Multivariate Analysis Midterm

DA 410

Marjorie Blanco

Note:

Round to the THIRD decimal place, unless otherwise noted in the instruction.

## Problem 1 - 3

Solved on paper (see pdf).

Problem 1: It has been shown that C is an orthogonal matrix since CC’ = C’C = I

Problem 2: It was shown that AB BA

Problem 3:

## Problem 4

In the following table, we have a comparison of four reagents. The first reagent is the one presently in use and the other three are less expensive reagents that we wish to compare with the first. All four reagents are used with a blood sample from each patient.

The three variables measured for each reagent are = white blood count, = red blood count, and = hemoglobin count.

The data for twenty subject from each of four reagents:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reagent | Subject | white.blood.count | red.blood.count | hemoglobin.count |
| 1 | 1 | 8.0 | 3.96 | 12.5 |
| 1 | 2 | 4.0 | 5.37 | 16.9 |
| 1 | 3 | 6.3 | 5.47 | 17.1 |
| 1 | 4 | 9.4 | 5.16 | 16.2 |
| 1 | 5 | 8.2 | 5.16 | 17.0 |
| 1 | 6 | 11.0 | 4.67 | 14.3 |
| 1 | 7 | 6.8 | 5.20 | 16.2 |
| 1 | 8 | 9.0 | 4.65 | 14.7 |
| 1 | 9 | 6.1 | 5.22 | 16.3 |
| 1 | 10 | 6.4 | 5.13 | 15.9 |
| 1 | 11 | 5.6 | 4.47 | 13.3 |
| 1 | 12 | 8.2 | 5.22 | 16.0 |
| 1 | 13 | 5.7 | 5.10 | 14.9 |
| 1 | 14 | 9.8 | 5.25 | 16.1 |
| 1 | 15 | 5.9 | 5.28 | 15.8 |
| 1 | 16 | 6.6 | 4.65 | 12.8 |
| 1 | 17 | 5.7 | 4.42 | 14.5 |
| 1 | 18 | 6.7 | 4.38 | 13.1 |
| 1 | 19 | 6.8 | 4.67 | 15.6 |
| 1 | 20 | 9.6 | 5.64 | 17.0 |
| 2 | 1 | 8.0 | 3.93 | 12.7 |
| 2 | 2 | 4.2 | 5.35 | 17.2 |
| 2 | 3 | 6.3 | 5.39 | 17.5 |
| 2 | 4 | 9.4 | 5.16 | 16.7 |
| 2 | 5 | 8.0 | 5.13 | 17.5 |
| 2 | 6 | 10.7 | 4.60 | 14.7 |
| 2 | 7 | 6.8 | 5.16 | 16.7 |
| 2 | 8 | 9.0 | 4.57 | 15.0 |
| 2 | 9 | 6.0 | 5.16 | 16.9 |
| 2 | 10 | 6.4 | 5.11 | 16.4 |
| 2 | 11 | 5.5 | 4.45 | 13.6 |
| 2 | 12 | 8.2 | 5.14 | 16.5 |
| 2 | 13 | 5.6 | 5.05 | 15.3 |
| 2 | 14 | 9.8 | 5.15 | 16.6 |
| 2 | 15 | 5.8 | 5.25 | 16.4 |
| 2 | 16 | 6.4 | 4.59 | 13.2 |
| 2 | 17 | 5.5 | 4.31 | 14.9 |
| 2 | 18 | 6.5 | 4.32 | 13.4 |
| 2 | 19 | 6.6 | 4.57 | 15.8 |
| 2 | 20 | 9.5 | 5.58 | 17.5 |
| 3 | 1 | 7.9 | 3.86 | 13.0 |
| 3 | 2 | 4.1 | 5.39 | 17.2 |
| 3 | 3 | 6.0 | 5.39 | 17.2 |
| 3 | 4 | 9.4 | 5.17 | 16.7 |
| 3 | 5 | 8.1 | 5.10 | 17.4 |
| 3 | 6 | 10.6 | 4.52 | 14.6 |
| 3 | 7 | 6.9 | 5.13 | 16.8 |
| 3 | 8 | 8.9 | 4.58 | 15.0 |
| 3 | 9 | 6.1 | 5.14 | 16.9 |
| 3 | 10 | 6.4 | 5.11 | 16.4 |
| 3 | 11 | 5.3 | 4.46 | 13.6 |
| 3 | 12 | 8.0 | 5.14 | 16.5 |
| 3 | 13 | 5.5 | 5.02 | 15.4 |
| 3 | 14 | 8.1 | 5.10 | 13.8 |
| 3 | 15 | 5.7 | 5.26 | 16.4 |
| 3 | 16 | 6.3 | 4.58 | 13.1 |
| 3 | 17 | 5.5 | 4.30 | 14.9 |
| 3 | 18 | 6.5 | 4.32 | 13.6 |
| 3 | 19 | 6.5 | 4.55 | 16.0 |
| 3 | 20 | 9.3 | 5.50 | 17.4 |
| 4 | 1 | 7.9 | 3.87 | 13.2 |
| 4 | 2 | 4.0 | 5.35 | 17.3 |
| 4 | 3 | 6.1 | 5.41 | 17.4 |
| 4 | 4 | 9.1 | 5.16 | 16.7 |
| 4 | 5 | 7.8 | 5.12 | 17.5 |
| 4 | 6 | 10.5 | 4.58 | 14.7 |
| 4 | 7 | 6.7 | 5.19 | 16.8 |
| 4 | 8 | 8.6 | 4.55 | 15.1 |
| 4 | 9 | 6.0 | 5.21 | 16.9 |
| 4 | 10 | 6.3 | 5.07 | 16.3 |
| 4 | 11 | 5.3 | 4.44 | 13.7 |
| 4 | 12 | 7.8 | 5.16 | 16.5 |
| 4 | 13 | 5.4 | 5.05 | 15.5 |
| 4 | 14 | 9.4 | 5.16 | 16.6 |
| 4 | 15 | 5.6 | 5.29 | 16.2 |
| 4 | 16 | 6.4 | 4.57 | 13.2 |
| 4 | 17 | 5.4 | 4.32 | 14.8 |
| 4 | 18 | 6.5 | 4.31 | 13.5 |
| 4 | 19 | 6.5 | 4.56 | 15.9 |
| 4 | 20 | 9.2 | 5.46 | 17.5 |

Compare the four reagents using all four MANOVA tests. State each hypotheses clearly, and interpret the results.

## $`1`  
## Reagent white.blood.count red.blood.count hemoglobin.count  
## Min. :1 Min. : 4.00 Min. :3.960 Min. :12.50   
## 1st Qu.:1 1st Qu.: 6.05 1st Qu.:4.650 1st Qu.:14.45   
## Median :1 Median : 6.75 Median :5.145 Median :15.85   
## Mean :1 Mean : 7.29 Mean :4.954 Mean :15.31   
## 3rd Qu.:1 3rd Qu.: 8.40 3rd Qu.:5.228 3rd Qu.:16.23   
## Max. :1 Max. :11.00 Max. :5.640 Max. :17.10   
##   
## $`2`  
## Reagent white.blood.count red.blood.count hemoglobin.count  
## Min. :2 Min. : 4.20 Min. :3.930 Min. :12.70   
## 1st Qu.:2 1st Qu.: 5.95 1st Qu.:4.570 1st Qu.:14.85   
## Median :2 Median : 6.55 Median :5.120 Median :16.40   
## Mean :2 Mean : 7.21 Mean :4.899 Mean :15.72   
## 3rd Qu.:2 3rd Qu.: 8.40 3rd Qu.:5.160 3rd Qu.:16.75   
## Max. :2 Max. :10.70 Max. :5.580 Max. :17.50   
##   
## $`3`  
## Reagent white.blood.count red.blood.count hemoglobin.count  
## Min. :3 Min. : 4.100 Min. :3.860 Min. :13.00   
## 1st Qu.:3 1st Qu.: 5.925 1st Qu.:4.543 1st Qu.:14.40   
## Median :3 Median : 6.500 Median :5.100 Median :16.20   
## Mean :3 Mean : 7.055 Mean :4.881 Mean :15.60   
## 3rd Qu.:3 3rd Qu.: 8.100 3rd Qu.:5.147 3rd Qu.:16.82   
## Max. :3 Max. :10.600 Max. :5.500 Max. :17.40   
##   
## $`4`  
## Reagent white.blood.count red.blood.count hemoglobin.count  
## Min. :4 Min. : 4.000 Min. :3.870 Min. :13.20   
## 1st Qu.:4 1st Qu.: 5.900 1st Qu.:4.558 1st Qu.:14.78   
## Median :4 Median : 6.500 Median :5.095 Median :16.25   
## Mean :4 Mean : 7.025 Mean :4.891 Mean :15.77   
## 3rd Qu.:4 3rd Qu.: 8.075 3rd Qu.:5.195 3rd Qu.:16.82   
## Max. :4 Max. :10.500 Max. :5.460 Max. :17.50

MANOVA analysis assumes both normality and homoscedasticity (equality of variance) of the experimental errors (residuals).

