HW2. for Multivariate Statistics ll

September 29, 2020

201611531/Department of Statistics/Jeong Hojae

Chapter 6. Discriminant and Classification analysis(DCA)

[Data 6.12.2](admission.txt) is the admission data for graduate school of business. These data are the GPA(undergraduate grade point average) and GMAT(graduate management aptitude test) scores of the three clusters which were classified as : admit, C_2:do not C_1 admit and C_3:borderline.

[DATA 6.12.2] Admission Data from the Graduate School of Business(admission.txt)								
admit			do not admit			borderline		
applicant	GPA	GMAT	applicant	GPA	GMAT	applicant	GPA	GMAT
1	2.96	596	32	2.54	446	60	2.86	494
2	3.14	473	33	2.43	425	61	2.85	496
3	3.22	482	34	2.20	474	62	3.14	419
4	3.29	527	35	2.36	531	63	3.28	371
5	3.69	505	36	2.57	542	64	2.89	447
6	3.46	693	37	2.35	406	65	3.15	313
7	3.03	626	38	2.51	412	66	3.50	402
8	3.19	663	39	2.51	458	67	2.89	485
9	3.63	447	40	2.36	399	68	2.80	444
10	3.59	588	41	2.36	482	69	3.13	416
11	3.30	563	42	2.66	420	70	3.01	471
12	3.40	553	43	2.68	414	71	2.79	490
13	3.50	572	44	2.48	533	72	2.89	431
14	3.78	591	45	2.46	509	73	2.91	446
15	3.44	692	46	2.63	504	74	2.75	546
16	3.48	528	47	2.44	336	75	2.73	467
17	3.47	552	48	2.13	408	76	3.12	463
18	3.35	520	49	2.41	469	77	3.08	440
19	3.39	543	50	2.55	538	78	3.03	419
20	3.28	523	51	2.31	505	79	3.00	509
21	3.21	530	52	2.41	489	80	3.03	438
22	3.58	564	53	2.19	411	81	3.05	399
23	3.33	565	54	2.35	321	82	2.85	483
24	3.40	431	55	2.60	394	83	3.01	453
25	3.38	605	56	2.55	528	84	3.03	414
26	3.26	664	57	2.72	399	85	3.04	446
27	3.60	609	58	2.85	381			
28	3.37	559	59	2.90	384			
29	3.80	521						
30	3.76	646						
31	3.24	467						

(1) Compute the mean vectors, covariance matrices, and joint covariance matrix of the three clusters.

mean vectors

```
> colMeans(group1)
       GPA       GMAT
  3.403871 561.225806
> colMeans(group2)
     GPA     GMAT
  2.4825 447.0714
> colMeans(group3)
       GPA       GMAT
  2.992692 446.230769
```

In each variable, the difference of between the first group and the second-third groups appears visible, but difference of the second and third groups appear unvisible.

covariance matrices

```
> list(S1, S2, S3)
[[1]]
            GPA         GMAT
GPA  0.04355785 5.809677e-02
GMAT 0.05809677 4.618247e+03

[[2]]
             GPA        GMAT
GPA   0.03364907   -1.192037
GMAT -1.19203704 3891.253968

[[3]]
             GPA        GMAT
GPA   0.02969246   -5.403846
GMAT -5.40384615 2246.904615
```

joint covariance matrix

```
> Sp
             GPA        GMAT
GPA   0.03606795   -2.018759
GMAT -2.01875915 3655.901121
```

(2) Consider the multivariate normal distribution and the homogeneity of the covariance matrices of the three clusters.

