STAT501
Multivariate Analysis

Bernard A. Ellem

December 15, 2005

Contents

List of Figures

List of Tables

B	Bilodeau and Brenner	Chs. 5, 7, 8,9,10,11 and 13.
DG	Dillon and Goldstein	Chs. 1, 2, 5, 9, 10 and 11.
J	Johnson D.E.	Chs. 1, 3, 5, 7 and 11.
JW	Johnson and Wichern	All but 9.
K	Krzanowski (Part 1)	Chs. 2, 4, 6 and 7.
S	Sharma	Chs. 1, 4, 7, 8, 9, 11, 12 and 13.

Table 1: KEY to Authors

Chapter 1

Introduction

(DG, Chapter 1, pp19–22)
(J, 1.1, pp1–7)
(JW Chapter 1, pp2–3)
(S, Chapter 1, pp4–12)

Multivariate analysis is the name given to the class of statistical techniques that attempt to describe the situation where each observation has more than one response variable. With the advent of the digital computer there has been an explosion in the availability of these techniques as well as an increase in the size of data sets collected. As it is impossible to cover all of the available methods, the rationale of the approach chosen in this unit is to demonstrate the common link between these methods via the mathematics of the multivariate inferential methods. This line of attack will provide the basis for any future work using multivariate methods.

The general objectives of this unit are :

- to develop an understanding of the theory underlying the definition, role and applications of multivariate methods, and,
- to emphasize the usefulness of the multivariate approach via applications.

To this end throughout these notes both the theoretical basis and practical application of methods will be covered. The computer package used will be R[1].

The specific objectives of this chapter are for you to be able to :

- recognise the multivariate equivalents of simple univariate problems, and
- describe the transition from univariate (single–response) to bivariate (two–response) models.

Later in the unit the generalization from bivariate to multivariate responses will be addressed, again using both theoretical and practical approaches.

[1]Ihaka R. and Gentleman R., (1996), *R : A Language for Data Analysis and Graphics*, Vol. 5, No. 3, pp299–314.

1.1 Definitions

The term 'multivariate' refers to *multiple* responses, and so multiple regression, for example, is not a multivariate method as it allows only for a single response with multiple predictors. Multiple regression could thus be called a *multivariable* method, where the term *multivariable* describes the multiplicity of predictor variables. Note the use of the terms 'response' and 'predictor' in contrast to the ambiguous terms 'dependent' and 'independent' variables. The term 'dependent' can also refer to correlation and so if the term 'dependent' is used for correlated responses these could be termed dependent dependent variables! The most potentially confusing situation occurs with correlated predictors which would then become dependent independent variables ...

For some multivariate problems the classification into 'responses' and 'predictors' is not appropriate, but then neither would the terms 'dependent' and 'independent' variables in such cases.

1.2 Broad Division of Methods

(Venables and Ripley, Chapter 11, page 329)

The broad division of multivariate methods is into two groups. The first group assumes a given structure, while the second attempts to find structure from the data alone. These correspond to the terms *dependence* and *interdependence* methods used by some authors (DG, p19 and S p4), while others (JW p2–3) recognize more than one division method. The terms *supervised* and *unsupervised* methods from the pattern–recognition literature correspond to this dichotomy. The interdependence methods are called *data mining* in some applications. Examples of the two types of methods would be discriminant analysis (dependent) and cluster analysis (interdependent). In discriminant analysis the groups are given *a priori* and the goal is to find a rule describing the division in terms of given predictors, while cluster analysis is concerned with finding the groupings based on (dis)similarities derived from given attributes variables alone.

1.3 Distance Measures

(S pp42–45)
(JW pp20–28)

Distances fall into three categories :

Euclidean

The squared *Euclidean* distance between two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in multivariate response space is defined as

$$(\boldsymbol{x}_i - \boldsymbol{x}_j)'(\boldsymbol{x}_i - \boldsymbol{x}_j)$$

where $\boldsymbol{x}' = (x_1, x_2, \ldots, x_p)$ for p responses.

This is an unscaled or unstandardised measure of distance between points.

Statistical (standard)

The squared *statistical* distance between two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is

$$\boldsymbol{Z}_{ij}{}'\boldsymbol{Z}_{ij}$$

where

$$\boldsymbol{Z}_{ij} = (\boldsymbol{x}_i - \boldsymbol{x}_j)/S_{ij}$$

where S_{ij} is a measure of the standard deviation. This can give a staggeringly different answer to the corresponding Euclidean distance (S, p43).

Mahalanobis

(JW calls this the (generalized) statistical distance, pp 26–27)

Standard distance can correct for differences in scale of measure, but not for plane rotations which for linear systems correspond to correlation between responses. Thus this measure incorporates not only the individual variance for each variable, but also the covariance between responses.

The squared *Mahalanobis* distance between two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is

$$(\boldsymbol{x}_i - \boldsymbol{x}_j)'\boldsymbol{S}^{-1}(\boldsymbol{x}_i - \boldsymbol{x}_j)$$

where $\boldsymbol{S}$ is the covariance matrix. The measure can also be interpreted in terms of a rotation of axes to a new coordinate system of uncorrelated variables.

Both previous distance measures can be shown to be special cases of this distance measure.

1.4 Generalized Variance

In moving from univariate to bivariate responses, the new concept in variability is *correlation*. Thus we now have 3 items : two variances σ_1^2, σ_2^2 and the covariance $\text{cov}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sigma_{12} = \sigma_1\sigma_2\rho_{12}$ where ρ_{12} is the correlation between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$.[2] So the covariance matrix is

$$\begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}$$

while the corresponding correlation matrix is

$$\begin{pmatrix} 1 & \rho_{12} \\ \rho_{12} & 1 \end{pmatrix}$$

The concept of *generalized variance* is an attempt to describe all these quantities by a single number(!). Contenders include :

[2] JW p128 use the notation $\sigma_{11} = \sigma_1^2, \ldots, \sigma_{22} = \sigma_2^2$.

1. the determinant of the sample covariance matrix $|\boldsymbol{S}|$ which in this case is

$$|\sigma_1^2\sigma_2^2 - \sigma_{12}^2| = \sigma_1^2\sigma_2^2|1 - \rho_{12}^2|.$$

2. the trace of $|\boldsymbol{S}| = \sigma_1^2 + \sigma_2^2$.

Probably the most important concept to be gleaned from this idea is that

$$|\boldsymbol{S}| = \lambda_1\lambda_2 \ldots \lambda_p$$

where $\lambda_1, \ldots, \lambda_p$ are the eigenvalues of $\boldsymbol{S}$. This result will be used later in PCA, together with other measures of generalized variance.

1.5 Bivariate Normal

(J, p19)
(JW, p128)

The path from the Univariate Normal to the Multivariate Normal is best taken via consideration of the Bivariate situation. The concept of Mahalanobis distance and generalized variance will now be seen to be natural elements inside the Bivariate Normal form. In fact, the Bivariate is best written initially in terms of the Multivariate, viz,

$$F(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}}\, e^{-(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})/2}$$

Notice a form of the generalized variance in $\boldsymbol{\Sigma}$ and the square of a Mahalanobis distance in the exponent. An expanded form for $p = 2$ is given in JW page 128, equation (4.5), but as commented by JW the "The (bivariate) expression is somewhat unwieldy and the compact general form ... is more informative...". The bivariate form is reproduced here for completeness :

$$f(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}}\, e^{-\frac{1}{2(1-\rho^2)}\left[\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} - 2\rho\frac{(x_1-\mu_1)}{\sigma_1}\frac{(x_2-\mu_2)}{\sigma_2}\right]}$$

This particular form can have other uses, such as examining independence and marginal distributions.

The real value of the Bivariate Normal is that it can be displayed graphically and all the generalizations to full Multivariate Normal from Univariate can be shown in two dimensions, making the jump to the multivariate case interpretable. Diagrams of the distribution surface are given on page 130 of JW (contour plots page133) and Bilodeau page 59 (contour plot page 60). Flury and Riedwyl (page109) shows the Bivariate Normal in a practical setting of displaying the two distributions of forged and genuine bank notes! Bilodeau page 61 shows the uninformative contour plot for a Trivariate Normal distribution ...

The contour diagrams from JW page 133 bear special attention.

The case $\rho_{12} = 0$ (Fig 4.4 (a)) shows independent Bivariate Normal distributions, ie, two independent Univariate Normals. Their independence can be verified by showing that any x_1 slice of the joint density produces the same conditional density for x_2 and conversely.

By contrast, in the case $\rho_{12} = 0.75$ (Fig 4.4 (b)), the conditional distribution for x_2 depends on the choice of x_1, and conversely. So x_1 and x_2 are not independent, ie, they are dependent. That is, the conditional distribution for any of the two variables depends on the specified value of the other.

1.6 Examples of Multivariate Problems

The following set of problems are given as examples having a multivariate nature. Some of them are simple extensions of (familiar?) univariate problems. Other introduce new paradigms. Those shown are but a small subset of a *wider* class of problems.

1.7 Examples 1 to 6

The following are several different types of problems/situations which demonstrate how the multivariate model can arise in practice.

Example 1:

8 features (8 trace elements) are taken on 606 samples.

	Zn	Cu	Pb	Ni	Co	Ag	Mn	Fe
Zn	1.000	.012	.472	.108	.272	.548	.108	.350
Cu	.012	1.000	-.004	.006	.038	.049	.069	.151
Pb	.472	-.004	1.000	-.016	.002	.932	.009	.036
Ni	.108	.006	-.016	1.000	.773	-.015	-.025	.205
Co	.272	.038	.002	.773	1.000	-.009	.117	.594
Ag	.548	.049	.932	-.015	-.009	1.000	.057	.068
Mn	.108	.069	.009	-.025	.117	.057	1.000	.252
Fe	.350	.151	.036	.205	.594	.068	.252	1.000

Table 1.1: Correlation matrix for 606 records with 8 variables

Source: Green, W. 1985, Computer-aided data analysis. Wiley, New York, p.59, Figure 4.5.

The following associations are suggested by the correlation matrix (using a cutoff of $\frac{1}{2}$ on r).

$$\begin{aligned} Zn &\rightarrow Ag,\ Ag \rightarrow Pb \\ Fe &\rightarrow Co,\ Ni \rightarrow Co...? \end{aligned}$$

So in fact two subgroups exist (at least).

Example 2:

Effluent study on Biochemical Oxygen Demand (BOD) and Suspended Solids (SS). Comparison of two groups (labs) on 2 features (BOD and SS). Note that the samples are paired across labs to reduce between sample differences, and so this is really like observations on one group ($\sim$ univariate paired t-test).

> Municipal waste water treatment plants are required by law to monitor their discharges into rivers and streams on a regular basis. Concern about the reliability of data from one of these self-monitoring programs led to a study in which samples of effluent were divided and sent to two laboratories for testing. One-half of each sample was sent to the Wisconsin State Laboratory of Hygiene and one-half was sent to a private commercial laboratory routinely used in the monitoring program. Measurements of biochemical oxygen demand (BOD) and suspended solids (SS) were obtained, for $n = 11$ sample splits, from the two laboratories. The data are displayed in Table 1.2.

	Commercial lab		State lat of hygiene	
Sample j	x_{11j} (BOD)	x_{12j} (SS)	X_{21j} (BOD)	X_{22j}(SS)
1	6	27	25	15
2	6	23	28	13
3	18	64	36	22
4	8	44	35	29
5	11	30	15	31
6	34	75	44	64
7	28	26	42	30
8	71	124	54	64
9	43	54	34	56
10	33	30	29	20
11	20	14	39	21

Table 1.2: Effluent data

Source: Data courtesy of S. Weber.

Example 3: Milk transportation - cost data

This is a MANOVA with 2 treatments (petrol vs diesel) where the response is on 3 measures of cost: fuel, repair and capital.

This is thus the multivariate extension of the 2 sample t–test.

In the first phase of a study of the cost of transporting milk form farms to dairy plants, a survey was taken of firms engaged in milk transportation. Cost data on X_1 = field, X_2 = repair, and X_3 = capital, all measured on a per-mile basis, are presented in Table 1.3 for $n_1 = 36$ gasoline and $n_2 = 23$ diesel trucks.

Gasoline trucks			Diesel trucks		
x_1	x_2	x_3	x_1	x_2	x_3
16.44	12.43	11.23	8.50	12.26	9.11
7.19	2.70	3.92	7.42	5.13	17.15
9.92	1.35	9.75	10.28	3.32	11.23
4.24	5.78	7.78	10.16	14.72	5.99
11.20	5.05	10.67	12.79	4.17	29.28
14.25	5.78	9.88	9.60	12.72	11.00
13.50	10.98	10.60	6.47	8.89	19.00
13.32	14.27	9.45	11.35	9.95	14.53
29.11	15.09	3.28	9.15	2.94	13.68
12.68	7.61	10.23	9.70	5.06	20.84
7.51	5.80	8.13	9.77	17.86	35.18
9.90	3.63	9.13	11.61	11.75	17.00
10.25	5.07	10.17	9.09	13.25	20.66
11.11	6.15	7.61	8.53	10.14	17.45
12.17	14.26	14.39	8.29	6.22	16.38
10.24	2.59	6.09	15.90	12.90	19.09
10.18	6.05	12.14	11.94	5.69	14.77
8.88	2.70	12.23	9.54	16.77	22.66
12.34	7.73	11.68	10.43	17.65	10.66
8.51	14.02	12.01	10.87	21.52	28.47
26.16	17.44	16.89	7.13	13.22	19.44
12.95	8.24	7.18	11.88	12.18	21.20
16.93	13.37	17.59	12.03	9.22	23.09
14.70	10.78	14.58			
10.32	5.16	17.00			
8.98	4.49	4.26			
9.70	11.59	6.83			
12.72	8.63	5.59			
9.49	2.16	6.23			
8.22	7.95	6.72			
13.70	11.22	4.91			
8.21	9.85	8.17			
15.86	11.42	13.06			
9.18	9.18	9.49			
12.49	4.67	11.94			
17.32	6.86	4.44			

Table 1.3: Milk Transportation–Cost Data

Source Data courtesy of M. Keaton.

Example 5: Riding mower owners

Consider two groups in a city - riding-mower owners, and those without riding mowers; that is nonowners. In order to identify the best sales prospects for an intensive sales campaign a riding-mower manufacturer is interested in classifying families as prospective owners or nonowners on the basis of x_1 = income and x_2 = lot size data. Random samples of $n_1 = 12$ current owners and $n_2 = 12$ current nonowners yield the values in Table 1.4.

Riding-mower owners		Nonowners	
x_1 (Income in \$1000s)	x_2 (Lot size in 1000 sq ft)	x_1 (Income in \$1000s)	x_2 (Lot size in 1000 sq $f(t)$
20.0	9.2	25.0	9.8
28.5	8.4	17.6	10.4
21.6	10.8	21.6	8.6
20.5	10.4	14.4	10.2
29.0	11.8	28.0	8.8
36.7	9.6	16.4	8.8
36.0	8.8	19.8	8.0
27.6	11.2	22.0	9.2
23.0	10.0	15.8	8.2
31.0	10.4	11.0	9.4
17.0	11.0	17.0	7.0
27.0	10.0	21.0	7.4

Table 1.4: Riding mower ownership

Example 6: Similarity in language

The meaning of words changes with the course of history. However, the meaning of the numbers 1,2, 3... represents one conspicuous exception. A first comparison of languages might be based on the numerals alone. Table 1.5 gives the first 10 numbers in English, Polish, Hungarian, and 8 other modern European languages. (Only languages that use the Roman alphabet are considered. Certain accent marks, such a cedillas, are omitted.) A cursory examination of the spelling of the numerals in Table 1.5 suggests that the first five languages (English, Norwegian, Danish, Dutch and German) are very much alike. French, Spanish and Italian are in even closer agreement. Hungarian and Finnish seem to stand by themselves, and Polish has some of the characteristics of the languages in each of the large subgroups.

The words for 1 in French, Spanish and Italian all begin with u. For illustrative purposes, we might compare languages by looking at the *first letters* of the numbers. We call the words for the same number in two different languages *concordant* if they have the same first letter and *discordant* if they do not. Using Table 1.5, the table of concordances (frequencies of matching first initials) for the numbers 1–10 is given in Table 1.6. We see that English and Norwegian have the same first letter for 8 of the 10 word pairs. The remaining frequencies were calculated in the same manner. The results in Table 1.6 confirm our initial visual impressions of Table 1.5 that is, English, Norwegian, Danish, Dutch, and German seem to form a group. French, Spanish, Italian, and Polish might be grouped together, while Hungarian and Finnish appear to stand alone.

English	Norwegian	Danish	Dutch	German	French	Spanish	Italian	Polish	Hungarian	Finnish
one	en	en	een	ein	un	uno	uno	jeden	egy	yksi
two	to	to	twee	zwei	deux	dos	due	dwa	ketto	kaksi
three	tre	tre	drie	drei	trois	tres	tre	trzy	harom	kolme
four	fire	fire	vier	vier	quatre	cuatro	quattro	cztery	negy	neua
five	fern	fem	vijf	funf	cinq	cinco	cinque	piec	ot	viisi
six	seks	seks	zes	sechs	six	seix	sei	szesc	hat	kuusi
seven	sju	syv	zeven	sieben	sept	siete	sette	siedem	het	seiseman
eight	atte	otte	acht	acht	huit	ocho	otto	osiem	nyolc	kahdeksan
nine	ni	ni	negen	neun	neuf	nueve	nove	dziewiec	kilenc	ybdeksan
ten	ti	ti	tien	zehn	dix	diez	dieci	dziesiec	tiz	kymmenen

Table 1.5: Numerals in Eleven Languages

	E	N	Da	Du	G	Fr	Sp	I	P	H	Fi
E	10										
N	8	10									
Da	8	9	10								
Du	3	5	4	10							
G	4	6	5	5	10						
Fr	4	4	4	1	3	10					
Sp	4	4	5	1	3	8	10				
I	4	4	5	1	3	9	9	10			
P	3	3	4	0	2	5	7	6	10		
H	1	2	2	2	1	0	0	0	0	10	
Fi	1	1	1	1	1	1	1	1	1	2	10

Table 1.6: Concordant First Letters For Numbers in Eleven Languages

Source: Johnson, R.A. & Wichern, D.W. 1992, 3rd edn. pp.582-584.

1.8 R packages

The following packages (at least!) in R will be used throughout this unit :

1. nlm
2. mva
   (cancor, biplot, prcomp)
3. MASS
   (lda, qda, mvnorm, parcoord, parallel, splom)

Other R packages may be downloaded as needed, eg `qqbeta` from Bilodeau.

You should now attempt Workshop 1.

Chapter 2

Multivariate Normal Distribution

This chapter deals with some of the results, properties, tests and extensions of the Multivariate Normal (MVN) distribution that will be needed in later chapters. The MVN is crucial to many of the methods used later on, so a full appreciation of the properties and tests for the MVN assumption is necessary.

2.1 Definition

(J, p19)
(JW p128)

The p dimensional vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_p]'$ follows the Multivariate Normal distribution if

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}}\, e^{-(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})/2}$$

where $-\infty < x_i < \infty,\ i = 1, 2, \ldots, p$. To denote this distribution the terminology

$$\boldsymbol{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

will be employed, where $\boldsymbol{\mu}$ is the mean vector $(\mu_1, \mu_2, \ldots, \mu_p)'$ and $\boldsymbol{\Sigma}$ is the population variance covariance matrix.

2.2 Properties

(JW p133)
(K p25)

The following is a list of properties of the MVN which are of use and interest in the practical application of multivariate dependence methods.

1. Linear combinations of MVN variables are also Normal.

2. All subsets of MVN variables are also (MV) Normal.

3. The vanishing of covariance implies independent variables.

4. Conditional distributions of MVN variables are also (MV) Normal.

The following quotes are provided with a view to later considering a test for Multivariate Normality in data analysis.

Fang and Zhang, page 43 " ... any marginal distributions of a multivariate normal distribution are still normal distributions. But the converse is not true in general; that is, the fact that each component of a random vector is (marginally) normal does not imply that the vector has a multivariate normal distribution. As a counterexample, assume the density of $\boldsymbol{x} = (X_1, X_2)'$ is

$$f(x_1, x_2) = \frac{1}{2\pi} e^{-\frac{1}{2}(x_1^2 + x_2^2)} \left[1 + x_1 x_2 e^{-\frac{1}{2}(x_1^2 + x_2^2)} \right].$$

It is easily seen that $X_1 \sim N(0,1)$ and $X_2 \sim N(0,1)$, but the joint distribution of X_1 and X_2 is not a binormal distribution."

K page 25 "All marginal and conditional distributions are multivariate normal. ... This property characterises the distribution. It is, of course, not true that normal marginal distributions ... imply multivariate normality;"

2.2.1 Summary

(JW p134, p140 and p144)

Below are a selection of some results on the MVN that will be useful later. Again $\boldsymbol{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

p134 Any linear combination $\boldsymbol{a}'\boldsymbol{x} \sim N_1(\boldsymbol{a}'\boldsymbol{\mu}, \boldsymbol{a}'\boldsymbol{\Sigma}\boldsymbol{a})$, and if $\boldsymbol{a}'\boldsymbol{x} \sim N_1(\boldsymbol{a}'\boldsymbol{\mu}, \boldsymbol{a}'\boldsymbol{\Sigma}\boldsymbol{a})$ for every $\boldsymbol{a}$ then $\boldsymbol{x}$ *must* be $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. See also Fang and Zhang page 44, where the same result is proved using characteristic functions.

p140 If $\boldsymbol{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then

$$(\boldsymbol{x} - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}) \sim \chi^2_p.$$

p144 If A is a symmetric matrix, then $tr(A) = \sum_i \lambda_i$ where λ_i are the eigenvalues of A.

2.3 Some Results

This section contains some results of use for general multivariate distributions and some results specific to the MVN.

2.3.1 General Background

(K, page 20)

Means and moments for a multivariate distribution function can be defined as

$$\boldsymbol{\mu} = E\boldsymbol{x}$$

and

$$\mu_{ijk,\ldots} = Ex_1^i x_2^j x_3^k, \ldots$$

The moment generating function is

$$M_x(\boldsymbol{t}) = Ee^{\boldsymbol{x}'\boldsymbol{t}}$$

while the characteristic function is given by

$$\phi_x(\boldsymbol{t}) = Ee^{i\boldsymbol{x}'\boldsymbol{t}}$$

and the cumulant generating function K is

$$K_x(\boldsymbol{t}) \stackrel{\text{def}}{=} \ln M_x(\boldsymbol{t}).$$

Note that $\boldsymbol{t} = (t_1, t_2, \ldots, t_p)$, and the coefficient of $t_1^i t_2^j t_3^k, \ldots$ in the expansion of $M_x(\boldsymbol{t})$ is $\mu_{ijk,\ldots}/i!j!k!, \ldots$.

The cumulant generating function for a Multivariate Normal distribution is

$$K(\boldsymbol{t}) = \boldsymbol{\mu}'\boldsymbol{t} + \frac{1}{2}\boldsymbol{t}'\boldsymbol{\Sigma}\boldsymbol{t}.$$

These results will be used later.

2.3.2 Specific to MVN

The results below are for a random sample $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ from a $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ population.

1. The sample mean $\overline{\boldsymbol{x}}$ is distributed as $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}/n)$.
2. The statistic $(n-1)\boldsymbol{S}$ is distributed as a Wishart on $n-1$ df.
3. The statistics $\overline{\boldsymbol{x}}$ and $\boldsymbol{S}$ are independently distributed. (Note that $\overline{\boldsymbol{x}}$ and $\boldsymbol{S}$ are MLE for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ and are sufficient statistics.

2.3.3 Central Limit Theorem

For a random sample of size n from *any* population with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ where $n-p$ is 'large'

$$\sqrt{n}(\overline{\boldsymbol{x}} - \boldsymbol{\mu}) \approx N_p(\boldsymbol{0}, \boldsymbol{\Sigma}).$$

A consequence of the CLT is that

$$(\overline{\boldsymbol{x}} - \boldsymbol{\mu})' \boldsymbol{S}^{-1} (\overline{\boldsymbol{x}} - \boldsymbol{\mu}) \approx \chi^2_p$$

for $n - p$ 'large'.

The following abridged extract from K (p24) summaries the conditions under which CLT may hold in practice.

"As with the univariate central limit theorem, the conditions may be relaxed somewhat. The $\boldsymbol{y}$ values need not be independent; ... need not be identically distributed, or have the same dispersion matrix; the distribution of the mean still tends to multivariate normality provided only that as $n \to \infty$ the distribution is not dominated by a few observations with high variances."

And so K (p24) concludes that :

"The practical conclusion is that inference about means based on the assumption of multivariate normality is unlikely to be misleading, provided distributions are not obviously skew or long–tailed, samples are reasonably large, and caution is exercised in making statements involving very small probabilities."

You should now attempt Workshop 2.

2.4 Tests for Multivariate Normality

Several practical options are open to the data analyst wishing to test the assumption of Multivariate Normality. The simplest first course is to examine the marginal distributions of each variate using univariate QQ plots. If any of these plots display non–normality, then multivariate normality is suspect. Alas even if all the univariate plots are satisfactory, there is no guarantee of multivariate normality. However, this is still a worthwhile exercise.

Gross errors or outliers can sometimes be traced by one dimensional screening methods such as histograms and graphics such as the generalized scatter plot, described in Chapter 3.

However, as pointed out by B (p170) most tests for multivariate normality are functions of the Mahalanobis distance. This is the basis of the graphic described in JW p158–163, where a χ^2 plot equivalent to a univariate QQ plot is devised to display conformity to (or lack of) multivariate normality. Briefly, the individual squared distances are ranked and plotted against the expected value based on the use of the χ^2 distribution. Deviation from a straight line is taken as evidence of departure from multivariate normality. Formal tests such as that due to Shapiro and Wilk (B, p169–171) are possible in some computer implementations. The S–plus function `qqbeta` (B, p184–188, and p262) produces a graphic for small sample sizes where the χ^2 approximation may not be valid. The question of how large the sample needs to be before the χ^2 approximation is valid is discussed in B page 186. It would

appear that the number of variates needs to be considered as well as the sample size. The overall opinion is that the beta distribution be employed rather than the χ^2 approximation.

2.4.1 Graphical tests of Multivariate Normality

(J, p67)

Two versions of graphical tests of Multivariate Normality both based on Mahalanobis distance are described together with R output demonstrating the computer implementation of each test.

χ^2 plot (JW p29, p158–161)

The following steps describe the production of the graphical test for Multivariate Normality using the χ^2 approximation to Mahalanobis distance.

1. Rank the distances into
$$d^2_{(1)} < d^2_{(2)} < \ldots < d^2_{(n)}$$
2. Produce pairs
$$(X_j, Y_j) = (d^2_{(j)}, \chi^2_p((j - 1/2)/n)),\ j = 1, \ldots, n$$
where $\chi^2_p((j - \frac{1}{2})/n)$ is the $100((j - \frac{1}{2})/n$ percentile of the χ^2_p distribution.
3. Plot Y versus X. If the data are MVN, the graph should exhibit a straight line.

The R code to implement this graphic is given below, recreating the example from JW p160–161.

```
> dat_read.table("jwp29.txt", header=T)
> d_as.data.frame(dat)
> d.cov_cov(d)
> dist_mahalanobis(d,mean(d),d.cov)
> print(dist)
       1         2         3         4         5         6         7         8
4.3429674 1.1988638 0.5944972 0.8295613 1.8794068 1.0128568 1.0235630 5.3330718
       9        10
0.8116529 0.9735590
> n_length(dist)
> u_((1:n)-0.5)/n
> p_qchisq(u,2)
> sd_sort(dist)
> xy_cbind(sd,p)
> print(xy)
```

```
          sd        p
3  0.5944972 0.1025866
9  0.8116529 0.3250379
4  0.8295613 0.5753641
10 0.9735590 0.8615658
6  1.0128568 1.1956740
7  1.0235630 1.5970154
2  1.1988638 2.0996442
5  1.8794068 2.7725887
1  4.3429674 3.7942400
8  5.3330718 5.9914645
> plot(sd, p)
```

The plot of the ordered Mahalanobis distances against their expected values under the assumption of Multivariate Normality is shown in Figure 2.1.

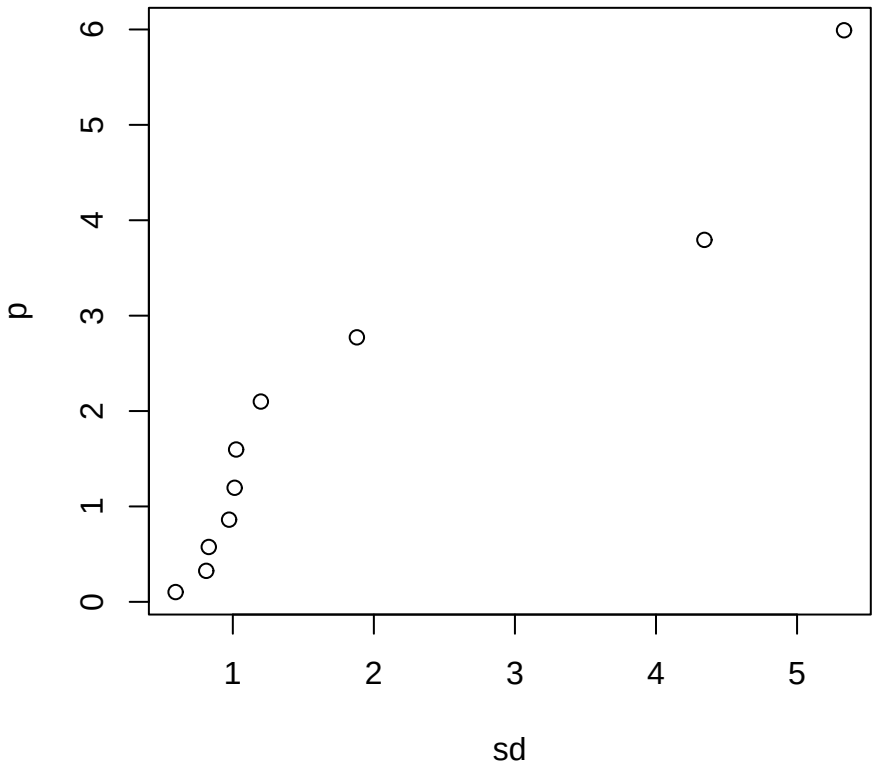

Figure 2.1: A χ^2 plot of ordered Mahalanobis distances.

As stated in JW p160 and B p186, the sampling distribution of the ranked squared distances is only *asymptotically* χ^2, and further the distances are not

independent for small sample sizes. This former restriction leads to the use of the beta distribution.

`qqbeta` (B p169–171, 184–188 and p261–262)

Bilodeau p185 shows that the sampling distribution of the squared distances is in fact a beta distribution, which can be approximated by a χ^2 for large sample sizes. The result assumes independence of distances, but as the correlation is

$$\text{corr}(d_i^2, d_j^2) = -\frac{1}{n-1},\ i \neq j$$

or $O(n^{-1})$ this is claimed to be small for moderate to large n. So the modification is simply to plot

$$(d_{(i)}^2, Ed_{(i)}^2),\ i = 1, \ldots, n$$

where

$$Ed_{(i)}^2 \sim \frac{(n-1)^2}{n}\text{beta}(p/2; (n-p-1)/2)$$

This plot is produced by the routine `qqbeta` which is contained in a file "multivariate"
at the STAT501 web site

Simply download the file `multivariate` and, at the R prompt,

type :

source("multivariate")

to compile the function. The R code to produce the beta plot is :

```
> dat_read.table("jwp29.txt", header=T)
> d_as.data.frame(dat)
> d
    sales profits
1  126974    4224
2   96933    3835
3   86656    3510
4   63438    3758
5   55264    3939
6   50976    1809
7   39069    2946
8   36156     359
9   35209    2480
10  32416    2413
> source("multivariate")
> qqbeta(d)
           x         y
```

```
3  0.5944972 0.2251653
9  0.8116529 0.4676867
4  0.8295613 0.7311691
10 0.9735590 1.0205690
6  1.0128568 1.3430015
7  1.0235630 1.7092784
2  1.1988638 2.1371779
5  1.8794068 2.6595817
1  4.3429674 3.3505890
8  5.3330718 4.4674184
```

Note that the code has been modified to produce the ordered distances and the percentile in the same format as for the χ^2 plot. The differences can be seen in the second column, ie, the percentiles, as the sample size is small.

The resulting plot is shown in Figure 2.2.

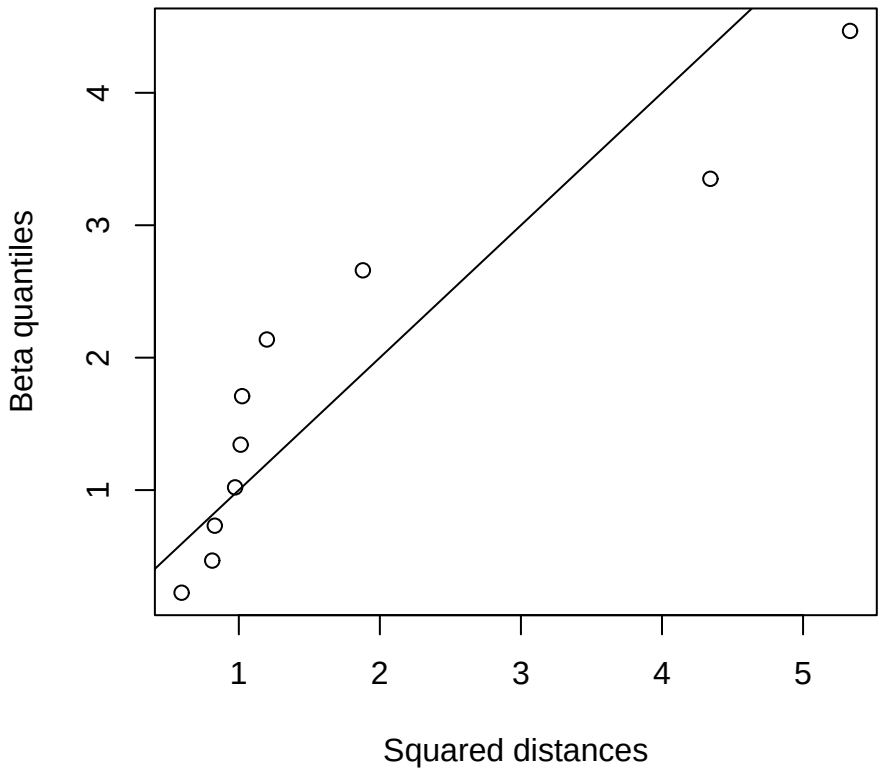

Figure 2.2: Ordered distances against expected using the beta approximation.

The plot should show a straight line if the data are MVN. The assessment of 'large' should include the value of p as well as n, as per the discussion on B p186. The overall suggestion is to use the beta approximation over the χ^2.

2.5 Transformations

(JW p164)
(S p383)

The standard transformations that can be used to attempt to transform data to normality are shown in Table 2.1.

Original	Transformed
Counts, c	$\sqrt{c}$
Proportions, p	$\frac{1}{2}\ln\left(\frac{p}{1-p}\right)$
Correlations, r	$\frac{1}{2}\ln\left(\frac{1+r}{1-r}\right)$

Table 2.1: Transformations to Normality

2.5.1 Power Transformations

For variables not falling under these types, the multivariate equivalent of the Box–Cox power transformation (JW, page 167–168 and K p61–63) is available.

Univariate Case

(JW p165)
(K p60)

The general family of power transformations is defined by $x^{(\lambda)} = x^{\lambda}$ with the proviso that $x^0 = \ln\lambda,\ x > 0$.

Box and Cox modified the definition to

$$x^{(\lambda)} = \begin{cases} \frac{x^{\lambda}-1}{\lambda} & \lambda \neq 0 \\ \ln\lambda & \lambda = 0 \end{cases}$$

Again $x > 0$.

The goal is to find λ such that the transformed data is as Normal as possible. Now if the transformed values are $N(\mu, \sigma^2)$ then the likelihood of the data $x_1, x_2, \ldots, x_n$ will be

$$(2\pi\sigma^2)^{-n/2} e^{-\sum_i (x^{(\lambda)}-\mu)^2/2\sigma^2} \Pi_i x_i^{\lambda-1}$$

For λ fixed this is maximised at

$$\overline{x}^{(\lambda)} = \sum_i x_i^{(\lambda)}/n$$

and

$$s^2(\lambda) = \sum_i (x_i^{(\lambda)} - \overline{x}^{(\lambda)})^2$$

The maximised value of the log–likelihood ℓ is thus proportional to

$$\ell(\lambda) = -n \ln(s^2(\lambda))/2 + (\lambda - 1) \sum_i \ln x_i$$

This function can then be maximised over values of λ numerically to find the 'best' value of λ. An R program is included for the sample problem from JW p166 (data p156), to demonstrate the approach.

```
> y<-c(.15,.09,.18,.1,.05,.12,.08,.05,.08,.10,.07,.02,.01,.1,.1,.1,.02,.1,.01,.4,.1,.05,.
> sumy<-sum(log(y))
> sumy
[1] -100.1325
> lambda<-seq(-1.0,1.5,by=0.1)
> nlam<-length(lambda)
> nlam
[1] 26
> nobs<-length(y)
> nobs
[1] 42
> dd<-c(0,0)
> for (i in 1:nlam) {
+ clam<-lambda[i]
+ sgn<-(abs(clam-0.0)< 1e-6)
+ yb<-sgn*log(y) +(1-sgn)*(y**clam -1 )/(clam+sgn)
+ s2<-var(yb)*(1-1/nobs)
+ llk<--(nobs/2)*log(s2)+(clam-1)*sumy
+ dat<-cbind(clam,llk)
+ dd<-rbind(dd,dat)
+ }
> dd[-1,]
 clam       llk
 -1.0  70.52270
 -0.9  75.64719
 -0.8  80.46258
 -0.7  84.94214
 -0.6  89.05872
 -0.5  92.78554
 -0.4  96.09746
 -0.3  98.97229
 -0.2 101.39233
 -0.1 103.34574
  0.0 104.82762
  0.1 105.84061
```

```
  0.2 106.39479
  0.3 106.50696
  0.4 106.19946
  0.5 105.49859
  0.6 104.43301
  0.7 103.03223
  0.8 101.32540
  0.9  99.34037
  1.0  97.10309
  1.1  94.63730
  1.2  91.96438
  1.3  89.10345
  1.4  86.07142
  1.5  82.88326
>
> # MASS routine
>
> library(MASS)
>
> day <- as.data.frame(y)
> boxcox(y~1,data=day)
$x
 [1] -2.0 -1.9 -1.8 -1.7 -1.6 -1.5 -1.4 -1.3 -1.2 -1.1 -1.0 -0.9 -0.8 -0.7 -0.6
[16] -0.5 -0.4 -0.3 -0.2 -0.1  0.0  0.1  0.2  0.3  0.4  0.5  0.6  0.7  0.8  0.9
[31]  1.0  1.1  1.2  1.3  1.4  1.5  1.6  1.7  1.8  1.9  2.0

$y
 [1] -71.376393 -64.303335 -57.355369 -50.544224 -43.882858 -37.385569
 [7] -31.068112 -24.947807 -19.043641 -13.376337  -7.968364  -2.843872
[13]   1.971514   6.451083  10.567656  14.294479  17.606395  20.481227
[19]  22.901268  24.854676  26.336562  27.349552  27.903725  28.015897
[25]  27.708398  27.007528  25.941946  24.541169  22.834340  20.849307
[31]  18.612028  16.146234  13.473321  10.612386   7.580360   4.392196
[37]   1.061075  -2.401376  -5.984889  -9.680372 -13.479744

> bx <-boxcox(y~1,data=day)
> print(cbind(bx$x,bx$y))
      [,1]       [,2]
 [1,] -2.0 -71.376393
 [2,] -1.9 -64.303335
 [3,] -1.8 -57.355369
 [4,] -1.7 -50.544224
 [5,] -1.6 -43.882858
 [6,] -1.5 -37.385569
 [7,] -1.4 -31.068112
 [8,] -1.3 -24.947807
```

```
 [9,] -1.2 -19.043641
[10,] -1.1 -13.376337
[11,] -1.0  -7.968364
[12,] -0.9  -2.843872
[13,] -0.8   1.971514
[14,] -0.7   6.451083
[15,] -0.6  10.567656
[16,] -0.5  14.294479
[17,] -0.4  17.606395
[18,] -0.3  20.481227
[19,] -0.2  22.901268
[20,] -0.1  24.854676
[21,]  0.0  26.336562
[22,]  0.1  27.349552
[23,]  0.2  27.903725
[24,]  0.3  28.015897
[25,]  0.4  27.708398
[26,]  0.5  27.007528
[27,]  0.6  25.941946
[28,]  0.7  24.541169
[29,]  0.8  22.834340
[30,]  0.9  20.849307
[31,]  1.0  18.612028
[32,]  1.1  16.146234
[33,]  1.2  13.473321
[34,]  1.3  10.612386
[35,]  1.4   7.580360
[36,]  1.5   4.392196
[37,]  1.6   1.061075
[38,]  1.7  -2.401376
[39,]  1.8  -5.984889
[40,]  1.9  -9.680372
[41,]  2.0 -13.479744
> boxcox(y~1,data=day,plotit=T,interp=T)
```

Note the correspondence with the table of $\lambda, \ell(\lambda)$ pairs in JW p166.

The routine `boxcox` from the `MASS` library of Venables and Ripley has been used to verify the R code. Also the plot from `boxcox` shows the optimal value together with the 95% confidence interval, which corroborates the choice of $y^{0.25}$ of JW. This routine could only be used to perform the marginal analysis in the multivariate case.

Exercise

Derive the form of $\ell(\lambda)$.

