*Multivariate Analysis for the Behavioral Sciences,*
Second Edition (Chapman and Hall/CRC, 2019)

# Examples of Chapter 18: Grouped Multivariate Data

*Kimmo Vehkalahti and Brian S. Everitt*

*20 December 2018*

## Contents

## Examples

## Table 18.1: Wechsler Adult Intelligence IQ Scores for Five Men and Five Women

**Source of data**: Willerman, L., Schultz, R., Rutledge, J. N. and Bigler, E. D. (1991). In vivo brain size and intelligence. *Intelligence*, 15, 223–228. https://doi.org/10.1016/0160-2896(91)90031-8

```
iq_scores <- read.csv("data/iq.csv")
iq_scores[1:5, ]
```

```
##   FSIQ VIQ PIQ
## 1  140 150 124
## 2  139 123 150
## 3  133 129 128
## 4   89  93  84
## 5  133 114 147
```

```
iq_scores[21:25, ]
```

```
##    FSIQ VIQ PIQ
## 21  133 132 124
## 22  137 132 134
## 23   99  90 110
## 24  138 136 131
## 25   92  90  98
```

```
n1 = 20; n2 = 20; q = 3
sex <- rep(c(1, 2), c(n1, n2))

# Hotelling's T2 test
m1 <- apply(iq_scores[sex==1, ], 2, mean)
m2 <- apply(iq_scores[sex==2, ], 2, mean)
S1 <- var(iq_scores[sex==1, ])
S2 <- var(iq_scores[sex==2, ])

S <- ((n1-1)*S1 + (n2-1)*S2) / (n1+n2-2)
T2 <- t(m1-m2) %*% solve(S) %*% (m1-m2)
F <- (n1+n2-q-1)*T2/((n1+n2-2)*q)
pvalue <- 1 - pf(F, q, n1+n2-q-1)
c(T2, F, pvalue) # raw output of the above statistics (see below for an alternative!)
```

```
## [1] 0.2707262 0.0854925 0.9675321
```

Quoting our own text on p.367 (here achieved with LaTeX and inline **R** code of **R Markdown**):

Here, Hotelling's $T^2$ takes the value 0.27, with the corresponding $F$-statistic being 0.09, having 3 and 36 degrees of freedom; the associated $p$-value is 0.97. There is no evidence of a gender difference on the three measures of IQ.

