实验五：判别分析

林泽钦 3160104013 统计 1601

目录

1 实验目的和要求

1.1 实验目的

通过本试验项目，使学生理解并掌握如下内容：

- 处理判别分析的基本步骤；
- 熟悉各类判别方法；

1.2 实验内容

1.2.1 第一部分

利用第五章的数据和上机指导四，熟悉 R 在判别分析中的应用（请动手操作）；

1.2.2 第二部分

实验四采用 “肝胆病患者检查数据”。这是一组医院病人的资料，基本包括了四变量分别为：总胆红素 (umol/L)，白蛋白 (g/L)，碱性磷酸酶，谷丙转氨酶和医生诊断结果，希望通过这组数据 (学习样本) 建立判别肝、胆疾病的判别函数，并应用于 “体检数据” 中，根据体检资料分析是否有得肝胆疾病的可能性。具体步骤如下：

- 利用判别分析的统计方法建立肝、胆疾病的判别函数

(a) 检验组间均值显著性差异
(b) 分析组间方差显著性差异
(c) 建立判别函数
(d) 分析判别效果

- 根据判别函数去分析体检数据

1.3 实验环境

- R-3.5.1
- RStudio

2 实验过程与结果

2.1 程序包导入

```
library(mice)
```

2.2 数据导入与预处理

使用所给的“肝胆病患者检查数据.xls”数据，由于已经有类别标签group，故删去临床诊断一列，然后保存成逗号分隔文件“Pro5Data1.csv”。而对于“体检资料.xls”，则是筛选出有关变量 T_BIL, Alb, ALP 以及 ALT，并将其余信息筛去。处理后的数据保存至逗号分隔文件“Pro5Data2.csv”。

之后便可以导入数据：

```
Train <- read.csv("Pro5Data1.csv")
Test <- read.csv("Pro5Data2.csv")
```

使用 mice 包的 md.pattern() 函数统计缺失值：

```
md.pattern(Train, plot = FALSE)
```

```
##  /\     /\
## {  `---'  }
## {  0   0  }
## ==>  V <==  No need for mice. This data set is completely observed.
##  \  \|/  /
##   `-----'
```

```
##     T_BIL Alb ALP ALT group
## 344     1   1   1   1     1 0
##         0   0   0   0     0 0
```

Train 数据中没有缺失值，而对于 Test 数据：

```
md.pattern(Test, plot = FALSE)
```

```
##     T_BIL ALP ALT Alb
## 222     1   1   1   1 0
## 1       1   1   1   0 1
##         0   0   0   1 1
```

可以看到有一个观测值的 Alb 缺失，这里我们用均值代替缺失值：

```
attach(Test)
Alb[is.na(Alb)] = mean(Alb, na.rm = TRUE)
detach(Test)
```

2.3 检验组间均值显著性差异

使用多元方差分析方法分析组间均值的显著性差异：

```
G <- factor(Train$group)
MX <- as.matrix(Train[1:4])
fit <- manova(MX~G)
summary(fit, test = "Wilks")
```

```
##            Df   Wilks approx F num Df den Df    Pr(>F)
## G           4 0.16343    51.95     16 1027.1 < 2.2e-16 ***
## Residuals 339
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

根据上面的结果，p-value 十分小，故拒绝原假设，认为组间均值具有较显著的差异。

2.4 分析组间方差显著性差异

这里使用了实验指导中的 varcomp 函数，原假设为各组协方差阵相等。

```
source("varcomp.R")
```

首先分离出各组数据，之后计算各组协方差矩阵，以及统计各组观测值数目：

```
GroupList <- lapply(1:5, function(i) {subset(Train, group == i)[1:4]})
CovList <- lapply(1:5, function(i) {cov(GroupList[[i]])})
SampleNum <- sapply(1:5, function(i) {nrow(GroupList[[i]])})
```

之后就可以使用 varcomp() 函数进行检验：

```
varcomp(CovList, SampleNum)
```

```
##
##  Equality of Covariances Matrices Test
##
## data:  CovList
## corrected lambda* = NaN, df = 40, p-value = NA
```

GG，检查程序发现，在计算行列式的幂次过程中数值过大以致溢出，于是对根据课本 3.4 节的检验统计量对程序进行修改，检验统计量为：

$$M = -2\ln\lambda^* = (n-k)\ln\left|\frac{A}{n-k}\right| - \sum_{i=1}^{k}(n_t-1)\ln\left|\frac{A_t}{n_t-1}\right|$$

其中 A_t 为各个总体的样本离差阵，A 为总样本离差阵。当样本容量 n 较大时，其有近似分布：

$$(1-d)M = -2(1-d)\ln\lambda^* \sim \chi^2(f)$$

其中

$$f = \frac{1}{2}p(p+1)(k-1)$$

$$d = \begin{cases} \frac{2p^2+3p-1}{6(p+1)(k-1)}\left[\sum_{i=1}^{k}\frac{1}{n_i-1} - \frac{1}{n-k}\right], & if\ n_i \neq n_j\ for\ some\ i,j \\ \frac{(2p^2+3p-1)(k+1)}{6(p+1)(n-k)}, & otherwise \end{cases}$$

具体代码如下所示：