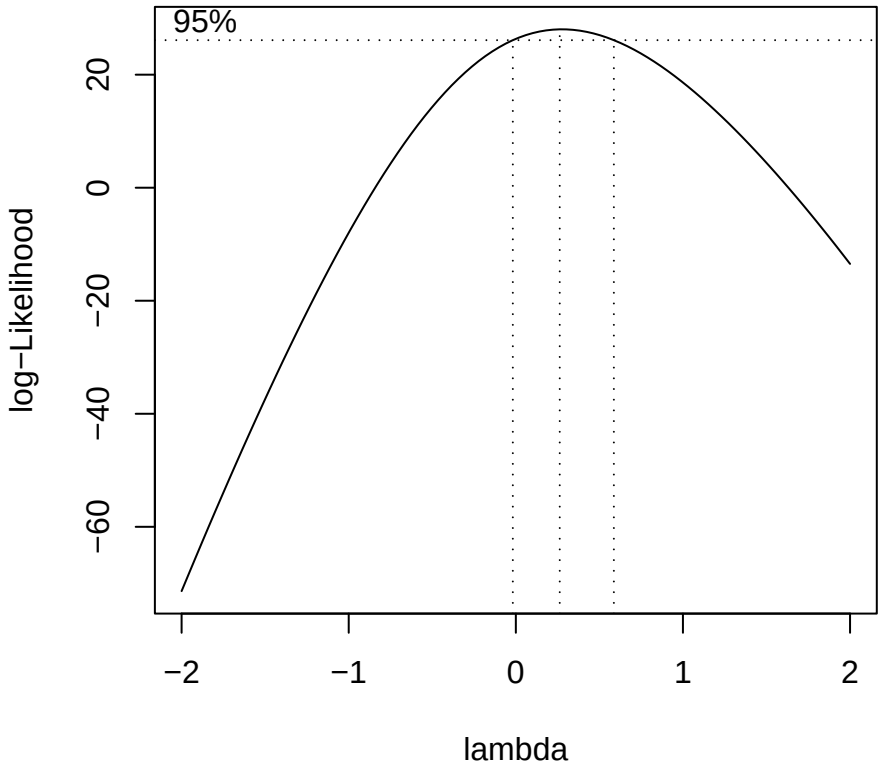

Figure 2.3: Log–likelihood versus λ; Oven data (JW)

Multivariate Case

(K p61–64)
(JW p167–171)

When responses are multivariate, the choice is between a *marginal* or a *joint* analysis.

Marginal A marginal analysis consists of undertaking a power transformation for each individual response, without regard to any correlation between responses. As seen already, the normality of individual responses does not necessarily guarantee joint normality. To allow an assessment of this method with the joint approach, the second response from the data in JW (p169) will now be undertaken. Notice also that for this second response, a minimisation algorithm in R (`nlm`) has been used. The results are :

```
> flk_function(lam)
+ {
+ y_c(.3,.09,.3,.1,.1,.12,.09,.1,.09,.1,.07,.05,.01,
```

```
.45,.12,.2,.04,.1,.01,.6,.12,.1,.05,.05,.15,.3,.15,
.09,.09,.28,.1,.1,.1,.3,.12,.25,.2,.4,.33,.32,.12,.12)
+ sumy_sum(log(y))
+ nobs_length(y)
+ sgn_(abs(lam-0.0)< 1e-6)
+ yb_sgn*log(y) +(1-sgn)*(y**lam -1 )/(lam+sgn)
+ s2_var(yb)*(1-1/nobs)
+ nllk_(nobs/2)*log(s2)-(lam-1)*sumy
+ nllk
+ }
> nlm(flk,0.5)
$minimum
[1] -97.51551

$estimate
[1] 0.2587415

$gradient
[1] 6.237144e-05

$code
[1] 1

$iterations
[1] 4
```

Thus the suggested transformation to Normality for the 'open door' case is

$$\frac{y^{0.2587} - 1}{0.2587}$$

or effectively $y^{1/4}$ as suggested in JW p167,169. Table 2.2 shows the effect of optimizing over λ for each response. A fourth root transformation is probably acceptable for each response. The results for the approximate procedure are shown in parentheses.

Response	λ	ℓ_{max}
Door Closed	0.2757 (0.3)	106.519 (106.507)
Door Open	0.2587 (0.3)	97.516 (97.474)

Table 2.2: Box–Cox Transformations to Normality

Notice that the R routine `nlm` performs function minimisation, hence the change of sign in the variable `nllk` to produce a *negative* log–likelihood.

Joint (K p63)

(JW p167–168)

To transform the responses jointly, the same modification is applied to each response as was used in the marginal case, except that each response now has its own power parameter. The goal is to produce a set of transformed responses that are jointly multivariate Normal. Following an argument similar to the univariate case leads to the log–likelihood being proportional to

$$\ell(\lambda_1, \ldots, \lambda_p) = -(n/2) \ln |\boldsymbol{S}| + \sum_j^p [(\lambda_j - 1) \sum_i^n \ln x_{ij}]$$

where $\boldsymbol{S}$ is the sample covariance matrix of the transformed variables. The following R code implements the joint estimation of $\boldsymbol{\lambda}$ for the bivariate data set from JW p171.

```
> mvflk_function(lam)
+ {
+ y1_c(.15,.09,.18,.1,.05,.12,.08,.05,.08,.10,.07,.02,.01,.1,.1,
.1,.02,.1,.01,.4,.1,.05,.03,.05,.15,.1,.15,.09,.08,.18,.1,.2,.11,
.3,.02,.2,.2,.3,.3,.4,.3,.05)
+ y2_c(.3,.09,.3,.1,.1,.12,.09,.1,.09,.1,.07,.05,.01,.45,.12,.2,
.04,.1,.01,.6,.12,.1,.05,.05,.15,.3,.15,.09,.09,.28,.1,.1,.1,.3,
.12,.25,.2,.4,.33,.32,.12,.12)
+ sumy1_sum(log(y1))
+ sumy2_sum(log(y2))
+ nobs_length(y1)
+ lam1_lam[1]
+ lam2_lam[2]
+ sgn1_(abs(lam1-0.0)< 1e-6)
+ sgn2_(abs(lam2-0.0)< 1e-6)
+ yb1_sgn1*log(y1) +(1-sgn1)*(y1**lam1 -1 )/(lam1+sgn1)
+ yb2_sgn2*log(y2) +(1-sgn2)*(y2**lam2 -1 )/(lam2+sgn2)
+ yb_cbind(yb1,yb2)
+ s2_det(cov(yb))*(1-1/nobs)
+ negllk_(nobs/2)*log(s2)-(lam1-1)*sumy1-(lam2-1)*sumy2
+ negllk
+ }
> nlm(mvflk, c(0.276,0.259))
$minimum
[1] -226.4339

$estimate
[1] 0.1607884 0.1511717

$gradient
[1]  5.741185e-06 -4.519052e-06
```

```
$code
[1] 1

$iterations
[1] 6
```

Table 2.3 shows the results of the optimisation procedure versus the graphical method given in JW p171.

Method	$\boldsymbol{\lambda}$	ℓ_{max}
`nlm`	(0.1607,0.1512)	226.4339
JW p171	(0.2, 0.2)	225.83

Table 2.3: Joint Box–Cox Transformations to Normality

The results of the joint procedure can now evaluated by performing a graphical plot using `qqbeta` to test the effectiveness of the joint power transformations. The quantile plot of the original data is shown in Figure 2.4.

Following the transformation described in Table 2.3 the change in the quantile plot given in Figure 2.5 shows the transformations have induced Normality. Notice that the function `qqbeta` has been constructed so that a line at 45 degrees in the QQ plot is consistent with MVN (B p186).

The R code to produce the two quantile plots was :

```
> y1_c(.15,.09,.18,.1,.05,.12,.08,.05,.08,.10,.07,.02,.01,
.1,.1,.1,.02,.1,.01,.4,.1,.05,.03,.05,.15,.1,.15,.09,.08,
.18,.1,.2,.11,.3,.02,.2,.2,.3,.3,.4,.3,.05)
> y2_c(.3,.09,.3,.1,.1,.12,.09,.1,.09,.1,.07,.05,.01,.45,
.12,.2,.04,.1,.01,.6,.12,.1,.05,.05,.15,.3,.15,.09,.09,.28,
.1,.1,.1,.3,.12,.25,.2,.4,.33,.32,.12,.12)
> y_cbind(y1,y2)
> y
        y1   y2
 [1,] 0.15 0.30
 [2,] 0.09 0.09
 [3,] 0.18 0.30
 [4,] 0.10 0.10
 [5,] 0.05 0.10
 [6,] 0.12 0.12
 [7,] 0.08 0.09
 [8,] 0.05 0.10
 [9,] 0.08 0.09
[10,] 0.10 0.10
[11,] 0.07 0.07
```

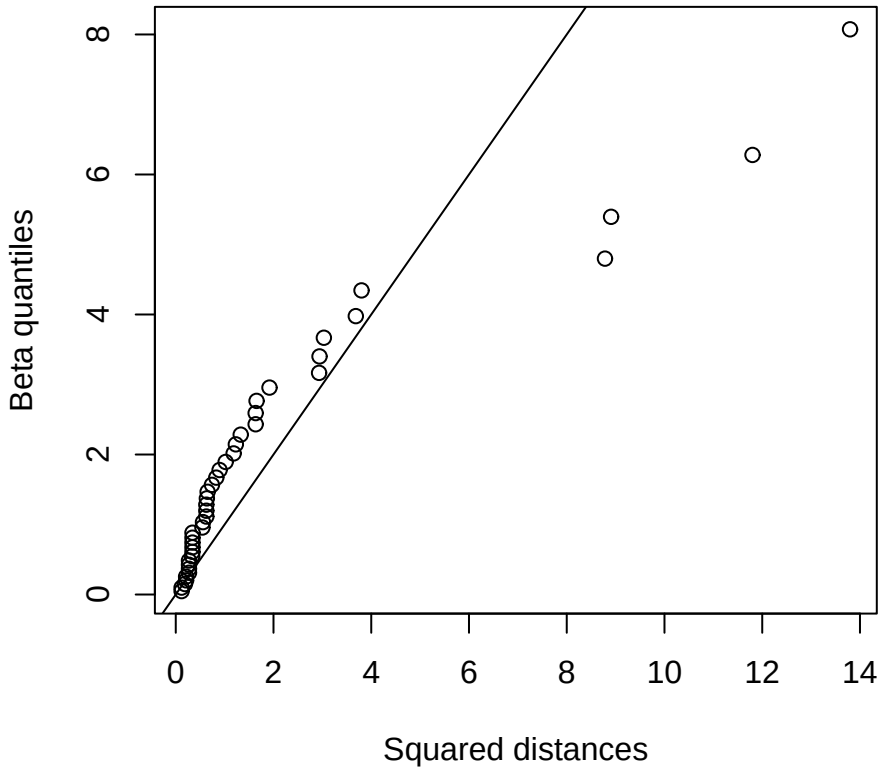

Figure 2.4: Quantile plot of original data

```
[12,] 0.02 0.05
[13,] 0.01 0.01
[14,] 0.10 0.45
[15,] 0.10 0.12
[16,] 0.10 0.20
[17,] 0.02 0.04
[18,] 0.10 0.10
[19,] 0.01 0.01
[20,] 0.40 0.60
[21,] 0.10 0.12
[22,] 0.05 0.10
[23,] 0.03 0.05
[24,] 0.05 0.05
[25,] 0.15 0.15
[26,] 0.10 0.30
[27,] 0.15 0.15
[28,] 0.09 0.09
[29,] 0.08 0.09
```

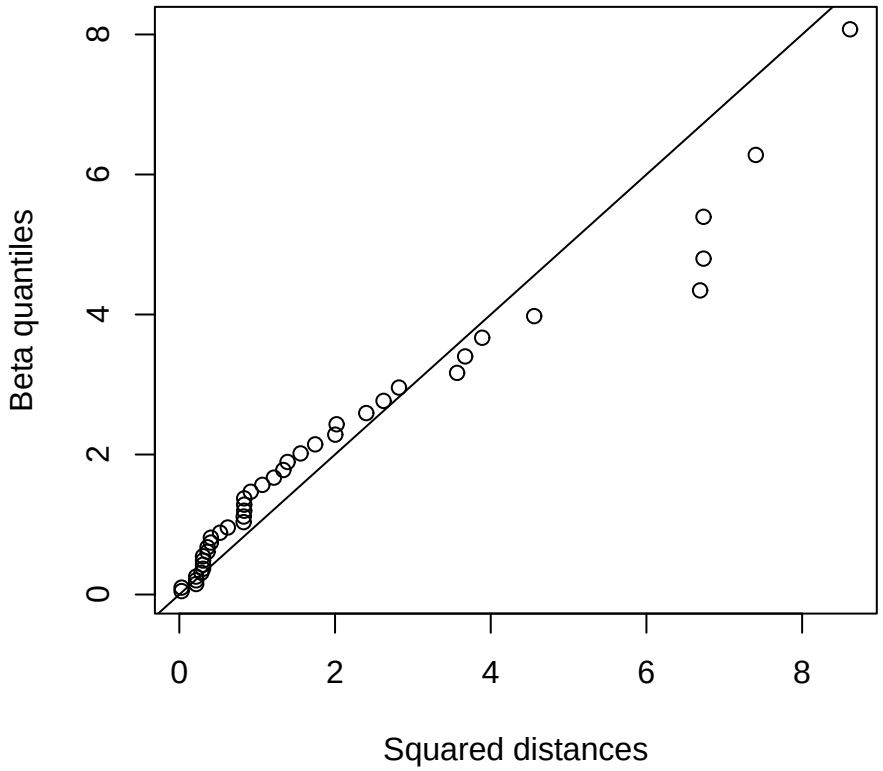

Figure 2.5: Quantile plot of transformed data

```
[30,] 0.18 0.28
[31,] 0.10 0.10
[32,] 0.20 0.10
[33,] 0.11 0.10
[34,] 0.30 0.30
[35,] 0.02 0.12
[36,] 0.20 0.25
[37,] 0.20 0.20
[38,] 0.30 0.40
[39,] 0.30 0.33
[40,] 0.40 0.32
[41,] 0.30 0.12
[42,] 0.05 0.12
> newy1_(y1^{0.1607} -1 )/0.1607
> newy2_(y2^{0.1512} -1 )/0.1512
> newy_cbind(newy1,newy2)
> newy
           newy1      newy2
```

```
 [1,] -1.6352129 -1.1007446
 [2,] -1.9967617 -2.0182894
 [3,] -1.4988131 -1.1007446
 [4,] -1.9246001 -1.9444950
 [5,] -2.3776668 -1.9444950
 [6,] -1.7968045 -1.8139868
 [7,] -2.0759983 -2.0182894
 [8,] -2.3776668 -1.9444950
 [9,] -2.0759983 -2.0182894
[10,] -1.9246001 -1.9444950
[11,] -2.1640340 -2.1896352
[12,] -2.9041392 -2.4090812
[13,] -3.2539536 -3.3172936
[14,] -1.9246001 -0.7521868
[15,] -1.9246001 -1.8139868
[16,] -1.9246001 -1.4285753
[17,] -2.9041392 -2.5485776
[18,] -1.9246001 -1.9444950
[19,] -3.2539536 -3.3172936
[20,] -0.8520226 -0.4915966
[21,] -1.9246001 -1.8139868
[22,] -2.3776668 -1.9444950
[23,] -2.6807023 -2.4090812
[24,] -2.3776668 -2.4090812
[25,] -1.6352129 -1.6492830
[26,] -1.9246001 -1.1007446
[27,] -1.6352129 -1.6492830
[28,] -1.9967617 -2.0182894
[29,] -2.0759983 -2.0182894
[30,] -1.4988131 -1.1579558
[31,] -1.9246001 -1.9444950
[32,] -1.4181487 -1.9444950
[33,] -1.8582610 -1.9444950
[34,] -1.0946633 -1.1007446
[35,] -2.9041392 -1.8139868
[36,] -1.4181487 -1.2506462
[37,] -1.4181487 -1.4285753
[38,] -1.0946633 -0.8556503
[39,] -1.0946633 -1.0207219
[40,] -0.8520226 -1.0466840
[41,] -1.0946633 -1.8139868
[42,] -2.3776668 -1.8139868
> source("multivariate")
> postscript(file="y.ps",onefile=F,horizontal=F,height=8,width=7)
> qqbeta(y)
               x          y
```

```
 [1,]  0.1199894 0.04881182
 [2,]  0.1199894 0.09875182
 [3,]  0.1873365 0.14987490
 [4,]  0.2119385 0.20224012
 [5,]  0.2119385 0.25591108
 [6,]  0.2707763 0.31095649
 [7,]  0.2707763 0.36745065
 [8,]  0.2707763 0.42547413
 [9,]  0.2707763 0.48511445
[10,]  0.3350118 0.54646698
[11,]  0.3418183 0.60963585
[12,]  0.3418183 0.67473506
[13,]  0.3418183 0.74188984
[14,]  0.3424485 0.81123810
[15,]  0.3424485 0.88293229
[16,]  0.5456976 0.95714144
[17,]  0.5573147 1.03405376
[18,]  0.6257666 1.11387960
[19,]  0.6257666 1.19685514
[20,]  0.6257666 1.28324677
[21,]  0.6364931 1.37335652
[22,]  0.6523064 1.46752878
[23,]  0.7404843 1.56615862
[24,]  0.8288197 1.66970229
[25,]  0.8972369 1.77869066
[26,]  1.0208562 1.89374652
[27,]  1.1844913 2.01560720
[28,]  1.2294244 2.14515466
[29,]  1.3293288 2.28345627
[30,]  1.6346824 2.43182099
[31,]  1.6346824 2.59187873
[32,]  1.6537157 2.76569542
[33,]  1.9182482 2.95594485
[34,]  2.9329147 3.16617515
[35,]  2.9416357 3.40123982
[36,]  3.0306589 3.66803399
[37,]  3.6845396 3.97683898
[38,]  3.8013379 4.34400336
[39,]  8.7828361 4.79796957
[40,]  8.9074764 5.39541890
[41,] 11.8008871 6.27910118
[42,] 13.7969703 8.07356122
> qqbeta(newy)
                x          y
 [1,] 0.02928426 0.04881182
 [2,] 0.02928426 0.09875182
```

```
 [3,] 0.21582881 0.14987490
 [4,] 0.21582881 0.20224012
 [5,] 0.21582881 0.25591108
 [6,] 0.28355218 0.31095649
 [7,] 0.30441468 0.36745065
 [8,] 0.30441468 0.42547413
 [9,] 0.30441468 0.48511445
[10,] 0.30441468 0.54646698
[11,] 0.36192536 0.60963585
[12,] 0.36192536 0.67473506
[13,] 0.40495424 0.74188984
[14,] 0.40495424 0.81123810
[15,] 0.52317250 0.88293229
[16,] 0.62255538 0.95714144
[17,] 0.82577651 1.03405376
[18,] 0.82679992 1.11387960
[19,] 0.83330750 1.19685514
[20,] 0.83330750 1.28324677
[21,] 0.83330750 1.37335652
[22,] 0.91694164 1.46752878
[23,] 1.06596475 1.56615862
[24,] 1.21513550 1.66970229
[25,] 1.33627526 1.77869066
[26,] 1.39010692 1.89374652
[27,] 1.55767959 2.01560720
[28,] 1.74496413 2.14515466
[29,] 2.00329713 2.28345627
[30,] 2.02025250 2.43182099
[31,] 2.40115872 2.59187873
[32,] 2.62331537 2.76569542
[33,] 2.82145043 2.95594485
[34,] 3.56772894 3.16617515
[35,] 3.67124674 3.40123982
[36,] 3.89053463 3.66803399
[37,] 4.55823993 3.97683898
[38,] 6.68987098 4.34400336
[39,] 6.73328665 4.79796957
[40,] 6.73328665 5.39541890
[41,] 7.40448798 6.27910118
[42,] 8.61552368 8.07356122
```

Exercise

Derive the form for $\ell(\lambda_1, \ldots, \lambda_p)$.

You should now attempt Workshop 3.

2.6 Robustness

(K p24, p27)
(B p206)

It is important to consider the robustness of statistical methods based on the MVN assumption, ie, how sensitive are inferences to departures from multivariate normality? Robust estimators are procedures that are designed *not* to be sensitive to departures from (multivariate) normality. The benefits of robust methods are summed up by Bilodeau, p206,

> "A robust analysis of data is useful in several ways. It can validate or rebuff data analysis done on classical assumptions of multivariate normality. It also comes into play in the identification of outliers, which is a challenging task for data sets with more than two variables".

The next section describes some of the theory behind the construction of such robust procedures. The basic idea is that these generalized distributions (on which such robust methods are based) contain the MVN as a special case, but they can display features that are departures from normality, eg, having heavy tails.

The motivation for examining generalized distributions is outlined in K page 24 :

> "The practical conclusion is that inference about means based on the assumption of multivariate normality is unlikely to be misleading, provided distributions are not obviously skew or long–tailed, samples are reasonably large, and caution is exercised in making statements involving very small probabilities."

2.7 Spherical Distributions

(K page 26)
(B page 207)

These rotationally invariant distributions are of the form

$$f(\boldsymbol{x}) \propto g[(\boldsymbol{x}-\boldsymbol{\mu})'(\boldsymbol{x}-\boldsymbol{\mu})]$$

ie, the mean is $\boldsymbol{\mu}$ and the covariance matrix is proportional to $\boldsymbol{I}$, the identity matrix.

An example is the spherical *normal*

$$f(\boldsymbol{z}) = \frac{1}{(2\pi)^{p/2}} e^{-\boldsymbol{z}'\boldsymbol{z}/2}$$

a special case which will be used frequently.

2.8 Elliptical Distributions

(B page 207)
(K p26–27)
(Fang and Zhang, Ch II)

These elliptically contoured distributions are of the form

$$f(\boldsymbol{x}) = |\Sigma^{-1}|^{-1/2} g[(\boldsymbol{x}-\boldsymbol{\mu})'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})]$$

ie, the mean is $\boldsymbol{\mu}$ and the covariance matrix is Σ. The rationale for these distributions is given by K p27 :

> "One important applications of distributions of this type is in examining the robustness of procedures based on the normal distribution. This involves, typically, examining the performance of these procedures with elliptical distributions with longer tails than the normal".

Examples of elliptical distributions include :

multivariate t

(K p27–28)

This distribution, due to Cornish, is obtained by dividing each variable in a MVN by the same variable y, where $\nu y^2 \sim \chi^2_\nu$. The centre is then transferred to $\boldsymbol{\mu}$. Thus

$$f(\boldsymbol{x}) = \frac{\Gamma[(\nu+p)/2]}{(\pi\nu)^{p/2}\Gamma(\nu/2)|\Sigma|^{1/2}} \left[1 + (\boldsymbol{x}-\boldsymbol{\mu})'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]^{-(\nu+p)/2}$$

The practical application of this distribution is to provide a method of generating long–tailed distributions for comparison with the Normal. The Hotelling's T^2 is the generalization of the t distribution that is used in inference for the multivariate normal.

power exponential

(B p209)

(K p27)

This distribution has pdf

$$f(\boldsymbol{x}) \propto |\Sigma|^{-1/2} e^{-[(\boldsymbol{x}-\boldsymbol{\mu})'\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})]^\alpha/2}$$

The advantage of this distribution is that it that can generate heavy and light tailed distributions depending on the choice of α, in contrast to many others which cannot generate light tails. The case $\alpha = 1$ generates the multivariate normal distribution.

2.9 Conclusion

Two comments may aid in your decisions about the MVN assumption during data analysis :

Symmetry Many multivariate methods appear to work when the data are symmetric rather than full MVN.

(K p16)

> " . . . many of the standard multivariate techniques work satisfactorily in the presence merely of symmetry of data. Transforming to marginal normality may achieve such symmetry even if it does not produce full normality, so it may still be worthwhile". "

Optimality The following caution should be taken when perusing computer output containing tests based on the assumption of MVN:

(B p238)

> "We conclude this analysis by guarding the practitioner against assuming indiscriminately the normality of the data and using the "optimal" test for normality. If the data came from an elliptical distribution . . . then what was supposed to be an $\alpha = 5\%$ significance level test . . . may be far from 5% as evidenced by Table 2.4".

	$\nu = 5$	$\nu = 6$	$\nu = 7$	$\nu = 8$	$\nu = 30$
$q = 1$	.26	.17	.13	.11	.06
$q = 2$	.37	.22	.17	.14	.06
$q = 3$	.46	.27	.20	.16	.06

Table 2.4: Asymptotic significance level of unadjusted LRT for α=5%.

Note that q is the df and ν is a function of the kurtosis for the Likelihood Ratio Test (LRT) on the sample variance.

Finally it should be pointed out that some multivariate methods originally based on the normality assumption, have been generalized using these elliptical generalizations of the MVN. An account is given in Fang and Zhang (1990).

Chapter 3

Multivariate Graphical Methods

(J, p55)
(K p 43–50)
(F and R p38–53)
(E = Everitt B.S., (1978), *Graphical techniques for multivariate data*, Heinmann, London.)
(E p6–8)
(V and R p335-336, Biplot)

This very short chapter is a brief introduction to multivariate graphical displays, with special emphasis on those methods that are implemented in R. Some specialist plots such as biplots and territorial maps will be omitted until later when their need naturally arises.

All multivariate graphical methods attempt to come to grips with the difficulty of representing efficiently and effectively multi–dimensional data in two dimensional space. A common problem is the confusion induced in the display when large numbers of data points are involved.

3.1 General multivariate plots

Enhanced scatter plot

Up to four dimensions can be displayed on a two dimensional plot by simply using North–South rays to represent the positive and negative of the third variable while the East–West does the same for the fourth. (E p7–8, K p44)

Intensity as the third dimension

On a graphical display a third dimension can be displayed using the intensity of the plotted point to represent the value of the third variable. (E p6)

Generalised scatterplot

This method produces all pairwise scatterplots of the variables aligned into a matrix with shared scales. The R implementation is called `splom` since it produces the conditional scatter plot matrix. (VR p9, JW p609–612)

Star plots

Assume that the data are non–negative with $p > 2$. In two dimensional space, construct circles for each data point with fixed radius and p equally spaced rays from the centre of each circle. The length of each ray represents the value of each response. The ends of the rays on each circle are then joined to form a star. Each star then represents a multivariate observation. It helps to standardise the observations and then use the centre of the circle as the smallest standardised observation.

(JW p593 and p615, K p49)

Glyphs

Each response vector(observation) is replaced by one glyph. A glyph is a circle of fixed radius with rays, corresponding to the characteristics (responses), emanating from it. The position of each ray labels each response, while the ray length shows its value. Thus a star plot is an example of a glyph.

(DG p192, K p46)

Weather vanes

This graphic can show 5 variables. The axes give the first two, while the diameter of the circle at each point shows the third, and the length of the ray from the centre of the circle gives variable four. The fifth variable is given by the direction of the ray.

(Gnanadesikan, R., (1977), *Methods for statistical data analysis of multivariate observations*, Wiley, New York, pp 65–66.)

Profile plots

Profile plots are type of glyphs, since the responses are shown on a line or profile. Thus star plots can be considered profile plots in polar coordinates.

(Chambers, J.M., (1983), *Graphical methods for data analysis*, Wadsworth, Belmont, Calif, p162)

Tree symbols

This is a type of glyph where the responses are assigned to branches of a tree with the proviso that the assignment of responses to branches is not arbitrary but is chosen using a clustering algorithm on the responses.

(Chambers, J.M., (1983), *Graphical methods for data analysis*, Wadsworth, Belmont, Calif, p165)

Andrews curves

The p dimensional vector $(x_1, x_2, \ldots, x_p)$ is represented by

$$f(t) = x_1/sqrt(2) + x_2 \sin t + x_3 \cos t + x_4 \sin 2t + x_5 \cos 2t + \ldots$$

The set of observations appear as a set of curves. This approach has nice properties, viz,

1. The function preserves means.

2. The function preserves Euclidean distances.
3. Variability of the plotted function is almost constant.
4. The function preserves linear relationships.

The data should be standardised and the number of observations plotted on each graph can be limiting due to potential crowding.

(K p48, JW p614–617)

Chernoff faces

In this graphic, the p dimensional observations are shown as a face with the facial characteristics showing the measurements on each response.

(K p49, JW p619, FR p38–53)

Parallel coordinate display

This is an ingenious method of displaying multidimensional data in an ordinary two dimensional display. The basic idea is quite simple. Given multivariate data $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_p$ a series of parallel coordinates in two dimensions is represented as a series of points joined by lines. Each parallel coordinate or level shows one of the multivariate responses. This method has the property that structure in high dimensions maps into structure in two dimensions. An example will be given later.

(Wegman, E.J., (1990), *Hyperdimensional data analysis using parallel coordinates*, Journal of the American Statistical Association, 85, p664–675.)

3.2 Multivariate plots implemented in R

3.2.1 splom

The London deaths data is used as an example to demonstrate the features of `splom`, the R implementation of the matrix scatterplot graphic. The data give the deaths in London over a two–week period in December 1952 with the corresponding measurements of smoke and SO_2 levels. The data are :

```
!
!Deaths in London 1-15 Dec 1952
!
! Col 1=Date Col 2=No. Deaths
! Col 3= Atmospheric Smoke mg/cu. m
! Col 4=Atmospheric SO2 ppm
!
 1 112 0.30 0.09
 2 140 0.49 0.16
 3 143 0.61 0.22
 4 120 0.49 0.14
 5 196 2.64 0.75
```

```
 6 294 3.45 0.86
 7 513 4.46 1.34
 8 518 4.46 1.34
 9 430 1.22 0.47
10 274 1.22 0.47
11 255 0.32 0.22
12 236 0.29 0.23
13 256 0.50 0.26
14 222 0.32 0.16
15 213 0.32 0.16
```

The R code to load the library and produce the plot is :

```
> dat <- read.table("deaths.dat", header=T)
> data <- as.data.frame(dat)
> data
   date deaths smoke  so2
1     1    112  0.30 0.09
2     2    140  0.49 0.16
3     3    143  0.61 0.22
4     4    120  0.49 0.14
5     5    196  2.64 0.75
6     6    294  3.45 0.86
7     7    513  4.46 1.34
8     8    518  4.46 1.34
9     9    430  1.22 0.47
10   10    274  1.22 0.47
11   11    255  0.32 0.22
12   12    236  0.29 0.23
13   13    256  0.50 0.26
14   14    222  0.32 0.16
15   15    213  0.32 0.16
> library(lattice)
> splom(~data)
```

The scatter plot matrix is shown in Figure 3.1.

3.2.2 parallel

The library `lattice` contains the entry `parallel` which is of the same format as `splom`. The following two artificial examples demonstrate the features of the parallel coordinate plot.

Example 1

As a baseline, this example shows the parallel display with data that is totally random. This data contains noise only, and this is reflected in the confusion shown in Figure 3.2.

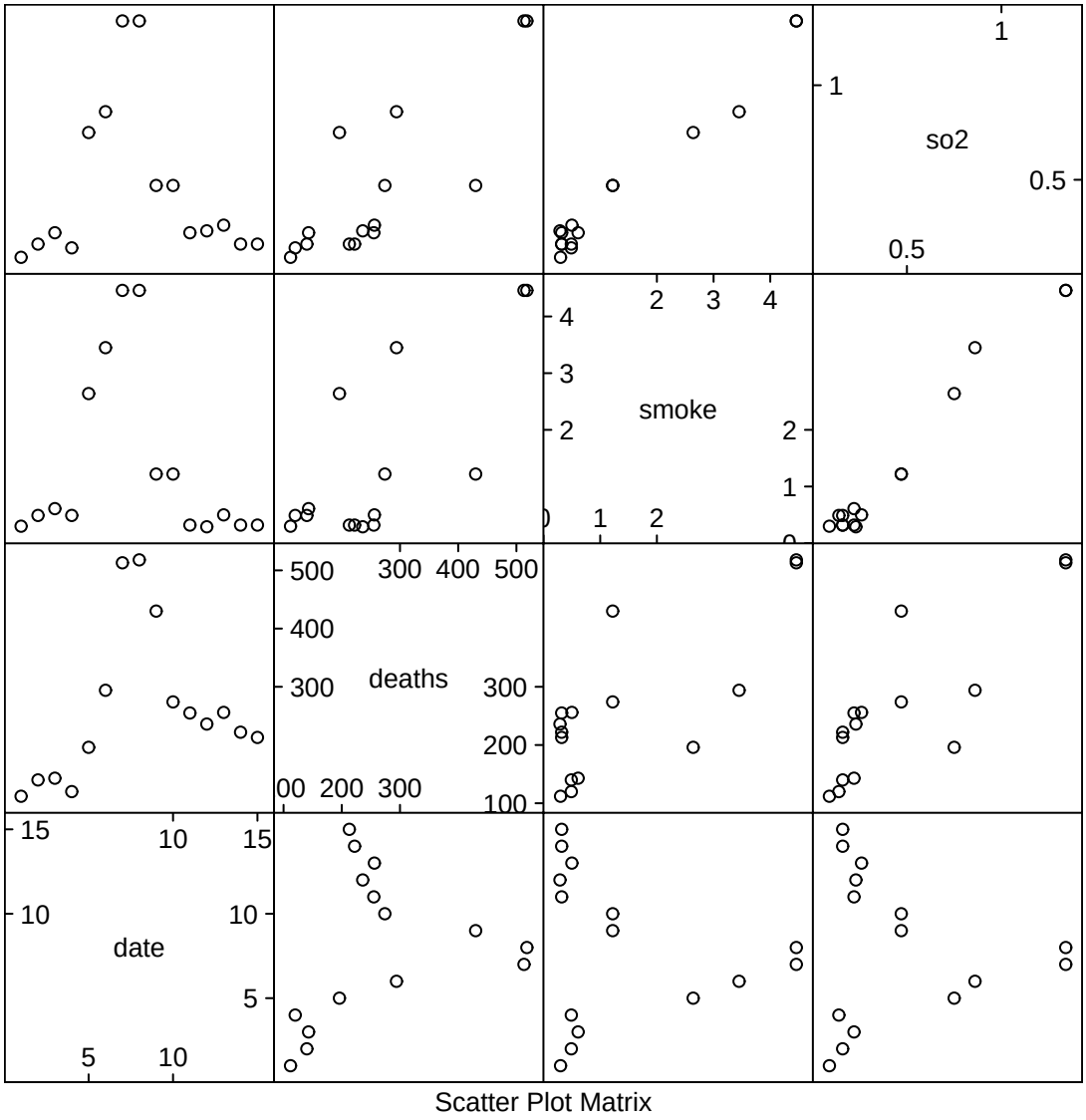

Figure 3.1: Scatter plot matrix of London deaths data

The 4 response vector is

$$x_1, x_2, x_3, x_4 \sim N(0, 1)$$

where all the x variables are unrelated, as clear from the R code.

```
> library(lattice)
> x1 <- rnorm(100)
> x2 <-  rnorm(100)
> x3 <-  rnorm(100)
> x4 <-  rnorm(100)
> y <- cbind(x1,x2,x3,x4)
> parallel(~y)
```

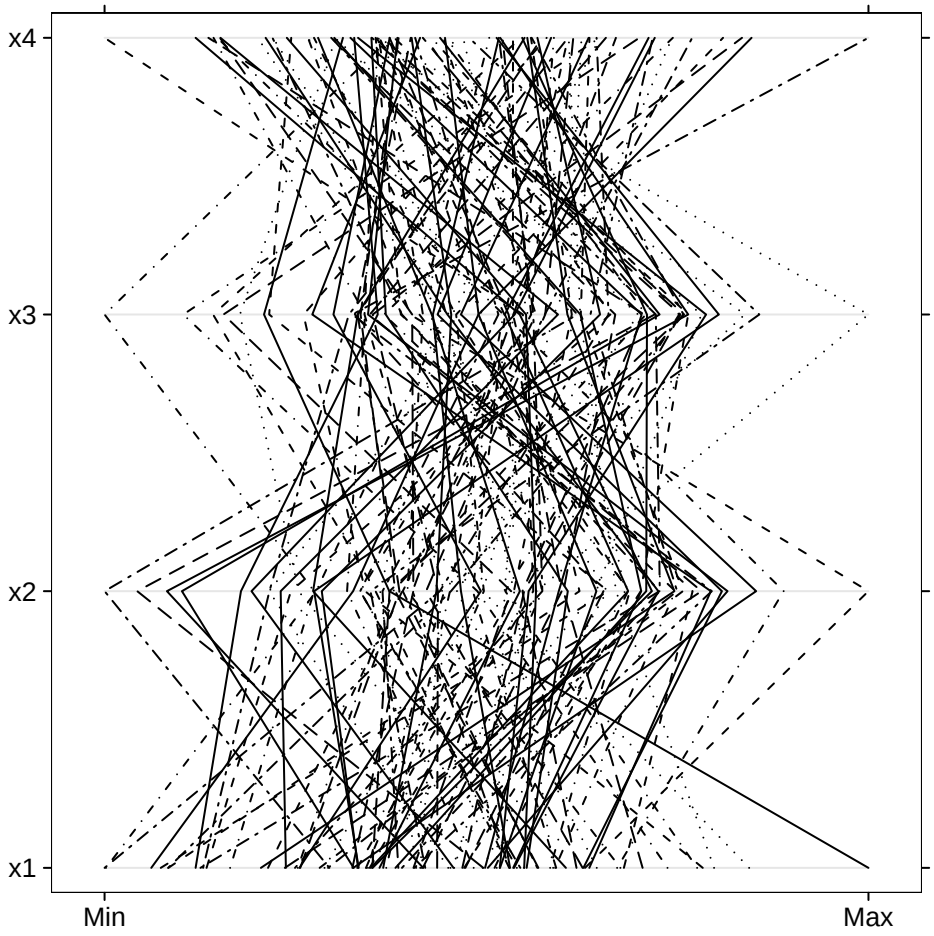

Figure 3.2: Parallel coordinate plot matrix of simulated data : Example 1

Example 2

In this example, a 4 response vector is constructed as follows :

$$x_1 \sim N(0,1),\ x_2 = 2x_1 + N(0,1),\ x_3 = 2x_2.x_1 + N(0,1),\ x_4 = x_3 + N(0,1)$$

The relationship between x_1 and x_2 is clear from the display in Figure 3.3, as is the relation between x_3 and x_4. It must be noted however, that the order of display has been chosen to correspond to the order of derivation, which is unknown in general.

R code :

```
> library(lattice)
> x1 <- rnorm(100)
> x2 <- x1 * 2 + rnorm(100)
> x3 <- 2 * x1 * x2 + rnorm(100)
> x4 <- x3 + rnorm(100)
> y <- cbind(x1,x2,x3,x4)
> parallel(~y)
```

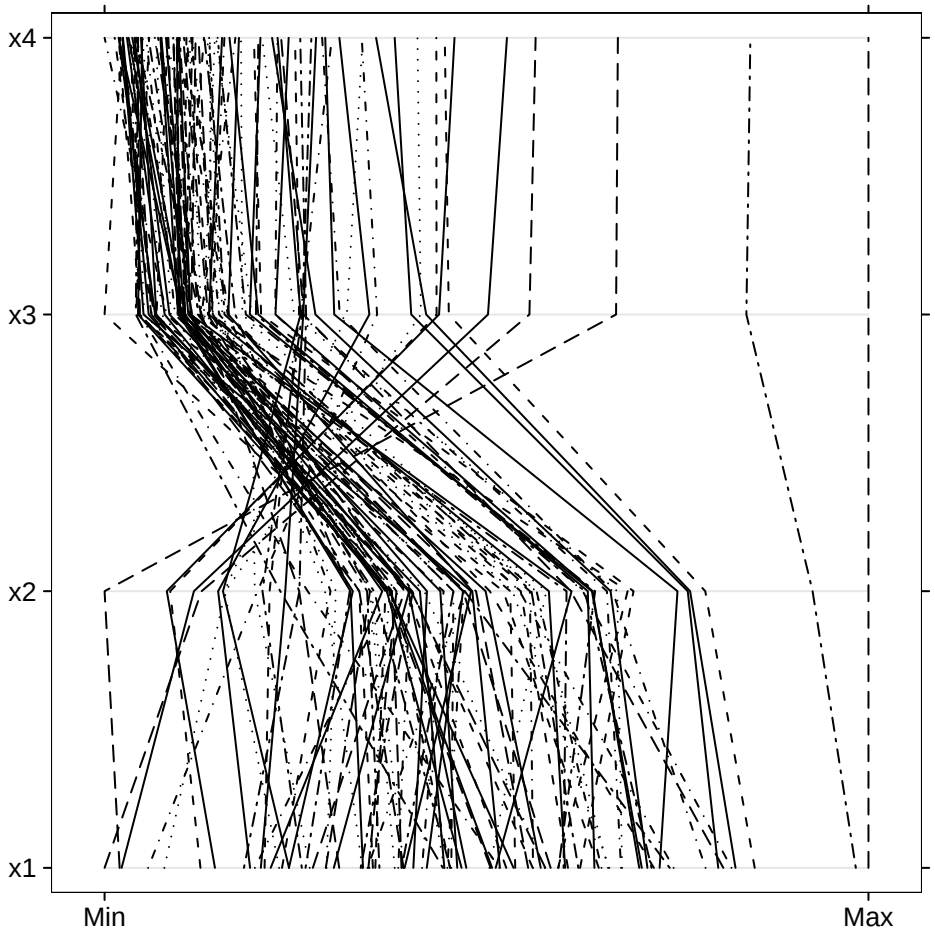

Figure 3.3: Parallel coordinate plot matrix of simulated data : Example 2

Example 3

This second simulation shows the effect of creating two clusters in multivariate data. The same construction is used for the first cluster, while the second is similar but displaced in mean by 5 standardised units. Note the clear distinction into two groups in the parallel coordinate plot in Figure 3.4, a feature when clusters are present. Again the caveat about the order of the variables holds.