## Table 18.2: SIDs Data

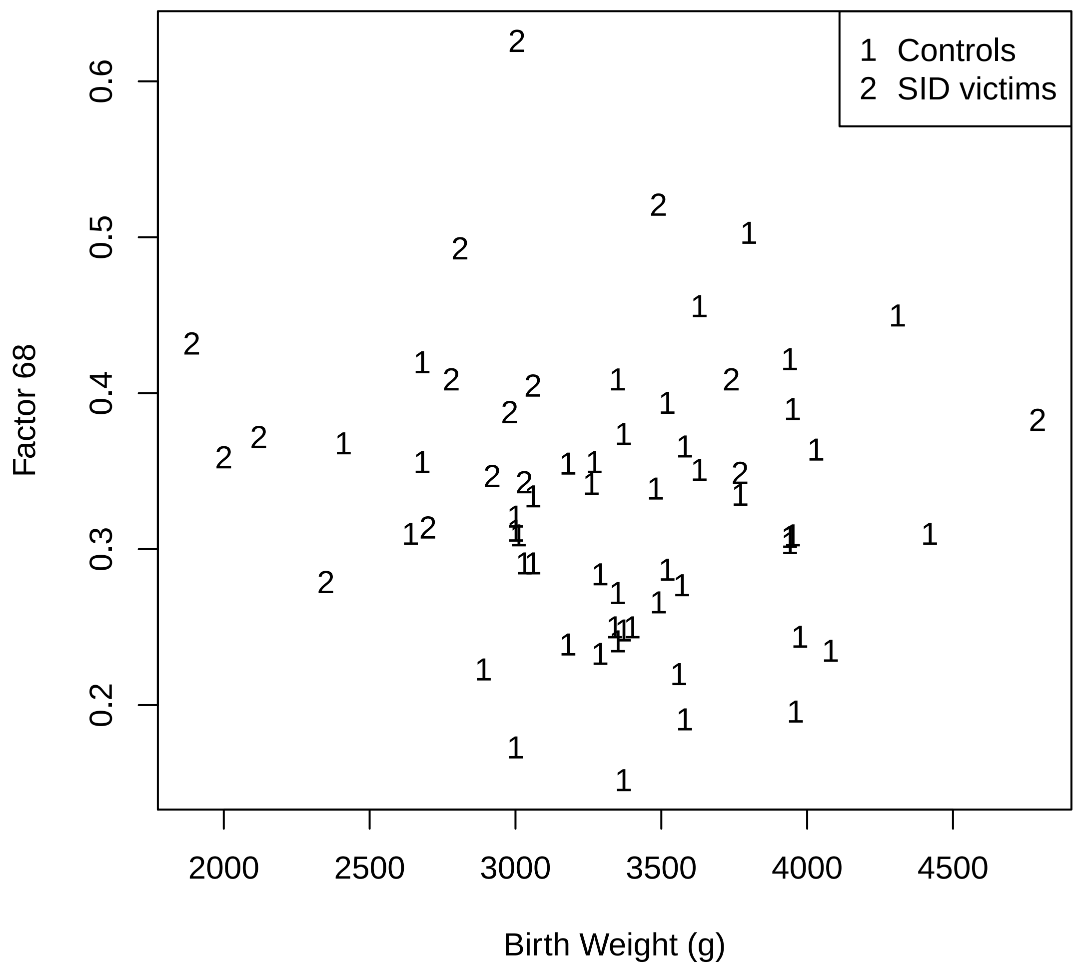
**Source of data**: Spicer, C. C., Lawrence, C. J. and Southall, D. P. (1987). Statistical analysis of heart rates and subsequent victims of sudden infant death syndrome. *Statistics in Medicine*, 6, 159–166. (Appendix II, p.165) https://doi.org/10.1002/sim.4780060208

```r
sid <- read.csv("data/sid.csv")
sid[45:54, ]
```

```
##    Group    HR   BW   F68 GA
## 45     1 108.2 3000 0.321 37
## 46     1 131.1 4310 0.450 40
## 47     1 129.7 3975 0.244 40
## 48     1 142.0 3000 0.173 40
## 49     1 145.5 3940 0.304 41
## 50     2 139.7 3740 0.409 40
## 51     2 121.3 3005 0.626 38
## 52     2 131.4 4790 0.383 40
## 53     2 152.8 1890 0.432 38
## 54     2 125.6 2920 0.347 40
```

**Figure 18.2**

```
plot(sid$BW, sid$F68, xlab = "Birth Weight (g)", ylab = "Factor 68", type = "n")
text(sid$BW, sid$F68, labels = sid$Group)
legend("topright", c("Controls", "SID victims"), pch = c("1", "2"))
```



### Table 18.3

```r
# Find discriminant function manually using only BW and F68:
sid1 <- sid[sid$Group == 1, -c(1, 2, 5)]
sid2 <- sid[sid$Group == 2, -c(1, 2, 5)]
m1 <- apply(sid1, 2, mean)
m2 <- apply(sid2, 2, mean)
S1 <- var(sid1)
S2 <- var(sid2)
n1 <- length(sid1[, 1])
n2 <- length(sid2[, 1])
S <- ((n1-1)*S1+(n2-1)*S2)/(n1+n2-2)
a <- solve(S)%*%(m1-m2)
z1 <- sum(a*m1)
z2 <- sum(a*m2)
z <- 0.5*(z1+z2)

# results of the Table briefly (could be more verbose):
m1; m2
```

```
##           BW          F68
## 3437.8571429    0.3108163
```

```
##           BW          F68
## 2964.6875000    0.4018125
```

```r
S1; S2
```

```
##               BW         F68
## BW  1.952708e+05 3.244494048
## F68 3.244494e+00 0.005842236
```

```
##               BW         F68
## BW  5.453049e+05 7.759604167
## F68 7.759604e+00 0.007274562
```

```r
S
```

```
##               BW         F68
## BW  278612.27679 4.319520266
## F68      4.31952 0.006183266
```

```r
a
```

```
##              [,1]
## BW    0.001947562
## F68 -16.077054121
```

```r
z1; z2; z
```

```
## [1] 1.698431
```

```
## [1] -0.6860473
```

```
## [1] 0.5061916
```

