely on the error E and hypothesis H matrices.  
n <- dim(blood\_d[1]) / length(unique(blood\_reagent))  
total.means <- colMeans(blood\_d[,3:5]) #variable means   
  
reagent.group <- split(blood\_d[,3:5], blood\_d$reagent) #each reagent patients data  
  
#mean by groups: colmeans for each variable in each reagent group   
reagent.means <- sapply(reagent.group, function(x)   
 {  
 apply(x, 2, mean)  
}, simplify = 'data.frame')

# Problem 5

# matrix for scores  
  
scores\_data<- matrix(c(90,60,90,90,90,30,60,60,60,60,60,90,30,30,30), nrow=5, ncol = 3, byrow = TRUE)  
colnames(scores\_data) <- c("Math","English","Art")  
  
Math <- c(90,90,60,60,30)  
English <- c(60,90,60,60,30)  
Art <- c(90,30,60,90,30)  
scores <- data.frame(Math,English,Art)  
  
**#a) sample covariance S**  
scores.cov <- cov(scores\_data)  
scores.cov

## Math English Art  
## Math 630 450 225  
## English 450 450 0  
## Art 225 0 900

**#b) correlation matrix R**  
scores.corr <- cor(scores\_data)  
scores.corr

## Math English Art  
## Math 1.0000000 0.8451543 0.2988072  
## English 0.8451543 1.0000000 0.0000000  
## Art 0.2988072 0.0000000 1.0000000

We can see that the scores for MATH and ENGLISH have a significant correlation, Math and Art are not significant correlated. It is interesting that the scores for English and Art don’t show any correlation, they are independent whatsoever.

#defining  
a <- matrix(c(-2, 3, 1), nrow = 3)  
ybar <- matrix(c(mean(Math), mean(English), mean(Art)), nrow = 3)  
ybar

## [,1]  
## [1,] 66  
## [2,] 60  
## [3,] 60

**# c) sample mean vector**  
zbar <- t(a) %\*% ybar  
zbar

## [,1]  
## [1,] 108

**#sample variance**  
samp\_var <- t(a) %\*% scores.cov %\*% a  
samp\_var

## [,1]  
## [1,] 1170

# Problem 6

library(Flury) #contains the flea beetles’ dataset

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

library(tidyverse)

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

## -- Attaching packages --------------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.8  
## v tidyr 0.8.2 v stringr 1.3.1  
## v readr 1.3.1 v forcats 0.3.0

## Warning: package 'ggplot2' was built under R version 3.5.2

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

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

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

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

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

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

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

## -- Conflicts ------------------------------------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(dplyr)  
library(ggplot2)  
  
#Loading and exploring the data  
data(flea.beetles)  
head(flea.beetles)

## Species TG Elytra Second.Antenna Third.Antenna  
## 1 oleracea 189 245 137 163  
## 2 oleracea 192 260 132 217  
## 3 oleracea 217 276 141 192  
## 4 oleracea 221 299 142 213  
## 5 oleracea 171 239 128 158  
## 6 oleracea 192 262 147 173

str(flea.beetles)

## 'data.frame': 39 obs. of 5 variables:  
## $ Species : Factor w/ 2 levels "oleracea","carduorum": 1 1 1 1 1 1 1 1 1 1 ...  
## $ TG : int 189 192 217 221 171 192 213 192 170 201 ...  
## $ Elytra : int 245 260 276 299 239 262 278 255 244 276 ...  
## $ Second.Antenna: int 137 132 141 142 128 147 136 128 128 146 ...  
## $ Third.Antenna : int 163 217 192 213 158 173 201 185 192 186 ...

names(flea.beetles)

## [1] "Species" "TG" "Elytra" "Second.Antenna"  
## [5] "Third.Antenna"

table(flea.beetles$Species)

##   
## oleracea carduorum   
## 19 20

summary(flea.beetles)

## Species TG Elytra Second.Antenna   
## oleracea :19 Min. :158.0 Min. :237.0 Min. :121.0   
## carduorum:20 1st Qu.:177.0 1st Qu.:262.5 1st Qu.:137.5   
## Median :184.0 Median :278.0 Median :146.0   
## Mean :186.8 Mean :279.2 Mean :147.5   
## 3rd Qu.:193.5 3rd Qu.:299.0 3rd Qu.:161.0   
## Max. :221.0 Max. :317.0 Max. :184.0   
## Third.Antenna   
## Min. :158.0   
## 1st Qu.:187.0   
## Median :197.0   
## Mean :197.9   
## 3rd Qu.:213.0   
## Max. :235.0