R code :

```
> library(lattice)
> x1 <- rnorm(100,mean=10)
> x2 <- x1 * 2 + rnorm(100,mean=10)
> x3 <- 2 * x1 * x2 + rnorm(100,mean=10)
> x4 <- x3 + rnorm(100,mean=10)
> y1 <- cbind(x1,x2,x3,x4)
>
> x11 <- rnorm(100,mean=15)
> x21 <- x11 * 2 + rnorm(100,mean=15)
> x31 <- 2 + x11 + x21 + rnorm(100,mean=15)
```

```
> x41 <- x31 + rnorm(100,mean=13)
> y2 <- cbind(x11,x21,x31,x41)
> y <- rbind(y1,y2)
> parallel(~y)
```

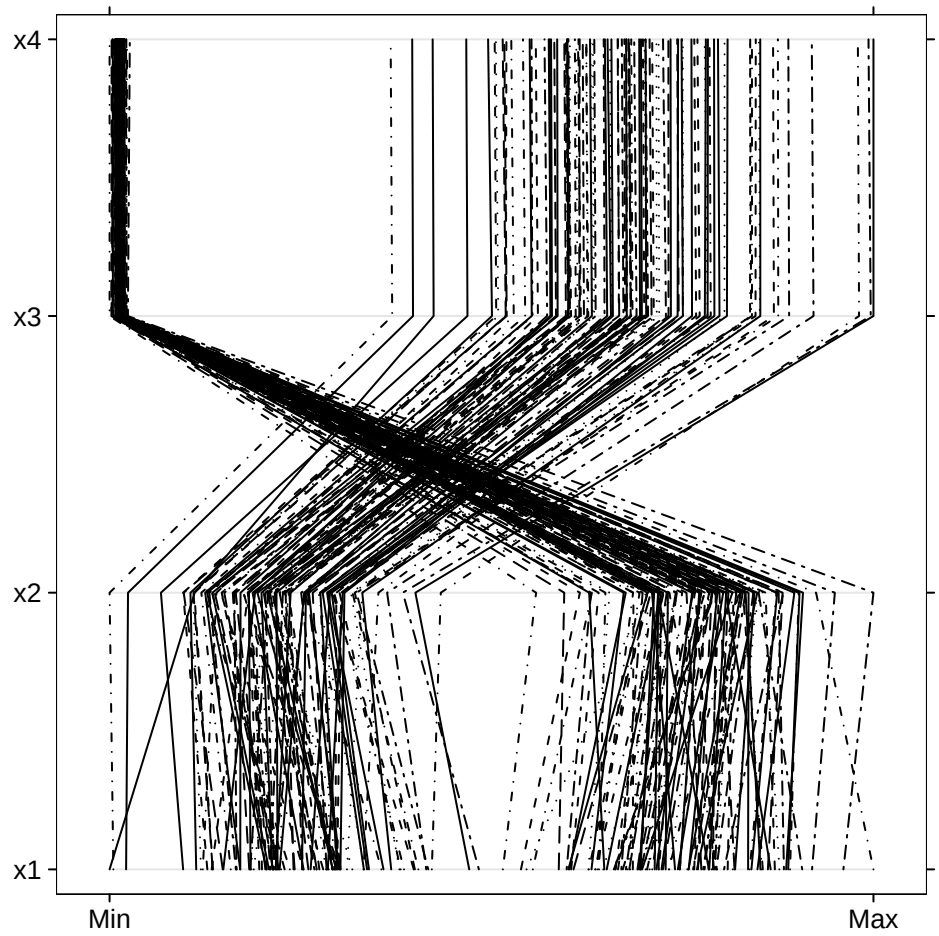

Figure 3.4: Parallel coordinate plot matrix of simulated data : Example 3

3.2.3 parallel

The results of the procedure `parallel` from the `lattice` library on the London deaths data are shown in Figure 3.5.

R code :

```
> dat <- read.table("deaths.dat", header=T)
> data <- as.data.frame(dat)
> data
   date deaths smoke  so2
1     1    112  0.30 0.09
2     2    140  0.49 0.16
```

```
3      3    143  0.61 0.22
4      4    120  0.49 0.14
5      5    196  2.64 0.75
6      6    294  3.45 0.86
7      7    513  4.46 1.34
8      8    518  4.46 1.34
9      9    430  1.22 0.47
10    10    274  1.22 0.47
11    11    255  0.32 0.22
12    12    236  0.29 0.23
13    13    256  0.50 0.26
14    14    222  0.32 0.16
15    15    213  0.32 0.16
> library(lattice)
> parallel(~data)
```

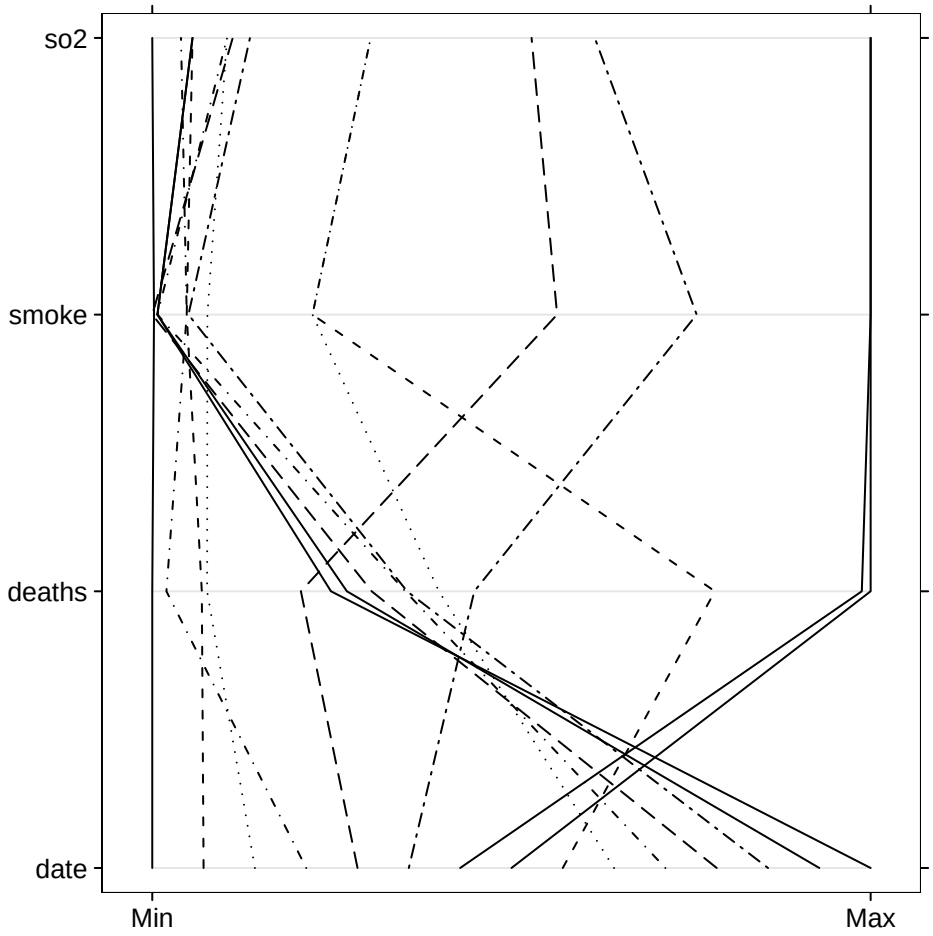

Figure 3.5: Parallel coordinate plot matrix of London data : `parallel`

3.2.4 parcoord

Finally, an alternative plot `parcoord` is provided in the `MASS` library of VR. The results of this procedure on the London deaths data are shown in Figure 3.6 for comparison with `parallel` from the `lattice` library.

R code :

```
> dat <- read.table("deaths.dat", header=T)
> data <- as.data.frame(dat)
> data
   date deaths smoke  so2
1     1    112  0.30 0.09
2     2    140  0.49 0.16
3     3    143  0.61 0.22
4     4    120  0.49 0.14
5     5    196  2.64 0.75
6     6    294  3.45 0.86
7     7    513  4.46 1.34
8     8    518  4.46 1.34
9     9    430  1.22 0.47
10   10    274  1.22 0.47
11   11    255  0.32 0.22
12   12    236  0.29 0.23
13   13    256  0.50 0.26
14   14    222  0.32 0.16
15   15    213  0.32 0.16
> library(MASS)
> parcoord(data)
```

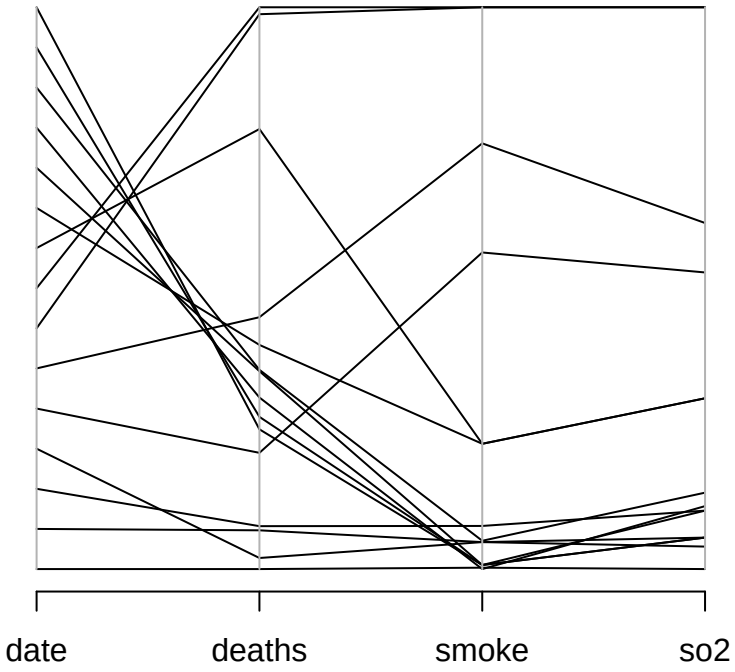

Figure 3.6: Parallel coordinate plot matrix of London data : `parcoord`

Chapter 4

Principal Component Analysis

4.1 Introduction

(J, p93)

Principal Component Analysis (PCA) is a technique for *data reduction* that can be used on responses or predictors. One such use can be to avoid multicollinearity in multiple regression by creating synthetic uncorrelated variables. The main potential problem with the technique can be in the interpretation of components. An application of this method involved the analysis of a growth experiment on *banksia* seedlings. Botanical considerations required measuring the lengths of the top four leaves to cover all types of growth response to stress on the plant. Rather than using four related responses to assess seedling growth, a single measure (first principal component) was used as since it accounted for 87% of the total variation in leaf growth.

4.2 Definition

Given variables $x_1, \ldots, x_p$ find a linear combination

$$y = \boldsymbol{a}'\boldsymbol{x}$$

such that its variance is maximised. The new variables $\boldsymbol{y}$ are called *principal components*, since the usual goal of the process is to replace the high dimensional set of variables $\boldsymbol{x}$ with a smaller efficient subset of the variables $\boldsymbol{y}$.

The determination of $\boldsymbol{y}$ is equivalent to a rotation of axes to define a new set of variables of the same dimensionality. The term *principal* derives from the desire to reduce dimensionality, and so only the principal $\boldsymbol{y}$ variables are retained, ie, those that are needed.

4.3 Derivation

The condition

$$\boldsymbol{a}'\boldsymbol{a} = 1$$

needs to be imposed, else the variance can be made infinitely large by scaling the data, ie, simply letting $\boldsymbol{a} \to \infty$.

So the problem becomes :

maximise (wrt $\boldsymbol{a}$) $V(\boldsymbol{a}'\boldsymbol{x})$ subject to $\boldsymbol{a}'\boldsymbol{a} = 1$.

Note that

$$V(\boldsymbol{a}'\boldsymbol{x}) = \boldsymbol{a}'\Sigma\boldsymbol{a}.$$

To incorporate the equality constraint, use a Lagrange multiplier to make the problem (Jo p4)

$$\text{Max } \boldsymbol{a}'\Sigma\boldsymbol{a} + \lambda(1 - \boldsymbol{a}'\boldsymbol{a}).$$

The reference Jo stands for

> Jollife I.T., (1986), *Principal Component Analysis*, Springer–Verlag, New York.

So

$$S = \boldsymbol{a}'\Sigma\boldsymbol{a} + \lambda(1 - \boldsymbol{a}'\boldsymbol{a})$$

and

$$\frac{\partial S}{\partial \boldsymbol{a}} = \Sigma\boldsymbol{a} + \lambda(-\boldsymbol{a}) = 0$$

if

$$(\Sigma - \lambda I) = 0$$

and so assuming that $\boldsymbol{a} \neq 0$, then the quantities λ are the eigenvalues of Σ, the covariance matrix.

4.3.1 Example

In two dimensions

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}, \quad \sigma_{12} = \rho\sigma_1\sigma_2$$

$$|\Sigma - \lambda I| = 0$$

gives

$$\begin{vmatrix} (\sigma_1^2 - \lambda) & \sigma_{12} \\ \sigma_{12} & (\sigma_1^2 - \lambda) \end{vmatrix} = 0$$

ie

$$(\sigma_1^2 - \lambda)(\sigma_1^2 - \lambda) - \sigma_{12}^2 = 0$$

This becomes

$$\lambda^2 - \lambda(\sigma_1^2 + \sigma_2^2) + \sigma_1^2\sigma_2^2 - \sigma_{12}^2 = 0$$

where

$$\lambda_1 + \lambda_2 = \sigma_1^2 + \sigma_1^2 = \text{trace}(\Sigma)$$

and

$$\lambda_1 \cdot \lambda_2 = \sigma_1^2 + \sigma_2^2 - \sigma_{12}^2 = |\Sigma|$$

correspond to the two forms of generalized variance, the trace being used in PCA.

See Workshop 2 Question 1 where it is shown that

$$|\Sigma| = \sigma_1^2\sigma_2^2 - \rho\sigma_1^2\sigma_2^2 = \sigma_1^2\sigma_2^2(1-\rho^2)$$

and

$$2\lambda = \sigma_1^2 + \sigma_2^2 \pm \sqrt{(\sigma_1^4 + \sigma_2^4 - s\sigma_1^2\sigma_2^2 + 4\rho^2\sigma_1^2\sigma_2^2)}$$

4.4 Rationale of the method

In two dimensions there is an alternative interpretation of PCA via a linear functional relation between the two variables. For a functional relation, both variables are subject to error and so the goal for a *linear* functional relation is to find a linear combination that minimises the variability about the line of the relational.[1] Since both variables are subject to error, minimising the variability about the relation reduces to minimising the residual *perpendicular* to the line of the fitted relation, as shown in Figure 4.2. Thus the *triangle of errors* (Figure 4.1) describes this incorporation of the two sources of variability, giving $E^2 = {X_1}^2 + {X_2}^2$.

[1] The analysis assumes that the variances in both variables are similar.

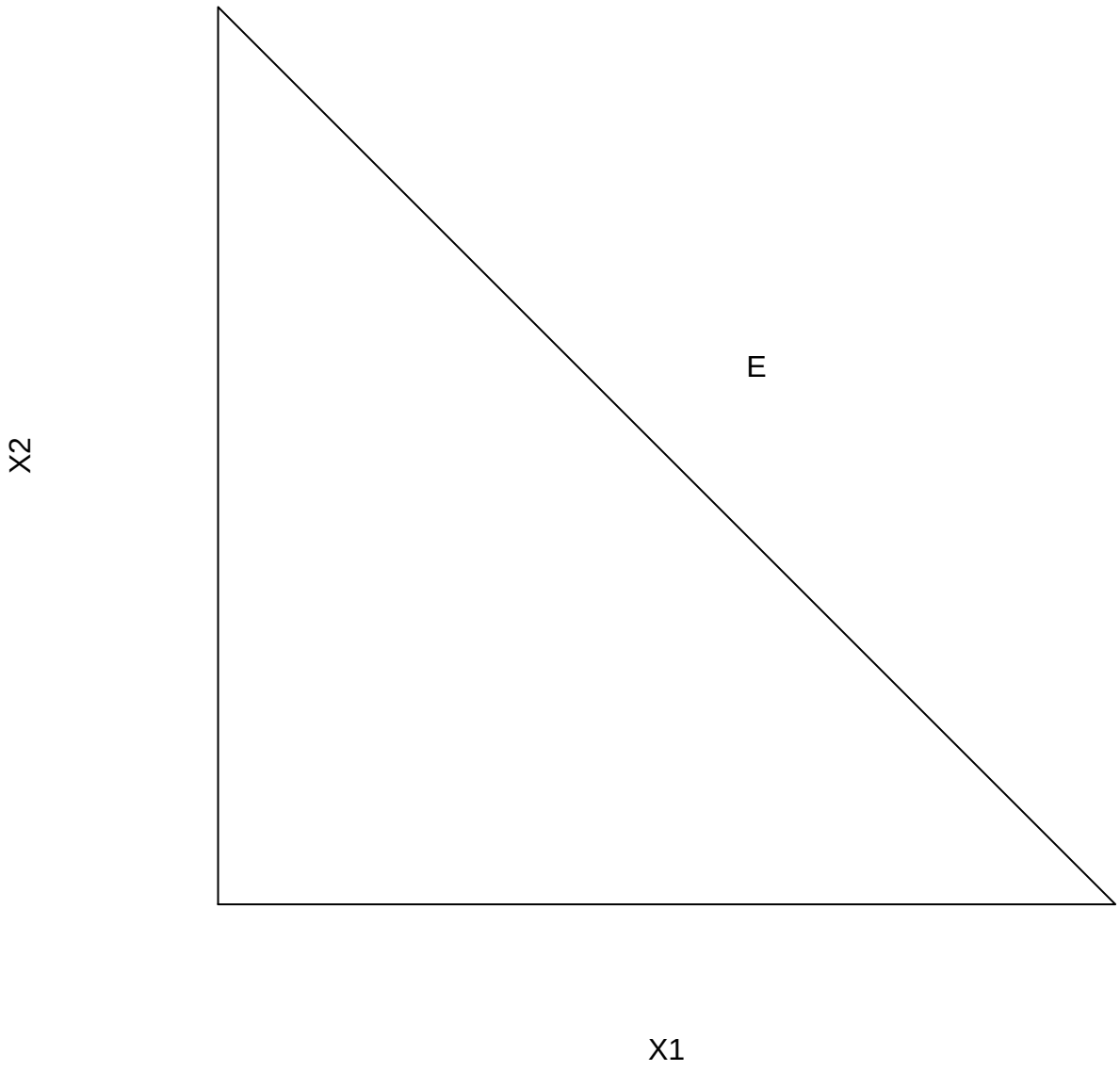

Figure 4.1: Triangle of errors for a functional relation

(For simple linear regression, only the direction X_2 would show any variability, as X_1 would be assumed to be measured without error, to give $E \equiv X_2$.)

So the error structure about the line of the relation is as shown in Figure 4.2.

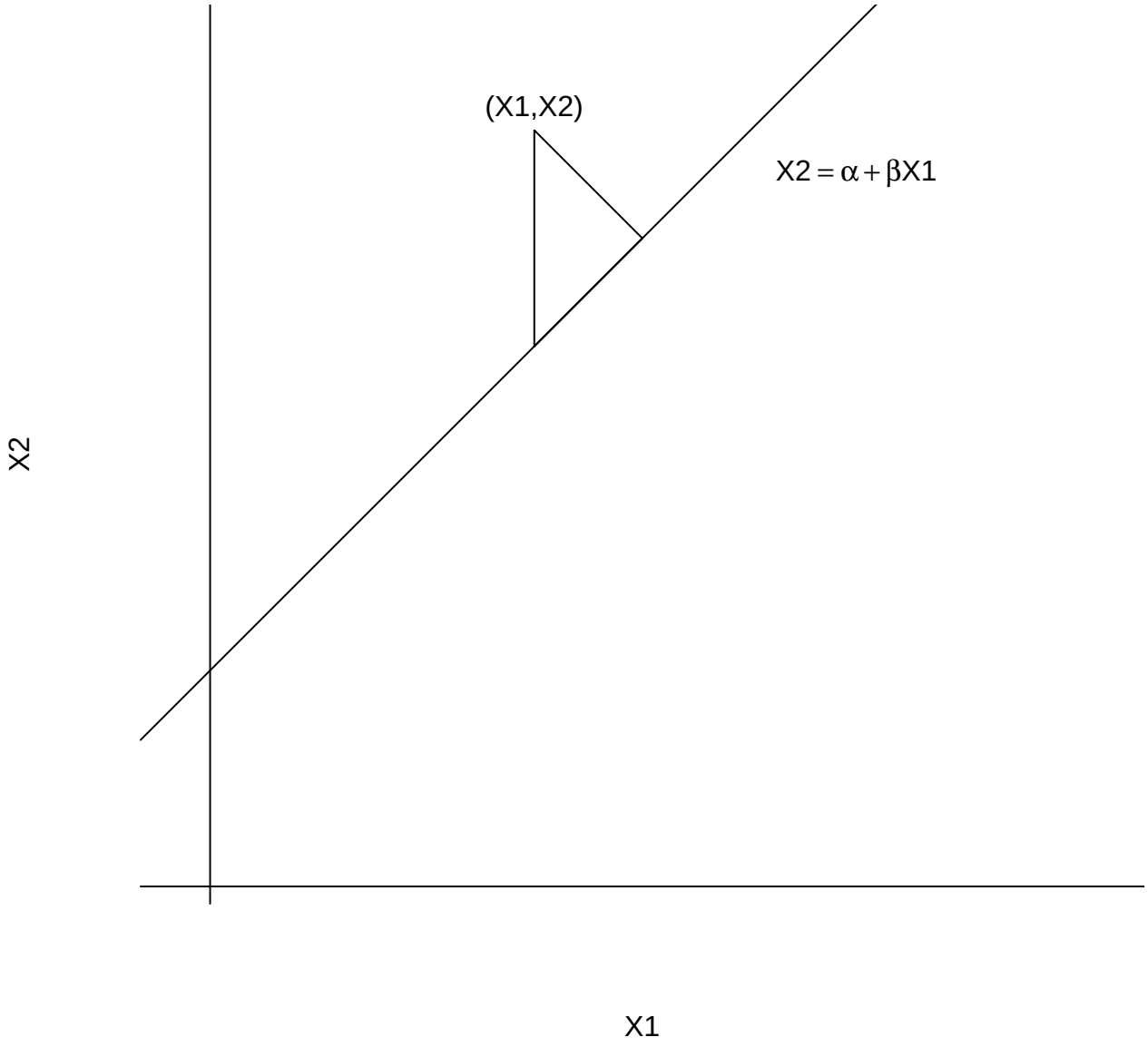

Figure 4.2: Error structure for a functional relation

Consideration of the triangle in Figure 4.2 with X_1, X_2 at the apex, gives the decomposition of errors as shown in Figure 4.3.

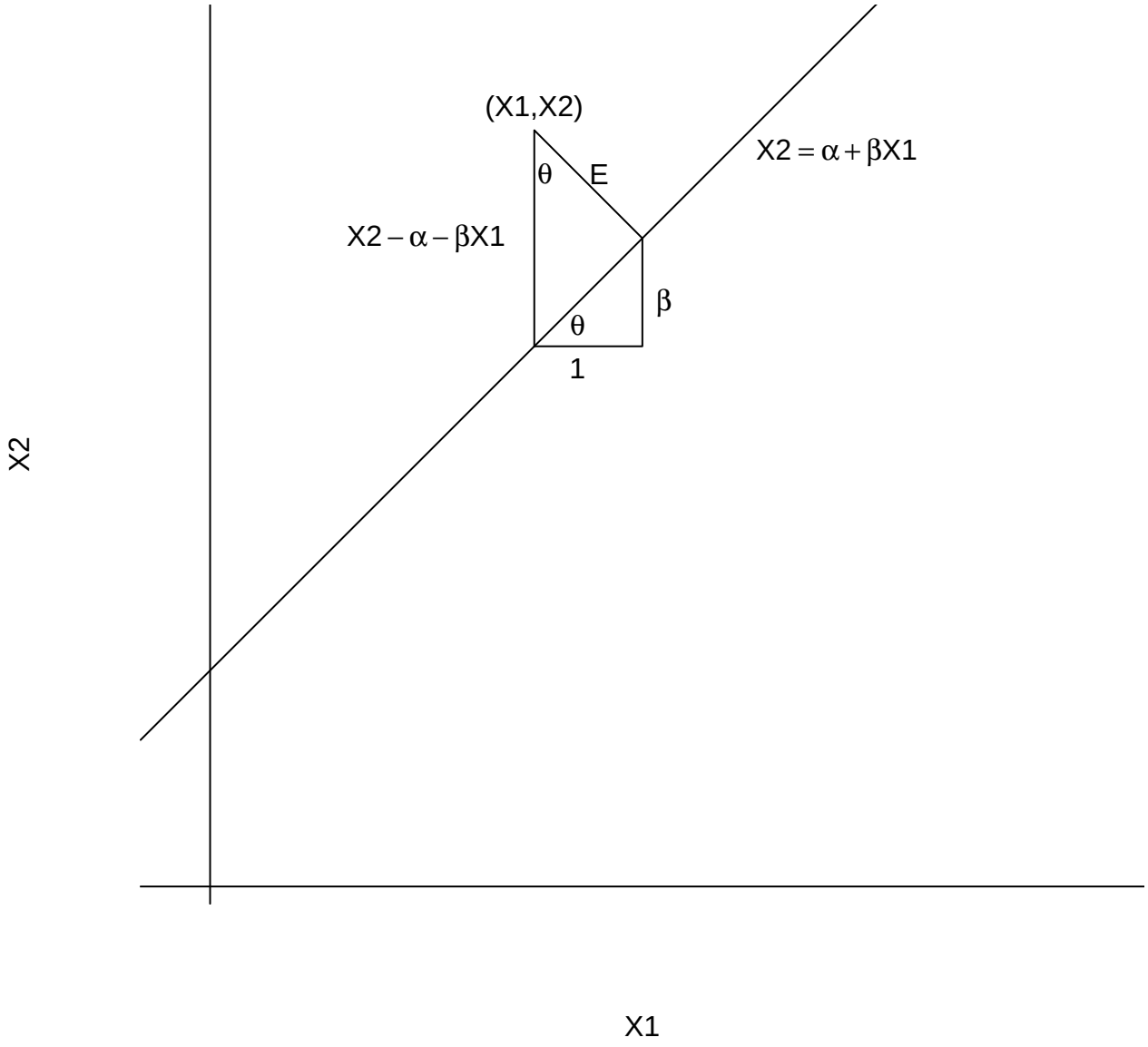

Figure 4.3: Error decomposition for a functional relation

Using similar triangles gives

$$\frac{\sqrt{1+\beta^2}}{1} = \frac{(X_2 - \alpha - \beta X_1)}{E}$$

and so

$$E = \frac{(X_2 - \alpha - \beta X_1)}{\sqrt{1+\beta^2}}.$$

The sum of squares of errors is then

$$S = \sum E^2 = \frac{1}{1+\beta^2} \sum (X_2 - \alpha - \beta X_1)^2.$$

The estimates of α and β are then chosen by solving

$$\frac{\partial S}{\partial \alpha} = 0$$

and

$$\frac{\partial S}{\partial \beta} = 0$$

Now

$$\frac{\partial S}{\partial \alpha} = \frac{2}{1+\beta^2} \sum (X_2 - \alpha - \beta X_1)(-1) = 0$$

gives

$$\widehat{\alpha} = \overline{X}_2 - \widehat{\beta}\overline{X}_1.$$

Meanwhile

$$\frac{\partial S}{\partial \beta} = \frac{-2\beta}{(1+\beta^2)^2} \sum (X_2 - \alpha - \beta X_1)^2 + \frac{1}{(1+\beta^2)} 2 \sum (X_2 - \alpha - \beta X_1)(-X_1) = 0$$

which gives

$$-\beta \sum (X_2 - \alpha - \beta X_1)^2 = (1+\beta^2) \sum (X_2 - \alpha - \beta X_1) X_1$$

$$-\beta \sum \left[(X_2 - \alpha)^2 + \beta^2 X_1^2 - 2(X_2 - \alpha)\beta X_1 \right] = (1+\beta^2) \sum (X_2 X_1 - \alpha X_1 - \beta X_1^2)$$

to become

$$-\beta^3 \sum X_1^2 - \beta \sum (X_2 - \alpha)^2 + 2\beta^2 \sum (X_2 - \alpha) X_1$$
$$= \sum (X_2 - \alpha) X_1 - \sum \beta X_1^2 + \beta^2 \sum (X_2 - \alpha) X_1 - \beta^3 \sum X_1^2$$

ie

$$-\beta \sum (X_2 - \alpha)^2 + \beta^2 \sum (X_2 - \alpha) X_1 = \sum (X_2 - \alpha) X_1 - \beta \sum X_1^2.$$

This reduces to

$$\beta^2 \sum (X_2 - \alpha) X_x + \beta (\sum X_1^2 - \sum (X_2 - \alpha)^2) - \sum (X_2 - \alpha) X_1 = 0$$

ie

$$\beta^2 + \beta \frac{\sum X_1^2 - \sum (X_2 - \alpha)^2}{\sum (X_2 - \alpha) X_1} - 1 = 0.$$

This is a quadratic in β. Therefore two solutions exist. One will give the optimal first combination.

Exercise

Reconcile this version of the 2D problem with the eigenvalue method. (Hint : use $\alpha = \overline{X}_2 - \widehat{\beta}\overline{X}_1$)

4.4.1 An extension

The analysis given in for 2D can be extended to higher dimensions. To introduce this generalisation a quote from the introduction of a recent article is given:

> Principal Component analysis (PCA) is a hugely popular dimensionality reduction technique that attempts to find a low–dimensional subspace passing close to a given set of points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in I\!R^d$. More specifically, in PCA, we find a lower dimensional subspace that minimizes the sum of squared distances from the data points $\boldsymbol{x}_i$ to their projections $\boldsymbol{\theta}_i$ in the subspace, i.e.,
>
> $$\sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{\theta}_i\|^2.$$
>
> This turns out to be equivalent to choosing a subspace that maximises the sum of the squared lengths of the projections $\boldsymbol{\theta}_i$, which is the same as the (empirical) variance of these projections if the data happen to be centered at the origin (so that $\sum_i \boldsymbol{x}_i = \boldsymbol{0}$).

Source :
A Generalization of Principal Component Analysis to the Exponential Family, Michael Collins, Sanjoy Dasgupta and Robert E. Schapire, AT and T Labs, NJ.

Let the data be $\boldsymbol{X}$ so that centering the data on the mean via $\boldsymbol{x} = \boldsymbol{X} - \bar{\boldsymbol{X}}$ places the origin at $\bar{\boldsymbol{X}}$. The situation can be shown via vectors in Figure 4.4 where $\boldsymbol{e} = \boldsymbol{a}'\boldsymbol{x}$, the linear components defined earlier.

Now

$$\|\boldsymbol{x}\|^2 = \|\boldsymbol{x} - \boldsymbol{e}\|^2 + \|\boldsymbol{e}\|^2$$

or

$$\|\boldsymbol{e}\|^2 = \|\boldsymbol{x}\|^2 - \|\boldsymbol{x} - \boldsymbol{e}\|^2$$

where the term $\|\boldsymbol{x}\|^2$ is fixed. Thus

$$\max\|\boldsymbol{e}\|^2 \equiv \min\|\boldsymbol{x} - \boldsymbol{e}\|^2$$

as quoted. The second form is the one most easily recognised by statisticians and users of multivariate methods.

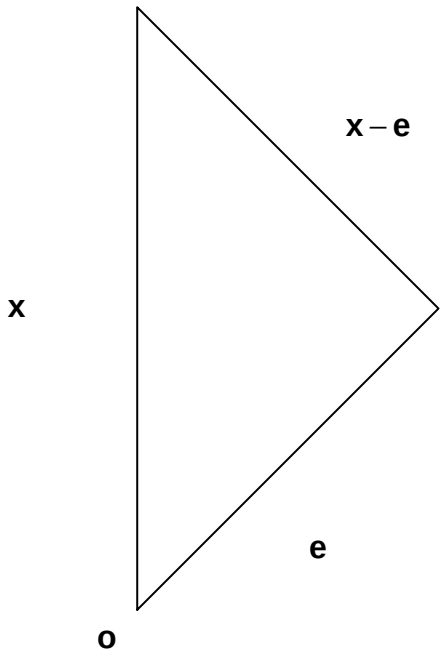

Figure 4.4: Generalisation of PCA triangle

4.5 Correlation Matrix

In line with the comment made earlier about equal variability, the use of PCA on a covariance matrix assumes that variables are measured on similar scales. Thus a

single variable can dominate the (covariance) PCA solution simply by increasing its scale of measure arbitrarily. When the scales are unequal, PCA should be performed on the correlation matrix $\mathcal{R}$. The data ($\boldsymbol{X}$) are usually standardised to $\boldsymbol{Z}$ so that

$$E(\boldsymbol{Z}) = 0, \;\; V(\boldsymbol{Z}) = \mathcal{R} = E\boldsymbol{Z}\boldsymbol{Z}'.$$

So now the problem to be solved is

$$max\, V(\boldsymbol{Z}^*) \;\; s.t. \;\; \boldsymbol{a}'\boldsymbol{a} = 1$$

where

$$\boldsymbol{Z}^* = \boldsymbol{a}'\boldsymbol{Z}$$

This problem can be cast as maximising the functional

$$S = V(\boldsymbol{Z}^*) + \lambda(1 - \boldsymbol{a}'\boldsymbol{a})$$

wrt to $\boldsymbol{a}$. Thus

$$S = \boldsymbol{a}'V(\boldsymbol{Z})\boldsymbol{a} + \lambda(1 - \boldsymbol{a}'\boldsymbol{a}) = \boldsymbol{a}'\mathcal{R}\boldsymbol{a} + \lambda(1 - \boldsymbol{a}'\boldsymbol{a})$$

Maximising S wrt $\boldsymbol{a}$ gives

$$\frac{\partial S}{\boldsymbol{a}} = \mathcal{R}\boldsymbol{a} - \lambda\boldsymbol{a} = 0$$

ie, if

$$(\mathcal{R} - \lambda I)\boldsymbol{a} = 0$$

which is an eigenvalue problem, as for the covariance matrix.

Example

In two dimensions,

$$\mathcal{R} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

giving

$$|\mathcal{R} - \lambda I| = 0$$

ie

$$\begin{vmatrix} 1-\lambda & \rho \\ \rho & 1-\lambda \end{vmatrix}$$

which becomes

$$(1 - \lambda)^2 = \rho^2$$

or

$$1 - \lambda = \pm\rho \implies \lambda = 1 \pm \rho$$

and so

$$\lambda = 1 + \rho, \; 1 - \rho$$

Finally we can note that the generalized variance forms give

$$\lambda_1 + \lambda_2 = 2 = tr(\mathcal{R})$$

and

$$\lambda_1\lambda_2 = 1 - \rho^2 = |\mathcal{R}|.$$

4.6 Eigenvectors

To determine the new directions (principal components) produce the eigenvectors from

$$\Sigma \boldsymbol{e} = \lambda \boldsymbol{e} \ \rightarrow \ |\Sigma - \lambda I| = 0$$

for the covariance matrix, or

$$\mathcal{R} \boldsymbol{e} = \lambda \boldsymbol{e} \ \rightarrow \ |\mathcal{R} - \lambda I| = 0$$

for the correlation matrix.

These directions are associated with decreasing shares of the total variance as given by the trace.

4.7 Calculations in R

The are two routines in the R library *mva*, `prcomp` and `princomp`. To access simply type :

```
library(mva)
help(prcomp)
```

The routine `prcomp` is preferred over `princomp` as it uses the singular value decomposition rather than direct calculations of eigenvalues from the input matrix, and so is more stable numerically.

4.7.1 Example

The data for this example are the deaths in London from Dec 1 to 15 1952, together with the levels of smoke and sulphur dioxide over the same period.

date	deaths	smoke	so2
01	112	0.30	0.09
02	140	0.49	0.16
03	143	0.61	0.22
04	120	0.49	0.14
05	196	2.64	0.75
06	294	3.45	0.86
07	513	4.46	1.34
08	518	4.46	1.34
09	430	1.22	0.47
10	274	1.22	0.47
11	255	0.32	0.22
12	236	0.29	0.23
13	256	0.50	0.26
14	222	0.32	0.16
15	213	0.32	0.16

The predictors `smoke` and `so2` are highly collinear, and if used in a multiple regression model to explain deaths, severe multicollinearity is induced.[2] Could PCA be used to create a synthetic variable ("pollution"?) that uses the information from both predictors, but avoids the pitfalls of collinearity, such as inflated standard errors, regression coefficients being of the wrong sign etc? A PCA on the *covariance* matrix should produce one component aligned with the 'sliver' connecting `smoke` and `so2`. Note that the data is first centered on the means. The first component, given by

$$C1 = 0.965(smoke - 1.406) + 0.261(so2 - 0.458)$$

explains 99.8% of the combined variance, which is not surprising given the data plot (Figure4.5) and the corresponding correlation ($r_{\text{smoke,so2}} = 0.987$). Is $C1$ appropriate? From Figure 4.5, the ratio of `smoke` to `so2` is approximately 3/1 which agrees with the ratio of the leading coefficients.

A similar results holds for $C2$.

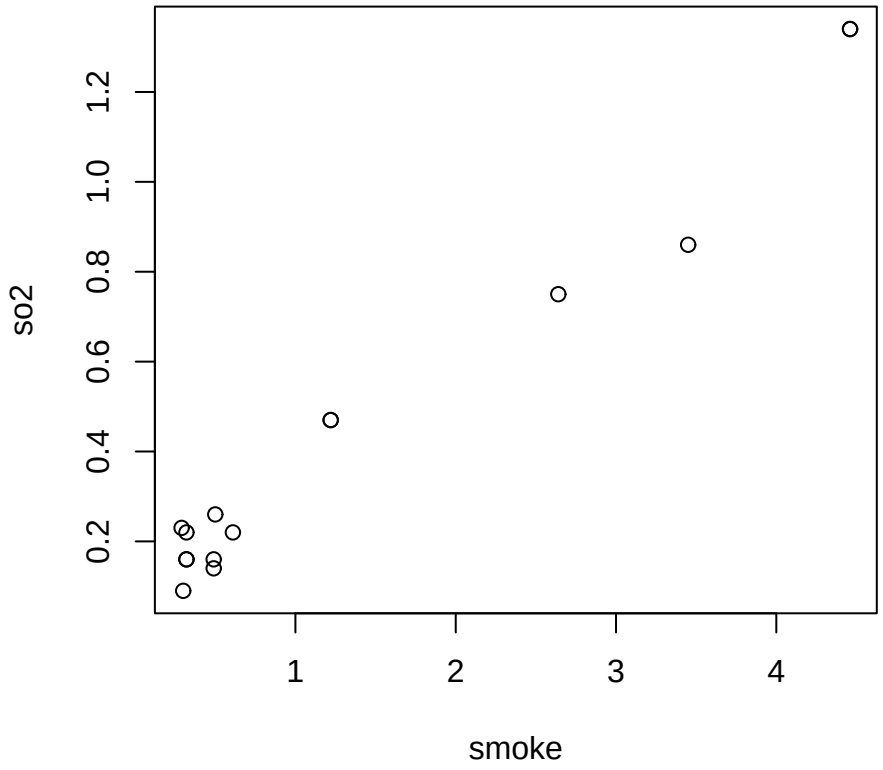

Figure 4.5: Plot of so2 versus smoke : London data

R Output : London data example

[2]This should be verified.

```
> dat <- read.table("deaths.dat", header=T)
> dat
   date deaths smoke  so2
1     1    112  0.30 0.09
2     2    140  0.49 0.16
3     3    143  0.61 0.22
4     4    120  0.49 0.14
5     5    196  2.64 0.75
6     6    294  3.45 0.86
7     7    513  4.46 1.34
8     8    518  4.46 1.34
9     9    430  1.22 0.47
10   10    274  1.22 0.47
11   11    255  0.32 0.22
12   12    236  0.29 0.23
13   13    256  0.50 0.26
14   14    222  0.32 0.16
15   15    213  0.32 0.16
> data <- as.data.frame(dat[,c(3,4)])
> data
   smoke  so2
1   0.30 0.09
2   0.49 0.16
3   0.61 0.22
4   0.49 0.14
5   2.64 0.75
6   3.45 0.86
7   4.46 1.34
8   4.46 1.34
9   1.22 0.47
10  1.22 0.47
11  0.32 0.22
12  0.29 0.23
13  0.50 0.26
14  0.32 0.16
15  0.32 0.16
> mean(data)
smoke   so2
1.406 0.458
> sd(data)
    smoke       so2
1.5483438 0.4234585
> var(data)
          smoke       so2
smoke 2.3973686 0.6475057
so2   0.6475057 0.1793171
```

```
> plot(data)
> cor(data)
          smoke       so2
smoke 1.0000000 0.9875641
so2   0.9875641 1.0000000
> library(mva)
> prcomp(data,center=T)
Standard deviations:
[1] 1.60391874 0.06426804

Rotation:
            PC1        PC2
smoke 0.9652938 -0.2611663
so2   0.2611663  0.9652938
> o1 <- prcomp(data,retx=T,center=T)
> o1$sdev
[1] 1.60391874 0.06426804
> o1$rotation
            PC1        PC2
smoke 0.9652938 -0.2611663
so2   0.2611663  0.9652938
> # scores are in o1$x
> o1$x

            PC1          PC2
  1  -1.1637242 -0.066378151
  2  -0.9620367 -0.048429188
  3  -0.8305315 -0.021851521
  4  -0.9672600 -0.067735065
  5   1.2674331 -0.040413471
  6   2.0780494 -0.145775887
  7   3.1783560  0.053787140
  8   3.1783560  0.053787140
  9  -0.1764107  0.060160465
  10 -0.1764107  0.060160465
  11 -1.1104667  0.053886718
  12 -1.1368138  0.071374646
  13 -0.9262671  0.045488529
  14 -1.1261366 -0.004030911
  15 -1.1261366 -0.004030911
> summary(o1)
Importance of components:
                          PC1    PC2
Standard deviation     1.604 0.0643
Proportion of Variance 0.998 0.0016
Cumulative Proportion  0.998 1.0000
```

```
> cor(data,o1$x)
            PC1          PC2
smoke 0.9999412 -0.01084039
so2   0.9892104  0.14650203
> cor(o1$x)
              PC1           PC2
PC1 1.000000e+00 6.391582e-16
PC2 6.391582e-16 1.000000e+00
```

Checking the trace

The trace is defined by

$$tr(S) = s_1^2 + s_2^2 = \lambda_1 + \lambda_2.$$

Now

$$s_1^2 + s_2^2 = 1.548344^2 + 0.423459^2 = 2.397369 + 0.179318 = 2.576687$$

while

$$\lambda_1 + \lambda_2 = 1.60391874^2 + 0.06426804^2 = 2.572555 + 0.00413 = 2.576685.$$

Scores

The scores are found from the new rotated variables, so

$$C1 = 0.965294(smoke - 1.406) + 0.261166(so2 - 0.458)$$

and for unit 1 this gives

$$0.965294(0.30 - 1.406) + 0.261166(0.09 - 0.458) = -1.067615 - 0.096109 = -1.163724$$

as expected.

Note that $C1$ is orthogonal to $C2$ as shown by the R output (`cor(o1$x)`).

Correlation of new variables with the original variables

As expected, the new synthetic variable ($C1$) is highly correlated with the original variables, as shown in Table 4.1.

	$C1$
smoke	0.999412
so2	0.9892104

Table 4.1: Correlations between $C1$ and original variables : London data.

Alas $C2$ is not so.

4.8 Correlation Input

If the variables are measured on vastly different scales, standardised variables should be used, viz,

$$Z_i = \frac{X_i - m_i}{s_i}$$

where X_i represents the ith original variable, and its mean is m_i and its standard deviation is s_i. This means that the PCA will now be performed on the correlation matrix rather than the covariance matrix. To implement this procedure in R, use `prcomp` with the `scale = T`.

This can give vastly different results to a PCA on the covariance matrix, as per JW p363–365.

Trace

$Trace(R) = 2 = 1.40981^2 + 0.1115163^2 = 1.987564 + 0.012436 = 2.000000 = \lambda_1 + \lambda_2$.