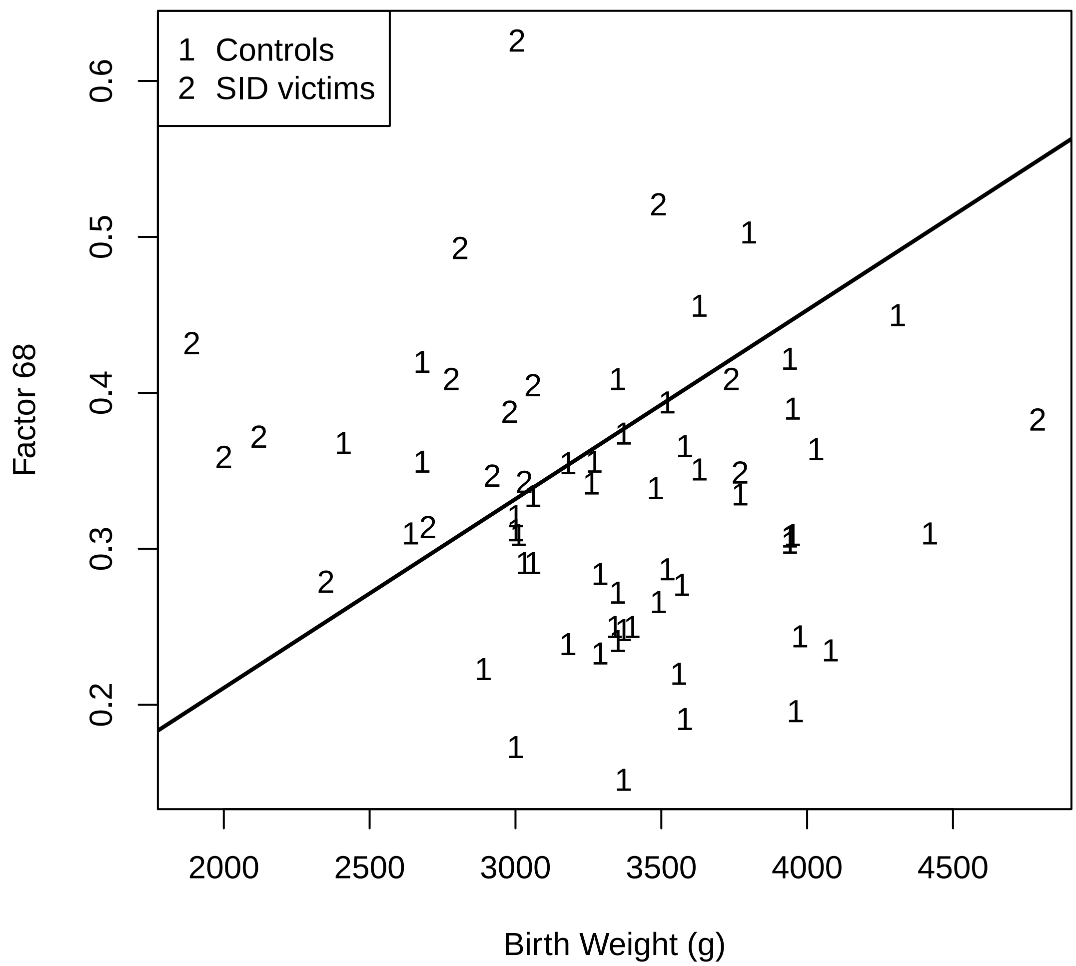
```r
# (mis)classification table:

#install.packages("MASS")
library("MASS")

sid_ldf <- lda(Group ~ BW + F68 + HR + GA, data = sid, prior = c(0.5, 0.5))
table(sid$Group, predict(sid_ldf)$class)
```

```
##
##      1  2
##   1 41  8
##   2  3 13
```

So, the (too optimistic) percentage of misclassifications is $(100 \times (8 + 3)/65)\% = 16.9\%$.