* Descriptive statistics by dependent variable

White blood count:

## reagents$Reagent: 1  
## median mean SE.mean CI.mean.0.95 var   
## 6.7500000 7.2900000 0.3976841 0.8323624 3.1630526   
## std.dev coef.var   
## 1.7784973 0.2439640   
## --------------------------------------------------------   
## reagents$Reagent: 2  
## median mean SE.mean CI.mean.0.95 var   
## 6.5500000 7.2100000 0.3923546 0.8212075 3.0788421   
## std.dev coef.var   
## 1.7546630 0.2433652   
## --------------------------------------------------------   
## reagents$Reagent: 3  
## median mean SE.mean CI.mean.0.95 var   
## 6.5000000 7.0550000 0.3715172 0.7775944 2.7605000   
## std.dev coef.var   
## 1.6614752 0.2355032   
## --------------------------------------------------------   
## reagents$Reagent: 4  
## median mean SE.mean CI.mean.0.95 var   
## 6.5000000 7.0250000 0.3773784 0.7898621 2.8482895   
## std.dev coef.var   
## 1.6876876 0.2402402

Red blood count:

## reagents$Reagent: 1  
## median mean SE.mean CI.mean.0.95 var   
## 5.14500000 4.95350000 0.09768444 0.20445588 0.19084500   
## std.dev coef.var   
## 0.43685810 0.08819180   
## --------------------------------------------------------   
## reagents$Reagent: 2  
## median mean SE.mean CI.mean.0.95 var   
## 5.1200000 4.8985000 0.0986975 0.2065763 0.1948239   
## std.dev coef.var   
## 0.4413887 0.0901069   
## --------------------------------------------------------   
## reagents$Reagent: 3  
## median mean SE.mean CI.mean.0.95 var   
## 5.10000000 4.88100000 0.09980745 0.20889939 0.19923053   
## std.dev coef.var   
## 0.44635247 0.09144693   
## --------------------------------------------------------   
## reagents$Reagent: 4  
## median mean SE.mean CI.mean.0.95 var   
## 5.09500000 4.89150000 0.10021103 0.20974409 0.20084500   
## std.dev coef.var   
## 0.44815734 0.09161961

Hemoglobin count:

## reagents$Reagent: 1  
## median mean SE.mean CI.mean.0.95 var   
## 15.85000000 15.31000000 0.32894568 0.68849123 2.16410526   
## std.dev coef.var   
## 1.47108982 0.09608686   
## --------------------------------------------------------   
## reagents$Reagent: 2  
## median mean SE.mean CI.mean.0.95 var   
## 16.40000000 15.72500000 0.34441980 0.72087893 2.37250000   
## std.dev coef.var   
## 1.54029218 0.09795181   
## --------------------------------------------------------   
## reagents$Reagent: 3  
## median mean SE.mean CI.mean.0.95 var   
## 16.20000000 15.59500000 0.34155026 0.71487291 2.33313158   
## std.dev coef.var   
## 1.52745919 0.09794544   
## --------------------------------------------------------   
## reagents$Reagent: 4  
## median mean SE.mean CI.mean.0.95 var   
## 16.25000000 15.76500000 0.32905567 0.68872142 2.16555263   
## std.dev coef.var   
## 1.47158168 0.09334486

* Multivariate normality

# run normality test  
mshapiro.test(t(reagents1))

##   
## Shapiro-Wilk normality test  
##   
## data: Z  
## W = 0.62978, p-value = 0.001241

mshapiro.test(t(reagents2))

##   
## Shapiro-Wilk normality test  
##   
## data: Z  
## W = 0.62978, p-value = 0.001241

mshapiro.test(t(reagents3))

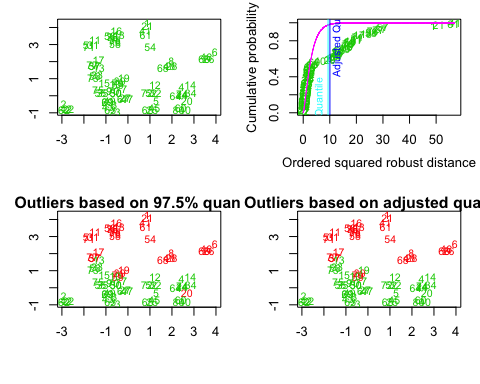
##   
## Shapiro-Wilk normality test  
##   
## data: Z  
## W = 0.62978, p-value = 0.001241

mshapiro.test(t(reagents4))

##   
## Shapiro-Wilk normality test  
##   
## data: Z  
## W = 0.62978, p-value = 0.001241

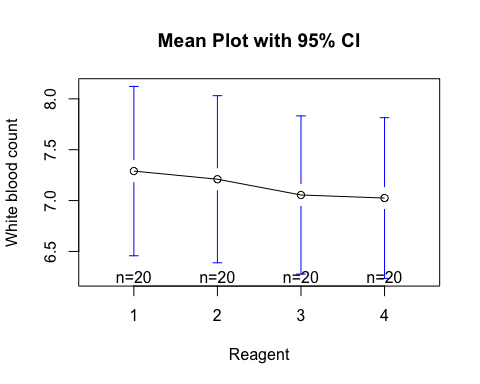
aq.plot(reagents[,3:5])

## Projection to the first and second robust principal components.  
## Proportion of total variation (explained variance): 0.9590544

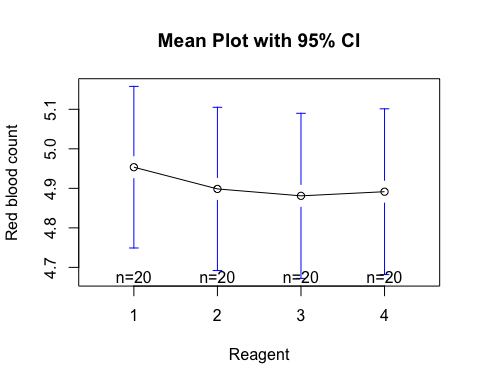


## $outliers  
## [1] TRUE FALSE FALSE FALSE FALSE TRUE FALSE TRUE FALSE FALSE TRUE  
## [12] FALSE FALSE FALSE FALSE TRUE TRUE TRUE FALSE FALSE TRUE FALSE  
## [23] FALSE FALSE FALSE TRUE FALSE TRUE FALSE FALSE TRUE FALSE FALSE  
## [34] FALSE FALSE TRUE TRUE TRUE TRUE FALSE TRUE FALSE FALSE FALSE  
## [45] FALSE TRUE FALSE TRUE FALSE FALSE TRUE FALSE FALSE TRUE FALSE  
## [56] TRUE TRUE TRUE TRUE FALSE TRUE FALSE FALSE FALSE FALSE TRUE  
## [67] FALSE TRUE FALSE FALSE TRUE FALSE FALSE FALSE FALSE TRUE TRUE  
## [78] TRUE TRUE FALSE

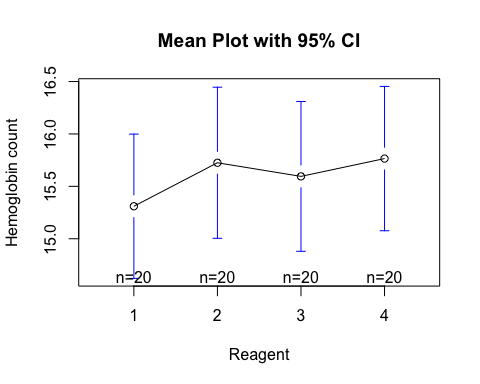
* Mean plot by dependent variable



This plot shows that the mean white blood count does not appear to be different for the four reagents.