Scores

For unit 1 :

$$C1 = 0.7071068\left(\frac{0.30 - 1.406}{1.548344}\right) + 0.7071068\left(\frac{0.09 - 0.458}{0.423459}\right) = -0.505095 - 0.614499 = -1.119594.$$

R Output : Correlation matrix input PCA

```
> dat <- read.table("deaths.dat", header=T)
> dat
   date deaths smoke  so2
1     1    112  0.30 0.09
2     2    140  0.49 0.16
3     3    143  0.61 0.22
4     4    120  0.49 0.14
5     5    196  2.64 0.75
6     6    294  3.45 0.86
7     7    513  4.46 1.34
8     8    518  4.46 1.34
9     9    430  1.22 0.47
10   10    274  1.22 0.47
11   11    255  0.32 0.22
12   12    236  0.29 0.23
13   13    256  0.50 0.26
14   14    222  0.32 0.16
15   15    213  0.32 0.16
> data <- as.data.frame(dat[,c(3,4)])
> data
   smoke  so2
```

```
1   0.30 0.09
2   0.49 0.16
3   0.61 0.22
4   0.49 0.14
5   2.64 0.75
6   3.45 0.86
7   4.46 1.34
8   4.46 1.34
9   1.22 0.47
10  1.22 0.47
11  0.32 0.22
12  0.29 0.23
13  0.50 0.26
14  0.32 0.16
15  0.32 0.16
> library(mva)
> prcomp(data,center=T,scale=T)
Standard deviations:
[1] 1.4098100 0.1115163

Rotation:
            PC1        PC2
smoke 0.7071068 -0.7071068
so2   0.7071068  0.7071068
> o1 <- prcomp(data,retx=T,center=T,scale=T)
> o1$sdev
[1] 1.4098100 0.1115163
> o1$rotation
            PC1        PC2
smoke 0.7071068 -0.7071068
so2   0.7071068  0.7071068
> o1$x

             PC1          PC2
  1  -1.11959464 -0.109405440
  2  -0.91593573 -0.079287166
  3  -0.76094319 -0.033899249
  4  -0.94933247 -0.112683907
  5   1.05114282 -0.075957971
  6   1.60474047 -0.262191469
  7   2.86751502  0.078077569
  8   2.86751502  0.078077569
  9  -0.06490553  0.104981621
  10 -0.06490553  0.104981621
  11 -0.89338210  0.098539660
  12 -0.89038431  0.128938608
```

```
  13 -0.74438516  0.083129682
  14 -0.99357233 -0.001650564
  15 -0.99357233 -0.001650564
> summary(o1)
Importance of components:
                          PC1     PC2
Standard deviation     1.410 0.11152
Proportion of Variance 0.994 0.00622
Cumulative Proportion  0.994 1.00000
> par(pty="s")
> plot(data)
> cor(data,o1$x)
            PC1         PC2
smoke 0.9968862 -0.07885396
so2   0.9968862  0.07885396
> cor(o1$x)
              PC1           PC2
PC1  1.000000e+00 -1.528639e-15
PC2 -1.528639e-15  1.000000e+00
```

4.9 Example

JW p369, 392.

This example is chosen as a typical case where a small number of components is used to describe a larger set of variables. The first two components account for 93% of the overall variation, as measured by the trace. Notice the agreement between the `prcomp` output from R and the reported values in JW p370. Only the correlations between the original variables and the first two principal components are given, since all the remaining correlations are low except for `value` in component 3. Note also the use of `cor` to verify that the components are uncorrelated with each other (off diagonals $\sim 10^{-15}$). The question of how to decide when a component coefficient is significant will be addressed later, eg, to assess `value` in component 3.

R Output

```
> dat <- read.table("census.dat", header=T)
> dat
   tract  popn school employ health value
1      1 5.935   14.2  2.265   2.27  2.91
2      2 1.523   13.1  0.597   0.75  2.62
3      3 2.599   12.7  1.237   1.11  1.72
4      4 4.009   15.2  1.649   0.81  3.02
5      5 4.687   14.7  2.312   2.50  2.22
6      6 8.044   15.6  3.641   4.51  2.36
7      7 2.766   13.3  1.244   1.03  1.97
8      8 6.538   17.0  2.618   2.39  1.85
```

```
9      9 6.451   12.9  3.147   5.52  2.01
10    10 3.314   12.2  1.606   2.18  1.82
11    11 3.777   13.0  2.119   2.83  1.80
12    12 1.530   13.8  0.798   0.84  4.25
13    13 2.768   13.6  1.336   1.75  2.64
14    14 6.585   14.9  2.763   1.91  3.17
> data <- as.data.frame(dat[,-1])
> data
    popn school employ health value
1  5.935   14.2  2.265   2.27  2.91
2  1.523   13.1  0.597   0.75  2.62
3  2.599   12.7  1.237   1.11  1.72
4  4.009   15.2  1.649   0.81  3.02
5  4.687   14.7  2.312   2.50  2.22
6  8.044   15.6  3.641   4.51  2.36
7  2.766   13.3  1.244   1.03  1.97
8  6.538   17.0  2.618   2.39  1.85
9  6.451   12.9  3.147   5.52  2.01
10 3.314   12.2  1.606   2.18  1.82
11 3.777   13.0  2.119   2.83  1.80
12 1.530   13.8  0.798   0.84  4.25
13 2.768   13.6  1.336   1.75  2.64
14 6.585   14.9  2.763   1.91  3.17
> library(mva)
> prcomp(data,center=T)
Standard deviations:
[1] 2.6326932 1.3360929 0.6242194 0.4790918 0.1189747

Rotation:
               PC1         PC2          PC3         PC4          PC5
popn   -0.78120807  0.07087183  0.003656607  0.54171007  0.302039670
school -0.30564856  0.76387277 -0.161817438 -0.54479937  0.009279632
employ -0.33444840 -0.08290788  0.014841008  0.05101636 -0.937255367
health -0.42600795 -0.57945799  0.220453468 -0.63601254  0.172145212
value   0.05435431  0.26235528  0.961759720  0.05127599 -0.024583093
> o1 <- prcomp(data,center=T,retx=T)
> o1$sdev
[1] 2.6326932 1.3360929 0.6242194 0.4790918 0.1189747
> o1$rotation
               PC1         PC2          PC3         PC4          PC5
popn   -0.78120807  0.07087183  0.003656607  0.54171007  0.302039670
school -0.30564856  0.76387277 -0.161817438 -0.54479937  0.009279632
employ -0.33444840 -0.08290788  0.014841008  0.05101636 -0.937255367
health -0.42600795 -0.57945799  0.220453468 -0.63601254  0.172145212
value   0.05435431  0.26235528  0.961759720  0.05127599 -0.024583093
> o1$x
```

```
           PC1         PC2         PC3         PC4          PC5
  1  -1.4376565  0.29260180  0.44050065  0.74853291  0.201197601
  2   3.5348762  0.08263869 -0.03638751 -0.17543886  0.167201310
  3   2.4002270 -0.64443800 -0.74444828  0.38289849 -0.027262233
  4   0.5952725  1.84591442  0.04643003  0.06319255 -0.047938015
  5  -0.7667385  0.26789299 -0.25718390 -0.37918649 -0.158603360
  6  -4.9574860 -0.04487018  0.20693717 -0.25439106 -0.039346660
  7   2.1317042 -0.06291364 -0.61802055  0.21054156  0.002267993
  8  -2.9913377  2.09728323 -0.99815525 -0.56291943  0.125174851
  9  -3.1718449 -2.85634591  0.51672992 -0.33189825  0.099924038
  10  1.4206830 -1.60007856 -0.32338757  0.38604008  0.019946152
  11  0.3649005 -1.38059323 -0.31947552 -0.18724994 -0.201211729
  12  3.2984865  0.97666914  1.44085808 -0.51641343 -0.037154372
  13  1.7373697 -0.08266930  0.13791241 -0.37069545  0.026902349
  14 -2.1584560  1.10890855  0.50769034  0.98698731 -0.131097925
> plot(o1)
> summary(o1)
Importance of components:
                          PC1   PC2    PC3    PC4     PC5
Standard deviation     2.633 1.336 0.6242 0.4791 0.11897
Proportion of Variance 0.741 0.191 0.0417 0.0245 0.00151
Cumulative Proportion  0.741 0.932 0.9739 0.9985 1.00000
> cor(data,o1$x)
              PC1         PC2          PC3         PC4           PC5
popn   -0.9909495  0.04562416  0.001099766  0.12504610  0.0173142311
school -0.6052660  0.76768201 -0.075977724 -0.19632652  0.0008304416
employ -0.9840179 -0.12379586  0.010353193  0.02731504 -0.1246195660
health -0.7991766 -0.55167514  0.098057095 -0.21712467  0.0145940026
value   0.2014908  0.49356853  0.845327227  0.03459025 -0.0041182450
> cor(o1$x)
              PC1           PC2           PC3           PC4           PC5
PC1  1.000000e+00 -4.818329e-17  7.119101e-17  1.303429e-15 -1.624428e-15
PC2 -4.818329e-17  1.000000e+00 -1.284979e-17  9.775148e-16 -2.175510e-15
PC3  7.119101e-17 -1.284979e-17  1.000000e+00 -9.802486e-17 -3.544149e-16
PC4  1.303429e-15  9.775148e-16 -9.802486e-17  1.000000e+00 -2.061601e-16
PC5 -1.624428e-15 -2.175510e-15 -3.544149e-16 -2.061601e-16  1.000000e+00
```

Exercise

Calculate the PCs for the correlation matrix and comment.

4.10 Choosing the Number of Components

K p82–83, S p76–79.

Procedures for choosing the number of components divide into methods for standardised (correlation matrix) or unstandardised (covariance matrix) data.

4.10.1 Scree Plot

This method can be used on both types of input data, ie, for both unstandardised and standardised data. The method uses a plot of λ_i versus i. The (empirical) rule is to judge the change in slope, and to discard components in the 'flat' section, as shown in Figure 4.6.

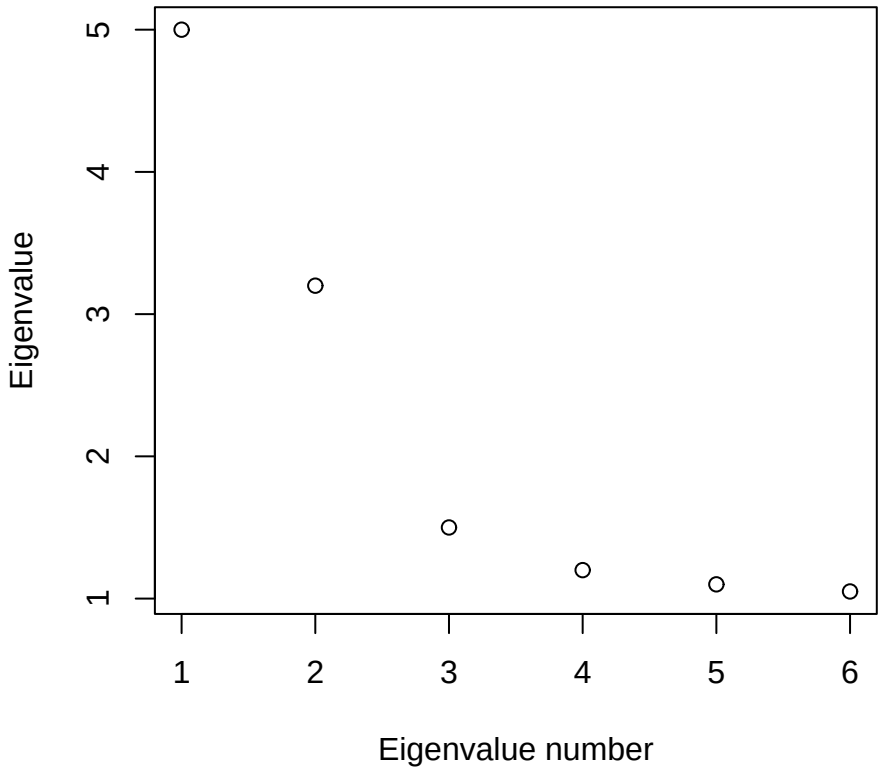

Figure 4.6: Scree plot showing slope on the left and rubble on the right.

This procedure fails if the change in λ_i is smooth,eg, as shown in Figure 4.7.

4.10.2 Horn's Procedure

This procedure replaces the data with an similar data structure in terms of number of variables and observations. Simulated data is generated under the assumption of no dependence between any of the response variables. The resulting scree plot is superimposed over the original scree plot and the region of intersection is taken as

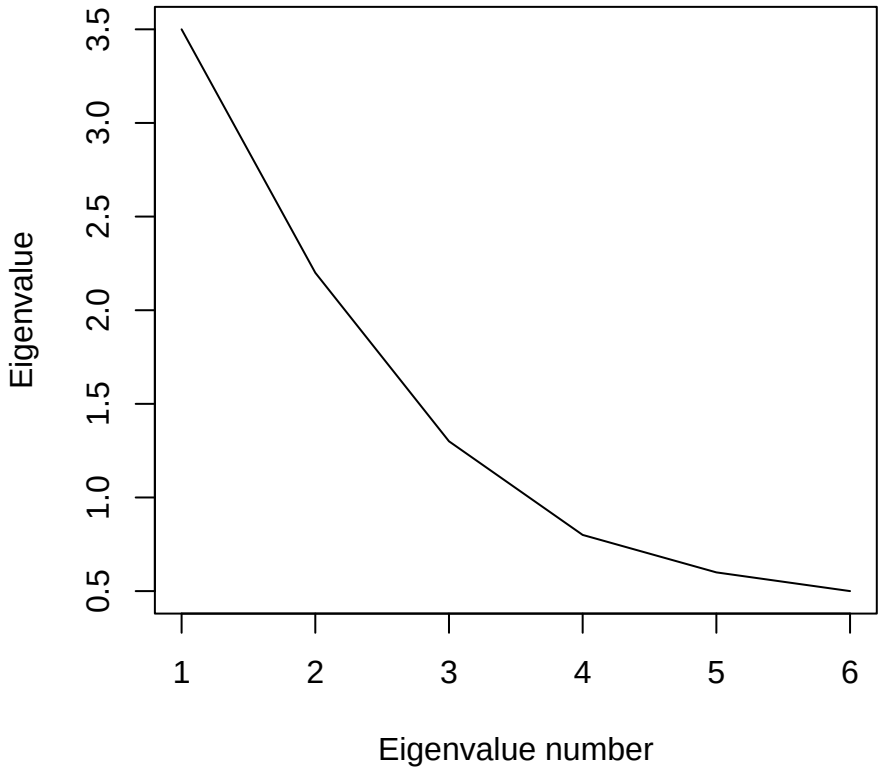

Figure 4.7: Scree plot showing smooth transition.

the cut–off. Thus in Figure 4.8 only the first two components would be retained. Thus this method effectively supersedes the scree plot.

The simulation can be replaced by an equation determined by regression for correlation input data, S p77–79. For standardised data, this procedure has been shown to outperform other procedures, as is thus recommended, S p79. For unstandardised data, the full simulation approach is required (S, p77 footnote 11).

For both types of data, an approximate implementation of Horn's method is the *broken stick* model; Jackson p47 , Jollife p95. If a line segment of unit length is broken into p segments, the expected length of the kth longest segment is

$$\ell_k = \frac{1}{p} \sum_{i=k}^{p} \left(\frac{1}{i} \right).$$

Retain the component k as long as the actual proportion explained (a_k) is longer than the expected proportion ℓ_k.

So for example if $p = 4$, the expected proportions are $0.0625, 0.14583^{\cdot}, 0.27083^{\cdot}$ and $0.52083^{\cdot}$ based on manual calculations.

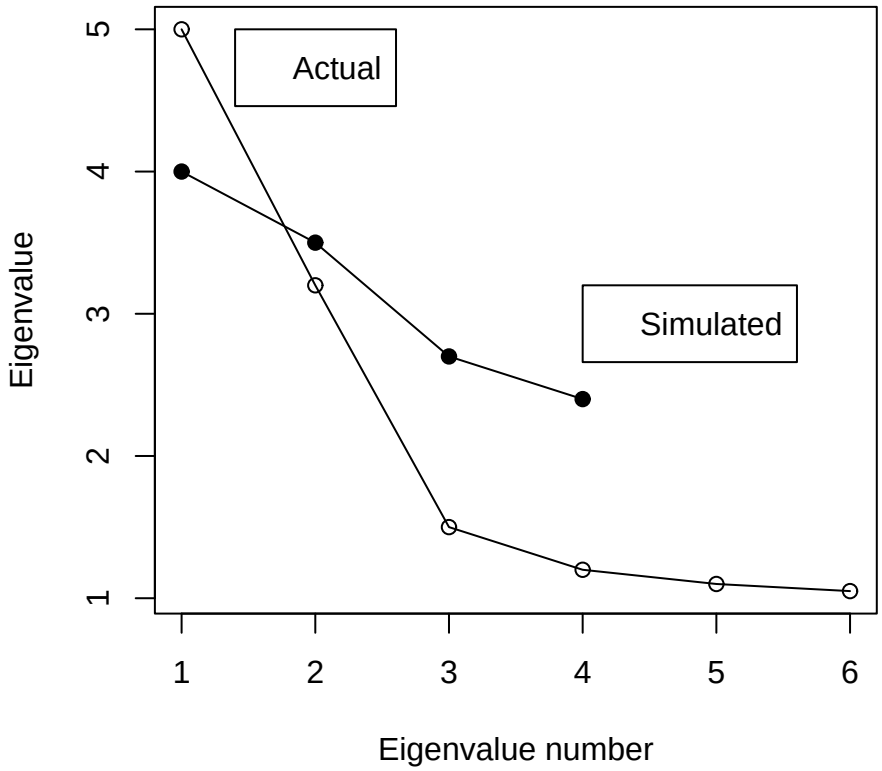

Figure 4.8: Scree plot with simulation superimposed.

R Implementation

The following R code shows the calculations needed to determine the expected proportions from the broken stick model for the case $p = 4$ described in Section 4.10.2.

The term `evn` gives the eigenvalue number, while `lk` gives the expected proportion of variance explained by each eigenvalue, based on the model of fractures when all eigenvalues are theoretically equal. Notice the reverse order for `evn` and thus `lk` in the final printout.

R Output

```
> p <- 4
> pone <- p-1
> temp <- 1/(p*p)
> lk <- temp
> for (k in pone:1){
+ temp<- temp + 1/(k*p)
+ lk <- rbind(lk,temp[1])
+ }
```

```
> #print(lk)
> evn<-(p:1)
> #evn
> #cbind(evn,lk)
> bs <-cbind(evn,lk)
> bs
   evn
lk   4 0.0625000
     3 0.1458333
     2 0.2708333
     1 0.5208333
```

4.10.3 % of Variance

This subjective method discards lower order components once the total % of variance accounted for reaches a preset limit, say 90%. The discarded components may still be informative however. This method does have its detractors, eg, Jackson p44 as the rule is somewhat arbitrary.

4.11 Correlation Input

The methods described in section 4.11.2 and section 4.11.1 are used to determine the number of eigenvalues to retain for a PCA based on standardised data.

4.11.1 Regression Method

Sharma p 77 reports a regression method due to Allen and Hubbard (1986) that obviates the necessity of simulations for standardised data. using Horn's procedure. The equation gives the estimate of the kth eigenvalue from uncorrelated data as

$$\ln \lambda_k = a_k + b_k \ln(n-1) + c_k \ln\{(p-k-1)(p-k+2)/2\} + d_k \ln(\lambda_{k-1})$$

where p is the number of variables, n is the number of observations, a_k, b_k, c_k and d_k are the regression coefficients given in Table 4.2. The eigenvalue λ_0 is assumed to be unity.

The regression equation produces the eigenvalue trail from the uncorrelated data which is then compared to the actual scree plot from the actual data. The cross over when the simulated and actual are plotted on the same diagram is taken as the point at which to discard components.

The sole limitation of the method is that the last two eigenvalues cannot be estimated for $p \leq 43$. This is not usually a concern since most of the time low order numbers of components are desired. If this limitation was critical, then the simulation implementation of Horn's procedure would need to be invoked.

k	a	b	c	d
1	.9794	-.2059	.1226	0.0000
2	-.3781	.0461	.0040	1.0578
3	-.3306	.0424	.0003	1.0805
4	-.2795	.0364	-.0003	1.0714
5	-.2670	.0360	-.0024	1.0899
6	-.2632	.0368	-.0040	1.1039
7	-.2580	.0360	-.0039	1.1173
8	-.2544	.0373	-.0064	1.1421
9	-.2111	.0329	-.0079	1.1229
10	-.1964	.0310	-.0083	1.1320
11	-.1858	.0288	-.0073	1.1284
12	-.1701	.0276	-.0090	1.1534
13	-.1697	.0266	-.0075	1.1632
14	-.1226	.0229	-.0113	1.1462
15	-.1005	.0212	-.0133	1.1668
16	-.1079	.0193	-.0088	1.1374
17	-.0866	.0177	-.0110	1.1718
18	-.0743	.0139	-.0081	1.1571
19	-.0910	.0152	-.0056	1.0934
20	-.0879	.0145	-.0051	1.1005

Table 4.2: Table of coefficients for the regression relation giving the random eigenvalues

R code

The regression equation to produce a simulated scree plot via Horn's procedure has been coded in R. The example from Sharma p80 has been reproduced to verify the correct operation of the implementation, by reproducing the simulated eigenvalues for the first three components.

Eigenvalue	Sharma	R
1	1.845	1.844 732
2	1.520	1.519 774
3	1.288	1.287 786

Table 4.3: Sharma vs R : regression method

Note that the file "abcd.dat" in the R code is simply the entries for `a, b, c, d` from Table 4.2.

R Output : regression method

```
> abcd <- read.table("abcd.dat",header=T)
> #abcd
```

```
> a <- abcd[,1]
> b <- abcd[,2]
> c <- abcd[,3]
> d <- abcd[,4]
> #print(cbind(a,b,c,d))
> llambda <- exp(1)
> lambda <- exp(1)
> n <- 23
> p <-  5
> pend <- p - 2
> for (i in 1:pend) {
+ ollambda <- llambda
+ llambda <- a[i] + b[i]*log(n - 1) + c[i]*log( (p - i - 1)*(p - i + 2)/2 ) +
  d[i]*ollambda
+ lambda <- rbind(lambda,exp(llambda))
+ #print(cbind(i,exp(llambda)))
+ }
> lambda[-1]

1.844732 1.519774 1.287786
```

4.11.2 Eigenvalues greater than Unity

This rule states that only those eigenvalues greater than unity be retained. The rationale is that if the correlation matrix is the identity, then

$$\lambda_i \equiv 1 \; \forall i$$

This rule is based on deterministic arguments and in practice does not always retain the correct number of components, according to the results of simulation experiments (S p 76).

4.12 Covariance Input

The methods described in this section are used to determine the number of eigenvalues to retain for a PCA based on unstandardised data. Again, the use of Horn's procedure is recommended based on empirical evidence from simulation studies, S p79.

4.12.1 Horn's procedure

This procedure is one of the methods encompassed in *parallel analysis*, by which is meant methods for comparing random and structured data, Jackson p47. The procedure for generating the simulated data is to produce normal uncorrelated data with the same number of observations and variables as the original, having the same variances for each variable. The criterion for retention of components is the same as for standardised data ; the broken stick calculation could be used as a guide and a check against what is a visual interpretation of the scree plot.

4.12.2 Tests on Eigenvalues

Bartlett's test can test if the remaining eigenvalues are equal. This may be used in deciding which components to discard. The test is rarely used due to its sensitivity to sample size (S p79).

4.12.3 SE of component coefficients

FR p204

This calculation assumes the MVN and that the PCA is performed on the *covariance* matrix. The given formula is asymptotically correct, ie, it assumes that the sample size is large. For b_{hj}, the jth coefficient in component h, the corresponding standard error is

$$SE(b_{hj}) = \sqrt{\left(\frac{1}{n-1}\lambda_h \sum_{k=1,k\neq h}^{p} \frac{\lambda_k}{(\lambda_k - \lambda_h)^2} b_{kj}^2\right)}$$

where λ_k is the kth eigenvalue, p is the number of variables and n is the number of observations. Note that the coefficients $b_{kj},\ k = 1, \ldots p$ are associated with the variable X_j, the jth column of the coefficient matrix.

These standard errors $SE(b)$ can aid in the interpretation of the coefficients in b, and consequently help decide the number of components to retain.

The following R code has been used on the data from FR p204 to verify the results obtained.

R Output : SE of coefficients

```
> ell <- c(0.6891,0.3593,0.1856,0.0872,0.0802,0.0420)
> b1 <- c(0.0613,0.3784,0.4715,-0.7863,0.1114,0.0114)
> ell
[1] 0.6891 0.3593 0.1856 0.0872 0.0802 0.0420
> n <- 100
> b1
[1]  0.0613  0.3784  0.4715 -0.7863  0.1114  0.0114
> e1 <- ell
> e1
[1] 0.6891 0.3593 0.1856 0.0872 0.0802 0.0420
> for (i in 1:6)
+ {
+ div <- 0
+ e1 <- 0
+ #print(ell[i])
+ div <- ell - ell[i]
+ #print(div)
+ div[abs(div)<1e-6] <- 1
+ e1 <- ell
+ #print(e1)
```

```
+ e1[i] <- 0
+ #print(div)
+ #e1[div=1] <- 0
+ #print(e1)
+ b1i <- ell[i] *sum(e1*b1*b1/(div*div))
+ #print(b1i)
+ seb <- sqrt(b1i/(n-1))
+ print(seb)
+ }
[1] 0.07402602
[1] 0.08797132
[1] 0.1175712
[1] 0.1492145
[1] 0.9459797
[1] 0.1120393
> #seb
```

Choosing the first variable X_1 from FR p204, the correspondence between the two is given in Table 4.4.

Component	$SE(b_{h1})$	'R'
U_1	0.074	0.0740
U_2	0.088	0.0879
U_3	0.118	0.1175
U_4	0.149	0.1492
U_5	0.943	0.9459
U_6	0.112	0.1120

Table 4.4: FR example p204 vs R code

4.12.4 SE of eigenvalues

The hth eigenvalue, λ_h, has the corresponding standard error

$$SE(\lambda_h) = \lambda_h \sqrt{\frac{2}{n-1}}$$

where n is the number of observations. This standard error can aid in the interpretation of the significance of the eigenvalue, and hence in the decision to retain the corresponding component. Note that if the MVN assumption does not hold, then this SE may be higher than stated by the formula.

To check the calculations for the FR example (n=100), the calculations are given in Table 4.5.

4.13 Worked example

Census data example from JW p392.

λ	$SE(\lambda)$ (FR)	hand calculation
0.6891	0.097944	0.0979
0.3593	0.051069	0.0511
0.1856	0.026380	0.0264
0.0872	0.012394	0.0124
0.0802	0.011399	0.0114
0.0420	0.005970	0.0060

Table 4.5: SE of eigenvalues : FR data

4.13.1 Correlation Matrix

```
> dat <- read.table("census.dat", header=T)
> dat
   tract  popn school employ health value
1      1 5.935   14.2  2.265   2.27  2.91
2      2 1.523   13.1  0.597   0.75  2.62
3      3 2.599   12.7  1.237   1.11  1.72
4      4 4.009   15.2  1.649   0.81  3.02
5      5 4.687   14.7  2.312   2.50  2.22
6      6 8.044   15.6  3.641   4.51  2.36
7      7 2.766   13.3  1.244   1.03  1.97
8      8 6.538   17.0  2.618   2.39  1.85
9      9 6.451   12.9  3.147   5.52  2.01
10    10 3.314   12.2  1.606   2.18  1.82
11    11 3.777   13.0  2.119   2.83  1.80
12    12 1.530   13.8  0.798   0.84  4.25
13    13 2.768   13.6  1.336   1.75  2.64
14    14 6.585   14.9  2.763   1.91  3.17
> data <- as.data.frame(dat[,-1])
> data
    popn school employ health value
1  5.935   14.2  2.265   2.27  2.91
2  1.523   13.1  0.597   0.75  2.62
3  2.599   12.7  1.237   1.11  1.72
4  4.009   15.2  1.649   0.81  3.02
5  4.687   14.7  2.312   2.50  2.22
6  8.044   15.6  3.641   4.51  2.36
7  2.766   13.3  1.244   1.03  1.97
8  6.538   17.0  2.618   2.39  1.85
9  6.451   12.9  3.147   5.52  2.01
10 3.314   12.2  1.606   2.18  1.82
11 3.777   13.0  2.119   2.83  1.80
12 1.530   13.8  0.798   0.84  4.25
13 2.768   13.6  1.336   1.75  2.64
14 6.585   14.9  2.763   1.91  3.17
```

```
> library(mva)
> prcomp(data,center=T,scale=T)
Standard deviations:
[1] 1.7403724 1.1362825 0.7566080 0.3088664 0.1100538

Rotation:
              PC1          PC2          PC3         PC4          PC5
popn   -0.5583589 -0.131392987  0.007945807 -0.55055321  0.606464575
school -0.3132830 -0.628872546 -0.549030533  0.45265380 -0.006564747
employ -0.5682577 -0.004262264  0.117280380 -0.26811649 -0.769040874
health -0.4866246  0.309560576  0.454923806  0.64798227  0.201325679
value   0.1742664 -0.701005911  0.691224986 -0.01510711 -0.014203097
> o1 <- prcomp(data,center=T,retx=T,scale=T)
> o1$sdev
[1] 1.7403724 1.1362825 0.7566080 0.3088664 0.1100538
> slam<-o1$sdev
> lam<-slam*slam
> print(lam)
[1] 3.02889606 1.29113796 0.57245566 0.09539848 0.01211184
> o1$rotation
              PC1          PC2          PC3         PC4          PC5
popn   -0.5583589 -0.131392987  0.007945807 -0.55055321  0.606464575
school -0.3132830 -0.628872546 -0.549030533  0.45265380 -0.006564747
employ -0.5682577 -0.004262264  0.117280380 -0.26811649 -0.769040874
health -0.4866246  0.309560576  0.454923806  0.64798227  0.201325679
value   0.1742664 -0.701005911  0.691224986 -0.01510711 -0.014203097
> o1$x

            PC1         PC2         PC3         PC4          PC5
  1  -0.5983117 -0.61944480  0.44595617 -0.42218507  0.206299772
  2   2.3630458  0.13910622 -0.11026946  0.17778409  0.143830557
  3   1.4157171  1.22491126 -0.61633603 -0.25023623 -0.020187378
  4   0.6086406 -1.39823366 -0.42134387 -0.06268978 -0.043651985
  5  -0.6592990 -0.04537044 -0.35615703  0.18590026 -0.154442903
  6  -3.2811300 -0.38476758  0.24703862  0.12870817 -0.034614757
  7   1.3140407  0.66607802 -0.64517372 -0.13460249  0.003155881
  8  -1.9462339 -0.91102625 -1.65457187  0.34338212  0.103700351
  9  -2.3387020  1.56386702  1.27796982  0.25380032  0.089695046
  10  0.7603588  1.55171940  0.08543445 -0.22878447  0.025570341
  11 -0.1088036  1.30466197  0.01530825  0.06761506 -0.190341141
  12  2.4373195 -1.78246564  1.24265263  0.36091778 -0.050017792
  13  1.0991242  0.02109529  0.12849966  0.25763727  0.013079454
  14 -1.0657666 -1.33013082  0.36099240 -0.67724703 -0.092075446
> plot(o1)
> summary(o1)
Importance of components:
```

```
                          PC1   PC2   PC3    PC4     PC5
Standard deviation     1.740 1.136 0.757 0.3089 0.11005
Proportion of Variance 0.606 0.258 0.114 0.0191 0.00242
Cumulative Proportion  0.606 0.864 0.979 0.9976 1.00000
> cor(data,o1$x)
               PC1          PC2          PC3          PC4           PC5
popn   -0.9717524 -0.149299554  0.006011861 -0.170047415  0.0667437294
school -0.5452291 -0.714576879 -0.415400893  0.139809572 -0.0007224753
employ -0.9889800 -0.004843136  0.088735273 -0.082812187 -0.0846358685
health -0.8469081  0.351748270  0.344198990  0.200139982  0.0221566555
value   0.3032884 -0.796540761  0.522986353 -0.004666078 -0.0015631047
> cor(o1$x)
               PC1           PC2           PC3           PC4           PC5
PC1  1.000000e+00  8.559088e-17  2.137875e-16 -8.039992e-16 -1.444690e-15
PC2  8.559088e-17  1.000000e+00 -2.020007e-16  4.965258e-16  9.770794e-16
PC3  2.137875e-16 -2.020007e-16  1.000000e+00 -1.393760e-16 -1.695007e-16
PC4 -8.039992e-16  4.965258e-16 -1.393760e-16  1.000000e+00 -1.616055e-15
PC5 -1.444690e-15  9.770794e-16 -1.695007e-16 -1.616055e-15  1.000000e+00
```

Regression Method

The results of using the regression method are reported in Table 4.6.

Eigenvalue	Actual	'Simulated'	
1	3.029	2.055	
2	1.291	1.663	
3	0.572	1.388	
4	0.095	–	cannot be
5	0.012	–	calculated

Table 4.6: Actual vs simulated : Horn's procedure via regression method

So choose components 1 and 2. The R code for the regression method follows.

R Output : regression method

```
> abcd <- read.table("abcd.dat",header=T)
> #abcd
> a <- abcd[,1]
> b <- abcd[,2]
> c <- abcd[,3]
> d <- abcd[,4]
> #print(cbind(a,b,c,d))
> llambda <- exp(1)
> lambda <- exp(1)
> n <- 14
> p <-  5
```

```
> pend <- p - 2
> for (i in 1:pend) {
+ ollambda <- llambda
+ llambda <- a[i] + b[i]*log(n - 1) + c[i]*log( (p - i - 1)*(p - i + 2)/2 ) +
   d[i]*ollambda
+ lambda <- rbind(lambda,exp(llambda))
+ #print(cbind(i,exp(llambda)))
+ }
> lambda[-1]

2.055783 1.663448 1.388495
```

The approximate implementation of Horn's procedure is the broken stick method. The results for $p = 5$ is given in Table 4.7.

% of variance	1	2	3	4	5
Actual	60.6	25.8	11.4	1.9	0.2
Broken Stick	45.6	25.6	15.6	9.0	4.0

Table 4.7: Actual % vs Broken Stick : Correlation matrix

R Output : Broken stick Model

```
> p <- 5
> pone <- p-1
> temp <- 1/(p*p)
> lk <- temp
> for (k in pone:1){
+ temp<- temp + 1/(k*p)
+ lk <- rbind(lk,temp[1])
+ }
> #print(lk)
> evn<-(p:1)
> #evn
> #cbind(evn,lk)
> bs <-cbind(evn,lk)
> bs
   evn
lk   5 0.0400000
     4 0.0900000
     3 0.1566667
     2 0.2566667
     1 0.4566667
```

So on the basis of the broken stick model, choose the first two components (just!).

Eigenvalues less than unity

From Table 4.6, choose components 1 and 2 only, as the remaining eigenvalues are all below unity.

% of Variance

As stated before, this criterion is not considered reliable (Jo p44), but the first two components account for 86% of the total variance.

4.13.2 Covariance Matrix

JW Census data p392

```
> dat <- read.table("census.dat", header=T)
> dat
   tract  popn school employ health value
1      1 5.935   14.2  2.265   2.27  2.91
2      2 1.523   13.1  0.597   0.75  2.62
3      3 2.599   12.7  1.237   1.11  1.72
4      4 4.009   15.2  1.649   0.81  3.02
5      5 4.687   14.7  2.312   2.50  2.22
6      6 8.044   15.6  3.641   4.51  2.36
7      7 2.766   13.3  1.244   1.03  1.97
8      8 6.538   17.0  2.618   2.39  1.85
9      9 6.451   12.9  3.147   5.52  2.01
10    10 3.314   12.2  1.606   2.18  1.82
11    11 3.777   13.0  2.119   2.83  1.80
12    12 1.530   13.8  0.798   0.84  4.25
13    13 2.768   13.6  1.336   1.75  2.64
14    14 6.585   14.9  2.763   1.91  3.17
> data <- as.data.frame(dat[,-1])
> data
    popn school employ health value
1  5.935   14.2  2.265   2.27  2.91
2  1.523   13.1  0.597   0.75  2.62
3  2.599   12.7  1.237   1.11  1.72
4  4.009   15.2  1.649   0.81  3.02
5  4.687   14.7  2.312   2.50  2.22
6  8.044   15.6  3.641   4.51  2.36
7  2.766   13.3  1.244   1.03  1.97
8  6.538   17.0  2.618   2.39  1.85
9  6.451   12.9  3.147   5.52  2.01
10 3.314   12.2  1.606   2.18  1.82
11 3.777   13.0  2.119   2.83  1.80
12 1.530   13.8  0.798   0.84  4.25
13 2.768   13.6  1.336   1.75  2.64
```

```
14 6.585   14.9  2.763   1.91  3.17
> library(mva)
> prcomp(data,center=T)
Standard deviations:
[1] 2.6326932 1.3360929 0.6242194 0.4790918 0.1189747

Rotation:
               PC1         PC2          PC3         PC4          PC5
popn   -0.78120807  0.07087183  0.003656607  0.54171007  0.302039670
school -0.30564856  0.76387277 -0.161817438 -0.54479937  0.009279632
employ -0.33444840 -0.08290788  0.014841008  0.05101636 -0.937255367
health -0.42600795 -0.57945799  0.220453468 -0.63601254  0.172145212
value   0.05435431  0.26235528  0.961759720  0.05127599 -0.024583093
> o1 <- prcomp(data,center=T,retx=T)
> o1$sdev
[1] 2.6326932 1.3360929 0.6242194 0.4790918 0.1189747
> o1$rotation
               PC1         PC2          PC3         PC4          PC5
popn   -0.78120807  0.07087183  0.003656607  0.54171007  0.302039670
school -0.30564856  0.76387277 -0.161817438 -0.54479937  0.009279632
employ -0.33444840 -0.08290788  0.014841008  0.05101636 -0.937255367
health -0.42600795 -0.57945799  0.220453468 -0.63601254  0.172145212
value   0.05435431  0.26235528  0.961759720  0.05127599 -0.024583093
> o1$x

            PC1         PC2         PC3         PC4          PC5
  1  -1.4376565  0.29260180  0.44050065  0.74853291  0.201197601
  2   3.5348762  0.08263869 -0.03638751 -0.17543886  0.167201310
  3   2.4002270 -0.64443800 -0.74444828  0.38289849 -0.027262233
  4   0.5952725  1.84591442  0.04643003  0.06319255 -0.047938015
  5  -0.7667385  0.26789299 -0.25718390 -0.37918649 -0.158603360
  6  -4.9574860 -0.04487018  0.20693717 -0.25439106 -0.039346660
  7   2.1317042 -0.06291364 -0.61802055  0.21054156  0.002267993
  8  -2.9913377  2.09728323 -0.99815525 -0.56291943  0.125174851
  9  -3.1718449 -2.85634591  0.51672992 -0.33189825  0.099924038
  10  1.4206830 -1.60007856 -0.32338757  0.38604008  0.019946152
  11  0.3649005 -1.38059323 -0.31947552 -0.18724994 -0.201211729
  12  3.2984865  0.97666914  1.44085808 -0.51641343 -0.037154372
  13  1.7373697 -0.08266930  0.13791241 -0.37069545  0.026902349
  14 -2.1584560  1.10890855  0.50769034  0.98698731 -0.131097925
> plot(o1)
> summary(o1)
Importance of components:
                          PC1   PC2    PC3    PC4     PC5
Standard deviation     2.633 1.336 0.6242 0.4791 0.11897
Proportion of Variance 0.741 0.191 0.0417 0.0245 0.00151
```

```
Cumulative Proportion  0.741 0.932 0.9739 0.9985 1.00000
> cor(data,o1$x)
              PC1         PC2          PC3         PC4           PC5
popn   -0.9909495  0.04562416  0.001099766  0.12504610  0.0173142311
school -0.6052660  0.76768201 -0.075977724 -0.19632652  0.0008304416
employ -0.9840179 -0.12379586  0.010353193  0.02731504 -0.1246195660
health -0.7991766 -0.55167514  0.098057095 -0.21712467  0.0145940026
value   0.2014908  0.49356853  0.845327227  0.03459025 -0.0041182450
> cor(o1$x)
              PC1           PC2           PC3           PC4           PC5
PC1  1.000000e+00 -4.818329e-17  7.119101e-17  1.303429e-15 -1.624428e-15
PC2 -4.818329e-17  1.000000e+00 -1.284979e-17  9.775148e-16 -2.175510e-15
PC3  7.119101e-17 -1.284979e-17  1.000000e+00 -9.802486e-17 -3.544149e-16
PC4  1.303429e-15  9.775148e-16 -9.802486e-17  1.000000e+00 -2.061601e-16
PC5 -1.624428e-15 -2.175510e-15 -3.544149e-16 -2.061601e-16  1.000000e+00
```

Horn's procedure

Sharma (p77) in a footnote alludes to the possibility of applying Horn's procedure when the input data are unstandardised. This is supported by Jackson p45–47, where Horn's procedure is described in general for a covariance matrix. Jackson p47 (2.8.7) further describes the broken–stick method as a "quick and dirty" version of Horn's method. Since this broken–stick method is used on the proportion of variance, it applies to both standardised and unstandardised data, thereby implying that Horn's procedure can be used in both situations. Also the direct use of the simulation method will produce *all* of the eigenvalues, which may be of interest if the number of variables (p) is small, say approximately 6. The following R code shows the implementation of Horn's procedure for the census data from JW assuming covariance matrix input.

R Output : Horn's procedure

```
> l <- c(0,0,0,0,0)
> library(mva)
> #
> # JW example cov matrix
> #
> for (i in 1:100)
+ {
+ x1 <- rnorm(14,mean=4.323286,sd=2.0754652)
+ x2 <- rnorm(14,mean=14.014286,sd=1.3294632)
+ x3 <- rnorm(14,mean=1.952286,sd=0.8948008)
+ x4 <- rnorm(14,mean=2.171429,sd=1.4033798)
+ x5 <- rnorm(14,mean=2.454286,sd=0.7101973)
+ data <- cbind(x1,x2,x3,x4,x5)
+ out <- prcomp(data,center=T)
+ lam <- t(out$sdev*out$sdev)
+ l <- rbind(l,lam)
```

```
+ }
> lambda <- l[-1,]
> #lambda
> mean(lambda[,1])
[1] 4.919402
> mean(lambda[,2])
[1] 2.163980
> mean(lambda[,3])
[1] 1.277632
> mean(lambda[,4])
[1] 0.5931993
> mean(lambda[,5])
[1] 0.2539806
```

The results of the simulation are reported in Table 4.8.

Eigenvalue	Actual	Simulated
1	6.931	4.919
2	1.785	2.163
3	0.389	1.277
4	0.229	0.593
5	0.014	0.253

Table 4.8: Actual and simulated eigenvalues : JW data - unstandardised

The corresponding scree plot with the simulated values superimposed is given in Figure 4.9.