**Figure 18.3**

```
plot(sid$BW, sid$F68, xlab = "Birth Weight (g)", ylab = "Factor 68", type = "n")
text(sid$BW, sid$F68, labels = sid$Group)
slope <- -a[1]/a[2]
intercept <- z/a[2]
abline(a=intercept, b=slope, lwd=2)
legend("topleft", c("Controls", "SID victims"), pch = c("1", "2"))
```



## Table 18.4: Data from Investigation of Risk Taking

**Source of data**: Timm, N. H. (2002). *Applied Multivariate Analysis*. Springer, New York. https://doi.org/10.1007/b98963

```
Timm <- read.csv("data/Timm.csv")
str(Timm)
```

```
## 'data.frame':    86 obs. of  3 variables:
##  $ Group: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Low  : int  8 18 8 12 15 12 12 18 29 6 ...
##  $ High : int  28 28 23 20 30 32 20 31 25 28 ...
```

```
head(Timm, n = 10)
```

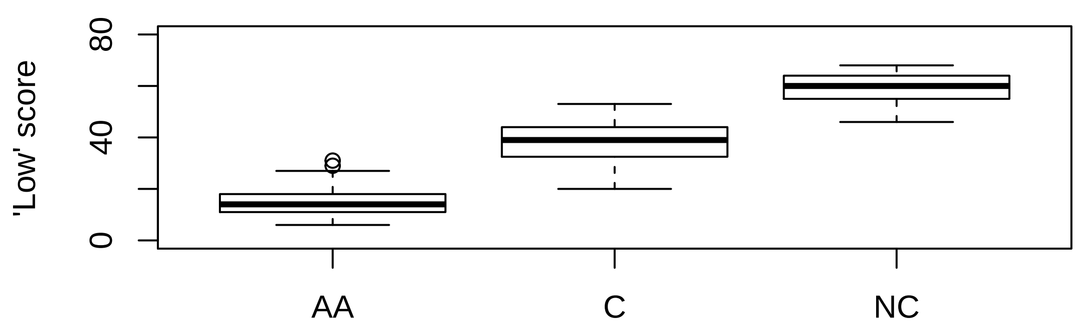
```
##    Group Low High
## 1      1   8   28
## 2      1  18   28
## 3      1   8   23
## 4      1  12   20
## 5      1  15   30
## 6      1  12   32
## 7      1  12   20
## 8      1  18   31
## 9      1  29   25
## 10     1   6   28
```

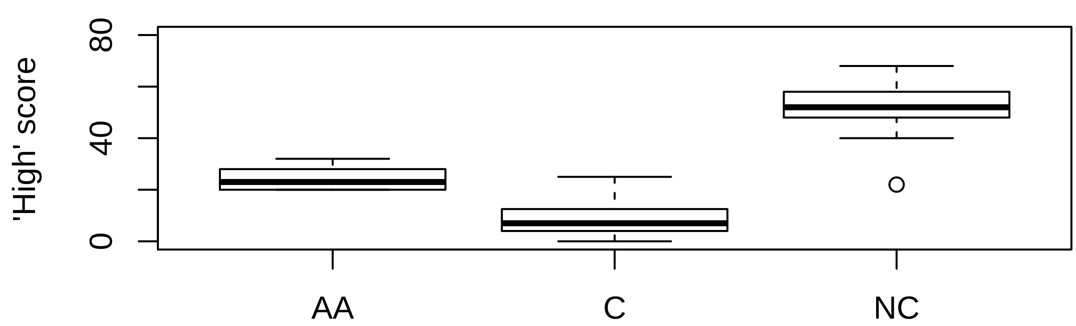
```
tail(Timm, n = 10)
```

```
##    Group Low High
## 77     3  65   58
## 78     3  46   53
## 79     3  46   49
## 80     3  47   40
## 81     3  64   22
## 82     3  64   54
## 83     3  63   64
## 84     3  63   56
## 85     3  64   44
## 86     3  63   40
```

### Figure 18.4

```
par(mfrow=c(2,1))
treatments <- c("AA", "C", "NC")
ylim <- c(0, 80)
with(Timm, {
boxplot(Low[Group==1], Low[Group==2], Low[Group==3],
        names = treatments, ylab = "'Low' score",  ylim = ylim)
  boxplot(High[Group==1], High[Group==2], High[Group==3],
        names = treatments, ylab = "'High' score", ylim = ylim)
})
```