This plot shows that the mean red blood count does not appear to be different for the four reagents.



This plot shows that the mean hemoglobin count does not appear to be different for the four reagents.

n <- dim(reagents)[1] / length(unique(reagents$Reagent))  
total.means <- colMeans(reagents[,3:5])

The overall mean vector:

|  |  |  |
| --- | --- | --- |
| white.blood.count | red.blood.count | hemoglobin.count |
| 7.145 | 4.906 | 15.599 |

The mean vector for each sample:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
| white.blood.count | 7.290 | 7.210 | 7.055 | 7.025 |
| red.blood.count | 4.954 | 4.899 | 4.881 | 4.891 |
| hemoglobin.count | 15.310 | 15.725 | 15.595 | 15.765 |

reagent1 <- reagents %>% filter(Reagent == 1) %>% dplyr::select(-c(Reagent, Subject))  
reagent2 <- reagents %>% filter(Reagent == 2) %>% dplyr::select(-c(Reagent, Subject))  
reagent3 <- reagents %>% filter(Reagent == 3) %>% dplyr::select(-c(Reagent, Subject))  
reagent4 <- reagents %>% filter(Reagent == 4) %>% dplyr::select(-c(Reagent, Subject))  
  
reagent1.bar <- colMeans(reagent1)  
reagent2.bar <- colMeans(reagent2)   
reagent3.bar <- colMeans(reagent3)   
reagent4.bar <- colMeans(reagent4)   
  
reagent.all.bar <- (reagent1.bar+reagent2.bar+reagent3.bar+reagent4.bar)/4  
  
reagent1.bar.diff <- reagent1.bar - reagent.all.bar   
reagent2.bar.diff <- reagent2.bar - reagent.all.bar   
reagent3.bar.diff <- reagent3.bar - reagent.all.bar   
reagent4.bar.diff <- reagent4.bar - reagent.all.bar   
  
H <- n \* unname(reagent1.bar.diff %\*% t(reagent1.bar.diff) +  
 reagent2.bar.diff %\*% t(reagent2.bar.diff) +  
 reagent3.bar.diff %\*% t(reagent3.bar.diff) +  
 reagent4.bar.diff %\*% t(reagent4.bar.diff))

H =

|  |  |  |
| --- | --- | --- |
| 0.955 | 0.208 | -1.065 |
| 0.208 | 0.063 | -0.340 |
| -1.065 | -0.340 | 2.539 |

"compute.within.matrix" <-function(data, mean) {  
 ret <- matrix(as.numeric(0), nrow=3, ncol=3)  
 for (i in 1:20) {  
 diff <- as.numeric(unname(data[i,] - mean))   
 ret <- ret + diff %\*% t(diff)}   
 return(ret)  
}  
E <- compute.within.matrix(reagent1, reagent1.bar) + compute.within.matrix(reagent2, reagent2.bar) +  
 compute.within.matrix(reagent3, reagent3.bar) + compute.within.matrix(reagent4, reagent4.bar)

E =

|  |  |  |
| --- | --- | --- |
| 225.163 | -0.911 | 6.020 |
| -0.911 | 14.929 | 44.549 |
| 6.020 | 44.549 | 171.671 |

# number of groups  
k <- length(unique(reagents$Reagent))  
# number of variables (dimension)  
p <- length(reagents[,3:5])  
# degrees of freedom for hypothesis  
vh <- k - 1  
# degrees of freedom for error  
ve <- dim(reagents)[1] - k

The number of groups: k = 4

The number of variables (dimension): p = 3

The degrees of freedom for hypothesis: = 3

The degrees of freedom for error: = 76

# MANOVA test  
reagents.manova <- manova(cbind(reagents$white.blood.count,   
 reagents$red.blood.count,  
 reagents$hemoglobin.count) ~ Reagent,   
 data = reagents)  
reagents.summary <- summary(reagents.manova)

We would then like to test if the properties (white blood, red blood and hemoglobin count) are the same across the four reagents.

At least two are unequal

E <- reagents.summary$SS$Residuals  
H <- reagents.summary$SS$Reagent  
# Wilk’s Lambda   
lambda <- det(E) / det(E + H)  
e1h.eigen <- eigen(solve(E) %\*% H)  
Vs <- sum(e1h.eigen$values / (1 + e1h.eigen$values))  
Us <- sum(e1h.eigen$values)  
s <- min(vh, p)  
roy.stat <- e1h.eigen$values[1]  
roy.omega <- roy.stat / (1 + roy.stat)

### Wilks’s test

reagents.summary <- summary(manova(cbind(reagents$white.blood.count,   
 reagents$red.blood.count,   
 reagents$hemoglobin.count) ~   
 reagents$Reagent),   
 test = "Wilks")  
reagents.summary

## Df Wilks approx F num Df den Df Pr(>F)   
## reagents$Reagent 1 0.91344 2.4007 3 76 0.07431 .  
## Residuals 78   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

= 0.913

The MANOVA model reports a Wilks test statistic of 0.913 and a p-value (0.074) > 0.05, thus fails to be rejected and it is concluded there are no significant differences in the means.

### Roy’s test

reagents.summary <- summary(manova(cbind(reagents$white.blood.count,   
 reagents$red.blood.count,   
 reagents$hemoglobin.count) ~   
 reagents$Reagent),   
 test = "Roy")  
reagents.summary

## Df Roy approx F num Df den Df Pr(>F)   
## reagents$Reagent 1 0.094764 2.4007 3 76 0.07431 .  
## Residuals 78   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

= 0.087

The MANOVA model reports a Roy test statistic of 0.095 and a p-value (0.074) > 0.05, thus fails to be rejected and it is concluded there are no significant differences in the means.

### Hotelling-Lawley’s test

reagents.summary <- summary(manova(cbind(reagents$white.blood.count,   
 reagents$red.blood.count,   
 reagents$hemoglobin.count) ~   
 reagents$Reagent),   
 test = "Hotelling-Lawley")  
reagents.summary

## Df Hotelling-Lawley approx F num Df den Df Pr(>F)   
## reagents$Reagent 1 0.094764 2.4007 3 76 0.07431 .  
## Residuals 78   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

= 0.095

The MANOVA model reports a Hotelling-Lawley test statistic of 0.095 and a p-value (0.074) > 0.05, thus fails to be rejected and it is concluded there are no significant differences in the means.

### Pillai’s test

reagents.summary <- summary(manova(cbind(reagents$white.blood.count,   
 reagents$red.blood.count,   
 reagents$hemoglobin.count) ~   
 reagents$Reagent),   
 test = "Pillai")  
reagents.summary

## Df Pillai approx F num Df den Df Pr(>F)   
## reagents$Reagent 1 0.086561 2.4007 3 76 0.07431 .  
## Residuals 78   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

= 0.087

The MANOVA model reports a Pillai test statistic of 0.087 and a p-value (0.074) > 0.05, thus fails to be rejected and it is concluded there are no significant differences in the means.