Thus the outcome of the procedure is to keep the first two components.

Broken Stick Model

The quick and dirty approximation via the broken stick model is given in Table 4.9. The outcome is to retain the first component only.

SE of coefficients

As an aid to interpretation of coefficients in a component, choose the original variable "Value", and examine the significance of the coefficient in component 3. The coefficients for "Value" and their corresponding SEs are given in Table 4.10.

Note that the incorrect sign for b_{55} in the R code is irrelevant. The T ratio for b_{53} is 18.8!

% of variance	1	2	3	4	5
Actual	74.1	19.1	4.2	2.45	0.15
Broken Stick	45.6	25.6	15.6	9.0	4.0

Table 4.9: Actual % vs Broken Stick : Covariance matrix

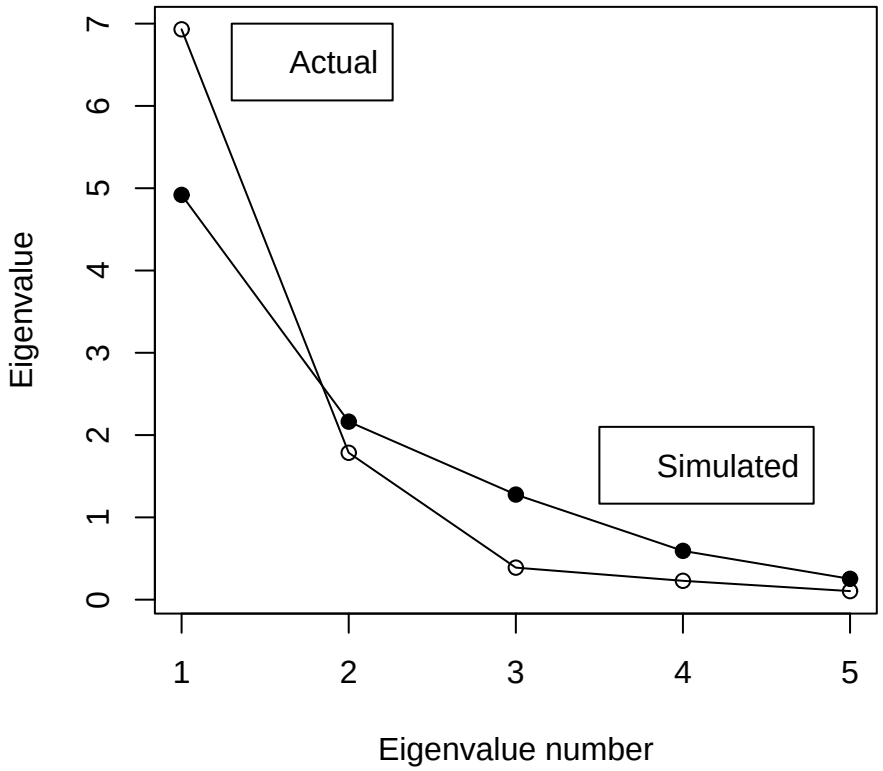

Figure 4.9: Scree plot with simulation superimposed. : JW data - unstandardised

```
> ell <- c(6.931,1.786,0.390,.230,.014)
> b1 <- c(.0543,.2623,.9617,.0512,0.0245)
> ell
[1] 6.931 1.786 0.390 0.230 0.014
> n <- 14
> b1
[1] 0.0543 0.2623 0.9617 0.0512 0.0245
> e1 <- ell
> e1
[1] 6.931 1.786 0.390 0.230 0.014
> for (i in 1:5)
+ {
+ div <- 0
+ e1 <- 0
+ #print(ell[i])
+ div <- ell - ell[i]
+ #print(div)
+ div[abs(div)<1e-6] <- 1
```

	PC1	PC2	PC3	PC4	PC5
Value	0.0543	0.2623	0.9617	0.0512	-0.0245
SE(b) [R]	0.083	0.1599	0.0511	0.500	0.0529

Table 4.10: Coefficients for "Value" and their corresponding SEs : JW data

$\sqrt{\lambda}$	λ	$SE(\lambda)$
2.633	6.931	2.718
1.336	1.785	0.700
0.624	0.389	0.152
0.479	0.229	0.090
0.118	0.014	0.005

Table 4.11: SE of eigenvalues : JW data

```
+ e1 <- ell
+ #print(e1)
+ e1[i] <- 0
+ #print(div)
+ #e1[div=1] <- 0
+ #print(e1)
+ b1i <- ell[i] *sum(e1*b1*b1/(div*div))
+ #print(b1i)
+ seb <- sqrt(b1i/(n-1))
+ print(seb)
+ }
[1] 0.08352809
[1] 0.1599018
[1] 0.05112992
[1] 0.5001897
[1] 0.05295391
> #seb
```

SE of eigenvalues

The results for JW data are given in Table 4.11.

These SEs are not much use in a decision about retaining components, since all the eigenvalues share the same T ratio of $\sqrt{\frac{n-1}{2}}$!

% of Variance

Components 1 and 2 account for 93% of the total variance.

Conclusion

The overall conclusion is to retain the first two components, but remove "Value" from the list of variables, as being a separate independent or unrelated variable.

You should now attempt Workshop 4.

Chapter 5

Discriminant Analysis

(J p217)

In Figure 5.1 is shown the situation where discriminant analysis is needed since neither x_1 nor x_2 alone discriminates between the two groups(o and Δ), but a linear combination of x_1 and x_2 does. The determination of that linear combination is the procedure called (linear) **discriminant analysis**. The figures on each axis are the projections of the data onto the axes.

The line shown joins the two centroids, ie, the means of each variable within each group.

The objective of discriminant analysis is prediction (classification) and explanation via discovery of the form of the underlying discriminant function(s).

5.1 Two group problem

K p1 **Part 2** Classification, Covariance Structures and Repeated Measurements

D and G p401.

Marlia p319

The variable means are defined as :

$$E(\boldsymbol{x}_1) = \begin{bmatrix} \mu_{11} \\ \vdots \\ \mu_{1p} \end{bmatrix} = \boldsymbol{\mu}_1,\ E(\boldsymbol{x}_2) = \begin{bmatrix} \mu_{21} \\ \vdots \\ \mu_{2p} \end{bmatrix} = \boldsymbol{\mu}_2.$$

The discrimination problem is to find a single variable $\mathbf{y} = \boldsymbol{b}'\boldsymbol{x}$ that separates individuals into Class 1 or 2.

Each variable is assumed to be MVN, ie,

$$\boldsymbol{x}_1 \sim N_p(\boldsymbol{\mu}_1, \Sigma)$$

and

$$\boldsymbol{x}_2 \sim N_p(\boldsymbol{\mu}_2, \Sigma).$$

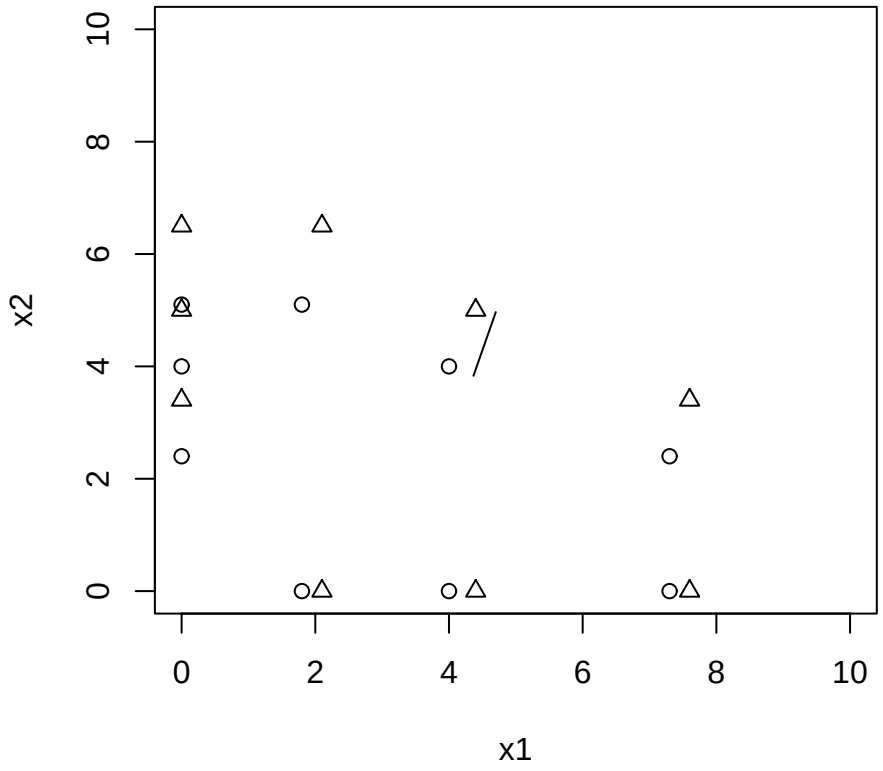

Figure 5.1: The need for a discriminant function

5.1.1 Graphical Representation

S p278, DG p 364.

The original derivation is due to Fisher (1936).

The idea is to find the discriminant function by maximising the ratio of between group distance to the within group distance. A graphical demonstration of the rationale of this approach is given in Figure 5.2, Figure 5.3 and Figure 5.4.

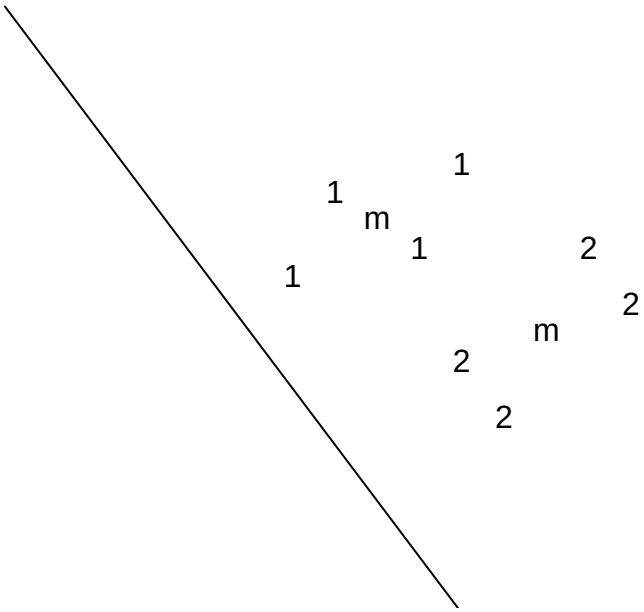

Figure 5.2: Group separation in the maximal configuration

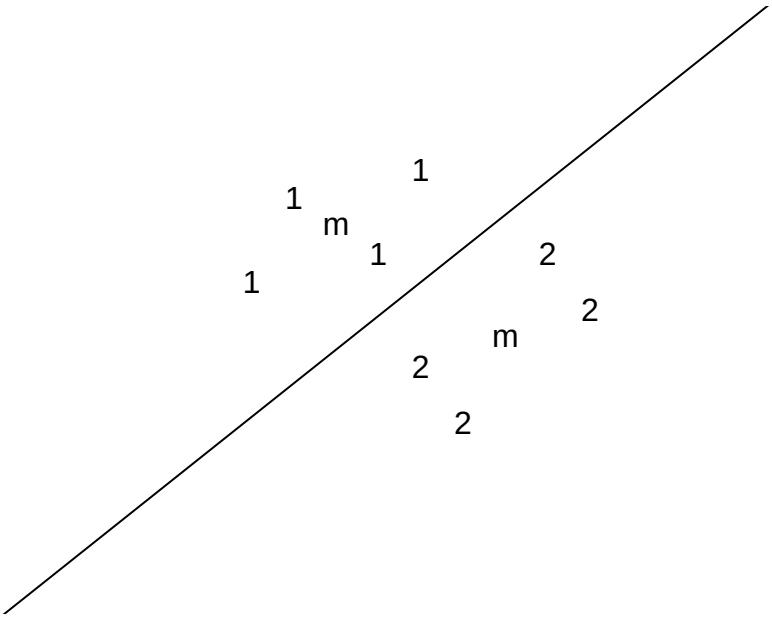

Figure 5.3: Group separations in the minimal configuration

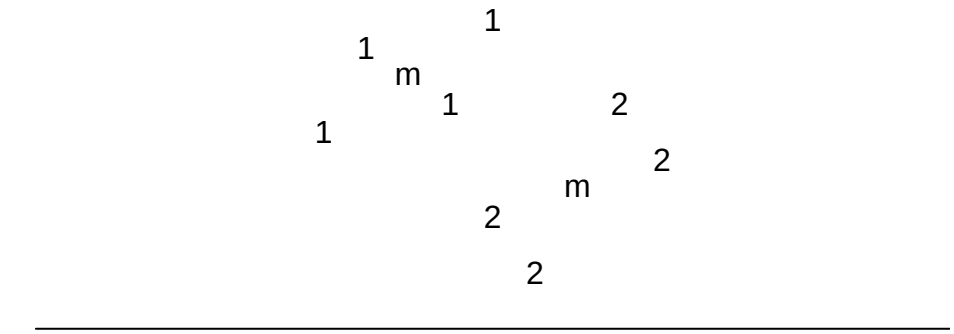

Figure 5.4: Group separation in the suboptimal configuration

The data points for groups 1 and 2 are shown by the numerals, while the group means are shown by the character `m`. The directions shown are approximate only. (It is left as an exercise to the student to verify that the third diagram shows a suboptimal configuration, assuming that the first is maximal.)

5.1.2 Derivation

The ratio Δ of the between to the within squared distance is formally denoted as

$$\Delta = \frac{D^2}{V(D)} = \frac{B}{W} = \frac{(\boldsymbol{b}'\boldsymbol{\mu}_{g1} - \boldsymbol{b}'\boldsymbol{\mu}_{g2})^2}{\boldsymbol{b}'\Sigma\boldsymbol{b}}$$

where the between distance $D = \boldsymbol{b}'\boldsymbol{\mu}_{g1} - \boldsymbol{b}'\boldsymbol{\mu}_{g2}$. Note that $\boldsymbol{\mu}_{g1} \equiv \boldsymbol{\mu}_1$ as per Section 5.1.

To verify the denominator note that

$$V(\boldsymbol{b}'\boldsymbol{X}) = E(\boldsymbol{b}'\boldsymbol{X}\boldsymbol{X}'\boldsymbol{b}) = \boldsymbol{b}'E(\boldsymbol{X}\boldsymbol{X}')\boldsymbol{b} = \boldsymbol{b}'\Sigma\boldsymbol{b}$$

as required.

To find the optimal value of $\boldsymbol{b}$ that produces the maximal ratio, rewrite the ratio Δ as

$$\Delta = \frac{[\boldsymbol{b}'(\boldsymbol{\mu}_{g1} - \boldsymbol{\mu}_{g2})]^2}{\boldsymbol{b}'\Sigma\boldsymbol{b}} = \frac{(\boldsymbol{b}'\boldsymbol{d})^2}{\boldsymbol{b}'\Sigma\boldsymbol{b}} = \frac{(b_1 d_1 + \ldots + b_p d_p)^2}{\boldsymbol{b}'\Sigma\boldsymbol{b}}$$

where $\boldsymbol{d} = \boldsymbol{\mu}_{g1} - \boldsymbol{\mu}_{g2}$.

Now Δ can be written as

$$\Delta = P^2/Q$$

where P and Q are scalars, since the numerator and denominator of Δ are scalars, being variances. So

$$\frac{\partial}{\partial \boldsymbol{b}}(P^2/Q) = \frac{2PQ\frac{\partial P}{\partial \boldsymbol{b}} - P^2\frac{\partial Q}{\partial \boldsymbol{b}}}{Q^2} = 0$$

when

$$2\frac{P}{Q}\frac{\partial P}{\partial \boldsymbol{b}} - \frac{P^2}{Q^2}\frac{\partial Q}{\partial \boldsymbol{b}} = 0$$

ie, when

$$\frac{1}{2}\frac{\partial Q}{\partial \boldsymbol{b}} = \frac{Q}{P}\frac{\partial P}{\partial \boldsymbol{b}}.$$

But, $b_1, \ldots, b_p$ are not unique, so choose $Q/P = 1$, to give

$$\frac{1}{2}\frac{\partial \boldsymbol{b}'\Sigma\boldsymbol{b}}{\partial \boldsymbol{b}} = \frac{\partial \boldsymbol{b}'\boldsymbol{d}}{\partial \boldsymbol{b}}.$$

This yields

$$\Sigma\boldsymbol{b} = \boldsymbol{d}$$

which becomes

$$\boldsymbol{b} = \Sigma^{-1}\boldsymbol{d} = \Sigma^{-1}(\boldsymbol{\mu}_{g1} - \boldsymbol{\mu}_{g2}) = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2).$$

Using *sample* values this gives the sample discriminant function to be defined by the coefficients

$$\widehat{\boldsymbol{b}} = \boldsymbol{S}^{-1}(\bar{\boldsymbol{x}}_{g1} - \bar{\boldsymbol{x}}_{g2}) = \boldsymbol{S}^{-1}(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)$$

again using the shorthand notation for the group means.

5.2 More than 2 groups

For the moment consider the situation of more than two groups, i.e., $g > 2$.

The problem is to find $\boldsymbol{a}$ in the linear function $\boldsymbol{a}'\boldsymbol{x}$ maximises the ratio (V) of between - group to within group variance. Thus the ratio is

$$V = \boldsymbol{a}'\boldsymbol{B}\boldsymbol{a}/\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a}$$

where

$\boldsymbol{B}$ = between group covariance matrix

$\boldsymbol{W}$ = within group covariance matrix [1]

Maximising V wrt $\boldsymbol{a}$ gives

$$\begin{aligned}\frac{\partial V}{\partial \boldsymbol{a}} &= \frac{2\boldsymbol{B}\boldsymbol{a}}{\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a}} + \boldsymbol{a}'\boldsymbol{B}\boldsymbol{a}\frac{(-1)}{(\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a})^2}(+2\boldsymbol{W}\boldsymbol{a}) = 0 \\ &= \frac{\boldsymbol{B}\boldsymbol{a}(\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a}) - (\boldsymbol{a}'\boldsymbol{B}\boldsymbol{a})\boldsymbol{W}\boldsymbol{a}}{(\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a})^2} = 0\end{aligned}$$

$$\begin{aligned}[(\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a})\boldsymbol{B} - (\boldsymbol{a}'\boldsymbol{B}\boldsymbol{a})\boldsymbol{W}]\,\boldsymbol{a} &= 0 \\ \left(\boldsymbol{B} - \frac{(\boldsymbol{a}'\boldsymbol{B}\boldsymbol{a})}{(\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a})}\boldsymbol{W}\right)\boldsymbol{a} &= \\ (\boldsymbol{B} - \lambda\boldsymbol{W})\boldsymbol{a} &= 0 \\ (\boldsymbol{W}^{-1}\boldsymbol{B} - \lambda\boldsymbol{I})\boldsymbol{a} &= 0\end{aligned}$$

So the optimal ratio is the largest eigenvalue of $\boldsymbol{W}^{-1}\boldsymbol{B}$.

Madia p479, 480.

Note that the # of eigenvalues is min $(p, g-1)$. This is also the number of discriminant functions, potentially.

Example

For the Lawn mower data as given in Example 5 in the set of 6 examples from Chapter 1, there are two variables, Income and Lotsize, therefore, p=2.

There are two groups, Owners and Non–owners, therefore g=2, and so there will be only *one* discriminant function.

5.3 Simple Worked Example

JW P566

Data:

group	x_1	x_2		
1	3	7		
1	2	4	(3, 6)	$= \bar{\boldsymbol{x}}'_{g1}$
1	4	7		
2	6	9		
2	5	7	(5, 8)	$= \bar{\boldsymbol{x}}'_{g2}$
2	4	8		

[1] Note that $\boldsymbol{a}'\boldsymbol{B}\boldsymbol{a}$ and $\boldsymbol{a}'\boldsymbol{W}\boldsymbol{a}$ are scalars.

Discriminant function is $\boldsymbol{b}'\boldsymbol{x}$ where

$$\boldsymbol{b} = S_p^{-1}(\bar{\boldsymbol{x}}_{g1} - \bar{\boldsymbol{x}}_{g2}) = S_p^{-1}(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)$$

and S_p is the pooled sample covariance matrix.

$$R \longrightarrow S_1 = \begin{bmatrix} 1 & 1.5 \\ 1.5 & 3 \end{bmatrix},\ S_2 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1.0 \end{bmatrix}$$

Check S_1:

$$\begin{array}{rrrr} & x_1 & x_2 & \\ 0 & 3 & 7 & 1 \\ -1 & 2 & 4 & -2 \\ 1 & 4 & 7 & 1 \end{array} \qquad s_{x_1}^2 = \frac{1+1}{2} = 1$$

$$s_{x_2}^2 = \frac{1+4+1}{2} = 3$$

$$s_{x_1x_2} = \frac{0\times 1 + 1\times 2 + 1\times 1}{2} = \frac{3}{2} = 1.5$$

Exercise

Check S_2 .

$$\begin{aligned} S_p &= \frac{S_1 + S_2}{2} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \\ S^{-1} &= \frac{\begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}}{|2-1|} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \\ SS^{-1}? &= \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ \boldsymbol{b} = S_p^{-1}\boldsymbol{d} &= \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 3 & - & 5 \\ 6 & - & 8 \end{bmatrix} \\ &= \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} -2 \\ -2 \end{bmatrix} = \begin{bmatrix} -4 & + & 2 \\ 2 & - & 2 \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \end{bmatrix} \\ R &\rightarrow \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \text{“}LD1\text{”} \end{aligned}$$

$$y = -2x_1 + 0x_2$$

Thus x_2 is useless as a discriminator, hence the coefficient of zero for x_2 in y. The efficacy of x_1 as a discriminator can be verified from the plot in Figure 5.5, where the best group separation at $x_1 = 4$ is shown in a direction parallel to the x_1 axis.

R Output

```
> dat <- read.table("jwlda.dat",header=T)
> dat
  group x1 x2
1     1  3  7
2     1  2  4
3     1  4  7
4     2  6  9
5     2  5  7
6     2  4  8
> x <- cbind(dat[,2],dat[,3])
> group <- factor(dat[,1])
> print(cbind(group,x))
     group
[1,]     1 3 7
[2,]     1 2 4
[3,]     1 4 7
[4,]     2 6 9
[5,]     2 5 7
[6,]     2 4 8
> library(MASS)
> lda(group~x)
Call:
lda.formula(group ~ x)

Prior probabilities of groups:
  1   2
0.5 0.5

Group means:
  x1 x2
1  3  6
2  5  8

Coefficients of linear discriminants:
             LD1
x1  1.000000e+00
x2 -1.364468e-16
> cov(x)
     [,1] [,2]
[1,]    2  2.0
[2,]    2  2.8
> #plot(x)
> x1 <- x[group==1]
> x2 <- x[group==2]
> dim(x1) <- c(3,2)
> x1
```

```
     [,1] [,2]
[1,]    3    7
[2,]    2    4
[3,]    4    7
> cov(x1)
     [,1] [,2]
[1,]  1.0  1.5
[2,]  1.5  3.0
> dim(x2) <- c(3,2)
> x2
     [,1] [,2]
[1,]    6    9
[2,]    5    7
[3,]    4    8
> cov(x2)
     [,1] [,2]
[1,]  1.0  0.5
[2,]  0.5  1.0
> plot(x1,x2,type="n")
> points(x1,pch=1)
> points(x2,pch=2)
```

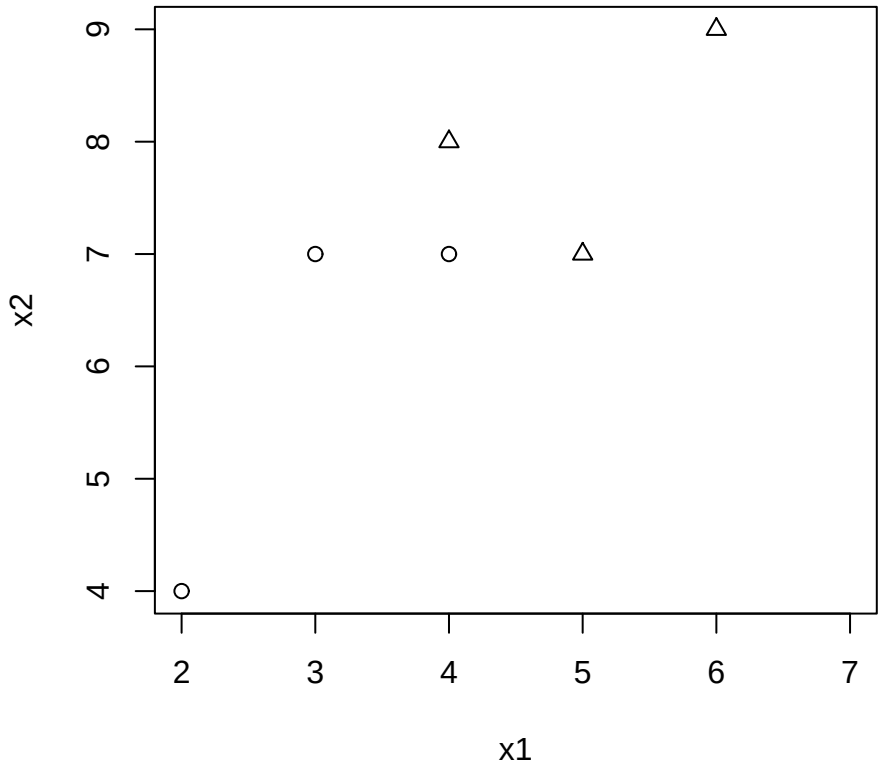

Figure 5.5: Plot of x_2 vs x_1 for JW p556

5.4 Lawn mower data

This data has been given as Example 5 in the set of 6 examples in Chapter 1. A plot of the data is shown in Figure 5.6.

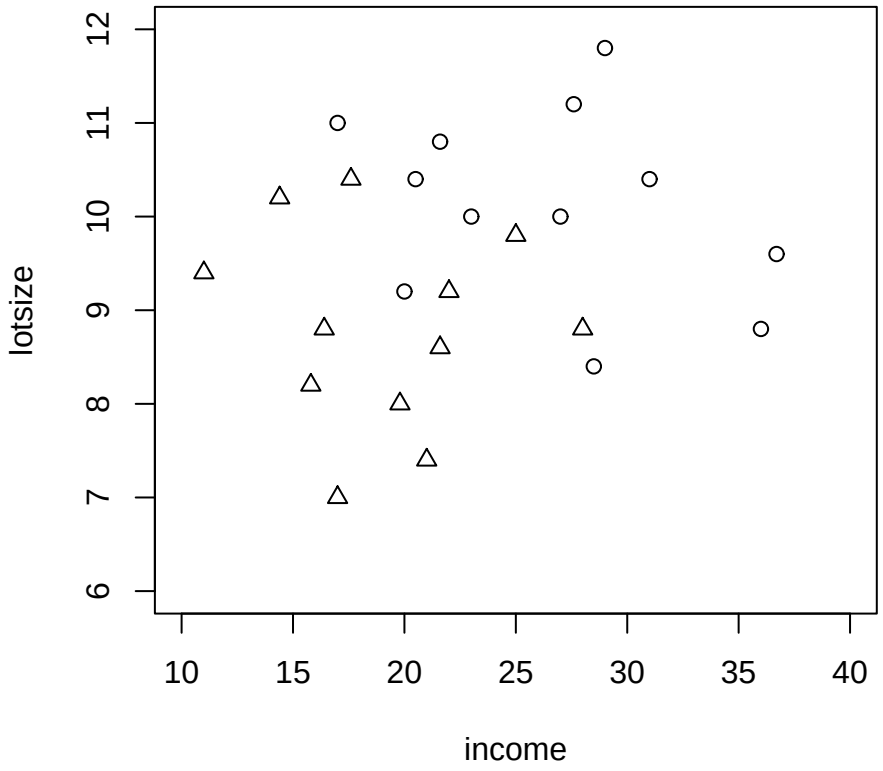

Figure 5.6: Lotsize vs Income : JW lawnmower data

$$
\begin{aligned}
R &\longrightarrow \boldsymbol{b} = \begin{bmatrix} -0.1453404 \\ -0.7590457 \end{bmatrix} \\
\boldsymbol{b} &= S_p^{-1}(\bar{\boldsymbol{x}}_{g1} - \bar{\boldsymbol{x}}_{g2}) \qquad (Fisher) \\
S_1 &= \begin{bmatrix} 39.182652 & -1.969697 \\ & 1.020606 \end{bmatrix} \\
S_2 &= \begin{bmatrix} 22.3006061 & -0.4315152 \\ & 1.1160606 \end{bmatrix} \\
S_p &= \frac{S_1 + S_2}{2} \qquad (n_1 = n_2) \\
S_p &= \begin{bmatrix} 30.741629 & -1.200606 \\ & 1.068333 \end{bmatrix} \\
S_p^{-1} &= \frac{\begin{bmatrix} 1.068333 & 1.200606 \\ & 30.741629 \end{bmatrix}}{|30.741629 \times 1.068333 - 1.200606^2| \longrightarrow 31.400842} \\
&= \begin{bmatrix} 0.034022 & 0.038235 \\ & 0.979007 \end{bmatrix} \\
\boldsymbol{d} &= \overline{\boldsymbol{x}}_{g1} - \overline{\boldsymbol{x}}_{g2} = \begin{bmatrix} 26.49167 & -19.13333 \\ 10.13333 & -8.816667 \\ (G1) & (G2) \end{bmatrix} \\
&= \begin{bmatrix} 7.358337 \\ 1.316666 \end{bmatrix} \\
\boldsymbol{b} = S^{-1}\boldsymbol{d} &= \begin{bmatrix} .034022 & .038235 \\ .038235 & .9790076 \end{bmatrix} \begin{bmatrix} 7.358337 \\ 1.316666 \end{bmatrix} \\
\boldsymbol{b} &= \begin{bmatrix} 0.250345 & +0.050343 \\ 0.281346 & +1.289025 \end{bmatrix} = \begin{bmatrix} 0.300688 \\ 1.570371 \end{bmatrix} \\
R &\longrightarrow \boldsymbol{b} = \begin{bmatrix} -0.1453404 \\ -0.7590457 \end{bmatrix} \\
\frac{0.300688}{1.570371} &= 0.191476, \text{ (Fisher)} \\
\frac{0.1453404}{0.7590457} &= 0.191478, \ (R)
\end{aligned}
$$

i.e., the two methods yield the same discriminant function. Brief inspection of Figure 5.6 yields an approximate value for the slope of the discriminant function as

$$\frac{L}{I} \approx \frac{12-6}{40-10} = \frac{6}{30} = 0.2$$

in agreement with the estimated discriminant function slope of 0.19.

R Output

```
> dat <- read.table("lawn.dat",header=T)
```

```
> dat
   own income lotsize
1    1   20.0     9.2
2    1   28.5     8.4
3    1   21.6    10.8
4    1   20.5    10.4
5    1   29.0    11.8
6    1   36.7     9.6
7    1   36.0     8.8
8    1   27.6    11.2
9    1   23.0    10.0
10   1   31.0    10.4
11   1   17.0    11.0
12   1   27.0    10.0
13   2   25.0     9.8
14   2   17.6    10.4
15   2   21.6     8.6
16   2   14.4    10.2
17   2   28.0     8.8
18   2   16.4     8.8
19   2   19.8     8.0
20   2   22.0     9.2
21   2   15.8     8.2
22   2   11.0     9.4
23   2   17.0     7.0
24   2   21.0     7.4
> y <- cbind(dat[,2],dat[,3])
> o <- factor(dat[,1])
> print(cbind(o,y))
      o
 [1,] 1 20.0  9.2
 [2,] 1 28.5  8.4
 [3,] 1 21.6 10.8
 [4,] 1 20.5 10.4
 [5,] 1 29.0 11.8
 [6,] 1 36.7  9.6
 [7,] 1 36.0  8.8
 [8,] 1 27.6 11.2
 [9,] 1 23.0 10.0
[10,] 1 31.0 10.4
[11,] 1 17.0 11.0
[12,] 1 27.0 10.0
[13,] 2 25.0  9.8
[14,] 2 17.6 10.4
[15,] 2 21.6  8.6
[16,] 2 14.4 10.2
```

```
[17,] 2 28.0  8.8
[18,] 2 16.4  8.8
[19,] 2 19.8  8.0
[20,] 2 22.0  9.2
[21,] 2 15.8  8.2
[22,] 2 11.0  9.4
[23,] 2 17.0  7.0
[24,] 2 21.0  7.4
> library(MASS)
> lda(o~y)
Call:
lda.formula(o ~ y)

Prior probabilities of groups:
  1   2
0.5 0.5

Group means:
        y1        y2
1 26.49167 10.133333
2 19.13333  8.816667

Coefficients of linear discriminants:
          LD1
y1 -0.1453404
y2 -0.7590457
> cov(y)
          [,1]     [,2]
[1,] 43.529837 1.379022
[2,]  1.379022 1.474130
> y1 <- y[o==1]
> y2 <- y[o==2]
> dim(y1) <- c(12,2)
> y1
      [,1] [,2]
 [1,] 20.0  9.2
 [2,] 28.5  8.4
 [3,] 21.6 10.8
 [4,] 20.5 10.4
 [5,] 29.0 11.8
 [6,] 36.7  9.6
 [7,] 36.0  8.8
 [8,] 27.6 11.2
 [9,] 23.0 10.0
[10,] 31.0 10.4
[11,] 17.0 11.0
```

```
[12,] 27.0 10.0
> cov(y1)
          [,1]      [,2]
[1,] 39.182652 -1.969697
[2,] -1.969697  1.020606
> dim(y2) <- c(12,2)
> y2
      [,1] [,2]
 [1,] 25.0  9.8
 [2,] 17.6 10.4
 [3,] 21.6  8.6
 [4,] 14.4 10.2
 [5,] 28.0  8.8
 [6,] 16.4  8.8
 [7,] 19.8  8.0
 [8,] 22.0  9.2
 [9,] 15.8  8.2
[10,] 11.0  9.4
[11,] 17.0  7.0
[12,] 21.0  7.4
> cov(y2)
           [,1]       [,2]
[1,] 22.3006061 -0.4315152
[2,] -0.4315152  1.1160606
> plot(y2,y1,type="n",ylim=c(6,12),xlim=c(10,40),xlab="income",ylab="lotsize")
> points(y1,pch=1)
> points(y2,pch=2)
> lawn.lda <- lda(o~y)
> plot(lawn.lda)
> lawn.lda$svd
[1] 5.067684
> coeff <- lawn.lda$scaling
> coeff <- as.matrix(coeff)
> ym <- as.matrix(y)
> scores <-  ym %*% coeff
> #scores
> scores1 <-  y1 %*% coeff
> scores1

              LD1
   [1,]  -9.890029
   [2,] -10.518186
   [3,] -11.337046
   [4,] -10.873554
   [5,] -13.171611
   [6,] -12.620832
```

```
   [7,] -11.911857
   [8,] -12.512707
   [9,] -10.933286
  [10,] -12.399628
  [11,] -10.820290
  [12,] -11.514648
> scores2 <-  y2 %*% coeff
> scores2

              LD1
   [1,] -11.072158
   [2,] -10.452066
   [3,]  -9.667146
   [4,]  -9.835168
   [5,] -10.749134
   [6,]  -9.063185
   [7,]  -8.950106
   [8,] -10.180709
   [9,]  -8.520553
  [10,]  -8.733774
  [11,]  -7.784107
  [12,]  -8.669087
> cor(scores,ym)
          [,1]       [,2]
LD1 -0.7761805 -0.7547179
> cor(scores1,y1)
          [,1]       [,2]
LD1 -0.6773521 -0.4880848
> cor(scores2,y2)
          [,1]       [,2]
LD1 -0.6112449 -0.7356054
> lawn.lda$means
        y1        y2
1 26.49167 10.133333
2 19.13333  8.816667
> p <- 2
> k <- 2
> S <- (cov(y1)+cov(y2))/2
> print(S)
          [,1]      [,2]
[1,] 30.741629 -1.200606
[2,] -1.200606  1.068333
> print(log(det(S)))
[1] 3.446835
> S1 <- cov(y1)
> print(log(det(S1)))
```

```
[1] 3.586579
> S2 <- cov(y2)
> print(log(det(S2)))
[1] 3.206909
> sn1 <- length(y1[,1]) -1
> sn2 <- length(y2[,1]) -1
> sn <- sn1 +sn2
> M <- log(det(S)) * sn - sn1*log(det(S1)) - sn2 * log(det(S2))
> print(M)
[1] 1.101993
> print(length(ym))
[1] 48
> print(length(y1))
[1] 24
> print(length(y2))
[1] 24
> f <- ( 2*p*p + 3 * p -1)/(6*(p+1)*(k-1))
> print(f)
[1] 0.7222222
> CM1 <- 1 - f *((1/sn1 + 1/sn2) - 1/sn)
> ch1 <- M*CM1
> print(ch1)
[1] 0.9934638
> df <- (k-1)*p*(p+1)/2
> print(df)
[1] 3
```

5.5 Centroids

The centroids $\bar{y}_1$ and $\bar{y}_2$ are simply defined by

$\bar{y}_1 = \boldsymbol{b}'\bar{\boldsymbol{x}}_{G1} \qquad \bar{y}_2 = \boldsymbol{b}'\bar{\boldsymbol{x}}_{G2}$

where the notation $\bar{\boldsymbol{x}}_{G1}$ denotes the first group means for x_1 and x_2.

5.6 Classification

When the groups sizes are equal, ie, when $n_1 = n_2$ then the cross–over point can be defined in terms of a the discriminant function via

$$m = \frac{\bar{y}_1 + \bar{y}_2}{2}.$$

Thus, to classify a new observation $\boldsymbol{x}_{new}$, choose

$G1$ if $y_{new} = \boldsymbol{b}'\boldsymbol{x}_{new} > m$
and $G2$ if $y_{new} = \boldsymbol{b}'\boldsymbol{x}_{new} < m$

assuming that $\bar{y}_1 > \bar{y}_2$.

Check:

$\boldsymbol{x}_{new} = \bar{\boldsymbol{x}}_{g1} \longrightarrow y_{new} = \bar{y}_1 \longrightarrow G1$

$\boldsymbol{x}_{new} = \bar{\boldsymbol{x}}_{g2} \longrightarrow y_{new} = \bar{y}_2 \longrightarrow G2$

5.6.1 JW data

This analysis relates to the problem described in Section 5.3, ie, the data from JW P556 :

Fisher's DF gives

$$\begin{aligned}
\bar{y}_1 &= -2(\bar{\boldsymbol{x}}_{g1}) = -2(3) = -6 \\
\bar{y}_2 &= -2(\bar{\boldsymbol{x}}_{g2}) = -2(5) = -10 \\
m &= \frac{-6-10}{2} = -8 \\
\bar{y}_1 &= -6 > m \longrightarrow G1 \\
\bar{y}_2 &= -10 < m \longrightarrow G2
\end{aligned}$$

Exercise:

Repeat using R discriminant function
i.e. $\boldsymbol{b} = \binom{1}{0}$.

5.6.2 Classifying a new observation

JW data (Lawn mower)

$$\begin{aligned}
b &= \begin{pmatrix} 0.300688 \\ 1.570371 \end{pmatrix} \qquad \text{Fisher} \\
\bar{y}_1 &= \boldsymbol{b}'\bar{\boldsymbol{x}}_{g1} = \begin{pmatrix} 0.300688 \\ 1.570371 \end{pmatrix}' \begin{pmatrix} 26.49167 \\ 10.13333 \end{pmatrix} \\
\bar{y}_1 &= (7.965729 + 15.913088) = 23.878817 \\
\bar{y}_2 &= (0.300688, 1.570371) \begin{pmatrix} 19.13333 \\ 8.816667 \end{pmatrix} \\
&= (5.753163 + 13.845438) = 19.598601 \\
m &= 21.738709
\end{aligned}$$

$$\begin{aligned}
y_{new} &> 21.738 \longrightarrow G1(\bar{y}_1) \\
y_{new} &< 21.738 \longrightarrow G2(\bar{y}_2)
\end{aligned}$$

$\bar{y}_2$	m	$\bar{y}_1$	type
-10	-8	-6	(Fisher's disc. fn)
-2	0	2	(centered on 0)
1	0	-1	R (histogram)
			Figure 5.7.

5.6.3 Example

Test outliers from scatter plot

$$
\begin{aligned}
G1 \quad &: \quad x_1 = 36.7 \qquad x_2 = 9.6 \\
G2 \quad &: \quad x_1 = 11 \qquad x_2 = 9.4 \\
&G1 \\
y_{new} \quad &= \quad (0.300688, 1.570371)\begin{pmatrix} 36.7 \\ 9.6 \end{pmatrix} \\
&= \quad (11.0352504 + 15.075562) = 26.11 > 21.73 \ \therefore \ G1 \text{ as expected} \\
&G2 \\
y_{new} \quad &= \quad (.300688, 1.570371)\begin{pmatrix} 11 \\ 9.4 \end{pmatrix} \\
&= \quad (3.307568 + 14.761487) = 18.07 < 21.73 \ \therefore \ (G2)
\end{aligned}
$$

Exercise * Verify these results using the R formulation. *

5.7 R implementation

A histogram of group membership can be produced in R via the command

`plot ('lda.object')`

where 'lda.object' is the result of a previous `lda` command.

An example for the small JW data set in Section 5.3 is shown in Figure 5.7.

5.7.1 Example 1

JW data p556:

Compare with the plot in Figure 5.5.

Note that G1 and G2 are reversed in the histogram formulation in R.

Since for R, $\boldsymbol{b} = \binom{1}{0}$, but for Fisher $\boldsymbol{b} = \binom{-2}{0}$ the signs are reversed and the value of the DF is halved. It can be noted from Figure 5.7 that on the DF scale the centroids are at +1 and -1. This can be seen to agree with the original variables, once the R DF is used rather than Fishers DF. Alternatively, the pattern can be worked directly from the R DF. Thus we have then the derivation given in Table 5.1.

5.7.2 Example 2

For the *JW* lawn mower data we have the results as shown in Table 5.2.

$\bar{y}_2$	m	$\bar{y}_1$	type
5	4	3	(R disc. fn)
1	0	-1	(centered on 0)
1	0	-1	R (histogram)
			Figure 5.7.