H.O <- flea.beetles[flea.beetles$Species=="oleracea",-1]  
H.A <- flea.beetles[flea.beetles$Species=="carduorum",-1]  
  
#mean vectors fro each species  
m1 <- apply(H.O, 2, mean)  
m2 <- apply(H.A, 2, mean)  
  
#Sp1  
Sp1<- ((19-1)\*var(H.O)+(20-1)\*var(H.A))/(19+20-2)  
Sp1 #Sp1

## TG Elytra Second.Antenna Third.Antenna  
## TG 143.55910 151.8034 42.52660 71.99253  
## Elytra 151.80341 367.7878 121.87653 106.24467  
## Second.Antenna 42.52660 121.8765 118.31408 42.06401  
## Third.Antenna 71.99253 106.2447 42.06401 208.07290

**#a) Classification function and cutoff**

## classification function   
class\_fun <- (m1 - m2) %\*% solve(Sp1)  
class\_fun

## TG Elytra Second.Antenna Third.Antenna  
## [1,] 0.345249 -0.1303878 -0.1064338 -0.1433533

## cutoff point  
cutoff <- 0.5\*(m1-m2) %\*% solve(Sp1) %\*% (m1+m2)  
cutoff

## [,1]  
## [1,] -15.80538

**# b) classification table using KNN**

#STEP 1 - normalizing numeric data  
##the normalization function is created  
nor <-function(x) {  
 return ((x -min(x))/(max(x)-min(x))) }  
  
Flea\_nor <- as.data.frame(lapply(flea.beetles[2:5], nor))  
  
summary(Flea\_nor)

## TG Elytra Second.Antenna Third.Antenna   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.3016 1st Qu.:0.3187 1st Qu.:0.2619 1st Qu.:0.3766   
## Median :0.4127 Median :0.5125 Median :0.3968 Median :0.5065   
## Mean :0.4575 Mean :0.5279 Mean :0.4212 Mean :0.5181   
## 3rd Qu.:0.5635 3rd Qu.:0.7750 3rd Qu.:0.6349 3rd Qu.:0.7143   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.0000

#STEP 2 - creating training and test dataset. we are using same dataset  
bee\_train <- Flea\_nor[1:39,]  
bee\_test <- Flea\_nor[1:39,]

#target value is species  
bee\_train\_label <-as.matrix(flea.beetles[1:39, 1])  
bee\_test\_label <- as.matrix(flea.beetles[1:39, 1])  
  
#STEP 3 - training model on data  
library(class)  
#k =3

fleaBeetle\_pred3 <- knn(train = bee\_train, test = bee\_test, cl=bee\_train\_label,k=3)  
  
  
#STEP 4 - evaluade model performance  
library(gmodels)

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

CrossTable(x=bee\_test\_label, y= fleaBeetle\_pred3,prop.chisq = FALSE)

##   
##   
## Cell Contents  
## |-------------------------|  
## | N |  
## | N / Row Total |  
## | N / Col Total |  
## | N / Table Total |  
## |-------------------------|  
##   
##   
## Total Observations in Table: 39   
##   
##   
## | fleaBeetle\_pred3   
## bee\_test\_label | carduorum | oleracea | Row Total |   
## ---------------|-----------|-----------|-----------|  
## carduorum | 19 | 1 | 20 |   
## | 0.950 | 0.050 | 0.513 |   
## | 1.000 | 0.050 | |   
## | 0.487 | 0.026 | |   
## ---------------|-----------|-----------|-----------|  
## oleracea | 0 | 19 | 19 |   
## | 0.000 | 1.000 | 0.487 |   
## | 0.000 | 0.950 | |   
## | 0.000 | 0.487 | |   
## ---------------|-----------|-----------|-----------|  
## Column Total | 19 | 20 | 39 |   
## | 0.487 | 0.513 | |   
## ---------------|-----------|-----------|-----------|  
##   
##

N <- nrow(Flea\_nor)  
  
accuracy\_k3 = sum(fleaBeetle\_pred3 == flea.beetles$Species)/N  
accuracy\_k3 #accuracy

## [1] 0.974359

Using the nearest neighbor method, the test data considered 39 observations with k=3. Out of which 19 flea beetles have been accurately predicted as H.oleracea. 19 out 39 flea beetles were accurately predicted as H. carduorum. There was 1 flea beetle that was actually H.carduorum(2) but got predicted as H.oleracea (1).