## Problem 5

The table below displays scores on math, English, and art tests for 5 students. Note that data from the table is represented in matrix A, where each column in the matrix shows scores on a test and each row shows scores for a student:

A <- matrix(c(90, 60, 90, 90, 90, 30, 60, 60, 60, 60, 60, 90, 30, 30, 30),   
 nrow = 5, ncol = 3, byrow = TRUE )  
colnames(A) <- c("Math", "English", "Art")

A =

Math, English, and Art tests for 5 students

|  |  |  |
| --- | --- | --- |
| Math | English | Art |
| 90 | 60 | 90 |
| 90 | 90 | 30 |
| 60 | 60 | 60 |
| 60 | 60 | 90 |
| 30 | 30 | 30 |

y <- data.frame(mean = colMeans(A))

=

Mean vector y

|  |  |  |  |
| --- | --- | --- | --- |
|  | Math | English | Art |
| mean | 66 | 60 | 60 |

1. Calculate the sample covariance matrix S.

S <- cov(A)

S =

Sample covariance matrix

|  |  |  |  |
| --- | --- | --- | --- |
|  | Math | English | Art |
| Math | 630 | 450 | 225 |
| English | 450 | 450 | 0 |
| Art | 225 | 0 | 900 |

Thus, 630 is the variance of the Math variable, 450 is the covariance between the Math and the English variables, 225 is the covariance between the Math and the Art variables, 450 is the variance of the English variable, 0 is the covariance between the English and Art variables and 900 is the variance of the Art variable.

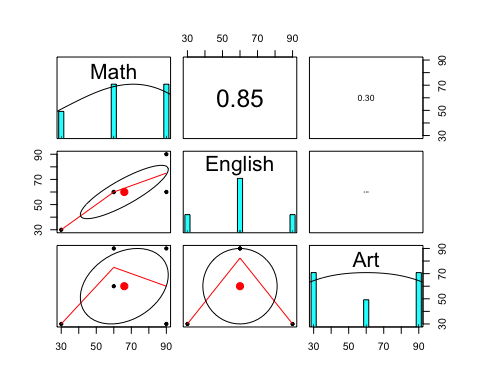
1. Calculate the sample correlation matrix R.

R <- cor(A)

R =

Sample correlation matrix

|  |  |  |  |
| --- | --- | --- | --- |
|  | Math | English | Art |
| Math | 1.0000000 | 0.8451543 | 0.2988072 |
| English | 0.8451543 | 1.0000000 | 0.0000000 |
| Art | 0.2988072 | 0.0000000 | 1.0000000 |



Based on the correlation matrix, we can see that Math and English score are highly correlated. Art and Math are weakly correlated, while English and Art are not correlated.

1. Now let’s define , where denotes Math scores, denotes English scores, and denotes Art scores. Find the sample mean vector and the sample variance .

A2 <- sweep(A, 2, c(-2, 3, 1), "\*")  
  
z <- data.frame(mean = rowSums(A2))  
rownames(z) <- paste(rep(c("z"), nrow(A)), rep(1:nrow(A)), sep="")

z =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | z1 | z2 | z3 | z4 | z5 |
| mean | 90 | 120 | 120 | 150 | 60 |

z\_bar <- sum(z) \* (1/nrow(A))

= 108

a <- c(-2, 3, 1)  
s2z<- t(a) %\*% as.matrix(S) %\*% a

= 1170

## Problem 6:

Use the beetle data, do the following:

Beetles

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Number | Species | y1 | y2 | y3 | y4 |
| 1 | 1 | 189 | 245 | 137 | 163 |
| 2 | 1 | 192 | 260 | 132 | 217 |
| 3 | 1 | 217 | 276 | 141 | 192 |
| 4 | 1 | 221 | 299 | 142 | 213 |
| 5 | 1 | 171 | 239 | 128 | 158 |
| 6 | 1 | 192 | 262 | 147 | 173 |
| 7 | 1 | 213 | 278 | 136 | 201 |
| 8 | 1 | 192 | 255 | 128 | 185 |
| 9 | 1 | 170 | 244 | 128 | 192 |
| 10 | 1 | 201 | 276 | 146 | 186 |
| 11 | 1 | 195 | 242 | 128 | 192 |
| 12 | 1 | 205 | 263 | 147 | 192 |
| 13 | 1 | 180 | 252 | 121 | 167 |
| 14 | 1 | 192 | 283 | 138 | 183 |
| 15 | 1 | 200 | 294 | 138 | 188 |
| 16 | 1 | 192 | 277 | 150 | 177 |
| 17 | 1 | 200 | 287 | 136 | 173 |
| 18 | 1 | 181 | 255 | 146 | 183 |
| 19 | 1 | 192 | 287 | 141 | 198 |
| 1 | 2 | 181 | 305 | 184 | 209 |
| 2 | 2 | 158 | 237 | 133 | 188 |
| 3 | 2 | 184 | 300 | 166 | 231 |
| 4 | 2 | 171 | 273 | 162 | 213 |
| 5 | 2 | 181 | 297 | 163 | 224 |
| 6 | 2 | 181 | 308 | 160 | 223 |
| 7 | 2 | 177 | 301 | 166 | 221 |
| 8 | 2 | 198 | 308 | 141 | 197 |
| 9 | 2 | 180 | 286 | 146 | 214 |
| 10 | 2 | 177 | 299 | 171 | 192 |
| 11 | 2 | 176 | 317 | 166 | 213 |
| 12 | 2 | 192 | 312 | 166 | 209 |
| 13 | 2 | 176 | 285 | 141 | 200 |
| 14 | 2 | 169 | 287 | 162 | 214 |
| 15 | 2 | 164 | 265 | 147 | 192 |
| 16 | 2 | 181 | 308 | 157 | 204 |
| 17 | 2 | 192 | 276 | 154 | 209 |
| 18 | 2 | 181 | 278 | 149 | 235 |
| 19 | 2 | 175 | 271 | 140 | 192 |
| 20 | 2 | 197 | 303 | 170 | 205 |

1. Find the classification function and cutoff point.

Let = Haltica oleracea and = Haltica carduorum

oleracea <- beetles\_df %>% dplyr::filter(Species == 1) %>% dplyr::select(-c(Species, Measurement.Number))  
carduorum <- beetles\_df %>% dplyr::filter(Species == 2) %>% dplyr::select(-c(Species, Measurement.Number))  
  
oleracea <- as.matrix(oleracea)  
carduorum <- as.matrix(carduorum)

### Mean vectors

y\_bar1 <- colMeans(oleracea)  
y\_bar2 <- colMeans(carduorum)

|  |  |  |  |
| --- | --- | --- | --- |
| transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| 194.474 | 267.053 | 137.368 | 185.947 |

|  |  |  |  |
| --- | --- | --- | --- |
| transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| 179.55 | 290.8 | 157.2 | 209.25 |

### Covariance matrices

S1 <- cov(oleracea)  
S2 <- cov(carduorum)

S1 =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| transverse.groove.dist | 187.596 | 176.863 | 48.371 | 113.582 |
| elytra.length | 176.863 | 345.386 | 75.980 | 118.781 |
| second.antennal.joint.length | 48.371 | 75.980 | 66.357 | 16.243 |
| third.antennal.joint.length | 113.582 | 118.781 | 16.243 | 239.942 |

S2 =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| transverse.groove.dist | 101.839 | 128.063 | 36.989 | 32.592 |
| elytra.length | 128.063 | 389.011 | 165.358 | 94.368 |
| second.antennal.joint.length | 36.989 | 165.358 | 167.537 | 66.526 |
| third.antennal.joint.length | 32.592 | 94.368 | 66.526 | 177.882 |

### Pooled covariance matrix

Spl <- (1/ (nrow(oleracea) + nrow(carduorum) - 2)) \*  
 ((nrow(oleracea) - 1) \* S1 + (nrow(carduorum) - 1) \* S2)

Spl =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| transverse.groove.dist | 143.559 | 151.803 | 42.527 | 71.993 |
| elytra.length | 151.803 | 367.788 | 121.877 | 106.245 |
| second.antennal.joint.length | 42.527 | 121.877 | 118.314 | 42.064 |
| third.antennal.joint.length | 71.993 | 106.245 | 42.064 | 208.073 |

=

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| transverse.groove.dist | 0.013 | -0.005 | 0.002 | -0.002 |
| elytra.length | -0.005 | 0.007 | -0.005 | -0.001 |
| second.antennal.joint.length | 0.002 | -0.005 | 0.013 | -0.001 |
| third.antennal.joint.length | -0.002 | -0.001 | -0.001 | 0.006 |

a\_prime <- t(y\_bar1 - y\_bar2) %\*% solve(Spl)

|  |  |  |  |
| --- | --- | --- | --- |
| transverse.groove.dist | elytra.length | second.antennal.joint.length | third.antennal.joint.length |
| 0.345 | -0.13 | -0.106 | -0.143 |

z\_bar1 <- a\_prime %\*% y\_bar1

|  |
| --- |
| -8.955 |

z\_bar2 <- a\_prime %\*% y\_bar2

|  |
| --- |
| -22.655 |

z <- (z\_bar1 + z\_bar2) /2

|  |
| --- |
| -15.805 |

Assign y to if z -15.805

Assign y to if z < -15.805

In other words, if z is greater than -15.805, the observation is assigned to Group 1. Otherwise, it is assigned to Group 2.