### Table 18.5

```r
Timm_manova <- manova(cbind(Low, High) ~ Group, data = Timm)
summary(Timm_manova, test = "Pillai")
```

```
##           Df  Pillai approx F num Df den Df    Pr(>F)    
## Group      1 0.86605   268.32      2     83 < 2.2e-16 ***
## Residuals 84                                             
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(Timm_manova, test = "Wilks")
```

```
##           Df   Wilks approx F num Df den Df    Pr(>F)    
## Group      1 0.13395   268.32      2     83 < 2.2e-16 ***
## Residuals 84                                             
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(Timm_manova, test = "Hotelling")
```

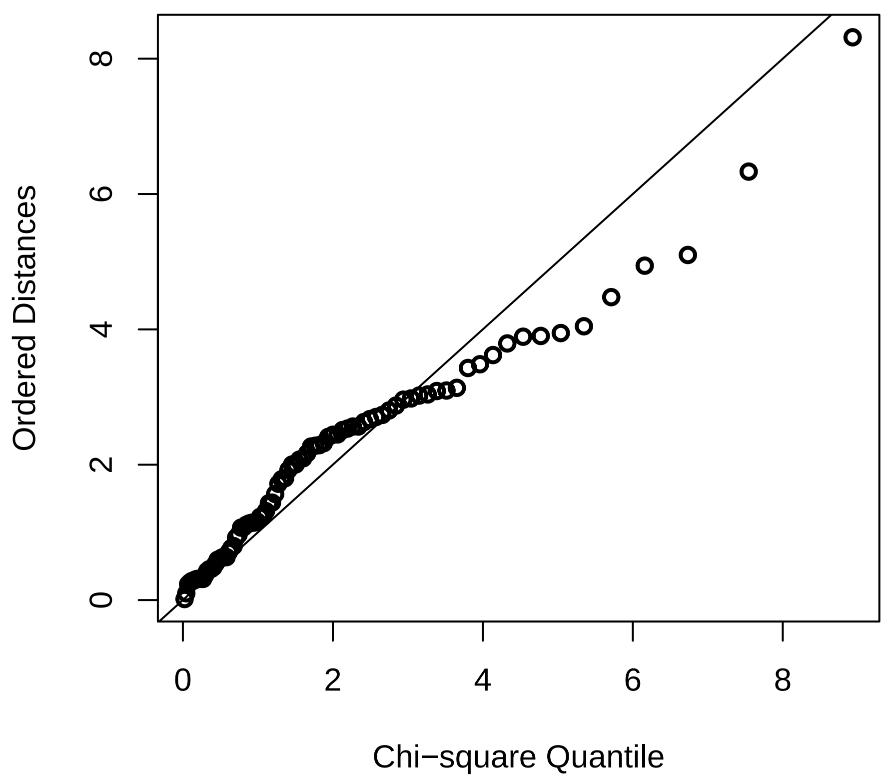
```
##           Df Hotelling-Lawley approx F num Df den Df    Pr(>F)    
## Group      1           6.4656   268.32      2     83 < 2.2e-16 ***
## Residuals 84                                                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(Timm_manova, test = "Roy")
```

```
##           Df    Roy approx F num Df den Df    Pr(>F)    
## Group      1 6.4656   268.32      2     83 < 2.2e-16 ***
## Residuals 84                                            
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### Figure 18.5

```r
# function copied here from Chapter 12 examples:
chisplot <- function(x) {
    if (!is.matrix(x)) stop("x is not a matrix")
    n <- nrow(x)
    p <- ncol(x)
    xbar <- apply(x, 2, mean)
    S <- var(x)
    S <- solve(S)
    index <- (1 : n) / (n + 1)
    xcent <- t(t(x) - xbar)
    di <- apply(xcent, 1, function(x,S) x %*% S %*% x, S)

    quant <- qchisq(index, p)
    plot(quant, sort(di),
         ylab = "Ordered Distances",
         xlab = "Chi-square Quantile",
         lwd = 2, pch = 1)
}

chisplot(residuals(Timm_manova))
abline(a = 0, b = 1)
```



## Table 18.6: fMRI Data