Table 5.1: Group membership for JW small data set : R formulation

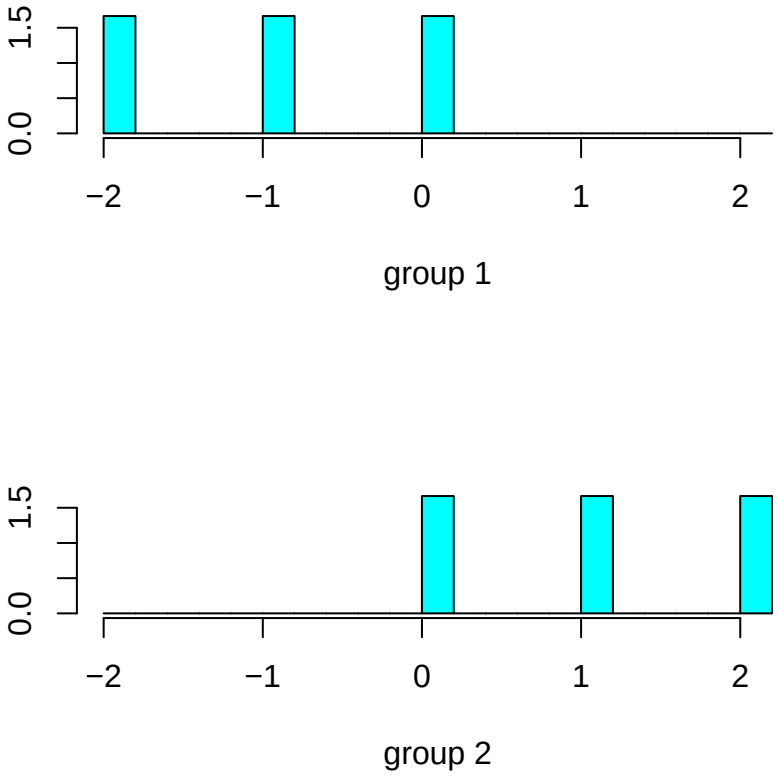

Figure 5.7: Histogram of group membership : JW small data set

	R	Fisher
$\boldsymbol{b}$	$\binom{-0.145}{-0.759}$	$\binom{0.300}{1.570}$

Table 5.2: Discriminant functions – lawn mower data

Therefore the R discriminant function is sign reversed and halved (again).

The rules for Group membership are given in Table 5.3.

G2	M	G1
19.599	21.739	23.879 (D.F $\rightarrow$)
-2.14	0	2.14 (Fisher)
1.07	0	-1.07 (R)

Table 5.3: Group membership : JW lawn mower data

Compare Table 5.3 with the histogram from R, as shown in Figure 5.8.

5.8 Classifying a new observation:

JW P556

small data set

Assume that the new observation is given as $X_{\text{new}} = (2, 7)$.

Fisher

Using FISHER's discriminant function,

$Y_{\text{new}} = -2(2) = -4 > -8 \rightarrow G1$

R

Using `lda` in R gives

R predict lda $\rightarrow 1$

NOTE LD1 is (-2), since $\boldsymbol{b} = (1, 0)'$.

R Output

```
> dat <- read.table("jwlda.dat",header=T)
> dat
  group x1 x2
1     1  3  7
2     1  2  4
3     1  4  7
```

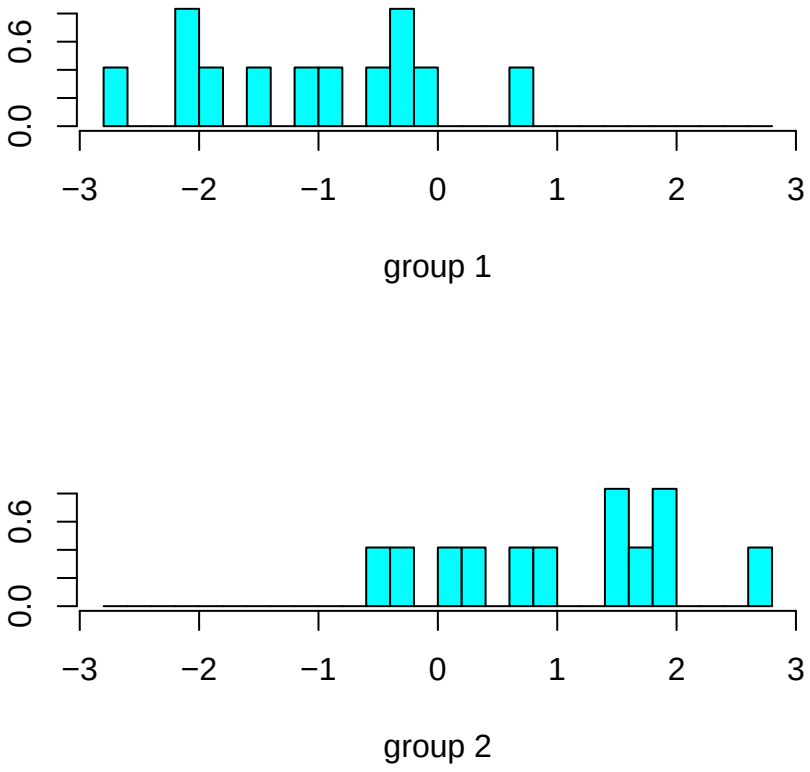

Figure 5.8: Histogram of group membership : JW data–lawn mowers

```
4     2  6  9
5     2  5  7
6     2  4  8
> x <- cbind(dat[,2],dat[,3])
> group <- factor(dat[,1])
> print(cbind(group,x))
     group
[1,]     1 3 7
[2,]     1 2 4
[3,]     1 4 7
[4,]     2 6 9
[5,]     2 5 7
[6,]     2 4 8
> library(MASS)
> lda(group~x)
Call:
lda.formula(group ~ x)

Prior probabilities of groups:
  1   2
0.5 0.5

Group means:
  x1 x2
1  3  6
2  5  8

Coefficients of linear discriminants:
             LD1
x1  1.000000e+00
x2 -1.364468e-16
> cov(x)
     [,1] [,2]
[1,]    2  2.0
[2,]    2  2.8
> x1 <- x[group==1]
> x2 <- x[group==2]
> dim(x1) <- c(3,2)
> x1
     [,1] [,2]
[1,]    3    7
[2,]    2    4
[3,]    4    7
> cov(x1)
     [,1] [,2]
[1,]  1.0  1.5
```

```
[2,]  1.5  3.0
> dim(x2) <- c(3,2)
> x2
     [,1] [,2]
[1,]    6    9
[2,]    5    7
[3,]    4    8
> cov(x2)
     [,1] [,2]
[1,]  1.0  0.5
[2,]  0.5  1.0
> jw.lda <- lda(x,group)
> plot(jw.lda)
> predict.lda(jw.lda,c(2,7))
$class
[1] 1
Levels:  1 2

$posterior
             1          2
[1,] 0.9820138 0.01798621

$x

       LD1
  [1,]  -2
```

5.8.1 Exercise

JW, Lawn Mower Data

Verify that $X_{new} = (29.0, 11.8)'$ gives G1, while $X_{new} = (17, 7)'$ gives G2, using both Fisher and R.

5.9 Importance of original variables in a discriminant function

(S p253)

Even a good discriminant may contain redundant variables. To assess the importance of each variable in the discriminant function, the correlation between discriminant scores and individual original variables is taken as a measure of the importance of those input variables. The only caveat is that such measures are highly influenced by multicollinearity between the original variables. Some packages provide these measures as part of their standard output.

5.10 Tests of Assumptions in `lda`

S p263, 264.

Apart from linearity, the main assumptions of `lda` are

1. MVN errors
2. Consequent equality of covariance matrices.

The test assumption can be assessed as per Ch2 although symmetry is probably more important. If normality cannot be induced by transformation or if the data are seriously non normal, e.g., categorical, then the alternative of logistic regression should be used. It is worth pointing out that if all the assumptions are satisfied, `lda` is the optimal procedure and so should be used. So use 'qqbeta' to assess overall MVN.

For the second assumption there is a test of equality of covariance matrices, Box's M test. Violation of this assumption can affect significance tests of classification results. The significance level can be inflated (false positives) when # of variables is large and the sample sizes of the groups differ. Quadratic methods can be used if $\sum_1 \neq \sum_2$ but a large # of parameters are involved and `lda` is thus superior for small sample sizes. Box's test also will reject $H_0 : \sum_i = \sum \ \forall i$ even for small differences if the sample sizes are large. Overall `lda` is robust to both the assumptions of MVN and equality of covariance matrices, esp if sample sizes are equal.

5.11 Box's M test

Morrison p252

N.B.
$n_i = N_i - 1$!
The null hypothesis is $H_0 : \sum_1 = \sum_2 = \sum$
The test requires the following :

$$\begin{aligned}
\mathbf{S} &= \frac{1}{\sum_i n_i} \sum_{i=1}^{k} n_i \times \mathbf{S}_i \\
M &= \sum_i n_i \ln |\mathbf{S}| - \sum_{i=1}^{k} n_i \ln |\mathbf{S}_i| \\
C^{-1} &= 1 - \frac{2p^2 + 3p - 1}{6(p+1)(k-1)} \left(\sum_{i=1}^{k} \frac{1}{n_i} - \frac{1}{\sum n_i} \right) \\
MC^{-1} &\sim \chi^2_(k-1)(p)(p+1)/2 \ \text{ as } n_i \to \infty
\end{aligned}$$

a as long as $K, p < 5 \qquad n_i = 20$, otherwise use F distribution to approximate χ^2.

This test has been implemented in R.

See the R Output for the lawn mower data. For this data set, $MC^{-1} = 0.993$ on 3 df, and since $\chi^2_{3,5\%} = 7.8$ we accept the assumption of equality of covariance matrices.

5.12 Other tests

There are several other tests from other multivariate techniques that are useful in two group discriminant analysis.

(a) $H_0 : \mu_1 = \mu_2$

(multivariate t test)

This is a special case of MANOVA (Multivariate Analysis of Variance). The importance of the test is in deciding if discriminant functions are worthwhile. If H_0 is true than `lda` is wasted. If H_0 is false than some form of discriminant can be found.

(b) Test on the significance of the discriminant function itself.

This is simply a test on the eigenvalue(s) and will be covered in multiple discriminant analysis. The procedure tests if the remaining eigenvalues are all equal.

(None of these tests appear to be implemented in R), apart from MANOVA of course which is the topic after multiple discriminant analysis. These tests can be formed from the R output of the routine for canonical correlation, `cancor`.

5.13 R implementation

The correlations between the original variables and the scores obtained from the discriminant function are given for the lawn mower data in Table 5.4. Note that the correlations are given overall and within group, as per the R output.

	Income	Lot size
overall	-0.776	-0.754
g1 (own)	-0.677	-0.488
g2 (non–own)	-0.611	-0.735

Table 5.4: Correlations between discriminant function and original variables

This enables the importance of each variable in the discriminant function to be assessed.

Checking the actual value of the score for the discriminant function gives :
Unit 1 (G1)

LD = (-0.1453404, -.07590457)'

Income = 20 ls = 9.2

Therefore Score = (-0.1453404 (20) - 0.7590457 (9.2))

= -2.906 808 - 6.983220

= -9.890028

as per the R output.

**You should now attempt Workshop 5,
and then Assignment 1.**

5.14 Multiple Group Discriminant Analysis

Two examples are given showing the extremes possibilities when more than two groups are involved. (S p288)

5.15 Example 1

(After S p288) Panel I

This data set involves 4 groups but only 1 discriminant function is needed to classify the data properly. This is shown in Figure 5.9.

Remember that it is the projection of the data onto the discriminant that provides the classifying rule. The plot in discriminant space (Figure 5.10) clearly shows that the 2nd potential discriminant LD2 is redundant.

	LD1	LD2
% variance =	99.99,	0.01

R Output (Example 1)

```
> dat <- read.table("s1.dat",header=T)
> dat
   group x1 x2
1      1  2  4
2      1  4  3
3      1  3  3
4      1  3  2
5      2  6  8
6      2  8  6
7      2  9  7
8      2  7  8
9      3 12 14
10     3 14 13
11     3 13 12
12     3 12 12
13     4 16 18
14     4 19 16
```

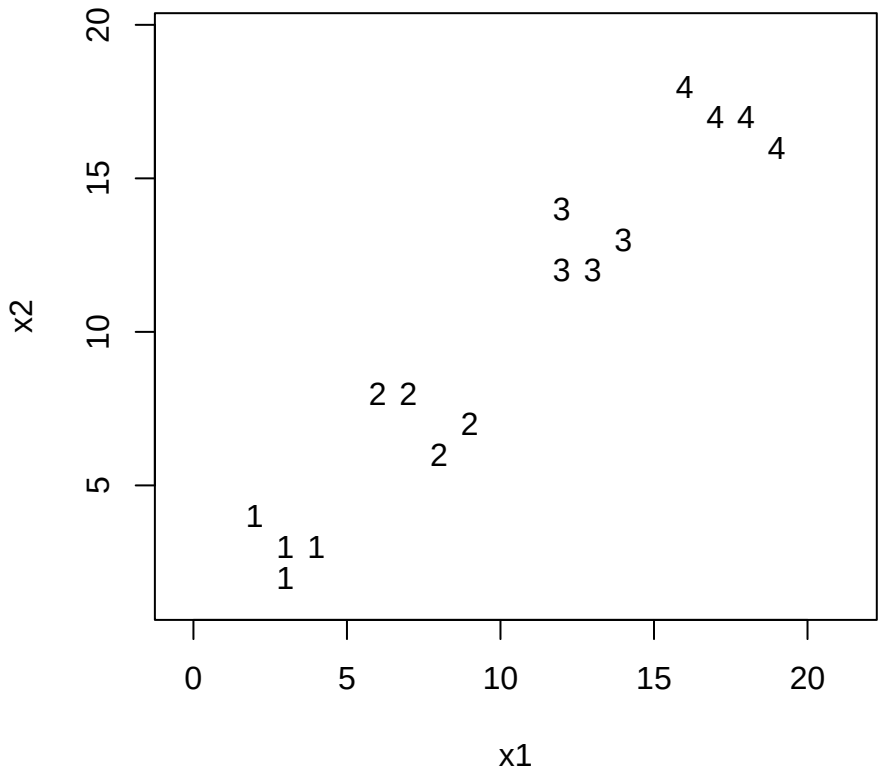

Figure 5.9: Sharma example Panel I (p288)

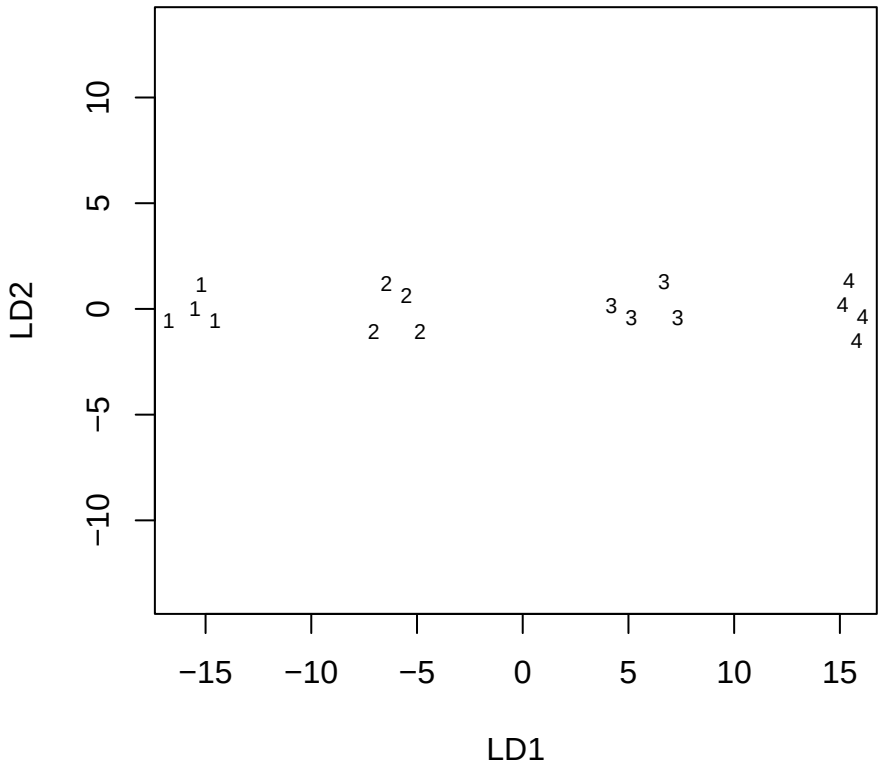

Figure 5.10: Discriminant plot : Sharma example Panel I (p288)

```
15     4 18 17
16     4 17 17
> y <- cbind(dat[,2],dat[,3])
> o <- factor(dat[,1])
> print(cbind(o,y))
      o
 [1,] 1  2  4
 [2,] 1  4  3
 [3,] 1  3  3
 [4,] 1  3  2
 [5,] 2  6  8
 [6,] 2  8  6
 [7,] 2  9  7
 [8,] 2  7  8
 [9,] 3 12 14
[10,] 3 14 13
[11,] 3 13 12
[12,] 3 12 12
[13,] 4 16 18
[14,] 4 19 16
[15,] 4 18 17
[16,] 4 17 17
> library(MASS)
> lda(o~y)
Call:
lda.formula(o ~ y)

Prior probabilities of groups:
   1    2    3    4
0.25 0.25 0.25 0.25

Group means:
     y1    y2
1  3.00  3.00
2  7.50  7.25
3 12.75 12.75
4 17.50 17.00

Coefficients of linear discriminants:
         LD1        LD2
y1 0.9457708 -0.5577360
y2 1.2429010  0.5718173

Proportion of trace:
   LD1    LD2
0.9999 0.0001
```

```
> grp <- factor(dat[,1])
> y
      [,1] [,2]
 [1,]    2    4
 [2,]    4    3
 [3,]    3    3
 [4,]    3    2
 [5,]    6    8
 [6,]    8    6
 [7,]    9    7
 [8,]    7    8
 [9,]   12   14
[10,]   14   13
[11,]   13   12
[12,]   12   12
[13,]   16   18
[14,]   19   16
[15,]   18   17
[16,]   17   17
> x1 <- y[grp==1]
> x1
[1] 2 4 3 3 4 3 3 2
> x2 <- y[grp==2]
> x2
[1] 6 8 9 7 8 6 7 8
> x3 <- y[grp==3]
> x3
[1] 12 14 13 12 14 13 12 12
> x4 <- y[grp==4]
> x4
[1] 16 19 18 17 18 16 17 17
> dim(x1) <- c(4,2)
> x1
     [,1] [,2]
[1,]    2    4
[2,]    4    3
[3,]    3    3
[4,]    3    2
> dim(x2) <- c(4,2)
> x2
     [,1] [,2]
[1,]    6    8
[2,]    8    6
[3,]    9    7
[4,]    7    8
> dim(x3) <- c(4,2)
```

```
> x3
     [,1] [,2]
[1,]   12   14
[2,]   14   13
[3,]   13   12
[4,]   12   12
> dim(x4) <- c(4,2)
> x4
     [,1] [,2]
[1,]   16   18
[2,]   19   16
[3,]   18   17
[4,]   17   17
> eqscplot(c(1,20),c(1,20),type="n",xlab="x1",ylab="x2")
> points(x1,pch="1")
> points(x2,pch="2")
> points(x3,pch="3")
> points(x4,pch="4")
> s1.lda <- lda(y,o)
> plot(s1.lda)
> predict(s1.lda)
$class
 [1] 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4
Levels: 1 2 3 4

$posterior
                   1             2             3             4
 [1,]  1.000000e+00  2.719924e-19  8.574883e-97 5.668451e-207
 [2,]  1.000000e-00  1.512955e-16  6.982263e-91 3.786887e-198
 [3,]  1.000000e+00  1.748874e-20  1.294732e-99 6.004922e-211
 [4,]  1.000000e+00  1.299849e-25 3.625059e-111 1.008731e-227
 [5,]  2.003136e-18  1.000000e+00  2.364084e-33 2.255292e-106
 [6,]  4.845147e-16  1.000000e-00  1.303655e-36 6.121886e-112
 [7,]  4.162687e-25  1.000000e+00  2.157300e-25  1.974508e-91
 [8,]  2.315491e-22  1.000000e+00  1.473709e-28  1.644032e-97
 [9,] 1.659439e-107  2.059918e-35  1.000000e+00  5.229925e-18
[10,] 2.037948e-113  1.407185e-38  1.000000e-00  4.290870e-15
[11,]  3.925313e-93  2.328622e-27  1.000000e+00  2.201504e-24
[12,]  2.116853e-84  1.451603e-22  1.000000e+00  1.882612e-28
[13,] 1.597489e-208 3.639647e-100  1.324954e-20  1.000000e+00
[14,] 2.257236e-213 1.839371e-103  2.301758e-22  1.000000e+00
[15,] 2.391222e-217 3.030474e-106  1.614920e-23  1.000000e+00
[16,] 1.507978e-204  2.209114e-97  1.888468e-19  1.000000e+00

$x
```

```
              LD1          LD2
   [1,] -15.200904  1.135560035
   [2,] -14.552264 -0.551729360
   [3,] -15.498035  0.006006685
   [4,] -16.740936 -0.565810621
   [5,]  -6.446217  1.191885077
   [6,]  -7.040478 -1.067221623
   [7,]  -4.851806 -1.053140363
   [8,]  -5.500446  0.634149032
   [9,]   6.685814  1.276372640
  [10,]   7.334454 -0.410916755
  [11,]   5.145782 -0.424998015
  [12,]   4.200012  0.132738029
  [13,]  15.440501  1.332697682
  [14,]  15.792011 -1.484145063
  [15,]  16.089141 -0.354591713
  [16,]  15.143371  0.203144332

> s1.lda$svd
[1] 27.1710210  0.2056504
> coeff <- s1.lda$scaling
> coeff
          LD1        LD2
[1,] 0.9457708 -0.5577360
[2,] 1.2429010  0.5718173
> scores <- y%*%coeff
> scores

              LD1         LD2
   [1,]  6.863146  1.17179713
   [2,]  7.511786 -0.51549226
   [3,]  6.566015  0.04224378
   [4,]  5.323114 -0.52957352
   [5,] 15.617833  1.22812217
   [6,] 15.023572 -1.03098453
   [7,] 17.212244 -1.01690327
   [8,] 16.563604  0.67038613
   [9,] 28.749863  1.31260974
  [10,] 29.398504 -0.37467966
  [11,] 27.209832 -0.38876092
  [12,] 26.264061  0.16897513
  [13,] 37.504551  1.36893478
  [14,] 37.856061 -1.44790797
  [15,] 38.153191 -0.31835462
  [16,] 37.207420  0.23938143
> cor(scores)
```

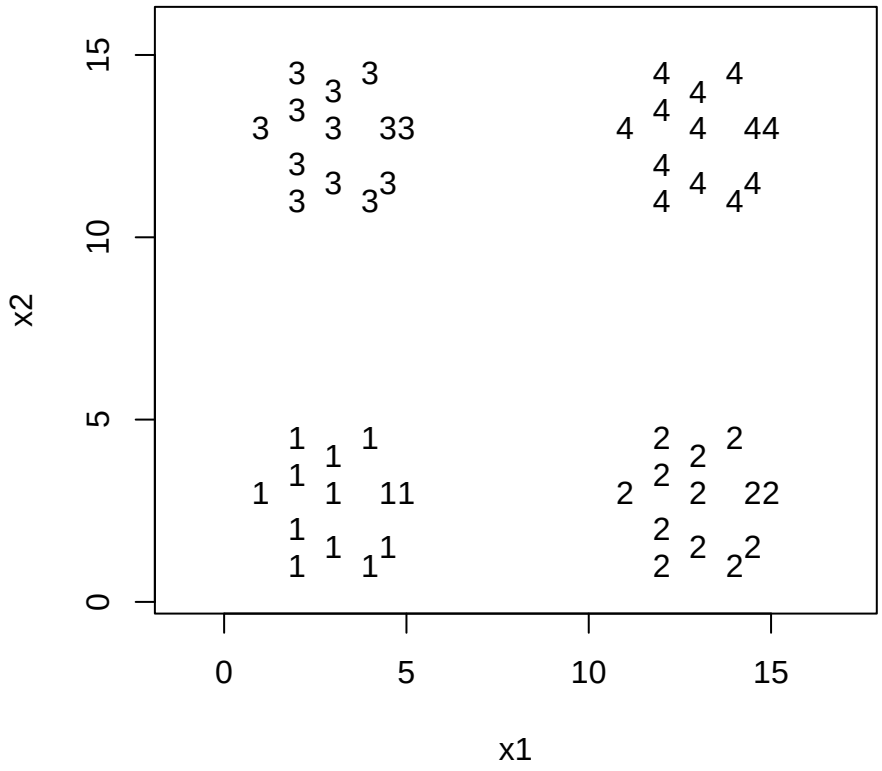

Figure 5.11: Sharma example Panel II (p288)

```
              LD1           LD2
LD1  1.000000e+00 -6.497992e-16
LD2 -6.497992e-16  1.000000e+00
> cor(scores,y)
          [,1]      [,2]
LD1  0.9873784 0.9922609
LD2 -0.1583789 0.1241703
```

5.16 Example 2

S p288 Panel II data S p290

Again there are 4 groups but now **both** potential discriminants are required to properly classify the population. This is shown in the data plot given in Figure 5.11.

The plot in discriminant space (Figure 5.12) shows the necessity for **two** discriminant functions, with LD1 separating g4 and g1 while LD2 separates g3 and g2.

This can be verified from the data plot in variable space (Figure 5.11), by imposing

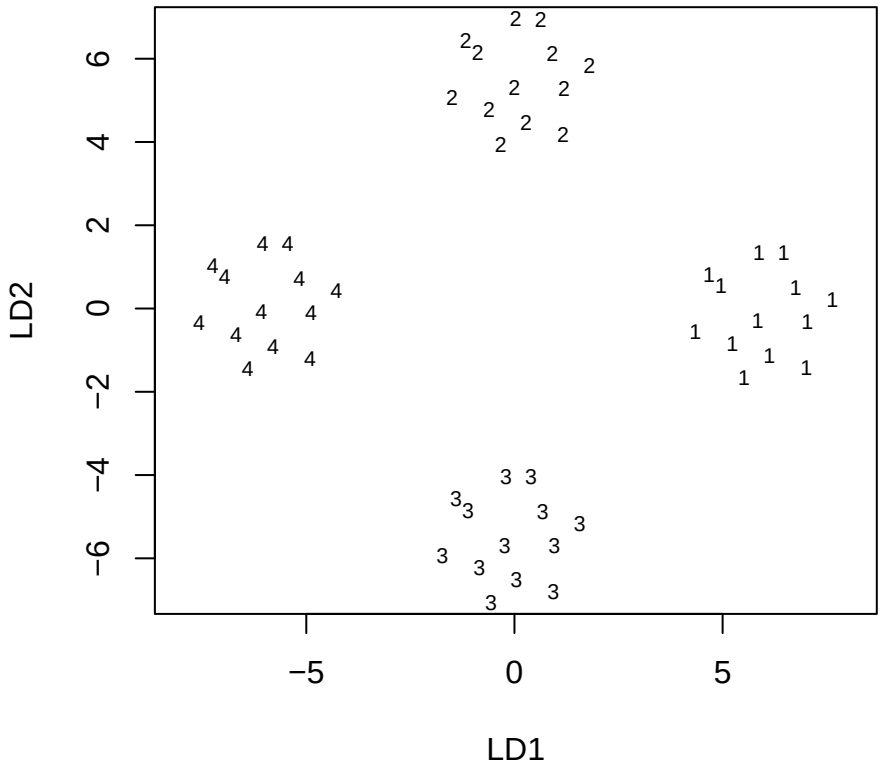

Figure 5.12: Discriminant plot : Sharma example Panel II (p288)

the directions specified by LD1 and LD2. Again the projection of the data onto the discriminant axis (Figure 5.12) is the key to classification. It can be seen that

	LD1	LD2
% variance =	54.03	45.97

and so both discriminants are approximately of equal value, as expected from the plot of the data in Figure 5.11.

R Output (Example 2)

```
> dat <- read.table("s2.dat",header=T)
> dat
   group   x1   x2
1      1  1.0  3.0
2      1  2.0  1.0
3      1  4.0  1.0
4      1  5.0  3.0
5      1  4.0  4.5
6      1  2.0  4.5
7      1  3.0  1.5
8      1  4.5  1.5
9      1  4.5  3.0
10     1  3.0  4.0
11     1  2.0  3.5
12     1  2.0  2.0
13     1  3.0  3.0
14     2 11.0  3.0
15     2 12.0  1.0
16     2 14.0  1.0
17     2 15.0  3.0
18     2 14.0  4.5
19     2 12.0  4.5
20     2 13.0  1.5
21     2 14.5  1.5
22     2 14.5  3.0
23     2 13.0  4.0
24     2 12.0  3.5
25     2 12.0  2.0
26     2 13.0  3.0
27     3  1.0 13.0
28     3  2.0 11.0
29     3  4.0 11.0
30     3  5.0 13.0
31     3  4.0 14.5
32     3  2.0 14.5
33     3  3.0 11.5
```

```
34     3  4.5 11.5
35     3  4.5 13.0
36     3  3.0 14.0
37     3  2.0 13.5
38     3  2.0 12.0
39     3  3.0 13.0
40     4 11.0 13.0
41     4 12.0 11.0
42     4 14.0 11.0
43     4 15.0 13.0
44     4 14.0 14.5
45     4 12.0 14.5
46     4 13.0 11.5
47     4 14.5 11.5
48     4 14.5 13.0
49     4 13.0 14.0
50     4 12.0 13.5
51     4 12.0 12.0
52     4 13.0 13.0
> y <- cbind(dat[,2],dat[,3])
> o <- factor(dat[,1])
> print(cbind(o,y))
      o
 [1,] 1  1.0  3.0
 [2,] 1  2.0  1.0
 [3,] 1  4.0  1.0
 [4,] 1  5.0  3.0
 [5,] 1  4.0  4.5
 [6,] 1  2.0  4.5
 [7,] 1  3.0  1.5
 [8,] 1  4.5  1.5
 [9,] 1  4.5  3.0
[10,] 1  3.0  4.0
[11,] 1  2.0  3.5
[12,] 1  2.0  2.0
[13,] 1  3.0  3.0
[14,] 2 11.0  3.0
[15,] 2 12.0  1.0
[16,] 2 14.0  1.0
[17,] 2 15.0  3.0
[18,] 2 14.0  4.5
[19,] 2 12.0  4.5
[20,] 2 13.0  1.5
[21,] 2 14.5  1.5
[22,] 2 14.5  3.0
[23,] 2 13.0  4.0
```

```
[24,] 2 12.0  3.5
[25,] 2 12.0  2.0
[26,] 2 13.0  3.0
[27,] 3  1.0 13.0
[28,] 3  2.0 11.0
[29,] 3  4.0 11.0
[30,] 3  5.0 13.0
[31,] 3  4.0 14.5
[32,] 3  2.0 14.5
[33,] 3  3.0 11.5
[34,] 3  4.5 11.5
[35,] 3  4.5 13.0
[36,] 3  3.0 14.0
[37,] 3  2.0 13.5
[38,] 3  2.0 12.0
[39,] 3  3.0 13.0
[40,] 4 11.0 13.0
[41,] 4 12.0 11.0
[42,] 4 14.0 11.0
[43,] 4 15.0 13.0
[44,] 4 14.0 14.5
[45,] 4 12.0 14.5
[46,] 4 13.0 11.5
[47,] 4 14.5 11.5
[48,] 4 14.5 13.0
[49,] 4 13.0 14.0
[50,] 4 12.0 13.5
[51,] 4 12.0 12.0
[52,] 4 13.0 13.0
> library(MASS)
> lda(o~y)
Call:
lda.formula(o ~ y)

Prior probabilities of groups:
   1    2    3    4
0.25 0.25 0.25 0.25

Group means:
         y1        y2
1  3.076923  2.730769
2 13.076923  2.730769
3  3.076923 12.730769
4 13.076923 12.730769

Coefficients of linear discriminants:
```

```
          LD1        LD2
y1 -0.5844154 -0.5604264
y2 -0.6076282  0.5390168

Proportion of trace:
   LD1    LD2
0.5403 0.4597
> grp <- factor(dat[,1])
> y
      [,1] [,2]
 [1,]  1.0  3.0
 [2,]  2.0  1.0
 [3,]  4.0  1.0
 [4,]  5.0  3.0
 [5,]  4.0  4.5
 [6,]  2.0  4.5
 [7,]  3.0  1.5
 [8,]  4.5  1.5
 [9,]  4.5  3.0
[10,]  3.0  4.0
[11,]  2.0  3.5
[12,]  2.0  2.0
[13,]  3.0  3.0
[14,] 11.0  3.0
[15,] 12.0  1.0
[16,] 14.0  1.0
[17,] 15.0  3.0
[18,] 14.0  4.5
[19,] 12.0  4.5
[20,] 13.0  1.5
[21,] 14.5  1.5
[22,] 14.5  3.0
[23,] 13.0  4.0
[24,] 12.0  3.5
[25,] 12.0  2.0
[26,] 13.0  3.0
[27,]  1.0 13.0
[28,]  2.0 11.0
[29,]  4.0 11.0
[30,]  5.0 13.0
[31,]  4.0 14.5
[32,]  2.0 14.5
[33,]  3.0 11.5
[34,]  4.5 11.5
[35,]  4.5 13.0
[36,]  3.0 14.0
```

```
[37,]  2.0 13.5
[38,]  2.0 12.0
[39,]  3.0 13.0
[40,] 11.0 13.0
[41,] 12.0 11.0
[42,] 14.0 11.0
[43,] 15.0 13.0
[44,] 14.0 14.5
[45,] 12.0 14.5
[46,] 13.0 11.5
[47,] 14.5 11.5
[48,] 14.5 13.0
[49,] 13.0 14.0
[50,] 12.0 13.5
[51,] 12.0 12.0
[52,] 13.0 13.0
> x1 <- y[grp==1]
> x1
 [1] 1.0 2.0 4.0 5.0 4.0 2.0 3.0 4.5 4.5 3.0 2.0 2.0 3.0 3.0 1.0 1.0 3.0 4.5 4.5
[20] 1.5 1.5 3.0 4.0 3.5 2.0 3.0
> x2 <- y[grp==2]
> x2
 [1] 11.0 12.0 14.0 15.0 14.0 12.0 13.0 14.5 14.5 13.0 12.0 12.0 13.0  3.0  1.0
[16]  1.0  3.0  4.5  4.5  1.5  1.5  3.0  4.0  3.5  2.0  3.0
> x3 <- y[grp==3]
> x3
 [1]  1.0  2.0  4.0  5.0  4.0  2.0  3.0  4.5  4.5  3.0  2.0  2.0  3.0 13.0 11.0
[16] 11.0 13.0 14.5 14.5 11.5 11.5 13.0 14.0 13.5 12.0 13.0
> x4 <- y[grp==4]
> x4
 [1] 11.0 12.0 14.0 15.0 14.0 12.0 13.0 14.5 14.5 13.0 12.0 12.0 13.0 13.0 11.0
[16] 11.0 13.0 14.5 14.5 11.5 11.5 13.0 14.0 13.5 12.0 13.0
> dim(x1) <- c(13,2)
> x1
      [,1] [,2]
 [1,]  1.0  3.0
 [2,]  2.0  1.0
 [3,]  4.0  1.0
 [4,]  5.0  3.0
 [5,]  4.0  4.5
 [6,]  2.0  4.5
 [7,]  3.0  1.5
 [8,]  4.5  1.5
 [9,]  4.5  3.0
[10,]  3.0  4.0
[11,]  2.0  3.5
```

```
[12,]  2.0  2.0
[13,]  3.0  3.0
> dim(x2) <- c(13,2)
> x2
      [,1] [,2]
 [1,] 11.0  3.0
 [2,] 12.0  1.0
 [3,] 14.0  1.0
 [4,] 15.0  3.0
 [5,] 14.0  4.5
 [6,] 12.0  4.5
 [7,] 13.0  1.5
 [8,] 14.5  1.5
 [9,] 14.5  3.0
[10,] 13.0  4.0
[11,] 12.0  3.5
[12,] 12.0  2.0
[13,] 13.0  3.0
> dim(x3) <- c(13,2)
> x3
      [,1] [,2]
 [1,]  1.0 13.0
 [2,]  2.0 11.0
 [3,]  4.0 11.0
 [4,]  5.0 13.0
 [5,]  4.0 14.5
 [6,]  2.0 14.5
 [7,]  3.0 11.5
 [8,]  4.5 11.5
 [9,]  4.5 13.0
[10,]  3.0 14.0
[11,]  2.0 13.5
[12,]  2.0 12.0
[13,]  3.0 13.0
> dim(x4) <- c(13,2)
> x4
      [,1] [,2]
 [1,] 11.0 13.0
 [2,] 12.0 11.0
 [3,] 14.0 11.0
 [4,] 15.0 13.0
 [5,] 14.0 14.5
 [6,] 12.0 14.5
 [7,] 13.0 11.5
 [8,] 14.5 11.5
 [9,] 14.5 13.0
```

```
[10,] 13.0 14.0
[11,] 12.0 13.5
[12,] 12.0 12.0
[13,] 13.0 13.0
> eqscplot(c(0,16),c(0,16),type="n",xlab="x1",ylab="x2")
> points(x1,pch="1")
> points(x2,pch="2")
> points(x3,pch="3")
> points(x4,pch="4")
> s2.lda <- lda(y,o)
> plot(s2.lda)
> predict(s2.lda)
$class
 [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 3 3 3 3 3 3 3 3 3 3 3
[39] 3 4 4 4 4 4 4 4 4 4 4 4 4
Levels: 1 2 3 4

$posterior
                 1            2            3            4
 [1,] 1.000000e-00 8.161197e-21 9.263831e-15 3.763283e-37
 [2,] 1.000000e+00 1.988274e-18 2.927940e-20 2.897748e-40
 [3,] 1.000000e-00 9.842646e-13 8.455858e-20 4.142780e-34
 [4,] 1.000000e-00 1.999978e-09 7.726480e-14 7.691823e-25
 [5,] 1.000000e-00 6.297295e-12 9.027731e-10 2.829795e-23
 [6,] 1.000000e-00 1.272092e-17 3.125958e-10 1.979355e-29
 [7,] 1.000000e-00 1.823656e-15 1.347384e-18 1.223085e-35
 [8,] 1.000000e-00 3.403452e-11 2.984957e-18 5.056851e-31
 [9,] 1.000000e-00 7.539913e-11 5.926971e-14 2.224444e-26
[10,] 1.000000e-00 6.865740e-15 1.961775e-11 6.704385e-28
[11,] 1.000000e-00 7.485496e-18 4.263045e-13 1.588411e-32
[12,] 1.000000e+00 3.378889e-18 2.146966e-17 3.610951e-37
[13,] 1.000000e-00 4.040076e-15 2.675384e-14 5.380196e-31
[14,] 4.121617e-09 1.000000e-00 7.670717e-21 9.263831e-15
[15,] 1.691786e-11 1.000000e-00 9.951423e-29 2.927940e-20
[16,] 3.417509e-17 1.000000e+00 5.805572e-34 8.455858e-20
[17,] 1.681885e-20 1.000000e-00 2.610692e-31 7.726480e-14
[18,] 5.341552e-18 1.000000e-00 9.687769e-25 9.027731e-10
[19,] 2.644254e-12 1.000000e-00 1.660596e-19 3.125958e-10
[20,] 1.844500e-14 1.000000e-00 4.992841e-30 1.347384e-18
[21,] 9.883297e-19 1.000000e+00 5.926764e-34 2.984957e-18
[22,] 4.461236e-19 1.000000e-00 5.312093e-30 5.926971e-14
[23,] 4.899302e-15 1.000000e-00 1.930906e-23 1.961775e-11
[24,] 4.493668e-12 1.000000e-00 3.848563e-22 4.263045e-13
[25,] 9.955145e-12 1.000000e-00 4.293886e-26 2.146966e-17
[26,] 8.325915e-15 1.000000e-00 4.475029e-26 2.675384e-14
[27,] 2.402028e-15 3.938308e-33 1.000000e-00 8.161197e-21
```

```
[28,] 7.599878e-10 3.035711e-25 1.000000e-00 1.988274e-18
[29,] 2.631547e-10 5.203561e-20 1.000000e-00 9.842646e-13
[30,] 2.879964e-16 1.157151e-22 1.000000e-00 1.999978e-09
[31,] 2.464848e-20 3.118329e-29 1.000000e-00 6.297295e-12
[32,] 7.118454e-20 1.819205e-34 1.000000e+00 1.272092e-17
[33,] 1.651495e-11 6.050592e-24 1.000000e-00 1.823656e-15
[34,] 7.454708e-12 5.097157e-20 1.000000e-00 3.403452e-11
[35,] 3.754360e-16 5.686957e-24 1.000000e-00 7.539913e-11
[36,] 1.134278e-18 1.564532e-30 1.000000e-00 6.865740e-15
[37,] 5.219739e-17 7.849592e-32 1.000000e+00 7.485496e-18
[38,] 1.036438e-12 7.035502e-28 1.000000e-00 3.378889e-18
[39,] 8.317304e-16 6.750716e-28 1.000000e-00 4.040076e-15
[40,] 4.927971e-26 2.402028e-15 4.121617e-09 1.000000e-00
[41,] 6.399916e-23 7.599878e-10 1.691786e-11 1.000000e-00
[42,] 4.476546e-29 2.631547e-10 3.417509e-17 1.000000e-00
[43,] 2.411047e-38 2.879964e-16 1.681885e-20 1.000000e-00
[44,] 6.553600e-40 2.464848e-20 5.341552e-18 1.000000e+00
[45,] 9.369387e-34 7.118454e-20 2.644254e-12 1.000000e-00
[46,] 1.516276e-27 1.651495e-11 1.844500e-14 1.000000e-00
[47,] 3.667371e-32 7.454708e-12 9.883297e-19 1.000000e-00
[48,] 8.337071e-37 3.754360e-16 4.461236e-19 1.000000e-00
[49,] 2.766152e-35 1.134278e-18 4.899302e-15 1.000000e-00
[50,] 1.167541e-30 5.219739e-17 4.493668e-12 1.000000e-00
[51,] 5.135863e-26 1.036438e-12 9.955145e-12 1.000000e-00
[52,] 3.446965e-32 8.317304e-16 8.325915e-15 1.000000e-00

$x

                LD1         LD2
   [1,]  7.010411852  1.41613054
   [2,]  7.641252910 -0.22232956
   [3,]  6.472422090 -1.34318244
   [4,]  4.672750213 -0.82557521
   [5,]  4.345723272  0.54337648
   [6,]  5.514554092  1.66422935
   [7,]  6.753023383 -0.51324758
   [8,]  5.876400268 -1.35388724
   [9,]  4.964957918 -0.54536199
  [10,]  5.233952799  0.83429450
  [11,]  6.122182326  1.12521252
  [12,]  7.033624676  0.31668727
  [13,]  5.841581033  0.29527766
  [14,]  1.166257753 -4.18813383
  [15,]  1.797098811 -5.82659393
  [16,]  0.628267991 -6.94744681
  [17,] -1.171403887 -6.42983958
```

```
  [18,] -1.498430827 -5.06088789
  [19,] -0.329600007 -3.94003502
  [20,]  0.908869284 -6.11751195
  [21,]  0.032246169 -6.95815161
  [22,] -0.879196182 -6.14962636
  [23,] -0.610201300 -4.76996987
  [24,]  0.278028226 -4.47905185
  [25,]  1.189470577 -5.28757710
  [26,] -0.002573067 -5.30898670
  [27,]  0.934129516  6.80629886
  [28,]  1.564970573  5.16783876
  [29,]  0.396139754  4.04698589
  [30,] -1.403532124  4.56459311
  [31,] -1.730559064  5.93354480
  [32,] -0.561728244  7.05439767
  [33,]  0.676741047  4.87692074
  [34,] -0.199882068  4.03628108
  [35,] -1.111324419  4.84480633
  [36,] -0.842329537  6.22446282
  [37,]  0.045899989  6.51538084
  [38,]  0.957342340  5.70685559
  [39,] -0.234701304  5.68544599
  [40,] -4.910024583  1.20203449
  [41,] -4.279183526 -0.43642561
  [42,] -5.448014346 -1.55727848
  [43,] -7.247686223 -1.03967125
  [44,] -7.574713163  0.32928043
  [45,] -6.405882344  1.45013331
  [46,] -5.167413053 -0.72734363
  [47,] -6.044036167 -1.56798328
  [48,] -6.955478518 -0.75945803
  [49,] -6.686483637  0.62019845
  [50,] -5.798254110  0.91111647
  [51,] -4.886811759  0.10259122
  [52,] -6.078855403  0.08118162

> s2.lda$svd
[1] 17.54973 16.18643
> coeff <- s2.lda$scaling
> coeff
            LD1        LD2
[1,] -0.5844154 -0.5604264
[2,] -0.6076282  0.5390168
> scores <- y%*%coeff
> scores
```

	LD1	LD2
[1,]	-2.407300	1.05662406
[2,]	-1.776459	-0.58183604
[3,]	-2.945290	-1.70268891
[4,]	-4.744962	-1.18508169
[5,]	-5.071989	0.18387000
[6,]	-3.903158	1.30472287
[7,]	-2.664689	-0.87275406
[8,]	-3.541312	-1.71339372
[9,]	-4.452754	-0.90486847
[10,]	-4.183759	0.47478802
[11,]	-3.295530	0.76570604
[12,]	-2.384087	-0.04281921
[13,]	-3.576131	-0.06422881
[14,]	-8.251454	-4.54764031
[15,]	-7.620613	-6.18610041
[16,]	-8.789444	-7.30695328
[17,]	-10.589116	-6.78934605
[18,]	-10.916143	-5.42039437
[19,]	-9.747312	-4.29954150
[20,]	-8.508843	-6.47701843
[21,]	-9.385466	-7.31765808
[22,]	-10.296908	-6.50913284
[23,]	-10.027913	-5.12947635
[24,]	-9.139684	-4.83855833
[25,]	-8.228241	-5.64708358
[26,]	-9.420285	-5.66849318
[27,]	-8.483582	6.44679238
[28,]	-7.852741	4.80833228
[29,]	-9.021572	3.68747941
[30,]	-10.821244	4.20508664
[31,]	-11.148271	5.57403832
[32,]	-9.979440	6.69489120
[33,]	-8.740971	4.51741426
[34,]	-9.617594	3.67677461
[35,]	-10.529036	4.48529986
[36,]	-10.260042	5.86495634
[37,]	-9.371812	6.15587436
[38,]	-8.460370	5.34734912
[39,]	-9.652413	5.32593951
[40,]	-14.327737	0.84252802
[41,]	-13.696895	-0.79593208
[42,]	-14.865726	-1.91678496
[43,]	-16.665398	-1.39917773
[44,]	-16.992425	-0.03022604
[45,]	-15.823594	1.09062683

```
  [46,] -14.585125 -1.08685011
  [47,] -15.461748 -1.92748976
  [48,] -16.373190 -1.11896451
  [49,] -16.104196  0.26069198
  [50,] -15.215966  0.55161000
  [51,] -14.304524 -0.25691525
  [52,] -15.496567 -0.27832486
> cor(scores)
              LD1           LD2
LD1  1.00000e+00 -3.90811e-17
LD2 -3.90811e-17  1.00000e+00
> cor(scores,y)
          [,1]       [,2]
LD1 -0.6916502 -0.7192432
LD2 -0.7222327  0.6947584
> mx <- s2.lda$means
> yc <- mx %*% coeff
> yc

           LD1        LD2
  1  -3.457494 -0.2524585
  2  -9.301648 -5.8567228
  3  -9.533776  5.1377099
  4 -15.377930 -0.4665545
> plot(s2.lda)
> lines(yc)
```

5.17 Comparison

For comparison, the discriminant functions for each of the two examples is given in Table 5.5.