1. Find the classification table using the nearest neighbor method by setting k = 3.

### Nearest neighbor method with no cross-validation/scaling

beetles\_df$KNNprediction <- knn(train = train.data,   
 test = train.data,   
 cl = as.matrix(label.data),   
 k = 3)

Classification Table for the Beetle Data Using the k Nearest Neighbor Method with k = 3

cm <- confusionMatrix(beetles\_df$Species, beetles\_df$KNNprediction)  
cm$table

## Reference  
## Prediction 1 2  
## 1 19 0  
## 2 1 19

The KNN model predictions classified all of group 1’s observations correctly but incorrectly assigned a group 2 observation to group 1.

normalize <- function(x) {  
return ((x - min(x)) / (max(x) - min(x))) }  
  
trainN.data <- as.data.frame(lapply(train.data, normalize))  
beetles\_df$KNNprediction.n <- knn(train = trainN.data,   
 test = trainN.data,   
 cl = as.matrix(label.data),   
 k = 3)

Classification Table for the Beetle Data Using the k Nearest Neighbor Method with k = 3 with Scaling/Normalization

cm <- confusionMatrix(beetles\_df$Species, beetles\_df$KNNprediction.n)  
cm$table

## Reference  
## Prediction 1 2  
## 1 19 0  
## 2 1 19

The KNN model predictions classified all of group 1’s observations correctly but incorrectly assigned a group 2 observation to group 1.

1. Calculate misclassification rate.

### Nearest neighbor method with no scaling

cm <- confusionMatrix(beetles\_df$Species, beetles\_df$KNNprediction)  
cm

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 1 2  
## 1 19 0  
## 2 1 19  
##   
## Accuracy : 0.9744   
## 95% CI : (0.8652, 0.9994)  
## No Information Rate : 0.5128   
## P-Value [Acc > NIR] : 1.858e-10   
##   
## Kappa : 0.9488   
## Mcnemar's Test P-Value : 1   
##   
## Sensitivity : 0.9500   
## Specificity : 1.0000   
## Pos Pred Value : 1.0000   
## Neg Pred Value : 0.9500   
## Prevalence : 0.5128   
## Detection Rate : 0.4872   
## Detection Prevalence : 0.4872   
## Balanced Accuracy : 0.9750   
##   
## 'Positive' Class : 1   
##

The apparent correct classification rate =

The apparent error rate = 1 - 0.974 = 0.026 = 2.6%

### Nearest neighbor method with scaling/normalizing

cm <- confusionMatrix(beetles\_df$Species, beetles\_df$KNNprediction)  
cm

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 1 2  
## 1 19 0  
## 2 1 19  
##   
## Accuracy : 0.9744   
## 95% CI : (0.8652, 0.9994)  
## No Information Rate : 0.5128   
## P-Value [Acc > NIR] : 1.858e-10   
##   
## Kappa : 0.9488   
## Mcnemar's Test P-Value : 1   
##   
## Sensitivity : 0.9500   
## Specificity : 1.0000   
## Pos Pred Value : 1.0000   
## Neg Pred Value : 0.9500   
## Prevalence : 0.5128   
## Detection Rate : 0.4872   
## Detection Prevalence : 0.4872   
## Balanced Accuracy : 0.9750   
##   
## 'Positive' Class : 1   
##

The apparent correct classification rate =

The apparent error rate = 1 - 0.974 = 0.026 = 2.6%

### Nearest neighbor method using cross-validation with no scaling

correct <- rep(0, times=nrow(beetles\_df))  
for (j in 1:nrow(beetles\_df))  
{  
 mypred <- knn(train = train.data[-j, ],   
 test = train.data[j, ],   
 cl = as.matrix(label.data)[-j], k = 3)  
 correct[j] <- (mypred == beetles\_df$Species[j])  
}  
cv.missclass <- 1 - mean(correct)

The KNN training model correctly classified 92.3% of observations.

The KNN training model misclassification rate for KNN = 3 is 7.7%.

### Nearest neighbor method using cross-validation with scaling/normalizing

correct <- rep(0, times=nrow(beetles\_df))  
for (j in 1:nrow(beetles\_df))  
{  
 mypred <- knn(train = trainN.data[-j, ],   
 test = trainN.data[j, ],   
 cl = as.matrix(label.data)[-j], k = 3)  
 correct[j] <- (mypred == beetles\_df$Species[j])  
}  
cv.missclass <- 1 - mean(correct)

The KNN training model correctly classified 89.7% of observations.

The KNN training model misclassification rate for KNN = 3 is 10.3%.

This model has lower accuracy than the model without scaling/normalizing.

## Problem 7

Use the above beetle data, do the following:

1. Use LDA by setting probability of 50% and 50% to train model.

The mean vectors:

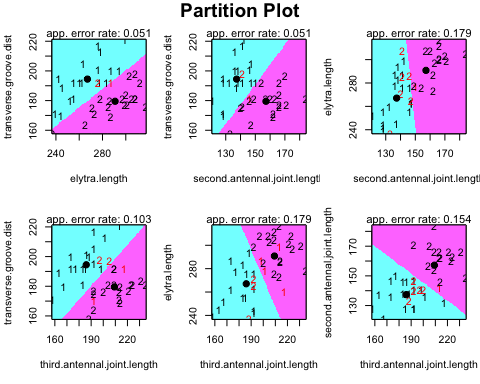
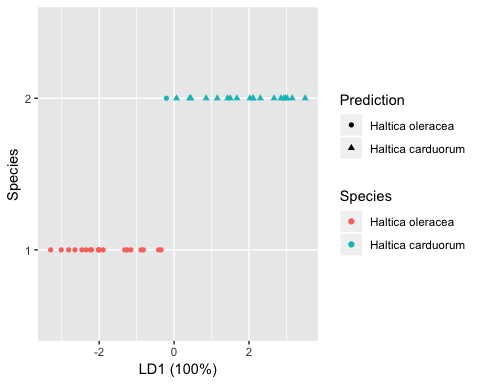
|  |  |  |
| --- | --- | --- |
|  | 1 | 2 |
| transverse.groove.dist | 194.474 | 179.55 |
| elytra.length | 267.053 | 290.80 |
| second.antennal.joint.length | 137.368 | 157.20 |
| third.antennal.joint.length | 185.947 | 209.25 |

beetles\_df$Measurement.Number <- NULL  
beetles.lda <- lda(Species ~ .,   
 prior = c(0.5,0.5),  
 data = beetles\_df[1:5])  
lda.pred <- predict(beetles.lda)$class

## Call:  
## lda(Species ~ ., data = beetles\_df[1:5], prior = c(0.5, 0.5))  
##   
## Prior probabilities of groups:  
## 1 2   
## 0.5 0.5   
##   
## Group means:  
## transverse.groove.dist elytra.length second.antennal.joint.length  
## 1 194.4737 267.0526 137.3684  
## 2 179.5500 290.8000 157.2000  
## third.antennal.joint.length  
## 1 185.9474  
## 2 209.2500  
##   
## Coefficients of linear discriminants:  
## LD1  
## transverse.groove.dist -0.09327642  
## elytra.length 0.03522706  
## second.antennal.joint.length 0.02875538  
## third.antennal.joint.length 0.03872998