multivariate normal distribution

```
> list(result_group1, result_group2, result_group3)
[[1]]
[[1]]$multivariateNormality
             Test          Statistic           p value Result
1 Mardia Skewness  0.471893695626844 0.976178819071029    YES
2 Mardia Kurtosis -0.816146237736216 0.414416501338956    YES
3             MVN               <NA>              <NA>    YES

[[1]]$univariateNormality
          Test  Variable Statistic   p value Normality
1 Shapiro-Wilk    GPA        0.9819    0.8640    YES
2 Shapiro-Wilk   GMAT        0.9775    0.7403    YES

[[1]]$Descriptives
      n       Mean     Std.Dev Median    Min   Max   25th   75th       Skew   Kurtosis
GPA   31   3.403871  0.2087052   3.39   2.96   3.8   3.27   3.54 0.08149089 -0.5619888
GMAT  31 561.225806 67.9576877 559.00 431.00 693.0 522.00 600.50 0.16697063 -0.6530970


[[2]]
[[2]]$multivariateNormality
             Test          Statistic           p value Result
1 Mardia Skewness   3.80540534067133 0.432981441551754    YES
2 Mardia Kurtosis -0.982183466405841 0.326009471362912    YES
3             MVN               <NA>              <NA>    YES

[[2]]$univariateNormality
          Test  Variable Statistic   p value Normality
1 Shapiro-Wilk    GPA        0.9800    0.8496    YES
2 Shapiro-Wilk   GMAT        0.9463    0.1595    YES

[[2]]$Descriptives
      n     Mean    Std.Dev Median    Min   Max   25th     75th        Skew   Kurtosis
GPA   28   2.4825  0.1834368   2.47   2.13   2.9   2.36   2.5775  0.27646115 -0.2726122
GMAT  28 447.0714 62.3799164 435.50 321.00 542.0 404.25 504.2500 -0.06529132 -1.0963701
```

```
[[3]]
[[3]]$multivariateNormality
             Test          Statistic              p value Result
1 Mardia Skewness 8.04014244073601 0.0901187440943936    YES
2 Mardia Kurtosis  2.0318152423983 0.0421723635318061     NO
3             MVN             <NA>                 <NA>     NO

[[3]]$univariateNormality
          Test  Variable Statistic   p value Normality
1 Shapiro-Wilk     GPA        0.9370    0.1136    YES
2 Shapiro-Wilk    GMAT        0.9685    0.5847    YES

[[3]]$Descriptives
      n       Mean    Std.Dev Median    Min    Max     25th     75th       Skew   Kurtosis
GPA   26   2.992692   0.172315   3.01   2.73    3.5   2.8675   3.0725  0.8064393 0.8235922
GMAT 26 446.230769 47.401525 446.00 313.00 546.0 419.0000 480.0000 -0.5036574 0.7583619
```

The first group and second group are satisfied with multivariate normality.
The third group is not satisfied with multivariate normality.
But in the case of skewness values, multivariate normality is satisfied.
So we can carry out the following processes; (3), (4), (5), (6)

the homogeneity of the covariance matrices

```
> boxM(admission[, -3], admission[, 3])

        Box's M-test for Homogeneity of Covariance Matrices

data:  admission[, -3]
Chi-Sq (approx.) = 16.074, df = 6, p-value = 0.01336
```

p-value = 0.01336 < 0.05 => reject H0 (Homogeneity of Covariance Matrices is satisfied)
So, It does not follow the homogeneity of the covariance matrix.

(3) Check whether the joint covariance matrix obtained in (1) is necessary by the result of (2).

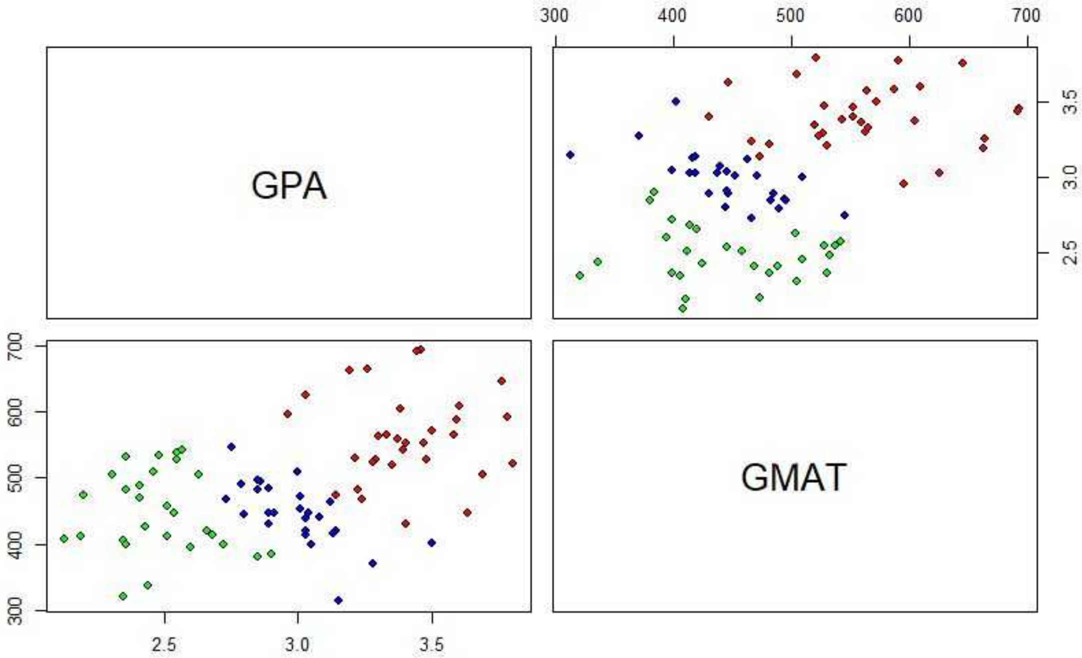
The results of (2) showed that the homogeneity of the covariance matrices was not followed.
The joint covariance matrix is used in the LDA method when each covariance matrix is homogeneous and multivariate normality is satisfied.
In this data, the QDA method is more appropriate because multivariate normality is satisfied and the covariance matrix is not homogeneous.
So, joint covariance matrix is not required.