Example 1	Example 2
LD1 (0.945, 1.2429)	(-0.5844, -0.6076)
LD2 (-0.557, 0.5718)	(-0.5604, 0.5390)

Table 5.5: Discriminant functions for Examples 1 and 2

The correlation between the discriminant scores and the original variables is shown in Figure 5.6. As expected, for Example 1 both of the original variables are highly correlated with the first discriminant function, as suggested by the plot in discriminant space. For example two, the pattern of correlation is roughly equivalent for each discriminant function in absolute value, with sign reversal in the second discriminant function.

Example 1

	X1	X2	Pnn Trace
LD1	0.987	0.992	0.9999
LD2	-0.158	0.124	0.0001

Example 2

	X1	X2	PPn Trace
LD1	-0.691	0.719	0.5403
LD2	-0.722	0.694	0.4597

Table 5.6: Tables of correlations for Examples 1 and 2

5.18 Number of discriminants

If g is the # of groups and p is the # of variables then r the number of possible discriminant functions is given by $r = \min(p, g-1)$. The first few values are given in the Table 5.7.

		$p \rightarrow$			
g	$(g-1)$	2	3	4	5
2	1	1	1	1	1
3	2	2	2	2	2
4	3	2	3	3	3
5	4	2	3	4	4
6	5	2	3	4	5
7	6	2	3	4	5

Table 5.7: The number of possible discriminant functions

Thus there is no loss of discriminant information by plotting in two dimensions if the conditions in Table 5.8 hold

variables	groups	max # of discriminants
any p	$g = 2$	1
any p	$g = 3$	2
$p = 2$	any g	2

Table 5.8: Maximum number of discriminants

5.19 Tests on the significance of the discriminant function

The question of objective tests on the significance of discriminant functions will be delayed until after the topic on Canonical correlation (and Manova). Again these tests

reduce to tests on eigenvalues. Note that these tests are sensitive to sample size and so % of variance is often used as a gauge of practical rather than statistical significance (S p302). Thus a very large sample size will show small differences as being statistically significant whereas the difference may not be of practical importance. So the same problem occurs as in PCA where the number of eigenvalues to be retained needs to be addressed. Tests similar to Horn's Procedure are needed.

5.20 Classification Rules

With only two groups, classification reduces to a dichotomous choice along the values of the single discriminant function (y) using a critical value based on a weighted value of the centroids $\bar{y}_1$ and $\bar{y}_2$; (S p278, JW p523). For multiple groups, an extension of such a procedure using the centroids is used. The simplest way to consider classification is in terms of the Euclidean distance in discriminant function space (y) choose

$$\underset{k}{\text{minimum}} \sum_{j=1}^{r} \left(Y_{\text{new}_j} - \bar{Y}_{Kj}\right)^2$$

to allocate $\mathbf{Y}$new to π_K i.e. group K. (K is the **group** index). That is allocate the new observation to that group whose centroid is nearest.

(JW p55)

When only two discriminants are involved, a **territorial map** can be produced showing the borders of group membership.

Such a map for the Sharma Panel II data is shown in Figure 5.13. The data in discriminant function space are the circles, while the numbers $1 \ldots 4$ denote the group centroids. The boundaries of group membership are shown by the solid lines.

5.21 Notes

1. A Euclidean distance metric in discriminant space corresponds to a quasi–Mahalanobis distance measure in variable space. The use of a strict Mahalanobis distance measure would imply MVN with constant Σ over groups for the original variables.

   JW p547–549, S p258.

2. Prediction can be cast in terms of general types of distributions (not just MVN) for the original data S p278, JW p525.

3. Posterior probabilities of group membership can also be computed (S p281), using Bayesian arguments based on prior distributions of group membership.

Discriminant functions are not necessarily orthogonal, but discriminant scores are uncorrelated.

(Recall that PCA functions are orthogonal *and* their scores are uncorrelated)

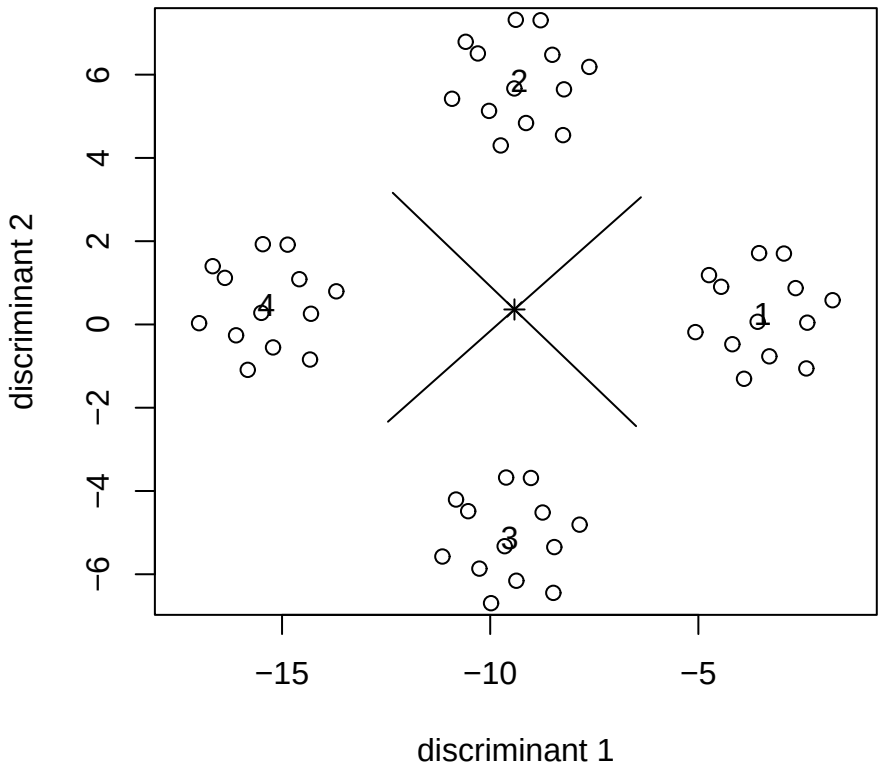

Figure 5.13: Territorial map Sharma Panel II data

5.22 Empirical demonstration

Ex1

Sharma Panel I

LD1		LD2	
0.9457708	×	-0.5577360	= -0.527490
1.2429010	×	0.5718173	= + 0.710712
			≠ 0

orthogonality is impossible!

Ex2

Sharma Panel II

LD1	LD2	
-0.5844154	-0.5604264	= + 0.327522
-0.6076282	0.5390168	= - 0.327 522
		0

This orthogonality of the discriminant functions is verified by the perpendicular lines in the territorial map in Figure 5.13.

R Output for the production of the territorial map

```
> cor(scores)
              LD1           LD2
LD1  1.000000e+00 -5.552055e-17
LD2 -5.552055e-17  1.000000e+00
> cor(scores,y)
          [,1]       [,2]
LD1 -0.6916502 -0.7192432
LD2  0.7222327 -0.6947584
> mx <- s2.lda$means
> yc <- mx %*% coeff
> yc

           LD1        LD2
  1  -3.457494  0.2524585
  2  -9.301648  5.8567228
  3  -9.533776 -5.1377099
  4 -15.377930  0.4665545
> dim(yc) <- c(4,2)
> yc1 <- yc[1,];dim(yc1) <- c(1,2)
> yc2 <- yc[2,];dim(yc2) <- c(1,2)
> yc3 <- yc[3,];dim(yc3) <- c(1,2)
> yc4 <- yc[4,];dim(yc4) <- c(1,2)
> ym1 <- mean(yc[,1])
> ym1
[1] -9.417712
> ym2 <- mean(yc[,2])
> ym2
[1] 0.3595065
> ym <- cbind(ym1,ym2)
```

```
> ym
           ym1       ym2
[1,] -9.417712 0.3595065
> y12 <- (yc[1,] + yc[2,])/2
> y24 <- (yc[2,] + yc[4,])/2
> y43 <- (yc[4,] + yc[3,])/2
> y31 <- (yc[3,] + yc[1,])/2
> summary(scores)
      LD1                LD2
 Min.   :-16.992   Min.   :-6.6949
 1st Qu.:-11.785   1st Qu.:-1.8977
 Median : -9.403   Median : 0.2676
 Mean   : -9.418   Mean   : 0.3595
 3rd Qu.: -6.983   3rd Qu.: 2.5205
 Max.   : -1.776   Max.   : 7.3177
> eqscplot(scores[,1],scores[,2],type="n",xlab="discriminant 1",ylab="discriminant 2")
> points(scores[,1],scores[,2],type="p")
> points(yc1,pch="1",type="p")
> points(yc2,pch="2",type="p")
> points(yc3,pch="3",type="p")
> points(yc4,pch="4",type="p")
> points(ym,pch=8,type="p")
> Y12 <- rbind(ym,y12)
> Y24 <- rbind(ym,y24)
> Y43 <- rbind(ym,y43)
> Y31 <- rbind(ym,y31)
> points(Y12,type="l")
> points(Y24,type="l")
> points(Y43,type="l")
> points(Y31,type="l")
```

Note the use of the `MASS` library function `eqscplot` to produce an equally scaled plot. This was necessary to show the orthogonality of the discriminant functions in the territorial map.

5.22.1 Exercise

Verify the rule used to produce the boundaries in the territorial map using the centroid averages and the grand mean.

5.23 Scores?

See R code where the discriminant scores for Ex1 are shown to be uncorrelated as $r = -6.49 \times 10^{-16}$!

(Verify for Ex2)

5.24 Orthogonality

S p293

Why are the discriminant scores always orthogonal, but the discriminant functions not necessarily so?

Answer :

The eigenvalue problem to determine the discriminants is

$$|W^{-1}B - \lambda I| = 0$$

where B and W are the between and within groups (SSCP) sum of squares and cross products matrices for the p variables. Alas $W^{-1}B$ is in general non symmetric, hence the resulting eigenvectors (discriminant functions) will not be orthogonal. Ex2 (Sharma Panel II) is a special case where $W^{-1}B$ was symmetric. See also JW P543.

For PCA the eigenvalues are obtained from $|\Sigma - \lambda I| = 0$ where Σ is symmetric being the covariance matrix. Thus for PCA, the functions are orthogonal as well as the scores.

You should now attempt Workshop 6.

Chapter 6

Multivariate Analysis of Variance

(J p439)

6.1 Introduction

Examples 4 and 5 from Chapter 1 show the multivariate generalisation of the two sample t-test. This is the simplest form of Multivariate AOV possible and it is used to start the development of the theory of the multivariate analysis of variance (MANOVA).

Now the multivariate hypothesis to be tested is

$$H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} = \boldsymbol{\mu}.$$

The assumptions are that $\boldsymbol{Y}$ is MVN and $\Sigma_1 = \Sigma_2 = \Sigma$.

The univariate t test for 2 independent samples is

$$t = \frac{\bar{x}_1 - \bar{x}_2}{s_p\sqrt{\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad \sim t_{n_1+n_2-2}$$

while Hotellings T^2 in the multivariate case is

$$T^2 = \left(\frac{n_1 n_2}{n_1 + n_2}\right)(\bar{x}_1 - \bar{x}_2)^T S_p^{-1}(\bar{x}_1 - \bar{x}_2)$$

with

$$\frac{(n_1 + n_2 - p - 1)}{p(n_1 + n_2 - 2)} T^2 \sim F_{p,n_1+n_2-p-1}.$$

Being a special case of MANOVA,the empirical results for T^2 should be identical to an analysis using MANOVA. This will be demonstrated later.

For the univariate t–test, the fitted values are simply the group means, as the test of equality can simply be cast as comparing a model with the same mean to

a model with different means for each group. For the multivariate case these fitted values are also the group means (for each variable), but the essential difference is that the pooled variance (s_p^2) for say two univariate t–tests on two variates would use the information from the group sample variances only, whereas the generalized estimate of variance (S_p) in the multivariate test includes the additional information from the sample correlation between the variates. This point leads into the question posed in the next section.

6.2 Why MANOVA?

Newcomers often ask "why not simply perform separate AOVS?". The empirical example from Sharma P353–355 demonstrates the dangers inherent in such an approach. Basically if the response variables are related, MANOVA will exploit such a correlation to heighten the test of differences in means; separate univariate analysis ignore such information since this approach assumes independence of response variates. However, if the responses are unrelated there is no advantage in using a MANOVA; in fact separate AOVS, properly used, give a **more** powerful form of test, when response variables are unrelated.

R Output : Sharma example p353–355

```
> dat <- read.table("sharma.dat",header=T)
> dat
   Group X1 X2
1      1  1  3
2      1  2  5
3      1  4  7
4      1  6 11
5      1  6 12
6      2  4  5
7      2  5  5
8      2  5  6
9      2  8  7
10     2  8  9
> y <- cbind(dat[,2],dat[,3])
> o <- factor(dat[,1])
> fit <- manova(y~o)
> summary.aov(fit)
 Response 1 :
            Df Sum Sq Mean Sq F value Pr(>F)
o            1  12.10   12.10  2.7816 0.1339
Residuals    8  34.80    4.35

 Response 2 :
            Df Sum Sq Mean Sq F value Pr(>F)
o            1    3.6     3.6  0.4091 0.5403
```

```
Residuals    8   70.4     8.8

> summary.manova(fit)
          Df  Pillai approx F num Df den Df   Pr(>F)
o          1  0.8099  14.9143      2      7 0.002994 **
Residuals  8
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> mv <- summary.manova(fit)
> mv$SS
$o
     [,1] [,2]
[1,] 12.1 -6.6
[2,] -6.6  3.6

[[2]]
     [,1] [,2]
[1,] 34.8 45.6
[2,] 45.6 70.4

> mv$stats
          Df    Pillai approx F num Df den Df      Pr(>F)
o          1 0.8099302 14.91429      2      7 0.002993612
Residuals  8        NA       NA     NA     NA          NA
> mv$Eigenvalues
      [,1]         [,2]
o 4.261226 -8.95551e-17
```

6.3 Assumption

Three items need consideration

1. Normality (multivariate) – test via `qqbeta`
2. Equality of covariance matrices - test via Box's M test.

   The assumption is that

   $$H_0 : \mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \ldots = \mathbf{\Sigma}$$

   is true, as per discriminant analysis.
3. Test for independence of observations or subjects. See S p387–388.

6.4 Two Sample Case

M p140, J W p237

$$H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} = \boldsymbol{\mu}$$

6.5 Univariate Case :

Under the null hypothesis $H_0 : \mu_1 = \mu_2$,

$$t = \frac{\bar{x}_1 - \bar{x}_2}{s_p\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \qquad \sim t_{n_1+n_2-2}$$

and s_p uses s_1^2 and s_2^2 for the single response variable.

6.6 Multivariate case :

Under the null hypothesis $H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$, Hotellings T^2 becomes

$$T^2 = \frac{n_1 n_2}{(n_1 + n_2)}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)^T S_p^{-1}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$$

and

$$\frac{(n_1 + n_2 - p_1)}{p(n_1 + n_2 - 2)}T^2 \sim F_{p,n_1+n_2-p-1}$$

but S_p uses s_1^2, s_2^2 and r_{12} for all responses pooled over groups.

(Note that paired multivariate samples could be treated as a one sample problem.)

6.7 Example

Lawn mower data with both groups: owners and nonowners.

$$
\begin{aligned}
& T^2 \\
T^2 &= \frac{n_1.n_2}{n_1+n_2}\left(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2\right)' S_p^{-1}\left(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2\right) \\
\bar{\mathbf{x}}_1 &= \begin{bmatrix} 26.491667 \\ 10.133333 \end{bmatrix} \bar{\mathbf{x}}_2 = \begin{bmatrix} 19.133333 \\ 8.816667 \end{bmatrix} \\
S_p &= \begin{bmatrix} 30.7416 & -1.2006 \\ -1.2006 & 1.0683 \end{bmatrix} \\
S_p^{-1} &= \begin{bmatrix} .034022 & .038236 \\ .038236 & 0.979038 \end{bmatrix} \qquad n_1 = n_2 = 12 \\
T^2 &= 6\left[(26.491667, 10.13) - (19.13, 8.816)\right] S^{-1} \begin{bmatrix} 26.4196 & -19.1 \\ 10.13 & -8.816 \end{bmatrix} \\
T_6^2 &= (7.358334, 1.316) \begin{bmatrix} .034022 & .038236 \\ .038236 & 0.979038 \end{bmatrix} \begin{bmatrix} 7.358334 \\ 1.316 \end{bmatrix} \\
T^2 &= (7.358334, 1.316) \begin{bmatrix} 0.300673 \\ 1.570420 \end{bmatrix} \times 6 \\
T^2 &= 25.681
\end{aligned}
$$

F?

$$
\begin{aligned}
F_0 &= \frac{n_1+n_2-p-1}{p(n_1+n_2-2)} T^2 \\
&= \frac{21}{2\times 22} 25.681 = 12.256 \\
F_{2,20,1\%} &= 5.85
\end{aligned}
$$

So Reject the null hypothesis of equality of means, as expected.
Note that an alternative calculation can be performed using

$$D^2 = \bar{y}_1 - \bar{y}_2 = \boldsymbol{b}'(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2)$$

from discriminant analysis where

$$D^2 = \left(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2\right)' S_p^{-1}\left(\bar{\mathbf{x}}_1-\bar{\mathbf{x}}_2\right)$$

and

$$y = \boldsymbol{b}'\boldsymbol{x} \qquad \boldsymbol{b} = S^{-1}(\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_2).$$

Note that from the R Output (discriminant analysis) we have

$$
\begin{aligned}
b_1 &= 0.3006909 \\
b_2 &= 1.5703694
\end{aligned}
$$

for comparison.

```
> dat <- read.table("lawn.dat",header=T)
> dat
   own income lotsize
1    1   20.0     9.2
2    1   28.5     8.4
3    1   21.6    10.8
4    1   20.5    10.4
5    1   29.0    11.8
6    1   36.7     9.6
7    1   36.0     8.8
8    1   27.6    11.2
9    1   23.0    10.0
10   1   31.0    10.4
11   1   17.0    11.0
12   1   27.0    10.0
13   2   25.0     9.8
14   2   17.6    10.4
15   2   21.6     8.6
16   2   14.4    10.2
17   2   28.0     8.8
18   2   16.4     8.8
19   2   19.8     8.0
20   2   22.0     9.2
21   2   15.8     8.2
22   2   11.0     9.4
23   2   17.0     7.0
24   2   21.0     7.4
> y <- cbind(dat[,2],dat[,3])
> o <- factor(dat[,1])
> fit <- manova(y~o)
> summary.aov(fit)
 Response 1 :
            Df Sum Sq Mean Sq F value   Pr(>F)
o            1 324.87  324.87  10.568 0.003665 **
Residuals   22 676.32   30.74


 Response 2 :
            Df  Sum Sq Mean Sq F value   Pr(>F)
o            1 10.4017 10.4017  9.7363 0.004983 **
Residuals   22 23.5033  1.0683


> summary.manova(fit)
          Df  Pillai approx F num Df den Df   Pr(>F)
```

```
o          1  0.5386  12.2570      2     21 0.000297 ***
Residuals 22
```

```
> mv <- summary.manova(fit)
> mv$SS
$o
          [,1]     [,2]
[1,] 324.87042 58.13083
[2,]  58.13083 10.40167

[[2]]
          [,1]      [,2]
[1,] 676.31583 -26.41333
[2,] -26.41333  23.50333
```

```
> mv$stats
          Df    Pillai approx F num Df den Df       Pr(>F)
o          1 0.5386044 12.25704      2     21 0.0002970103
Residuals 22        NA       NA     NA     NA           NA
> mv$Eigenvalues
      [,1]         [,2]
o 1.167337 4.553649e-17
> fitu <- aov(y~o)
> summary(fitu)
 Response 1 :
            Df Sum Sq Mean Sq F value   Pr(>F)
o            1 324.87  324.87  10.568 0.003665 **
Residuals   22 676.32   30.74


 Response 2 :
            Df  Sum Sq Mean Sq F value   Pr(>F)
o            1 10.4017 10.4017  9.7363 0.004983 **
Residuals   22 23.5033  1.0683
```

```
> print(fitu)
Call:
   aov(formula = y ~ o)

Terms:
                        o Residuals
resp 1          324.8704  676.3158
resp 2           10.4017   23.5033
Deg. of Freedom        1        22
```

```
Residual standard error: 5.544513 1.033602
Estimated effects may be unbalanced
> fitu$fitted
       [,1]      [,2]
1  26.49167 10.133333
2  26.49167 10.133333
3  26.49167 10.133333
4  26.49167 10.133333
5  26.49167 10.133333
6  26.49167 10.133333
7  26.49167 10.133333
8  26.49167 10.133333
9  26.49167 10.133333
10 26.49167 10.133333
11 26.49167 10.133333
12 26.49167 10.133333
13 19.13333  8.816667
14 19.13333  8.816667
15 19.13333  8.816667
16 19.13333  8.816667
17 19.13333  8.816667
18 19.13333  8.816667
19 19.13333  8.816667
20 19.13333  8.816667
21 19.13333  8.816667
22 19.13333  8.816667
23 19.13333  8.816667
24 19.13333  8.816667
```

6.8 Manova with Several means

JW p252

For One-way MANOVA the null hypothesis is equality of means, ie,

$$H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \cdots = \boldsymbol{\mu}_k = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} = \boldsymbol{\mu}.$$

The linear model is, in terms of effects,

$$\begin{aligned} \boldsymbol{Y}_{ij} = \boldsymbol{\mu} + \boldsymbol{t}_i + \boldsymbol{\epsilon}_{ij} \qquad j &= 1 \cdots n_i \\ i &= 1 \cdots k \end{aligned}$$

Thus H_0 : becomes

$$\boldsymbol{t}_1 = \boldsymbol{t}_2 = \cdots \boldsymbol{t}_k = \mathbf{0}$$

where

$$\boldsymbol{t}_i = \begin{bmatrix} t_{i1} \\ \cdots \\ t_{ip} \end{bmatrix} \text{ and } \boldsymbol{\mu}_i = \boldsymbol{\mu} + \boldsymbol{t}_i$$

6.9 Decomposition

The decomposition for MANOVA parallels that of ANOVA. For an observation, we have

$$\mathbf{y}_{ij} = \bar{\mathbf{y}} + (\bar{\mathbf{y}}_i - \bar{\mathbf{y}}) + (\bar{\mathbf{y}}_{ij} - \bar{\mathbf{y}}_i)$$

ie,

$$\text{obs} = \text{mean} + \text{treatment effect} + \text{Residual}$$

For the decomposition of the SSP matrix,

$$\text{SST} = \text{SSB} + \text{SSW}$$

$$\text{(Total SSP)} = \text{(Between SSP)(Within SSP)}$$

Now for MANOVA, the decomposition in **matrix** form is

$$\begin{aligned} \mathbf{T} &= \mathbf{B} + \mathbf{W} \\ \mathbf{W} &= (n_1 - 1)\mathbf{S}_1 + (n_2 - 1)\mathbf{S}_2 + (n_k - 1)\mathbf{S}_k \\ \mathbf{B} &= \sum_{i=1}^{k} n_i(\bar{\mathbf{y}}_i - \bar{\mathbf{y}})(\bar{\mathbf{y}}_i - \bar{\mathbf{y}})^T \\ \mathbf{W} &= \sum_{i=1}^{k}\sum_{j=1}^{n_i}(\mathbf{y}_{ij} - \bar{\mathbf{y}}_i)(\mathbf{y}_{ij} - \mathbf{y}_i)^T \end{aligned}$$

6.10 MANOVA Table

The MANOVA table in general form is shown in Table 6.1 where the term SSP is the matrix of sum of squares and cross products.

Source	SSP	DF
TREATMENT	$\mathbf{B}$	$k-1$
ERROR	$\mathbf{W}$	$\sum_{i=1}^{k} n_i - k$
TOTAL	$\mathbf{W}+\mathbf{B}$	$\sum_{i=1}^{k} n_i - 1$

Table 6.1: MANOVA Table

6.11 Test Statistic

The statistic Wilks lambda is defined as

$$\Lambda^* = \frac{|W|}{|B+W|}.$$

In the MANOVA table, special cases in which Simple Functions of Λ are exactly distributed as F are shown in Table 6.2.

Variables	Groups	Transformation	$\sim F$
Any number	2	$\left(\frac{1-\Lambda}{\Lambda}\right)\left(\frac{n-p-1}{p}\right)$	$F_{\alpha;(p,n-p-1)}$
Any number	3	$\left(\frac{1-\Lambda^{1/2}}{\Lambda^{1/2}}\right)\left(\frac{n-p-2}{p}\right)$	$F_{\alpha;(2p,2(n-p-2))}$
1	Any number	$\left(\frac{1-\Lambda}{\Lambda}\right)\left(\frac{n-k}{(k-1)}\right)$	$F_{\alpha;(k-1,n-k)}$
2	Any number	$\left(\frac{1-\Lambda^{1/2}}{\Lambda^{1/2}}\right)\left(\frac{n-k-1}{(k-1)}\right)$	$F_{\alpha;(2(k-1),2(n-k-1))}$

Table 6.2: Special Cases in the MANOVA Table

In general, the distribution of the test statistic is approximated by an F distribution.

6.12 Which test statistic?

The popular choices of test statistic in Manova are :

$$\begin{aligned} \text{Pillai's trace} &= \sum_i \frac{\lambda_i}{1+\lambda_i} \\ \text{Hotellings Trace} &= \sum_i \lambda_1 \\ \text{Wilks } \Lambda &= \prod_i \frac{1}{1+\lambda_i} \end{aligned}$$

Pillai is used as it is the most robust and has adequate power. So this explains why the R team chose it in their implementation.

6.13 Example

S p355

A study is designed with 20 subjects randomly divided into 4 equally sized groups of 5 subjects. Two drugs were used. The first group were given a placebo (o=`control`), the second a combination of both drugs (b=`ab`), while the remaining groups were given only one drug each (1=`a`, 2=`b`). The drug efficiency is measured by two responses. A plot of the dual responses is given in Figure 6.1, showing the four drug treatment combinations.

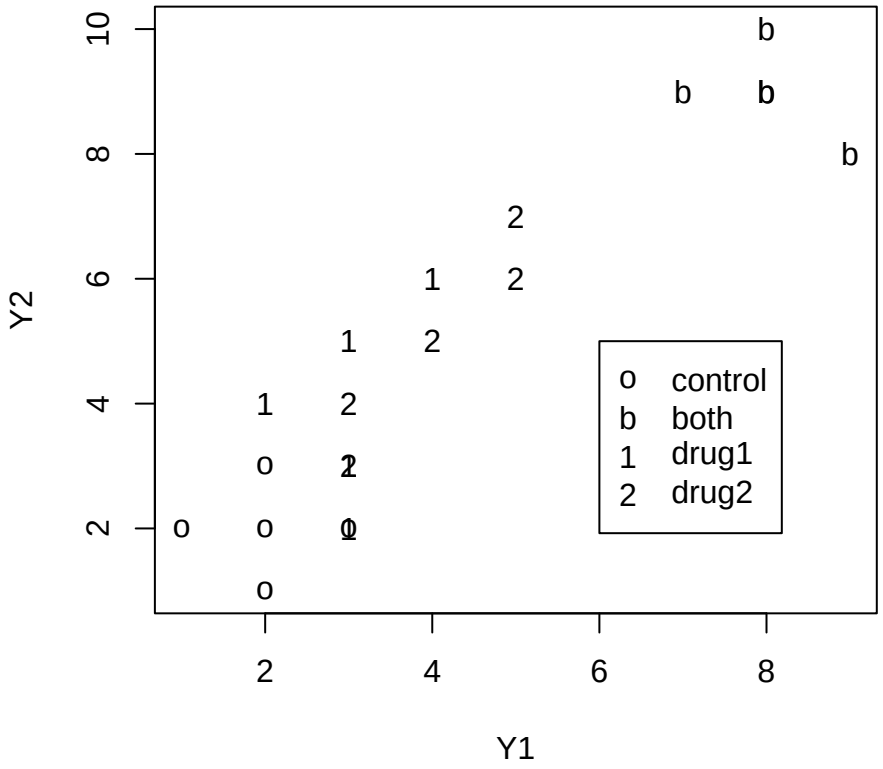

Figure 6.1: Dual responses to 4 drug treatments

Note the effect of the two drugs used in combination.

R Output

```
> dat <- read.table("drug.txt",header=T)
> dat
     Treat Y1 Y2
1  control  1  2
2  control  2  1
3  control  3  2
4  control  2  3
5  control  2  2
```

```
6       ab  8  9
7       ab  9  8
8       ab  7  9
9       ab  8  9
10      ab  8 10
11       a  2  4
12       a  3  2
13       a  3  3
14       a  3  5
15       a  4  6
16       b  4  5
17       b  3  3
18       b  3  4
19       b  5  6
20       b  5  7
> attach(dat)
> y <- cbind(Y1,Y2)
> fit <- manova(y~Treat)
> summary.aov(fit)
 Response Y1 :
            Df  Sum Sq Mean Sq F value    Pr(>F)
Treat        3 103.750  34.583  55.333 1.144e-08 ***
Residuals   16  10.000   0.625


 Response Y2 :
            Df  Sum Sq Mean Sq F value    Pr(>F)
Treat        3 130.000  43.333  28.889 1.078e-06 ***
Residuals   16  24.000   1.500


> summary.manova(fit)
          Df Pillai approx F num Df den Df    Pr(>F)
Treat      3 1.0272   5.6310      6     32 0.0004429 ***
Residuals 16

> mv <- summary.manova(fit)
> mv$SS
$Treat
       Y1  Y2
Y1 103.75 115
Y2 115.00 130

$Residuals
   Y1 Y2
Y1 10  7
```

```
Y2  7 24

> mv$stats
          Df   Pillai approx F num Df den Df       Pr(>F)
Treat      3 1.027150 5.631017      6     32 0.0004429130
Residuals 16       NA       NA     NA     NA           NA
> mv$Eigenvalues
          [,1]      [,2]
Treat 11.29190 0.1217107
> plot(Y1,Y2)
> # plot 1 2 3 4
>
> cov(y)
         Y1       Y2
Y1 5.986842 6.421053
Y2 6.421053 8.105263
> y1 <- y[Treat=="control"]
> y2 <- y[Treat=="ab"]
> y3 <- y[Treat=="a"]
> y4 <- y[Treat=="b"]
> y1 <- matrix(y1,ncol=2)
> cov(y1)
     [,1] [,2]
[1,]  0.5  0.0
[2,]  0.0  0.5
> y2 <- matrix(y2,ncol=2)
> cov(y2)
      [,1]  [,2]
[1,]  0.50 -0.25
[2,] -0.25  0.50
> y3 <- matrix(y3,ncol=2)
> cov(y3)
     [,1] [,2]
[1,]  0.5  0.5
[2,]  0.5  2.5
> y4 <- matrix(y4,ncol=2)
> cov(y4)
     [,1] [,2]
[1,]  1.0  1.5
[2,]  1.5  2.5
> plot(Y1,Y2,type="n",xlab="Y1",ylab="Y2")
> points(y1,pch="o")
> points(y2,pch="b")
> points(y3,pch="1")
> points(y4,pch="2")
> legend(6,5,legend=c("control","both","drug1","drug2"),pch=c("o","b","1","2"))
```

```
> p <- 2
> k <- 4
> S1 <- cov(y1)
> print(log(det(S1)))
[1] -1.386294
> S2 <- cov(y2)
> print(log(det(S2)))
[1] -1.673976
> S3 <- cov(y3)
> print(log(det(S3)))
[1] 2.220446e-16
> S4  <- cov(y4)
> print(log(det(S4)))
[1] -1.386294
> sn1 <- length(y1[,1]) -1
> sn2 <- length(y2[,1]) -1
> sn3 <- length(y3[,1]) -1
> sn4 <- length(y4[,1]) -1
> sn <- sn1 +sn2 +sn3 +sn4
> S <- (sn1*S1 +sn2*S2 +sn3*S3 +sn4*S4)/sn
> print(log(det(S)))
[1] -0.292904
> M <- log(det(S)) * sn - sn1*log(det(S1)) - sn2 * log(det(S2)) -sn3 * log(det(S3)) -
+ sn4*log(det(S4))
> print(M)
[1] 13.09980
> f <- ( 2*p*p + 3 * p -1)/(6*(p+1)*(k-1))
> print(f)
[1] 0.2407407
> CM1 <- 1 - f *((1/sn1 + 1/sn2 + 1/sn3 +1/sn4) - 1/sn)
> ch1 <- M*CM1
> print(ch1)
[1] 10.14325
> df <- (k-1)*p*(p+1)/2
> print(df)
[1] 9
> pchisq(ch1,df,lower.tail=F)
[1] 0.3390137
```

Note the implementation of Box's M test for more than two groups. The given R code also will handle unequal group sizes. We would accept the assumption of equality of covariance matrices (P=0.339) and overall the presence of a treatment effect (P=0.00044).

Compare the R output with Sharma p356.

6.13.1 Exercise

Verify the value obtained in R for the Pillai trace for the two data sets from Sharma and the lawn mower data.

6.14 Multivariate Regression Model

MANOVA is a special case of the *Multivariate Regression Model* or Multivariate Linear Model, which is an extension of the Classical Linear Regression Model.

The Classical Regression Model has a single response, r predictors and n observations.

1. The form of the model is (JW p287)

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$(n \times 1) = [n(r+1)] \times [(r+1) \times 1] + (n \times 1)$$

Now

$$E\boldsymbol{\varepsilon} = \mathbf{0}, \;\; \mathrm{Cov}(\boldsymbol{\varepsilon}) = \sigma^2 \boldsymbol{I} = E(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}')$$
$$n \times 1, \quad (n \times n)$$

where $\boldsymbol{\beta}$ and σ^2 are unknown parameters, and the jth row of the design matrix $\mathbf{Z}$ is $[Z_{j0}, Z_{j1}, \ldots, Z_{jr}]$. Usually $Z_{j0} = 1$.

In expanded form

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

becomes

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & Z_{11} & \ldots & Z_{1r} \\ \vdots & \vdots & & \vdots \\ 1 & Z_{n1} & \ldots & Z_{nr} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_r \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

2. The *Multivariate Regression Model* now has p responses in place of 1, and so the model becomes (JW p315)

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$(n \times p) = [n \times (r+1)][(r+1) \times p] + (n \times p)$$

with

$$E\boldsymbol{\varepsilon}_{(i)} = 0, \;\; \mathrm{Cov}(\boldsymbol{\varepsilon}_{(i)}, \boldsymbol{\varepsilon}_{(k)}) = \sigma_{ik}\boldsymbol{I} \qquad i, k = 1, 2, \ldots p$$

The matrix $\boldsymbol{I}$ is $n \times n$.

The p responses from the jth trial have covariance $\Sigma = \{\sigma_{ik}\}$, but observations from different trials are assumed to be uncorrelated. The parameters $\boldsymbol{\beta}$ and σ_{ik} are unknown, and the design matrix $\mathbf{Z}$ has jth row $[Z_{j0}, Z_{j1}, \ldots, Z_{jr}]$.

In expanded form

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

becomes

$$\begin{bmatrix} Y_{11} & \ldots & Y_{1p} \\ \vdots & & \vdots \\ Y_{n1} & \ldots & Y_{np} \end{bmatrix} = \begin{bmatrix} Z_{10} & Z_{11} & \ldots & Z_{1r} \\ \vdots & \vdots & & \vdots \\ Z_{n0} & Z_{n1} & \ldots & Z_{nr} \end{bmatrix} \begin{bmatrix} \beta_{01} & \ldots & \beta_{0p} \\ \vdots & & \vdots \\ \beta_{r1} & \ldots & \beta_{rp} \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} & \ldots & \varepsilon_{1p} \\ \vdots & & \vdots \\ \varepsilon_{n1} & \ldots & \varepsilon_{np} \end{bmatrix}$$

in compact form

$$[\boldsymbol{Y}_1|\boldsymbol{Y}_2|\ldots|\boldsymbol{Y}_n] = \boldsymbol{Z}\left[\boldsymbol{\beta}_0|\boldsymbol{\beta}_1|\ldots|\boldsymbol{\beta}_p\right] + [\boldsymbol{\varepsilon}_1|\boldsymbol{\varepsilon}_2|\ldots|\boldsymbol{\varepsilon}_n]$$

Note that this model assumes that the *same* predictors are used for each response.

The sth response follows a linear regression model

$$\boldsymbol{Y}_s = \boldsymbol{Z}\boldsymbol{\beta}_s + \boldsymbol{\varepsilon}_s$$

for $s = 1, \ldots, p$ with $\mathrm{Cov}(\boldsymbol{\varepsilon}_s) = \sigma_{ss}\boldsymbol{I}$ where $\boldsymbol{I}$ is $n \times n$. This is verified by the use of `lm` for each of the responses in the R Output for the JW problem (p317), given in Section 6.17.

6.15 Generalised Least Squares

The multivariate regression contains MANOVA as a special case (categorical predictors). The multivariate regression model is itself a special case of **generalised least squares** (GLS), since for the multivariate regression model the same model using the same predictors is used for all responses. This is not necessarily so in the full GLS model.

The GLS model is defined as [1]

$$\begin{aligned} \mathbf{Y} &= \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \\ E(\boldsymbol{\varepsilon}) &= 0 \qquad E(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}') = \boldsymbol{\Sigma} = \sigma^2\boldsymbol{\Omega} \end{aligned}$$

where $\boldsymbol{\Omega}$ is positive definite.

[1] See p358 and Section 17.2.1, p488, Greene W.H., Econometric Analysis, MacMillan 1993.

Recall that the ordinary least squares estimator is

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

where $E(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}') = \sigma^2\boldsymbol{I}$.

The GLS estimator

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{\Omega}\boldsymbol{Y}$$

is also called the **Aitken** estimator.