The first discriminant function is a linear combination of the variables:

The LDA probability of Haltica oleracea is 50% while Haltica carduorum is 50%.



cm <- confusionMatrix(beetles\_df$Species, as.factor(lda.data$LDAprediction))  
cm

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 1 2  
## 1 19 0  
## 2 1 19  
##   
## Accuracy : 0.9744   
## 95% CI : (0.8652, 0.9994)  
## No Information Rate : 0.5128   
## P-Value [Acc > NIR] : 1.858e-10   
##   
## Kappa : 0.9488   
## Mcnemar's Test P-Value : 1   
##   
## Sensitivity : 0.9500   
## Specificity : 1.0000   
## Pos Pred Value : 1.0000   
## Neg Pred Value : 0.9500   
## Prevalence : 0.5128   
## Detection Rate : 0.4872   
## Detection Prevalence : 0.4872   
## Balanced Accuracy : 0.9750   
##   
## 'Positive' Class : 1   
##

The LDA model predictions classified all of group 1’s observations correctly but incorrectly assigned a group 2 observation to group 1.

1. Predict new observation (189,245,138,164).

new.data <- data.frame(189,245,138,164)  
colnames(new.data) <- c("transverse.groove.dist", "elytra.length", "second.antennal.joint.length", "third.antennal.joint.length")  
plda <- predict(beetles.lda, newdata = new.data)  
plda

## $class  
## [1] 1  
## Levels: 1 2  
##   
## $posterior  
## 1 2  
## 1 0.9999818 1.818995e-05  
##   
## $x  
## LD1  
## 1 -2.94882

The new observation LD1 is -2.949 and it is predicted to be assigned to Group 1 (Haltica oleracea).

1. Calculate misclassification rate.

correct <- rep(0, times=nrow(beetles\_df))  
for (j in 1:nrow(beetles\_df))  
{  
 mydis<- lda(grouping = beetles\_df$Species[-j],  
 x=beetles\_df[-j, 2:5],  
 prior = c(0.5, 0.5))  
 mypred <- predict(mydis, newdata = beetles\_df[j, 2:5])$class  
 correct[j] <- (mypred == beetles\_df$Species[j])  
}  
cv.missclass <- 1 - mean(correct)

The LDA training model correctly classified 92.3% of observations.

The LDA training model misclassification rate is 7.7%.

## Problem 8

The following table contains data from O’Sullivan and Mahan with measurements of blood glucose levels on three occasions for 30 women. The y’s represent fasting glucose measurements on the three occasions; the x’s are glucose measurements 1 hour after sugar intake. Find the mean vector and covariance matrix for all six variables and partition them into , and

blood.glucose <- read.table('data/data\_problem8.txt')  
blood.glucose <- cbind(blood.glucose[1:30,], blood.glucose[31:60,])  
colnames(blood.glucose) <- c("y1", "y2", "y3", "x1", "x2", "x3")

kable(blood.glucose, format = "pandoc") %>%  
 kable\_styling(bootstrap\_options = "striped")

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| y1 | y2 | y3 | x1 | x2 | x3 |
| 62 | 75 | 68 | 116 | 130 | 91 |
| 74 | 64 | 70 | 109 | 101 | 103 |
| 64 | 71 | 66 | 77 | 102 | 130 |
| 73 | 70 | 64 | 115 | 110 | 109 |
| 68 | 67 | 75 | 76 | 85 | 119 |
| 69 | 82 | 74 | 72 | 133 | 127 |
| 60 | 67 | 61 | 130 | 134 | 121 |
| 70 | 74 | 78 | 150 | 158 | 100 |
| 66 | 74 | 78 | 150 | 131 | 142 |
| 83 | 70 | 74 | 99 | 98 | 105 |
| 68 | 66 | 90 | 119 | 85 | 109 |
| 78 | 63 | 75 | 164 | 98 | 138 |
| 103 | 77 | 77 | 160 | 117 | 121 |
| 77 | 68 | 74 | 144 | 71 | 153 |
| 66 | 77 | 68 | 77 | 70 | 109 |
| 70 | 70 | 72 | 114 | 93 | 122 |
| 75 | 65 | 71 | 77 | 82 | 89 |
| 91 | 74 | 93 | 114 | 93 | 122 |
| 66 | 75 | 73 | 77 | 70 | 109 |
| 75 | 82 | 76 | 153 | 132 | 115 |
| 74 | 71 | 66 | 143 | 105 | 100 |
| 76 | 70 | 64 | 114 | 113 | 129 |
| 74 | 82 | 76 | 73 | 106 | 116 |
| 74 | 71 | 66 | 116 | 81 | 77 |
| 67 | 70 | 64 | 63 | 87 | 70 |
| 67 | 90 | 86 | 63 | 87 | 70 |
| 78 | 77 | 80 | 105 | 132 | 80 |
| 64 | 71 | 69 | 83 | 94 | 133 |
| 71 | 76 | 80 | 81 | 87 | 86 |
| 63 | 73 | 71 | 120 | 89 | 59 |

y <- data.frame(mean = colMeans(blood.glucose))

=

|  |  |
| --- | --- |
|  | mean |
| y1 | 72.200 |
| y2 | 72.733 |
| y3 | 73.300 |
| x1 | 108.467 |
| x2 | 102.467 |
| x3 | 108.467 |

S =

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | y1 | y2 | y3 | x1 | x2 | x3 |
| y1 | 77.614 | 0.986 | 23.731 | 100.076 | 4.869 | 34.317 |
| y2 | 0.986 | 36.202 | 15.221 | -46.457 | 30.370 | -32.078 |
| y3 | 23.731 | 15.221 | 57.459 | 13.407 | -6.421 | 1.476 |
| x1 | 100.076 | -46.457 | 13.407 | 959.499 | 299.361 | 232.637 |
| x2 | 4.869 | 30.370 | -6.421 | 299.361 | 500.189 | 61.809 |
| x3 | 34.317 | -32.078 | 1.476 | 232.637 | 61.809 | 527.016 |

=

|  |  |  |  |
| --- | --- | --- | --- |
|  | y1 | y2 | y3 |
| y1 | 77.614 | 0.986 | 23.731 |
| y2 | 0.986 | 36.202 | 15.221 |
| y3 | 23.731 | 15.221 | 57.459 |

=

|  |  |  |  |
| --- | --- | --- | --- |
|  | x1 | x2 | x3 |
| y1 | 100.076 | 4.869 | 34.317 |
| y2 | -46.457 | 30.370 | -32.078 |
| y3 | 13.407 | -6.421 | 1.476 |

=

|  |  |  |  |
| --- | --- | --- | --- |
|  | y1 | y2 | y3 |
| x1 | 100.076 | -46.457 | 13.407 |
| x2 | 4.869 | 30.370 | -6.421 |
| x3 | 34.317 | -32.078 | 1.476 |

=

|  |  |  |  |
| --- | --- | --- | --- |
|  | x1 | x2 | x3 |
| x1 | 959.499 | 299.361 | 232.637 |
| x2 | 299.361 | 500.189 | 61.809 |
| x3 | 232.637 | 61.809 | 527.016 |

Syy <- matrix(c(cov(blood.glucose$y1, blood.glucose$y1),   
 cov(blood.glucose$y1, blood.glucose$y2),   
 cov(blood.glucose$y1, blood.glucose$y3),   
 cov(blood.glucose$y2, blood.glucose$y1),   
 cov(blood.glucose$y2, blood.glucose$y2),  
 cov(blood.glucose$y2, blood.glucose$y3),  
 cov(blood.glucose$y3, blood.glucose$y1),  
 cov(blood.glucose$y3, blood.glucose$y2),  
 cov(blood.glucose$y3, blood.glucose$y3)),   
 nrow = 3, byrow = TRUE)  
  