The total accuracy of the model is 97.43%.

## prediction equation  
pred\_rule <- as.data.frame(apply(flea.beetles[,2:5], 1, function(y){  
 z <- (m1-m2) %\*% solve(Sp1) %\*% y}))  
  
#classification function  
class\_fun <- ifelse(pred\_rule > as.numeric(cutoff),'oleracea','carduorum')  
  
####(Actual VS Prediction) using classification function   
beetle.pred <- table(flea.beetles$Species, class\_fun, dnn = c('Actual group','prediction group'))  
  
beetle.pred ## Classification table

## prediction group  
## Actual group carduorum oleracea  
## oleracea 0 19  
## carduorum 19 1

**# c) Misclassification rate**error\_rate = 1/length(flea.beetles$Species)  
error\_rate ##misclassification rate

## [1] 0.02564103

# Problem 7

# a) using LDA with probabilities of 50% and 50% to train the model  
library(MASS)

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

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

beetle.lda <- lda(Species ~ ., data = flea.beetles, prior=c(0.5,0.5))  
beetle.lda

## Call:  
## lda(Species ~ ., data = flea.beetles, prior = c(0.5, 0.5))  
##   
## Prior probabilities of groups:  
## oleracea carduorum   
## 0.5 0.5   
##   
## Group means:  
## TG Elytra Second.Antenna Third.Antenna  
## oleracea 194.4737 267.0526 137.3684 185.9474  
## carduorum 179.5500 290.8000 157.2000 209.2500  
##   
## Coefficients of linear discriminants:  
## LD1  
## TG -0.09327642  
## Elytra 0.03522706  
## Second.Antenna 0.02875538  
## Third.Antenna 0.03872998

Group means, which are the average of each predictor within each class. These values could suggest that the variable TG might have a slightly greater influence on oleracea beetles (194.47) than on cardourum beetles (179.55). On the other hand, the variables Elytra, and second.antenna, and third .antenna have a slightly greater influence on cardourum beetles (290.80 and 157.20 and 209.25) than on oleracea beetles (267.05 and 137.36 and 185.94).

The TG coefficient is much greater that the Elytra, second.antenna and thirs.antenna coefficients, suggesting that the TG variable has greater influence on the flea beetle’s species classification than the other variables.

lda.pred <- predict(beetle.lda)$class  
table(flea.beetles$Species, lda.pred, dnn = c('Actual Group','Predicted Group'))

## Predicted Group  
## Actual Group oleracea carduorum  
## oleracea 19 0  
## carduorum 1 19

the test data considered 39 observations with k=3. Out of which 19 flea beetles have been accurately predicted as H.oleracea. 19 out 39 flea beetles were accurately predicted as H. carduorum. There was 1 flea beetle that was actually H.carduorum(2) but got predicted as H.oleracea (1).

**# b) prediction for new flea beetle**  
new.obv <- predict(beetle.lda,newdata=data.frame(TG=189, Elytra=245, Second.Antenna=138, Third.Antenna=164))  
new.obv ## the new beetle should be classified as H. oleracea

## $class  
## [1] oleracea  
## Levels: oleracea carduorum  
##   
## $posterior  
## oleracea carduorum  
## 1 0.9999818 1.818995e-05  
##   
## $x  
## LD1  
## 1 -2.94882

According to our model the new flea beetle observation will be classified as oleracea.

**# c) Misclassification rate**beetle.cv <- lda(Species ~ ., CV = TRUE, data = flea.beetles)  
beetle.cv$class

## [1] oleracea oleracea oleracea oleracea oleracea oleracea oleracea   
## [8] oleracea oleracea oleracea oleracea oleracea oleracea oleracea   
## [15] oleracea oleracea oleracea oleracea oleracea carduorum oleracea   
## [22] carduorum carduorum carduorum carduorum carduorum oleracea carduorum  
## [29] carduorum carduorum carduorum carduorum carduorum carduorum carduorum  
## [36] oleracea carduorum carduorum carduorum  
## Levels: oleracea carduorum

table(flea.beetles$Species, beetle.cv$class, dnn = c('Actual Group','Predicted Group'))