The residual variance is

$$\hat{\sigma}^2 = \frac{\hat{\boldsymbol{\varepsilon}}'\hat{\boldsymbol{\varepsilon}}}{n-k} = \frac{(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}})'\boldsymbol{\Omega}^{-1}(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}})}{(n-k)}$$

6.16 Special Cases of GLS

There are several special cases of GLS which give results that are useful in multivariate modelling

(a) (JW p352, 7.11)

(GLS for the multivariate regression model).

Let $\mathbf{A}$ be a positive definite matrix so

$$d_j^2(\mathbf{B}) = (\mathbf{y}_i - \mathbf{B}'Z_j)'\mathbf{A}(\mathbf{y}_j - \mathbf{B}'\mathbf{Z}_j)$$

is a squared statistical distance from the jth observation $\mathbf{y}_j$ to its ith regression $\mathbf{B}'\mathbf{z}_j$ i.e., $\mathbf{y}_i - \mathbf{B}'\mathbf{z}_j$ is the jth residual. The choice

$$\mathbf{B} = \hat{\boldsymbol{\beta}} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\boldsymbol{Y}$$

minimises $\Sigma_j d_j^2(\mathbf{B})$ for any $\mathbf{A}$.

Choices for $\mathbf{A}$ include $\boldsymbol{\Sigma}^{-1}$ (manova and multivariate regression) and $\boldsymbol{I}$ (univariate regressions).

(b) (Greene, p488)

An equivalent set of statements is given by Greene, such that if,

(a) there is no correlation between responses, then GLS becomes OLS, and also if

(b) the same set of predictors are used in all equations, then GLS and OLS coincide for estimation purposes.

In the more general case of unrestricted correlation, the question of correlation between observation (subjects) arises. This is classed as *repeated measures* as frequently the cause of such correlation is repeated measurements on the same subject. Greene offers the following comments :

(a) The greater the correlation between observations (disturbances) the greater the benefit of GLS over OLS (Ordinary Least Squares), and

(b) the less correlation between predictor or design matrices ($\boldsymbol{Z}$) for each response, the greater the gain in using GLS.

The basic message from these results is that in the case of MANOVA (and GLS) there is **no** extra benefit of assuming the more complicated structure if the variables (responses) are essentially independent and uncorrelated. In fact, individual separate and independent models for each response are in fact more efficient when there is no correlation between responses.

Conversely if responses are correlated then, even though the estimates of coefficients are unchanged, the precision of their estimation certainly is not since the within SSCP determinant[2] can then be reduced considerably.

6.16.1 Example

In the case of Sharma (p353-355), ignoring the correlation inflates the error SS from 370.56 to 2449.92, i.e. a factor of 6.6.

Calculation from the MANOVA:

$$\text{ESS (ignoring correlation)} = \begin{vmatrix} 34.8 & 0 \\ 0 & 70.4 \end{vmatrix} = 2449.92$$

$$\text{ESS (using correlation)} = \begin{vmatrix} 34.8 & 45.6 \\ 56.6 & 70.4 \end{vmatrix} = 370.56$$

This huge reduction due to correlation between the two responses explains the phenomenon of no significance using univariate tests (which treat the responses as independent) compared to the significant difference found using the multivariate test (which exploits the correlation between the two responses).

6.17 Example

(Multivariate Regression) JW p.317 - 319

This simple problem with two responses and a single predictor demonstrates how to fit a multivariate regression model in R.

1. Pillai trace $= \sum \frac{\lambda_i}{1+\lambda_i}$

   $= \frac{15}{16} = 0.9375$

[2]Error SSq

2. Effect of correlation between responses

$$\begin{aligned} \text{SSE (ignoring correlation)} &= \begin{vmatrix} 6 & 0 \\ 0 & 4 \end{vmatrix} = 24 \\ \text{SSE (using correlation)} &= \begin{vmatrix} 6 & -2 \\ -2 & 4 \end{vmatrix} = 24 - 4 = 20 \\ \text{increase} &= \frac{24}{20} = \frac{12}{10} = 1.2(20\%) \end{aligned}$$

Compare the R output with the results from J&W P319.

3. The effect of the multivariate regression model can be seen in the P values

(i) Y_1 alone $P = 0.020$

(ii) Y_2 alone $P = 0.071$

(iii) Y_1 and Y_2 (multivariate regression) $P = 0.062$

This is not as dramatic as the Sharma example but at $\alpha = .05$, Y_1 by itself would have been called significant whereas together the claim for a true joint relation is weaker. Effectively the extra information from Y_2 has acted like a quasi replicate (being related to Y_1) and has improved precision of the estimates, leading to stronger inference.

R Output

```
> dat <- read.table("p317.dat",header=T)
> dat
  y1 y2 z1
1  1 -1  0
2  4 -1  1
3  3  2  2
4  8  3  3
5  9  2  4
> y <- cbind(dat[,1],dat[,2])
> z1 <- dat[,3]
> fit <- manova(y~z1)
> summary.aov(fit)
 Response 1 :
            Df Sum Sq Mean Sq F value  Pr(>F)
z1           1     40      40      20 0.02084 *
Residuals    3      6       2


 Response 2 :
            Df  Sum Sq Mean Sq F value  Pr(>F)
z1           1 10.0000 10.0000     7.5 0.07142 .
```

```
Residuals    3  4.0000  1.3333


> summary.manova(fit)
          Df  Pillai approx F num Df den Df Pr(>F)
z1         1  0.9375  15.0000      2      2 0.0625 .
Residuals  3

> mv <- summary.manova(fit)
> mv$SS
$z1
     [,1] [,2]
[1,]   40   20
[2,]   20   10

[[2]]
     [,1] [,2]
[1,]    6   -2
[2,]   -2    4

> mv$stats
          Df Pillai approx F num Df den Df Pr(>F)
z1         1 0.9375       15      2      2 0.0625
Residuals  3     NA       NA     NA     NA     NA
> mv$Eigenvalues
   [,1]         [,2]
z1   15 1.065988e-15
> attach(dat)
> fit1 <- lm(y1~z1)
> fit2 <- lm(y2~z1)
> summary.aov(fit1)
            Df Sum Sq Mean Sq F value  Pr(>F)
z1           1     40      40      20 0.02084 *
Residuals    3      6       2

> fit1$fitted
1 2 3 4 5
1 3 5 7 9
> fit1$coeff
(Intercept)          z1
          1           2
> summary.aov(fit2)
            Df  Sum Sq Mean Sq F value  Pr(>F)
z1           1 10.0000 10.0000     7.5 0.07142 .
Residuals    3  4.0000  1.3333
```

```
> fit2$fitted
            1             2             3             4             5
-1.000000e+00  2.220446e-16  1.000000e+00  2.000000e+00  3.000000e+00
> fit2$coeff
(Intercept)          z1
         -1           1
```

From the R Output and the form of the Multivariate Regression Model, we have

$$\boldsymbol{Y} = \widehat{\boldsymbol{Y}} + \boldsymbol{e}$$

$$\begin{bmatrix} 1 & -1 \\ 4 & -1 \\ 3 & 2 \\ 8 & 3 \\ 9 & 2 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 3 & 0 \\ 5 & 1 \\ 7 & 2 \\ 9 & 3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & -1 \\ -2 & 1 \\ 1 & 1 \\ 0 & -1 \end{bmatrix}$$

$$\widehat{\boldsymbol{Y}} = \boldsymbol{Z}\widehat{\boldsymbol{\beta}}$$

$$\begin{bmatrix} 1 & -1 \\ 3 & 0 \\ 5 & 1 \\ 7 & 2 \\ 9 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}$$

For verification, see JW p317–318.

6.18 Worked Example - MANOVA

The data in `skull.txt` consist of measurements on skulls of 13 ant–eaters belonging to the subspecies *chapadensis*, deposited in the British Museum from 3 different locations, 1 = Minas Graes (Brazil), 2 = Matto Grosso (Brazil) and 3 = Santa Cruz (Bolivia). On each skull 3 measurements were taken :

`x1`	basal length excluding the premaxilla
`x2`	occipito–nasal length
`x3`	maximum nasal length

A plot of the data (Figure 6.2) shows the relationship between the measurements but not the group structure.

From the R output we conclude that skull measures do not change over localities, since Pillai's trace gives P=0.5397. Note that Box's M test uses only the Brazilian groups as the Bolivian data is insufficient to estimate a covariance matrix. Box's test accepts the equality of covariance matrices from Brazil (P=0.999)

R Output

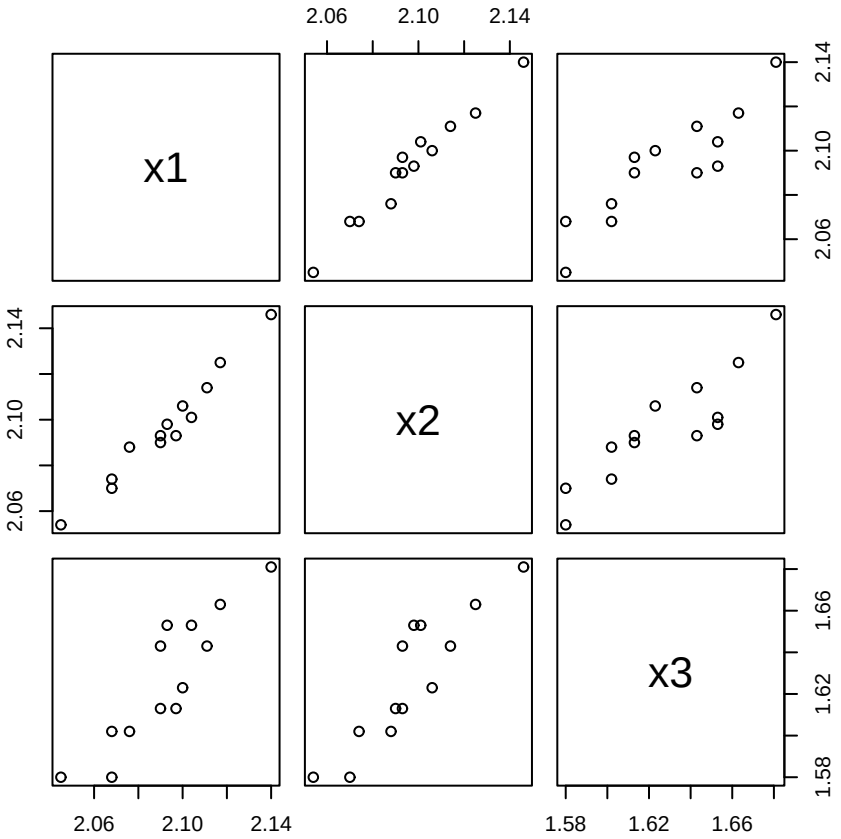

Figure 6.2: Skull measurements on ant–eaters

```
> dat <- read.table("skulls.txt",header=T)
> dat
   Location    x1    x2    x3
1         1 2.068 2.070 1.580
2         1 2.068 2.074 1.602
3         1 2.090 2.090 1.613
4         1 2.097 2.093 1.613
5         1 2.117 2.125 1.663
6         1 2.140 2.146 1.681
7         2 2.045 2.054 1.580
8         2 2.076 2.088 1.602
9         2 2.090 2.093 1.643
10        2 2.111 2.114 1.643
11        3 2.093 2.098 1.653
12        3 2.100 2.106 1.623
13        3 2.104 2.101 1.653
> dat$Location <- factor(dat$Location)
> attach(dat)
> x <- cbind(x1,x2,x3)
> fit <- manova(x~Location)
> summary.aov(fit)
 Response x1 :
            Df    Sum Sq   Mean Sq F value Pr(>F)
Location     2 0.0008060 0.0004030  0.6354 0.5498
Residuals   10 0.0063423 0.0006342

 Response x2 :
            Df    Sum Sq   Mean Sq F value Pr(>F)
Location     2 0.0004820 0.0002410  0.3794 0.6937
Residuals   10 0.0063528 0.0006353

 Response x3 :
            Df    Sum Sq   Mean Sq F value Pr(>F)
Location     2 0.0011844 0.0005922  0.5399 0.5989
Residuals   10 0.0109673 0.0010967

> summary.manova(fit)
          Df  Pillai approx F num Df den Df Pr(>F)
Location   2 0.44703  0.86356      6     18 0.5397
Residuals 10
> mv <- summary.manova(fit)
> mv$SS
$Location
             x1           x2           x3
x1 0.0008059744 0.0006232436 0.0007497949
x2 0.0006232436 0.0004820192 0.0005859487
```

```
x3 0.0007497949 0.0005859487 0.0011843590

$Residuals
            x1          x2          x3
x1 0.006342333 0.006241833 0.007615667
x2 0.006241833 0.006352750 0.007612667
x3 0.007615667 0.007612667 0.010967333

> mv$stats
          Df    Pillai  approx F num Df den Df    Pr(>F)
Location   2 0.4470284 0.8635608      6     18 0.5397193
Residuals 10        NA        NA     NA     NA        NA
> mv$Eigenvalues
              [,1]      [,2]          [,3]
Location 0.3553089 0.2267946 -7.21645e-16
> cov(x)
             x1           x2           x3
x1 0.0005956923 0.0005720897 0.0006971218
x2 0.0005720897 0.0005695641 0.0006832179
x3 0.0006971218 0.0006832179 0.0010126410
> y1 <- x[Location=="1"]
> y2 <- x[Location=="2"]
> y3 <- x[Location=="3"]
> y1 <- matrix(y1,ncol=3)
> cov(y1)
             [,1]         [,2]         [,3]
[1,] 0.0007958667 0.0008342667 0.001044933
[2,] 0.0008342667 0.0008930667 0.001135733
[3,] 0.0010449333 0.0011357333 0.001484267
> y2 <- matrix(y2,ncol=3)
> cov(y2)
          [,1]       [,2]     [,3]
[1,] 0.0007670 0.00068250 0.000807
[2,] 0.0006825 0.00061825 0.000688
[3,] 0.0008070 0.00068800 0.000982
> y3 <- matrix(y3,ncol=3)
> cov(y3)
          [,1]           [,2]     [,3]
[1,]  3.10e-05  1.150000e-05 -1.5e-05
[2,]  1.15e-05  1.633333e-05 -6.5e-05
[3,] -1.50e-05 -6.500000e-05  3.0e-04
> pairs(x)
> pairs(y1,pch="1")
> p <- 3
> k <- 2
> S1 <- cov(y1)
```

```
> print(log(det(S1)))
[1] -28.65235
> S2 <- cov(y2)
> print(log(det(S2)))
[1] -28.49674
> S3 <- cov(y3)
> print(det(S3))
[1] 0
> print(log(det(S3)))
[1] -Inf
> sn1 <- length(y1[,1]) -1
> sn2 <- length(y2[,1]) -1
> sn3 <- length(y3[,1]) -1
> sn <- sn1 +sn2
> S <- (sn1*S1 +sn2*S2)/sn
> print(log(det(S)))
[1] -27.07428
> M <- log(det(S)) * sn - sn1*log(det(S1)) - sn2 * log(det(S2))
> print(M)
[1] 12.15772
> f <- ( 2*p*p + 3 * p -1)/(6*(p+1)*(k-1))
> print(f)
[1] 1.083333
> CM1 <- 1 - f *((1/sn1 + 1/sn2 + 1/sn3) - 1/sn)
> ch1 <- M*CM1
> print(ch1)
[1] 0.1941859
> df <- (k-1)*p*(p+1)/2
> print(df)
[1] 6
> pchisq(ch1,df,lower.tail=F)
[1] 0.9998581
```

Exercise

Verify the value obtained for Pillai's trace.

You should now attempt Workshop 7, and then Assignment 2.

Chapter 7

Canonical Correlation

(J, p494)

7.1 Dependence method

Canonical Correlation can be classed as a *dependence* method. Arbitrarily we can designate the $\boldsymbol{Y}$ variable a vector of $p \times 1$ responses, and the $\boldsymbol{X}$ variable a vector of $m \times 1$ predictors. These vectors can be termed the Y–set and the X–set respectively.

7.1.1 Objective

The objective of canonical correlation is to determine *simultaneous* relationships between linear combinations of the original variables. These new *canonical* variables are denoted as $\boldsymbol{X}^*$ and $\boldsymbol{Y}^*$. The technique produces a sequence of t *uncorrelated* pairs of such canonical variables, where $t = \min(m, p)$.

7.2 Canonical correlation - the method

Before describing the method, some definitions and notation are required.

7.2.1 Notation

The respective means of the X–set and the Y–set are defined as

$$\boldsymbol{\mu}_X = E(\boldsymbol{X}) = \begin{bmatrix} \mu_{1_X} \\ \vdots \\ \mu_{m_X} \end{bmatrix}$$

and

$$\boldsymbol{\mu}_Y = E(\boldsymbol{Y}) = \begin{bmatrix} \mu_{1_Y} \\ \vdots \\ \mu_{m_Y} \end{bmatrix}.$$

The joint covariance between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is given by (S p412, DG p341)

$$\boldsymbol{\Sigma} = \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}$$

This can be found by examining the $(m+p) \times 1$ vector

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \end{bmatrix}.$$

The term

$$\Sigma_{XX} = E(\boldsymbol{X} - \boldsymbol{\mu}_X)(\boldsymbol{X} - \boldsymbol{\mu}_X)'$$

can be interpreted as a within–set covariance, while the term

$$\Sigma_{XY} = E(\boldsymbol{X} - \boldsymbol{\mu}_X)(\boldsymbol{Y} - \boldsymbol{\mu}_Y)'$$

is a between–set covariance.

Similar comments apply to Σ_{YY} and Σ_{YX}.

Note that the term Σ_{YX} is *not* the same as Σ_{XY}, although the items involved are identical.

The covariance Σ_{YX} is $p \times m$ while Σ_{XY} is $m \times p$, as shown in Figure 7.1.

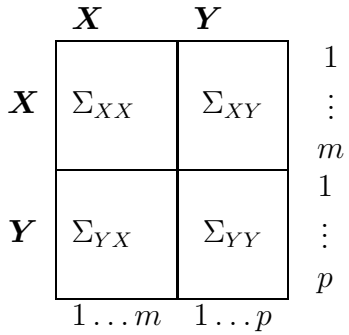

Figure 7.1: The covariance matrix for the X–set and the Y–set

Also,

$$\Sigma_{XY} = E(\boldsymbol{X} - \boldsymbol{\mu}_X)(\boldsymbol{Y} - \boldsymbol{\mu}_Y)'$$

but

$$\Sigma_{YX} = E(\boldsymbol{Y} - \boldsymbol{\mu}_Y)(\boldsymbol{X} - \boldsymbol{\mu}_X)'.$$

7.2.2 Derivation

The canonical variables $\boldsymbol{X}^*$ and $\boldsymbol{Y}^*$ are defined as

$$\boldsymbol{X}^* = \boldsymbol{a}'\boldsymbol{X} = a_1 X_1 + \ldots + a_m X_m$$

and

$$\boldsymbol{Y}^* = \boldsymbol{b}'\boldsymbol{Y} = b_1 Y_1 + \ldots + b_p Y_p.$$

If the variables $\boldsymbol{X}^*$ and $\boldsymbol{Y}^*$ are *normalized* then $\boldsymbol{a}$ and $\boldsymbol{b}$ are chosen so that

$$E(\boldsymbol{X}^*) = 0 = E(\boldsymbol{Y}^*)$$

and

$$V(\boldsymbol{X}^*) = 1 = V(\boldsymbol{Y}^*).$$

The canonical correlation between the new (canonical) variables $\boldsymbol{X}^*$ and $\boldsymbol{Y}^*$ is given by

$$\rho_{\boldsymbol{X}^*,\boldsymbol{Y}^*(\boldsymbol{a},\boldsymbol{b})} = \frac{\boldsymbol{a}'\Sigma_{XY}\boldsymbol{b}}{\left[(\boldsymbol{a}'\Sigma_{XX}\boldsymbol{a})(\boldsymbol{b}'\Sigma_{YY}\boldsymbol{b})\right]^{1/2}}$$

in general. Note that

$\boldsymbol{a}'\Sigma_{XY}\boldsymbol{b}$ is the covariance between $\boldsymbol{X}^*$ and $\boldsymbol{Y}^*$,

$\boldsymbol{a}'\Sigma_{XX}\boldsymbol{a}$ is the variance of $\boldsymbol{X}^*$, and

$\boldsymbol{b}'\Sigma_{YY}\boldsymbol{b}$ is the variance of $\boldsymbol{Y}^*$.

All three quantities are scalars.

If the canonical variables are normalized, then

$$\rho_{\boldsymbol{X}^*,\boldsymbol{Y}^*(\boldsymbol{a},\boldsymbol{b})} = \boldsymbol{a}'\Sigma_{XY}\boldsymbol{b}$$

as per Sharma p413.

The maximisation of $\rho_{\boldsymbol{X}^*,\boldsymbol{Y}^*(\boldsymbol{a},\boldsymbol{b})}$ wrt to $\boldsymbol{a}$ and $\boldsymbol{b}$ leads to the eigenvalue problems

$$\left(\Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} - \lambda\boldsymbol{I}\right)\boldsymbol{a} = 0$$

and

$$\left(\Sigma_{YY}^{-1}\Sigma_{YX}\Sigma_{XX}^{-1}\Sigma_{XY} - \lambda\boldsymbol{I}\right)\boldsymbol{b} = 0$$

after Sharma p412-415, DG p341. The solutions to these two problems are interchangeable since it can be shown that

$$\boldsymbol{a} = \frac{\Sigma_{XX}^{-1}\Sigma_{XY}\boldsymbol{b}}{\sqrt{\lambda}}$$

and

$$\boldsymbol{b} = \frac{\Sigma_{YY}^{-1}\Sigma_{YX}\boldsymbol{a}}{\sqrt{\lambda}}$$

as per DG p342. Hence $\widehat{\boldsymbol{a}}$ can be obtained from $\widehat{\boldsymbol{b}}$ and vice versa.

Exercise

Derive the DG p342 result for the relations between $\boldsymbol{a}$ and $\boldsymbol{b}$.

	Sharma p399	R
X_1	0.424 602	0.088 534
X_2	0.668 993	0.139 496
	X_1^*	X_1^*

Table 7.1: Canonical coefficients from S p399 versus R

	Sharma p399	R
r_1	0.924 475	0.924 475 4
r_2	0.012 376	0.102 377 7

Table 7.2: Squared canonical correlations from S p399 and R

7.2.3 Simple example

To implement the procedure in R, the library `mva` needs to be loaded, and the function `cancor` invoked. The example on S p392 is reproduced, and calculations performed to verify that the same results have been obtained.

The coefficients for the first canonical variable in the X–set are shown from S p399 and from the R Output in Table 7.1.

Manual calculations are also presented by S on P415,416.

The coefficients in Table 7.1 can be seen to be equivalent since

$$\frac{0.424\,602}{0.668\,993} = 0.634\,688, \quad \frac{0.088\,534}{0.139\,496} = 0.634\,671.$$

The squared canonical correlations from S p399 and R are given in Table 7.2, again verifying equivalence.

R Output :

```
> dat <- read.table("tab131.dat",header=T)
> dat
       x1     x2     y1     y2
1   1.051 -0.435  0.083  0.538
2  -0.419 -1.335 -1.347 -0.723
3   1.201  0.445  1.093 -0.112
4   0.661  0.415  0.673 -0.353
5  -1.819 -0.945 -0.817 -1.323
6  -0.899  0.375 -0.297 -0.433
7   3.001  1.495  1.723  2.418
8  -0.069 -2.625 -2.287 -1.063
9  -0.919  0.385 -0.547  0.808
10 -0.369 -0.265 -0.447 -0.543
11 -0.009 -0.515  0.943 -0.633
12  0.841  1.915  1.743  1.198
13  0.781  1.845  1.043  2.048
```

```
14  0.631 -0.495  0.413 -0.543
15 -1.679 -0.615 -1.567 -0.643
16 -0.229 -0.525 -0.777 -0.252
17 -0.709 -0.975  0.523 -0.713
18 -0.519  0.055 -0.357  0.078
19  0.051  0.715  0.133  0.328
20  0.221  0.245  0.403  0.238
21 -1.399 -0.645 -0.817 -1.133
22  0.651  0.385  1.063 -0.633
23 -0.469 -0.125 -0.557 -0.393
24  0.421  1.215 -0.017  1.838
> x <- cbind(dat[,1],dat[,2])
> y <- cbind(dat[,3],dat[,4])
> library(mva)
> cancor(x,y)
$cor
[1] 0.9614964 0.1112553

$xcoef
            [,1]       [,2]
[1,] 0.08853495 -0.2150704
[2,] 0.13949558  0.1914804

$ycoef
          [,1]       [,2]
[1,] 0.1125829 -0.2180069
[2,] 0.1207477  0.2156653

$xcenter
[1]  0.0001666667 -0.0004166667

$ycenter
[1]  8.333333e-05 -4.166667e-05

> xy.can <- cancor(x,y)
> xy.can$cor * xy.can$cor
[1] 0.92447542 0.01237774
```

While canonical correlation is a method in its own right, it is also a unifying concept that connects many other multivariate dependence methods.

7.3 Relation to other methods

Canonical correlation can be considered an 'umbrella' technique in that it is a generalisation of many of the classical multivariate inferential procedures. Some of these

Technique	condition
SLR	$p = m = 1$
MLR	$p = 1$
MANOVA	$\boldsymbol{X}$ variables categorical
DISCRIMINANT ANALYSIS	$\boldsymbol{Y}$ variables categorical
Multivariate regression	-

Table 7.3: Techniques related to canonical correlation

relations will be displayed empirically. A brief overview is given in Sharma (13.6) p 409. The subsumed techniques are shown in Table 7.3.

There is an obvious duality between discriminant analysis and MANOVA.

7.4 Empirical demonstration

This simple example from JW p531(3rd ed) will be used not only to demonstrate the correspondence between canonical correlation and the techniques in Section 7.3, but also to show how to obtain the underlying eigenvalues. The data are :

```
d1 d2 d3  g  x1  x2
 1  0  0  1  -2   5
 1  0  0  1   0   3
 1  0  0  1  -1   1
 0  1  0  2   0   6
 0  1  0  2   2   4
 0  1  0  2   1   2
 0  0  1  3   1  -2
 0  0  1  3   0   0
 0  0  1  3  -1  -4
```

where 'g' denotes group membership, and 'd1', 'd2' and 'd3' are the corresponding dummy variables. The bivariate response is given by 'x1' and 'x2'.

So a canonical correlation using two of the dummy variables and (x1,x2) as the second set should reproduce the results of a discriminant analysis and a MANOVA. R output is given showing all the required combinations. Some cross referencing calculations are presented.

```
> dat <- read.table("jw1010.dat",header=T)
> dat
  d1 d2 d3 g x1 x2
1  1  0  0 1 -2  5
2  1  0  0 1  0  3
3  1  0  0 1 -1  1
```

```
4  0  1  0 2  0  6
5  0  1  0 2  2  4
6  0  1  0 2  1  2
7  0  0  1 3  1 -2
8  0  0  1 3  0  0
9  0  0  1 3 -1 -4
> x <- cbind(dat[,5],dat[,6])
> y <- cbind(dat[,1],dat[,2])
> library(mva)
> cancor(x,y)
$cor
[1] 0.8610496 0.6891215

$xcoef
            [,1]        [,2]
[1,] -0.08005356  0.27749881
[2,] -0.10267684 -0.03311574

$ycoef
           [,1]       [,2]
[1,] -0.5032586 -0.6429599
[2,] -0.8084489  0.1143548

$xcenter
[1] 0.000000 1.666667

$ycenter
[1] 0.3333333 0.3333333

> cancor(y,x)
$cor
[1] 0.8610496 0.6891215

$xcoef
          [,1]       [,2]
[1,] 0.5032586 -0.6429599
[2,] 0.8084489  0.1143548

$ycoef
           [,1]        [,2]
[1,] 0.08005356  0.27749881
[2,] 0.10267684 -0.03311574

$xcenter
[1] 0.3333333 0.3333333
```

```
$ycenter
[1] 0.000000 1.666667

> gp <- factor(dat[,4])
> gp
[1] 1 1 1 2 2 2 3 3 3
Levels:  1 2 3
> manova(x~gp)
Call:
   manova(x ~ gp)

Terms:
                gp Residuals
resp 1           6         6
resp 2          62        24
Deg. of Freedom  2         6

Residual standard error: 1 2
Estimated effects may be unbalanced
> man.x <- manova(x~gp)
> summary(man.x)
          Df Pillai approx F num Df den Df  Pr(>F)
gp         2 1.2163   4.6559      4     12 0.01683 *
Residuals  6

> mv <- summary.manova(man.x)
> mv$Eigenvalues
       [,1]      [,2]
gp 2.867071 0.9043575
> library(MASS)
> lda(gp~x)
Call:
lda.formula(gp ~ x)

Prior probabilities of groups:
        1         2         3
0.3333333 0.3333333 0.3333333

Group means:
  x1 x2
1 -1  3
2  1  4
3  0 -2

Coefficients of linear discriminants:
          LD1        LD2
```

```
x1 -0.3856092  0.9380176
x2 -0.4945830 -0.1119397

Proportion of trace:
   LD1    LD2
0.7602 0.2398
> lda.x <- lda(gp~x)
> lda.x$svd
[1] 2.932783 1.647141
> lda.x$svd*lda.x$svd
[1] 8.601213 2.713072
```

7.4.1 Discriminant analysis

The results reported in Table 7.4 show the correspondence between the linear discriminant function (LD1) and the first canonical variable in the X–set, as determined from the R output from Section 7.4.

	Discriminant Function		Canonical variable	
	LD1	ratio (1/2)	xcoef(1)	ratio (1/2)
x1	-0.385 609	0.779 665	-0.080 053	0.779 666
x2	-0.494 583		-0.102 676	

Table 7.4: Discriminant function 1 vs first canonical variable in the X–set

Notice the *duality* between using `cancor` for discriminant analysis or MANOVA, ie,

$$\texttt{cancor(x,y)} \equiv \texttt{cancor(y,x)}.$$

7.4.2 Eigenvalues

To be able to conduct tests for discriminant functions and other test associated with MANOVA, we need the eigenvalues used to construct the discriminant function, ie, λ from the solution to

$$\left|\boldsymbol{W}^{-1}\boldsymbol{B} - \lambda\boldsymbol{I}\right| = 0$$

where $\boldsymbol{W}$ is the within–group covariance matrix, and $\boldsymbol{B}$ is the between–group covariance matrix. These eigenvalues can be obtained from the `cancor` output in R via the canonical correlations. Note from the SAS output (Sharma p399), the 'eigenvalues' are for "inv(E) * H", ie, $\boldsymbol{W}^{-1}\boldsymbol{B}$. Also, Wilks $\Lambda = 1 - c^2$ where c = the canonical correlation. Thus

$$\Lambda = \frac{1}{1+\lambda} = 1 - c^2$$

$$1 + \lambda = \frac{1}{1-c^2}$$

$$\lambda = \frac{1}{1-c^2} - 1 = \frac{1-(1-c^2)}{1-c^2} = \frac{c^2}{1-c^2}$$

which gives the SAS formula on p399 of Sharma.

Returning to the JW problem, we have

$$r_1 = 0.8610496, \rightsquigarrow \lambda_1 = 2.867070$$

while

$$r_2 = 0.6891215, \rightsquigarrow \lambda_2 = 0.904356$$

so the proportion of trace given in R becomes

$$\frac{2.867070}{2.867070 + 0.904356} = 0.7602$$

as required. Thus we can obtain the eigenvalues for discriminant analysis (and MANOVA) from the output of `cancor`.

7.4.3 Manova check

The Pillai trace in terms of the eigenvalues is

$$\sum_i \frac{\lambda_i}{1+\lambda_i} = \frac{2.867070}{3.867070} + \frac{0.905356}{1.904356} = 0.741406 + 0.474888 = 1.216294(1.2163\ R)$$

7.5 Tests using eigenvalues

The tests using eigenvalues relate to :

1. tests on the significance of the discriminant function, and
2. tests on the significance of canonical correlations.

7.5.1 Discriminant function tests

For a test on the significance of discriminant functions, the null hypothesis H_0 states that none of the discriminants are significant, ie, that they are all equal.

The test statistic V is

$$V = -\left[(n-1) - (p+k)/2\right] \ln \Lambda \sim \chi^2_{p(k-1)}$$

where

$$\Lambda = \prod_i \frac{1}{1+\widehat{\lambda}_i}$$

the number of variables is p and k = the number of groups.

References are DG p404 and S p252/299.

So operationally the test statistic becomes

$$V = \left[(n-1) - (p+k)/2\right] \sum_i \ln(1+\widehat{\lambda}_i)$$

An approximate test of significance for an individual discriminant (jth) is

$$V_j = [(n-1) - (p+k)/2] \ln(1+\widehat{\lambda}_j) \sim \chi^2_{p+k-2j}.$$

A sequence of tests gives the best form of test as shown in Table 7.5.

Term	df	Δ df $= p+k-2j$	j
V(all)	$p(k-1)$	Overall test	0
$V - V_1$	$(p-1)(k-2)$	$p+k-2$	1
$V - V_1 - V_2$	$(p-2)(k-3)$	$p+k-4$	2
	$\vdots$		
	Stop at NS		

Table 7.5: Test sequence for significance of discriminant functions

Example

The small 3 group problem from JW will be used to demonstrate the testing procedure. For this problem $p = 2$, $k = 3$ and $n = 9$. From the results of `cancor` we have

$$\widehat{\lambda}_1 = 2.867070,\ \widehat{\lambda}_2 = 0.904356$$

which gives

$$\ln(1+\widehat{\lambda}_1) = 1.352497,\ \ln(1+\widehat{\lambda}_2) = 0.644144$$

ie

$$\sum_i \ln(1+\widehat{\lambda}_i) = 1.996641.$$

Thus

(V) :

$$V = (8 - 5/2) \times 1.996641 = 10.981525$$

with

$$df = p(k-1) = 2(3-1) = 4.$$

Since

$$\chi^2_{4,5\%} = 9.48\ (P = 0.0268)$$

then at least one discriminant is significant.

$(V - V_1)$:

$$V - V_1 = [(n-1) - (p+k)/2] \ln(1+\widehat{\lambda}_2) = 5.5 \times 0.644144 = 3.542792$$

$$\chi^2_{(p-1)(k-2)} = \chi^2_1$$

and since $\chi^2_{1,5\%} = 3.84 (P = 0.0598)$, the second discriminant is not significant.

Thus we retain only the *first* discriminant function.

NOTE :

The approximate test for the first component gives

$$V_1 = [(n-1) - (p+k)/2] \ln(1 + \widehat{\lambda}_1) = 5.5 \times 1.352497 = 7.438$$
$$df = 2 + 3 - 2 = 3$$

and since $\chi^2_3 = 7.82$ the first discriminant appears to be NS, but the individual tests are not as valid as the sequence.

7.5.2 Canonical correlation tests

The statistical significance of canonical correlation can be cast as $H_0 : \Sigma_{XY} = 0$, ie, all the canonical correlations are zero, S p402, JW p and DG p353. The test statistic is

$$V = -[(n-1) - (m+p+1)] \ln \Lambda \sim \chi^2_{mp}$$

for the overall test. To test correlations after the $(r-1)$th, use the statistic

$$V_r = -[(n-1) - (m+p+1)/2] \ln \Lambda_r \sim \chi^2_{(m+1-r)(p+1-r)}$$

where

$$\Lambda = \prod_{j=1} (1 - \widehat{\lambda}_j)$$

and

$$\Lambda_r = \prod_{i=r} (1 - \widehat{\lambda}_i).$$

Again, a sequence of tests is used.

Example

For the example from S p392/399,

$$r_1 = 0.92447542.\ r_2 = 0.01237774$$

and since $\Lambda = \prod_i (1 - c_i^2)$

$$\Lambda = (1 - 0.92447542)(1 - 0.01237774) = 0.07552458 \times 0.98762226 = 0.074589756.$$

Thus

$$\ln \Lambda = -2.959752095.$$

Overall test:

$$V = -[24 - 1 - (2 + 2 + 1)/2] \times \ln \Lambda = 20.5 \times 2.595752095 = 53.2129$$

and $\chi^2_{4,5\%} = 9.48$ and so reject H_0 : all correlations are zero.

Now to test correlations after ρ_1;

$$V_2 = -20.5 \times \ln 0.98762226 = 0.255$$

and since $\chi^2_{1,5\%} = 3.84$ accept H_0 : all correlations after ρ_1 are zero. Thus only the first canonical correlation is significant, as per S p402/403.

7.6 Worked Example

Finally, a worked example of canonical correlation as a method *per se*, is presented.

The data in the file `sons.txt` are the measurements on the first and second adult sons in a sample of 25 families (Mardia, p121). For each son, two variables were measured, head length and head breadth. The first son's measurements are taken as the X–set and those of the second son form the Y–set.

From the R Output, the first canonical correlation only is significant. The test statistic for testing the significance of all correlations gives $V = 20.96$ on 4 df, while the test of correlations remaining after the first gives $V = 0.062$ on 1 df.

The first canonical variables are

$$X_1^* = 0.0115hl1 + 0.0144hb1$$

and

$$Y_1^* = 0.0103hl2 + 0.0164hb2.$$

These are roughly the length sum and breadth sum for each brother, say "girth". These new variables are highly correlated ($r_1 = 0.7885$), indeed more so than any of the individual measures between brothers (max $r = 0.7107$). The second canonical variables are measuring differences between length and breadth ("shape"?), but these measures have little correlation between brothers.

R Output

```
> dat <- read.table("sons.txt",header=T)
> dat
   hl1 hb1 hl2 hb2
1  191 155 179 145
2  195 149 201 152
3  181 148 185 149
4  183 153 188 149
5  176 144 171 142
6  208 157 192 152
7  189 150 190 149
8  197 159 189 152
9  188 152 197 159
10 192 150 187 151
11 179 158 186 148
12 183 147 174 147
13 174 150 185 152
14 190 159 195 157
15 188 151 187 158
16 163 137 161 130
17 195 155 183 158
18 186 153 173 148
19 181 145 182 146
20 175 140 165 137
```

```
21 192 154 185 152
22 174 143 178 147
23 176 139 176 143
24 197 167 200 158
25 190 163 187 150
> attach(dat)
> x <- cbind(dat[,1],dat[,2])
> y <- cbind(dat[,3],dat[,4])
> library(mva)
> cor(x)
          [,1]      [,2]
[1,] 1.0000000 0.7345555
[2,] 0.7345555 1.0000000
> cor(y)
         [,1]     [,2]
[1,] 1.000000 0.839252
[2,] 0.839252 1.000000
> cor(x,y)
          [,1]      [,2]
[1,] 0.7107518 0.7039807
[2,] 0.6931573 0.7085504
> cancor(x,y)
$cor
[1] 0.7885079 0.0537397

$xcoef
           [,1]        [,2]
[1,] 0.01154653 -0.02857148
[2,] 0.01443910  0.03816093

$ycoef
           [,1]        [,2]
[1,] 0.01025573 -0.03595605
[2,] 0.01637533  0.05349758

$xcenter
[1] 185.72 151.12

$ycenter
[1] 183.84 149.24

> xy.can <- cancor(x,y)
> xy.can$cor
[1] 0.7885079 0.0537397
> xy.can$cor * xy.can$cor
[1] 0.621744734 0.002887956
```

7.6.1 Correlation tests

For the sons example,

$$r_1 = 0.7885079.\ r_2 = 0.0537397$$

and since $\Lambda = \prod_i (1 - c_i^2)$

$$\Lambda = (1 - 0.621744734)(1 - 0.002887956) = 0.378255 \times 0.997112 = 0.377163.$$

Thus

$$\ln \Lambda = -0.975079.$$

Overall test:

$$V = -[25 - 1 - (2 + 2 + 1)/2] \times \ln \Lambda = 21.5 \times 0.975079 = 20.96$$

and $\chi^2_{4,5\%} = 9.48$ and so reject H_0 : all correlations are zero.

Now to test correlations after ρ_1;

$$V_2 = -21.5 \times \ln 0.997112 = 0.062$$

and since $\chi^2_{1,5\%} = 3.84$ accept H_0 : all correlations after ρ_1 are zero. Thus only the first canonical correlation is significant.

You should now attempt Workshop 8.

Bibliography

[1] Dillon W.R. and Goldstein M., *Multivariate analysis : Methods and applications*, Wiley, New York, 1984. (*)

[2] Dunteman G.H., *Principal Components Analysis*, Sage, California, 1989.

[3] Everitt B.S., *Graphical techniques for multivariate data*, Heinemann, London, 1978.

[4] Everitt B.S., Landau S. and Leese M., *Cluster Analysis*, 4th Ed., Arnold, London, 2001.

[5] Flury B. and Riedwyl H., *Multivariate Statistics : A Practical Approach*, Chapman and Hall, 1994. (*)

[6] Gower J.C. and Hand D.J., *Biplots*, Chapman and Hall, London, 1996.

[7] Jobson J.D., Applied multivariate data analysis, Vol II Categorical and multivariate methods, Springer Verlag, 1992.

[8] Johnson D.E., *Applied Multivariate Methods for Data Analysis*, Duxbury, 1998.

[9] Johnson R.A. and Wichern D.W., *Applied multivariate statistical analysis*, 3rd Ed., Prentice Hall, N.J., 1992.

[10] Krzanowski W.J. and Marriott F.H.C., *Multivariate Analysis : Part 1 : Distributions, Ordination and Inference*, Kendall's Library of Statistics 1, Edward Arnold, London, 1994.

[11] Krzanowski W.J. and Marriott F.H.C., *Multivariate Analysis : Part 2 : Classification, Covariance Structures and Repeated Measurements*, Kendall's Library of Statistics 2, Edward Arnold, London, 1994.

[12] Morrison D.F., *Multivariate statistical methods*, 3rd Ed., McGraw–Hill, New York, 1990.

[13] Sharma S., *Applied Multivariate Techniques*, Wiley, New York, 1996.

[14] Venables W.N. and Ripley B.D., *Modern Applied Statistics with S–PLUS*, 3rd Ed., Springer, New York, 1999.

General Reading

1. Bilodeau M and Brenner D, *Theory of Multivariate Statistics*, Springer, New York, 1999.

2. Eaton M.L., *Multivariate Statistics*, Wiley, New York, 1983.

3. Everitt B.S., *Latent Variable Models*, Chapman and Hall, London, 1984.

4. Everitt B.S., *An R and S-PLUS Companion to Multivariate Analysis*, Springer, London, 2005.

5. Fang K.T. and Zhang Y.T., *Generalized Multivariate Analysis*, Springer, New York, 1990.

6. Kaufman L. and Rousseeuw P.J., *Finding Groups in Data*, Wiley, New York, 1990.