```r
fMRI <- read.csv("data/fMRI.csv")
head(fMRI, n = 20)
```

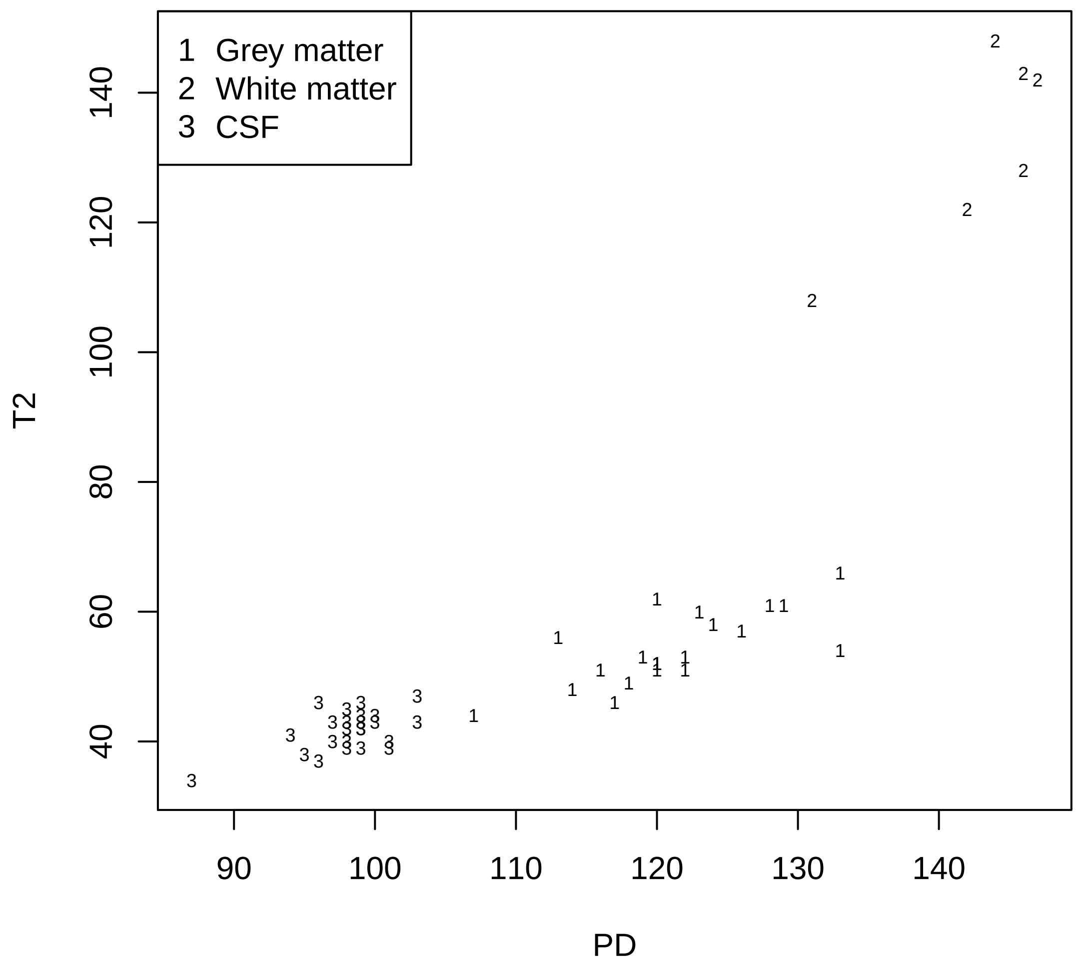
```
##      PD  T2 Class
## 1   124  58     1
## 2   107  44     1
## 3    98  45     3
## 4    87  34     3
## 5   129  61     1
## 6    99  42     3
## 7   142 122     2
## 8    96  37     3
## 9    99  44     3
## 10  144 148     2
## 11  133  66     1
## 12  122  53     1
## 13   98  40     3
## 14   99  46     3
## 15   97  43     3
## 16   99  42     3
## 17  103  47     3
## 18  120  62     1
## 19   97  40     3
## 20  117  46     1
```