```
myVarComp <- function(CovList, SampleNum) {
  if (is.list(CovList)) {
    if (length(CovList) < 2)
      stop("CovList must be a list with at least 2 elements")
    ps <- as.vector(sapply(CovList, dim))
    if (sum(ps[1] == ps) != length(ps))
```

```
    stop("All covariance matrices must have the same dimension")
  p <- ps[1]
  k <- length(CovList)
  if (length(SampleNum) < 2) {
    stop("need at least 2 populations")
  } else if (length(SampleNum) != k) {
    stop("n must be equal length(covmat)")
  }
  DNAME <- deparse(substitute(CovList))
}
else
  stop("covmat must be a list")

n.t <- SampleNum - 1
n <- sum(SampleNum)
A.t <- lapply(1:length(CovList),
              function(i, mat, n) { n[i] * mat[[i]] },
              mat=CovList, n=n.t)
A <- matrix(colSums(matrix(unlist(A.t), ncol=p^2, byrow=T)), ncol=p)
logS.t <- sapply(1:length(CovList),
                 function(i, mat, n) {n[i] * log(det(mat[[i]]))},
                 mat=CovList, n=n.t)
logS <- (n-k)*log(det(A/(n-k)))
f <- (k-1)*p*(p+1)/2

if(sum(n.t[1] == n.t) < length(n.t)) {
  d <- (2*p^2+3*p-1)/(6*(p+1)*(k-1))*(sum(1/n.t)-1/(n-k))
} else {
  d <- (2*p^2+3*p-1)*(k+1)/(6*(p+1)*(n-k))
}

STATISTIC <- (1-d)*(logS - sum(logS.t))
PVAL <- 1 - pchisq(STATISTIC, f)
```

```
  names(STATISTIC) <- "(1-d)*M"
  names(f) <- "df"
  RVAL <- structure(list(statistic = STATISTIC, parameter = f,
                         p.value = PVAL, data.name = DNAME,
                         method = "Equality of Covariances Matrices Test"),
                    class="htest")
  return(RVAL)
}
```

在上面的代码中，n.t 代表各类别样本的数量减一，n 代表总样本数量，A.t 代表样本离差阵，A 为总样本离差阵。而 logS.t 代表

$$(n_t - 1) \ln \left| \frac{A_t}{n_t - 1} \right|$$

logS 则为

$$(n - k) \ln \left| \frac{A}{n - k} \right|$$

利用新的函数重新进行检验，如下所示，可以发现得到的 p-value 十分小，于是拒绝原假设，认为各类别的协方差阵不相等。

```
myVarComp(CovList, SampleNum)
```

```
##
##  Equality of Covariances Matrices Test
##
## data:  CovList
## (1-d)*M = 1991.6, df = 40, p-value < 2.2e-16
```

2.5 建立判别函数

由上述检验，我们认为各个类别的协方差阵不相等，所以使用二次判别函数：

$$\hat{d}_i^Q(\mathbf{x}) = -\frac{1}{2} \ln |\mathbf{S}_i| - \frac{1}{2} (\mathbf{x} - \overline{\mathbf{x}}_i)' \mathbf{S}_i^{-1} (\mathbf{x} - \overline{\mathbf{x}}_i) + \ln p_i$$

这里我们用各个类别出现的频率估计先验概率 p_i，于是可以写出判别程序，这里 MuList 为各类别样本均值组成的 list。

```
discriminate <- function(New, MuList, CovList, SampleNum) {
  prior <- SampleNum/sum(SampleNum)
  score <- sapply(1:length(CovList),
                  function(i, mus, covs, ps) {-log(det(covs[[i]]))/2 -
                      mahalanobis(New, mus[[i]], covs[[i]]) + log(ps[i])},
                  mus=MuList, covs=CovList, ps=prior)
  return(which.max(score))
}
```

2.6 分析判别效果

首先计算各类别的均值：

```
MuList <- lapply(1:5, function(i) {colMeans(GroupList[[i]])})
```

利用上述判别函数计算判别结果：

```
correct = 0
for (i in 1:nrow(MX)) {
  res <- discriminate(MX[i, ], MuList, CovList, SampleNum)
  if(res == Train$group[i])
    correct <- correct + 1
}
correct
```

```
## [1] 243
```

2.7 根据判别函数去分析体检数据

对 Test 的每一个观测值调用判别函数：

```
dis.res <- sapply(1:nrow(Test),
                  function(i) {discriminate(Test[i, ], MuList, CovList, SampleNum)[1]})
```

Test 的前几个判别结果如下：

```
head(dis.res)
```

```
## 1 2 3 4 5 6
## 5 5 3 3 3 3
```