(4) Select LDA or QDA according to the results of (2).

In this data, the QDA method is more appropriate because multivariate normality is satisfied and the covariance matrix is not homogeneous.

(5) Conduct a discriminant analysis that was not applied in (4) and compare the two results of LDA or QDA.

pairs(admission[1:2], pch=21, bg=c("red", "green", "blue")[unclass(admission$group)])



Look at the picture, it will be well-classified.

QDA

```
> table(admission$group, qcluster)
   qcluster
     1  2  3
  1 30  0  1
  2  0 27  1
  3  1  0 25
> (1-mean(admission$group==qcluster))*100
[1] 3.529412
```

LDA

```
> table(admission$group, lcluster)
   lcluster
     1  2  3
  1 28  0  3
  2  0 26  2
  3  1  1 24
> (1-mean(admission$group==lcluster))*100
[1] 8.235294
```

QDA's method has a smaller misclassification rate than LDA's method.
In this data, the QDA method is seem to be more appropriate.

(6) Compare the results using RSM and CVM to evaluate performance of QDA.

```
> list(confusion_admission, EAER)
[[1]]

     1  2  3
  1 30  0  1
  2  0 27  1
  3  1  1 24

[[2]]
[1] 4.705882
```

Comparing the results of the RSM and CVM methods in the QDA process, the misclassification rate in the CVM was higher. This is because the all data was used to create the discriminant function in the RSM method and evaluate the discriminant function. On the other hand, the RSM method tends to estimate EAER smaller than its actual values. There is a downside that this may lead to overfitting.

In contrast, the CVM had higher EAERs than the RSM method. Because the entire sample was divided into training and test samples. And then training sample was used to create a discriminant function and test sample was used to evaluate the degree of classification rate. In CVM, the sample size must be large, and the classification function is not used all data when they create the classification function. So, created the classification function may not obtain the value we want to obtain.

In the RSM method is smaller than CVM. And its misclassification rate difference is about 1.17647 As with the RSM method, the CVM also shows high performance for classification because there is no significant difference between them.

```r
library(HDclassif)
library(MASS)
library(MVN)
library(biotools)

setwd("G:/학교/2020 2학기 정호재/다변량통계학2/실습/20200929/Rdata")

admission<-read.table("admission.txt", header=T)
attach(admission)
head(admission)
dim(admission)
pairs(admission[1:2], pch=21, bg=c("red", "green", "blue")[unclass(admission$group)])
str(admission)
unique(admission[,3])
group1 = admission[which(admission$group == 1),1:2]
group2 = admission[which(admission$group == 2),1:2]
group3 = admission[which(admission$group == 3),1:2]
#평균 벡터
colMeans(group1)
colMeans(group2)
colMeans(group3)
#공분산 행렬
S1=cov(group1)
S2=cov(group2)
S3=cov(group3)
#합동 공분산 행렬
Sp=(30*S1+27*S2+25*S3)/(85-3)
#(31-1)*S1+(28-1)*S2+(26-1)*S3)/(85-3)
list(S1, S2, S3)
Sp
##################################
result_group1 = mvn(group1)
result_group2 = mvn(group2)
result_group3 = mvn(group3)
list(result_group1, result_group2, result_group3)

dim(group1)
dim(group2)
dim(group3)

library(biotools)
boxM(admission[, -3], admission[, 3])
##################################
n1=dim(group1)[1]
n2=dim(group2)[1]
n3=dim(group3)[1]

QDA=qda(group~., data=admission, prior=c(n1,n2,n3)/(n1+n2+n3))
qcluster=predict(QDA, admission)$class
table(admission$group, qcluster)
(1-mean(admission$group==qcluster))*100


LDA=lda(group~., data=admission, prior=c(n1,n2,n3)/(n1+n2+n3))
lcluster=predict(LDA, admission)$class
table(admission$group, lcluster)
(1-mean(admission$group==lcluster))*100
##################################
QDA=qda(group~., data=admission, prior=c(n1,n2,n3)/(n1+n2+n3), CV=TRUE)
confusion_admission=table(admission$group, QDA$class)
confusion_admission

# Expected actual error rate : EAER
EAER=(1-sum(diag(prop.table(confusion_admission))))*100
list(confusion_admission, EAER)

pairs(admission[1:2], pch=21, bg=c("red", "green", "blue")[unclass(admission$group)])
```