## Predicted Group  
## Actual Group oleracea carduorum  
## oleracea 19 0  
## carduorum 3 17

#miss.rate  
beetle.er <-3/39  
beetle.er

## [1] 0.07692308

With the LDA model and prior probabilities of 50% and 50%. 19 flea beetles have been accurately predicted as H.oleracea and 17 flea beetles were accurately predicted as H. carduorum. There were 3 flea beetles that were actually H.carduorum(2) but got predicted as H.oleracea (1). Therefore, the misclassification rate is 0.077

# Problem 8

glucose<- read.table("~/Desktop/WINTER 2019/DA410-MULTIVARIATE-CHENG/multivariate\_analysis - 3rd Ed/multivariate\_analysis - 3rd Ed/glucose.txt", header=FALSE, col.names=c("y1", "y2", "y3", "x1", "x2", "x3"))  
  
colMeans(glucose)

## y1 y2 y3 x1 x2 x3   
## 72.20000 72.73333 73.30000 108.46667 102.46667 108.46667

cov(glucose)

## y1 y2 y3 x1 x2 x3  
## y1 77.6137931 0.9862069 23.731034 100.07586 4.868966 34.317241  
## y2 0.9862069 36.2022989 15.220690 -46.45747 30.370115 -32.078161  
## y3 23.7310345 15.2206897 57.458621 13.40690 -6.420690 1.475862  
## x1 100.0758621 -46.4574713 13.406897 959.49885 299.360920 232.636782  
## x2 4.8689655 30.3701149 -6.420690 299.36092 500.188506 61.809195  
## x3 34.3172414 -32.0781609 1.475862 232.63678 61.809195 527.016092

The variance of y1 is 77.61, the variance for y2 is 36.20, the variance for y3 is 57.46, the variance for x1 is 959.50, the variance for x2 is 500.19, the variance for x3 is 527.02

We can see that Syy and Sxx are symmetric and that Sxy is the transpose of Syx

# Problem 9

**HYPOTHESIS**

H0: mean score vector of consumer goods IS EQUAL TO the mean vector of producer goods.

HA: mean score vector of consumer goods IS DIFFERENT FROM the mean vector of producer goods.

econ<- read.table("~/Desktop/WINTER 2019/DA410-MULTIVARIATE-CHENG/multivariate\_analysis - 3rd Ed/multivariate\_analysis - 3rd Ed/T5\_8\_GOODS.DAT", header=FALSE, col.names=c("Item", "goods", "y1", "y2", "y3", "y4"))  
  
econ <- econ[,-1] #removing item number  
attach(econ)  
goods <- as.factor(econ$goods)  
  
consumer <- length(goods[goods ==1])  
consumer

## [1] 9

producer <- length(goods[goods==2])  
producer

## [1] 10

consumer\_data <- econ[goods==1, -1]  
producer\_data <- econ[goods==2, -1]  
  
my.q <- ncol(consumer\_data)  
  
#mean vector for consumer and producer  
m\_consumer <- apply(consumer\_data, 2, mean)  
m\_producer <- apply(producer\_data, 2, mean)  
  
# "pooled" sample covariance matrix:  
S.econ<- ((consumer-1)\*var(consumer\_data)+(producer-1)\*var(producer\_data))/(consumer+producer-2)  
S.econ

## y1 y2 y3 y4  
## y1 597.1169935 -54.303922 34.1761438 -0.4577778  
## y2 -54.3039216 73.676471 1.1568627 -1.0166667  
## y3 34.1761438 1.156863 17.9189542 0.7005229  
## y4 -0.4577778 -1.016667 0.7005229 0.1974444

# Hotelling T^2, the F-statistic, and the P-value:  
Hotelling <- ((consumer\*producer)/(consumer+producer))\* (t(m\_consumer-m\_producer) %\*% solve(S.econ) %\*% (m\_consumer-m\_producer) )   
  
Fstat <- ((consumer+producer-my.q-1)\*Hotelling)/((consumer+producer-2)\*my.q)  
  
pvalue <- 1-pf(Fstat, my.q, consumer+producer-my.q-1)  
  
print(paste("Hotelling T^2 =", round(Hotelling,4), "F=", round(Fstat,4), "P-value =", round(pvalue,4) ))

## [1] "Hotelling T^2 = 18.4625 F= 3.8011 P-value = 0.027"

T^2 =18.462557

F= 3.8011

P-value = 0.027 that is lower than alpha = 0.05

**we reject the null hypothesis and conclude that mean score vector of consumer goods is different from the mean vector of producer goods.**

**After concluding that the consumer goods can be distinguished from the producer goods the next step in the analysis is to determine upon which variables, they are different.**