Sxx <- matrix(c(cov(blood.glucose$x1, blood.glucose$x1),   
 cov(blood.glucose$x1, blood.glucose$x2),  
 cov(blood.glucose$x1, blood.glucose$x3),  
 cov(blood.glucose$x2, blood.glucose$x1),   
 cov(blood.glucose$x2, blood.glucose$x2),  
 cov(blood.glucose$x2, blood.glucose$x3),  
 cov(blood.glucose$x3, blood.glucose$x1),  
 cov(blood.glucose$x3, blood.glucose$x2),  
 cov(blood.glucose$x3, blood.glucose$x3)),   
 nrow = 3, byrow = TRUE)  
  
Syx <- matrix(c(cov(blood.glucose$y1, blood.glucose$x1),   
 cov(blood.glucose$y1, blood.glucose$x2),  
 cov(blood.glucose$y1, blood.glucose$x3),  
 cov(blood.glucose$y2, blood.glucose$x1),   
 cov(blood.glucose$y2, blood.glucose$x2),  
 cov(blood.glucose$y2, blood.glucose$x3),  
 cov(blood.glucose$y3, blood.glucose$x1),  
 cov(blood.glucose$y3, blood.glucose$x2),  
 cov(blood.glucose$y3, blood.glucose$x3)),   
 nrow = 3, byrow = TRUE)  
  
Sxy <- t(Syx)  
  
S <- cbind(rbind(Syy, Sxy), rbind(Syx, Sxx))

S =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 77.614 | 0.986 | 23.731 | 100.076 | 4.869 | 34.317 |
| 0.986 | 36.202 | 15.221 | -46.457 | 30.370 | -32.078 |
| 23.731 | 15.221 | 57.459 | 13.407 | -6.421 | 1.476 |
| 100.076 | -46.457 | 13.407 | 959.499 | 299.361 | 232.637 |
| 4.869 | 30.370 | -6.421 | 299.361 | 500.189 | 61.809 |
| 34.317 | -32.078 | 1.476 | 232.637 | 61.809 | 527.016 |

## Problem 9

Various aspects of economic cycles were measured for consumer goods and producer goods by Tintner.

The variables are:

= length of cycle

= percentage of rising prices

= cyclical amplitude

= rate of change

The data for several items are given in the following table:

goods <- read.table('Software-Files/T5\_8\_GOODS.DAT',   
 col.names = c('Item', 'Type', 'y1', 'y2', 'y3', 'y4'))

Economic cycles measurements for consumer goods and producer goods

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item | Type | y1 | y2 | y3 | y4 |
| 1 | 1 | 72.0 | 50 | 8.0 | 0.5 |
| 2 | 1 | 66.5 | 48 | 15.0 | 1.0 |
| 3 | 1 | 54.0 | 57 | 14.0 | 1.0 |
| 4 | 1 | 67.0 | 60 | 15.0 | 0.9 |
| 5 | 1 | 44.0 | 57 | 14.0 | 0.3 |
| 6 | 1 | 41.0 | 52 | 18.0 | 1.9 |
| 7 | 1 | 34.5 | 50 | 4.0 | 0.5 |
| 8 | 1 | 34.5 | 46 | 8.5 | 1.0 |
| 9 | 1 | 24.0 | 54 | 3.0 | 1.2 |
| 1 | 2 | 57.0 | 57 | 12.5 | 0.9 |
| 2 | 2 | 100.0 | 54 | 17.0 | 0.5 |
| 3 | 2 | 100.0 | 32 | 16.5 | 0.7 |
| 4 | 2 | 96.5 | 65 | 20.5 | 0.9 |
| 5 | 2 | 79.0 | 51 | 18.0 | 0.9 |
| 6 | 2 | 78.5 | 53 | 18.0 | 1.2 |
| 7 | 2 | 48.0 | 50 | 21.0 | 1.6 |
| 8 | 2 | 155.0 | 44 | 20.5 | 1.4 |
| 9 | 2 | 84.0 | 64 | 13.0 | 0.8 |
| 10 | 2 | 105.0 | 35 | 17.0 | 1.8 |

res <- t.test(y1 ~ Type, data = goods\_df)  
res

##   
## Welch Two Sample t-test  
##   
## data: y1 by Type  
## t = -3.818, df = 14.635, p-value = 0.001749  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -65.01293 -18.36485  
## sample estimates:  
## mean in group 1 mean in group 2   
## 48.61111 90.30000

The p-value is 0.0017494. The consumer goods and producer goods differ in their length of cycle.

res <- t.test(y2 ~ Type, data = goods\_df)  
res

##   
## Welch Two Sample t-test  
##   
## data: y2 by Type  
## t = 0.57097, df = 12.427, p-value = 0.5782  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -6.069915 10.403248  
## sample estimates:  
## mean in group 1 mean in group 2   
## 52.66667 50.50000

The p-value is 0.5782013. The consumer goods and producer goods does not differ in their percentage of rising prices.

res <- t.test(y3 ~ Type, data = goods\_df)  
res

##   
## Welch Two Sample t-test  
##   
## data: y3 by Type  
## t = -3.166, df = 12.152, p-value = 0.008016  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -10.704651 -1.984238  
## sample estimates:  
## mean in group 1 mean in group 2   
## 11.05556 17.40000

The p-value is 0.0080162. The consumer goods and producer goods differ in their cyclical amplitude.

res <- t.test(y4 ~ Type, data = goods\_df)  
res

##   
## Welch Two Sample t-test  
##   
## data: y4 by Type  
## t = -0.71867, df = 16.083, p-value = 0.4827  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.5835069 0.2879514  
## sample estimates:  
## mean in group 1 mean in group 2   
## 0.9222222 1.0700000

The p-value is 0.4826578. The consumer goods and producer goods does not differ in their rate of change.

Use Hotelling’s T^2 test to test for a difference in the mean measurements vector of the Consumers Goods and the mean vector of the Producer Goods. State each hypotheses clearly, and interpret the results.

We would then like to test if the properties (four economic cycles measurements) are the same across consumer and producer goods.

:

: The are unequal

Let = Consumer goods and = Producer goods

### Mean vectors

y\_bar1 <- colMeans(consumer)  
y\_bar2 <- colMeans(producer)

|  |  |  |  |
| --- | --- | --- | --- |
| y1 | y2 | y3 | y4 |
| 48.611 | 52.667 | 11.056 | 0.922 |

|  |  |  |  |
| --- | --- | --- | --- |
| y1 | y2 | y3 | y4 |
| 90.3 | 50.5 | 17.4 | 1.07 |

### Covariance matrices

S1 <- cov(consumer)  
S2 <- cov(producer)

The respective sample covariances matrices for the consumer goods:

S1 =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | y1 | y2 | y3 | y4 |
| y1 | 289.674 | 12.042 | 44.368 | -1.878 |
| y2 | 12.042 | 21.750 | 8.083 | -0.179 |
| y3 | 44.368 | 8.083 | 28.403 | 0.905 |
| y4 | -1.878 | -0.179 | 0.905 | 0.224 |

The respective sample covariances matrices for the producer goods:

S2 =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | y1 | y2 | y3 | y4 |
| y1 | 870.400 | -113.278 | 25.117 | 0.804 |
| y2 | -113.278 | 119.833 | -5.000 | -1.761 |
| y3 | 25.117 | -5.000 | 8.600 | 0.519 |
| y4 | 0.804 | -1.761 | 0.519 | 0.173 |

### Pooled covariance matrix

Spl <- (1/ (nrow(consumer) + nrow(producer) - 2)) \*  
 ((nrow(consumer) - 1) \* S1 + (nrow(producer) - 1) \* S2)

=

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | y1 | y2 | y3 | y4 |
| y1 | 597.117 | -54.304 | 34.176 | -0.458 |
| y2 | -54.304 | 73.676 | 1.157 | -1.017 |
| y3 | 34.176 | 1.157 | 17.919 | 0.701 |
| y4 | -0.458 | -1.017 | 0.701 | 0.197 |

Inverse matrix of the sample pool covariance matrix of the two samples:

=

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | y1 | y2 | y3 | y4 |
| y1 | 0.002 | 0.002 | -0.006 | 0.038 |
| y2 | 0.002 | 0.017 | -0.011 | 0.132 |
| y3 | -0.006 | -0.011 | 0.082 | -0.358 |
| y4 | 0.038 | 0.132 | -0.358 | 7.102 |

Hotelling T^2, the F-statistic, and the P-value:

l1 <- nrow(consumer)  
l2 <- nrow(producer)  
my.q <- ncol(producer)  
  
  
T2 <- ((l1\*l2)/(l1+l2))\* (t(y\_bar1-y\_bar2) %\*% solve(Spl) %\*% (y\_bar1-y\_bar2) )   
T2

## [,1]  
## [1,] 18.46248

Fstat <-((l1+l2-my.q-1)\*T2)/((l1+l2-2)\*my.q)  
Fstat

## [,1]  
## [1,] 3.801099

pvalue <-1-pf(Fstat, my.q, l1+l2-my.q-1)  
pvalue

## [,1]  
## [1,] 0.02701984

The null hypothesis of no group mean difference is rejected. The two groups, consumer and producer goods, have a statistically significant joint mean difference.

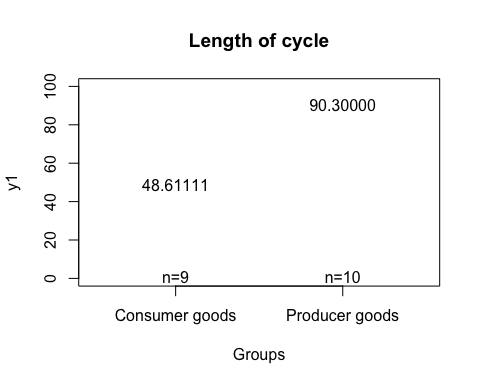
# MANOVA test  
goods.summary <- summary(manova(cbind(goods$y1,   
 goods$y2,  
 goods$y3,  
 goods$y4) ~ goods$Type), test="Hotelling")  
  
goods.summary

## Df Hotelling-Lawley approx F num Df den Df Pr(>F)   
## goods$Type 1 1.086 3.8011 4 14 0.02702 \*  
## Residuals 17   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

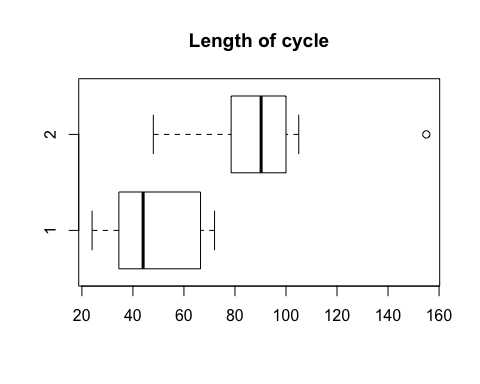
The Hotelling T2 multivariate t-test replaces each variable with a vector of means (, , and ) for each group.

Hotelling = 18.462, F = 3.801, P-value = 0.027

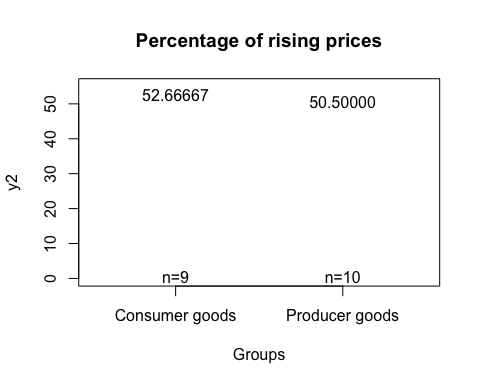
plotmeans(y1 ~ Type, data=goods, ylim=c(0,100), xlab="Groups", legends=c("Consumer goods","Producer goods"), main ="Length of cycle", connect=FALSE,mean.labels=TRUE, col=NULL, p=1.0)



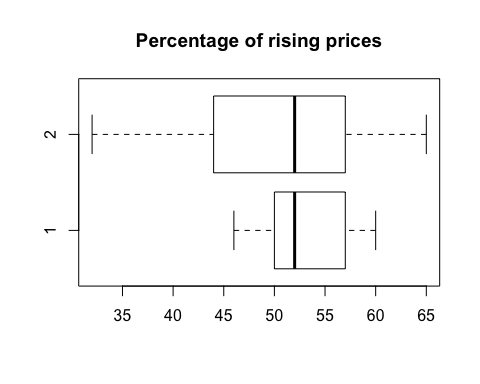
boxplot(y1 ~ Type, data = goods, main = "Length of cycle", horizontal = T)



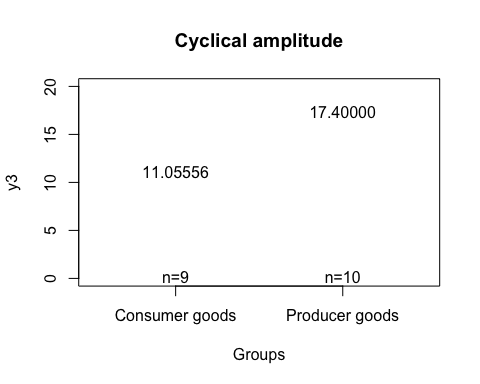
plotmeans(y2 ~ Type, data=goods, ylim=c(0,55), xlab="Groups", legends=c("Consumer goods","Producer goods"), main ="Percentage of rising prices", connect=FALSE,mean.labels=TRUE, col=NULL, p=1.0)



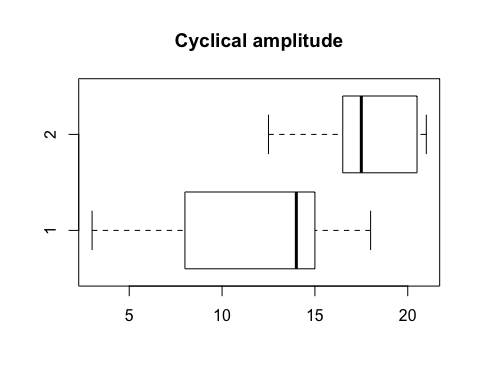
boxplot(y2 ~ Type, data = goods, main = "Percentage of rising prices", horizontal = T)



plotmeans(y3 ~ Type, data=goods, ylim=c(0,20), xlab="Groups", legends=c("Consumer goods","Producer goods"), main ="Cyclical amplitude", connect=FALSE,mean.labels=TRUE, col=NULL, p=1.0)



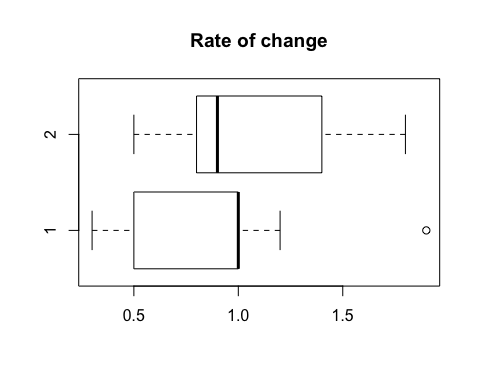
boxplot(y3 ~ Type, data = goods, main = "Cyclical amplitude", horizontal = T)



plotmeans(y4 ~ Type, data=goods, ylim=c(0,2), xlab="Groups", legends=c("Consumer goods","Producer goods"), main ="Rate of change", connect=FALSE,mean.labels=TRUE, col=NULL, p=1.0)



boxplot(y4 ~ Type, data = goods, main = "Rate of change", horizontal = T)



Compute Box M test of equal covariance matrices

heplots::boxM(goods [2:5],goods$Type)

##   
## Box's M-test for Homogeneity of Covariance Matrices  
##   
## data: goods[2:5]  
## Chi-Sq (approx.) = 10.565, df = 10, p-value = 0.3924

The Box M results indicated that the variance-covariance matrices of the two groups were equal (Chi-square = 10.565,, df = 10 p = 0.3924).