```r
tail(fMRI, n = 20)
```

```
##      PD  T2 Class
## 31  113  56     1
## 32  123  60     1
## 33   98  42     3
## 34   99  39     3
## 35  101  40     3
## 36  146 143     2
## 37   95  38     3
## 38  146 128     2
## 39   99  43     3
## 40  147 142     2
## 41  128  61     1
## 42  120  52     1
## 43  131 108     2
## 44  126  57     1
## 45  103  43     3
## 46  114  48     1
## 47   98  39     3
## 48  118  49     1
## 49  122  51     1
## 50  101  39     3
```

**Figure 18.6**

```
plot(fMRI$T2 ~ fMRI$PD, type = "n", pch = fMRI$Class, xlab = "PD", ylab = "T2")
text(fMRI$PD, fMRI$T2, labels = fMRI$Class, cex=0.6)
legend("topleft", c("Grey matter", "White matter", "CSF"), pch = c("1", "2", "3"))
```



### Table 18.7

```
Class <- fMRI$Class
# calculate means of each class
m1 <- apply(fMRI[Class==1, -3], 2, mean)
m2 <- apply(fMRI[Class==2, -3], 2, mean)
m3 <- apply(fMRI[Class==3, -3], 2, mean)

# numbers in each class
n1 <- length(fMRI[Class==1, 1])
n2 <- length(fMRI[Class==2, 1])
n3 <- length(fMRI[Class==3, 1])

# covariance matrices
S1 <- (n1-1)*var(fMRI[Class==1, -3])/(n1-1)
S2 <- (n2-1)*var(fMRI[Class==2, -3])/(n2-1)
S3 <- (n3-1)*var(fMRI[Class==3, -3])/(n3-1)

# pooled covariance matrix
S123 <- ((n1-1)*var(fMRI[Class==1, -3]) +
         (n2-1)*var(fMRI[Class==2, -3]) +
         (n3-1)*var(fMRI[Class==3,-3])) / (n1+n2+n3-3)

# results briefly:
m1; m2; m3
```

```
##     PD     T2
## 121.20  54.25
```

```
##       PD       T2
## 142.6667 131.8333
```

```
##       PD       T2
## 98.08333 41.66667
```

```
c(n1, n2, n3)
```

```
## [1] 20  6 24
```

```
S1; S2; S3
```

```
##          PD       T2
## PD 42.48421 26.94737
## T2 26.94737 33.25000
```

```
##          PD        T2
## PD 35.86667  74.93333
## T2 74.93333 233.76667
```

```
##           PD       T2
## PD 10.340580 5.768116
## T2  5.768116 9.623188
```

```
S123
```

```
##          PD       T2
## PD 26.05035 21.68794
## T2 21.68794 43.01950
```

## Table 18.8

```r
# coefficients for each classification class
invS <- solve(S123)
a1 <- invS%*%(m1-m2)
a2 <- invS%*%(m1-m3)
a3 <- invS%*%(m2-m3)

# thresholds
z12 <- (m1%*%a1+m2%*%a1)/2
z13 <- (m1%*%a2+m3%*%a2)/2
z23 <- (m2%*%a3+m3%*%a3)/2

# results (very) briefly:

a1; a2; a3
```

```
##          [,1]
## PD  1.167354
## T2 -2.391958
```

```
##           [,1]
## PD  1.1095706
## T2 -0.2668783
```

```
##            [,1]
## PD -0.05778304
## T2  2.12507943
```

```r
z12; z13; z23
```

```
##            [,1]
## [1,] -68.53887
```

```
##           [,1]
## [1,] 108.8561
```

```
##          [,1]
## [1,] 177.395
```

**Figure 18.7**

```
# code from previous figure:
plot(fMRI$T2 ~ fMRI$PD, type = "n", pch = fMRI$Class, xlab = "PD", ylab = "T2")
text(fMRI$PD, fMRI$T2, labels = fMRI$Class, cex=0.6)
legend("topleft", c("Grey matter", "White matter", "CSF"), pch = c("1", "2", "3"))

# add discriminant functions
abline(z12/a1[2], -a1[1]/a1[2])
abline(z13/a2[2], -a2[1]/a2[2], lty=2)
abline(z23/a3[2], -a3[1]/a3[2], lty=3)
legend("bottomright", c("Grey/White", "Grey/CSF", "White/CSF"), lty=1:3)
```