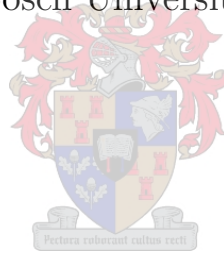


The identification and application of common principal components

Pieter Theo Pepler

Dissertation presented for the degree of
Doctor of Philosophy
in the Faculty of Economic and Management Sciences at
Stellenbosch University



Promotor: Dr. D.W. Uys
Co-promotor: Prof. D.G. Nel

December 2014

DECLARATION

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Pieter Theo Pepler

Date: December 2014

Opsomming

Wanneer die kovariansiematrikse van twee of meer populasies beraam word, word dikwels aanvaar dat die kovariansiematrikse óf gelyk, óf heeltemal onverwant is. Die gemeenskaplike hoofkomponente (GHK) model verskaf 'n alternatief wat tussen hierdie twee ekstreme aannames geleë is: Die aanname word gemaak dat die populasie kovariansiematrikse dieselfde versameling eievektore deel, maar verskillende versamelings eiewaardes het.

'n Belangrike vraag in die toepassing van die GHK model is om te bepaal of dit geskik is vir die data wat beskou word. Flury (1988) het twee metodes, gebaseer op aanneemlikheidsberaming, voorgestel om hierdie vraag aan te spreek. Die aanname van meerveranderlike normaliteit is egter ongeldig vir baie werklike datastelle, wat die toepassing van hierdie metodes bevraagteken. 'n Aantal nie-parametriese metodes, gebaseer op skoenlus-herhalings van eievektore, word voorgestel om 'n geskikte gemeenskaplike eievektor model te kies vir twee populasie kovariansiematrikse. Met die gebruik van simulatie eksperimente word aangetoon dat die voorgestelde seleksiemetodes beter vaar as die bestaande parametriese seleksiemetodes.

Indien toepaslik, kan die GHK model kovariansiematriks beramers verskaf wat minder sydig is as wanneer aanvaar word dat die kovariansiematrikse gelyk is, en waarvan die elemente kleiner standaardfoute het as die elemente van die gewone onsydige kovariansiematriks beramers. 'n Geregulariseerde kovariansiematriks beramer onder die GHK model word voorgestel, en Monte Carlo simulatie resultate toon dat dit meer akkurate beramings van die populasie kovariansiematrikse verskaf as ander mededingende kovariansiematriks beramers.

Kovariansiematriks beraming vorm 'n integrale deel van baie meerveranderlike statistiese metodes. Toepassings van die GHK model in diskriminantanalise, bi-stippings en regressie-analise word ondersoek. Daar word aangetoon dat, in gevalle waar die GHK model toepaslik is, GHK diskriminantanalise betekenisvol kleiner misklassifikasie foutkoerse lewer as beide gewone kwadratiese diskriminantanalise en lineêre diskriminantanalise. 'n Raamwerk vir die vergelyking van verskillende tipes bi-stippings vir data met verskeie groepe word ontwikkel, en word gebruik om GHK bi-stippings gekonstrueer vanaf gemeenskaplike eievektore met ander tipe hoofkomponent bi-stippings te vergelyk.

'n Deelversameling van data vanaf die Vermont Oxford Network (VON), van babas opgeneem in deelnemende neonatale intensiewe sorg eenhede in Suid-Afrika en Namibië gedurende 2009, word met behulp van die GHK model ontleed. Daar word getoon dat die voorgestelde nie-parametriese metodiek 'n verbetering op die bekende parametriese metodes bied in die

ontleding van hierdie datastel wat afkomstig is uit 'n nie-normaal verdeelde meerveranderlike populasie.

GHK regressie word vergelyk met hoofkomponent regressie en parsiiële kleinste kwadrate regressie in die passing van modelle om neonatale mortaliteit en lengte van verblyf te voorspel vir babas in die VON datastel. Die gepasde regressiemodelle, wat maklik bekombare dag-van-toelating data gebruik, kan deur mediese personeel en hospitaaladministrateurs gebruik word om ouers te adviseer en die toewysing van mediese sorg hulpbronne te verbeter. Voorspelde waardes vanaf hierdie modelle kan ook gebruik word in normwaarde oefeninge om die prestasie van neonatale intensiewe sorg eenhede in die Suider-Afrikaanse konteks, as deel van groter gehalteverbetering-programme, te evalueer.

Summary

When estimating the covariance matrices of two or more populations, the covariance matrices are often assumed to be either equal or completely unrelated. The common principal components (CPC) model provides an alternative which is situated between these two extreme assumptions: The assumption is made that the population covariance matrices share the same set of eigenvectors, but have different sets of eigenvalues.

An important question in the application of the CPC model is to determine whether it is appropriate for the data under consideration. Flury (1988) proposed two methods, based on likelihood estimation, to address this question. However, the assumption of multivariate normality is untenable for many real data sets, making the application of these parametric methods questionable. A number of non-parametric methods, based on bootstrap replications of eigenvectors, is proposed to select an appropriate common eigenvector model for two population covariance matrices. Using simulation experiments, it is shown that the proposed selection methods outperform the existing parametric selection methods.

If appropriate, the CPC model can provide covariance matrix estimators that are less biased than when assuming equality of the covariance matrices, and of which the elements have smaller standard errors than the elements of the ordinary unbiased covariance matrix estimators. A regularised covariance matrix estimator under the CPC model is proposed, and Monte Carlo simulation results show that it provides more accurate estimates of the population covariance matrices than the competing covariance matrix estimators.

Covariance matrix estimation forms an integral part of many multivariate statistical methods. Applications of the CPC model in discriminant analysis, biplots and regression analysis are investigated. It is shown that, in cases where the CPC model is appropriate, CPC discriminant analysis provides significantly smaller misclassification error rates than both ordinary quadratic discriminant analysis and linear discriminant analysis. A framework for the comparison of different types of biplots for data with distinct groups is developed, and CPC biplots constructed from common eigenvectors are compared to other types of principal component biplots using this framework.

A subset of data from the Vermont Oxford Network (VON), of infants admitted to participating neonatal intensive care units in South Africa and Namibia during 2009, is analysed using the CPC model. It is shown that the proposed non-parametric methodology offers an improvement over the known parametric methods in the analysis of this data set which originated from a non-normally distributed multivariate population.

CPC regression is compared to principal component regression and partial

least squares regression in the fitting of models to predict neonatal mortality and length of stay for infants in the VON data set. The fitted regression models, using readily available day-of-admission data, can be used by medical staff and hospital administrators to counsel parents and improve the allocation of medical care resources. Predicted values from these models can also be used in benchmarking exercises to assess the performance of neonatal intensive care units in the Southern African context, as part of larger quality improvement programmes.

Acknowledgements

Although the process of obtaining a PhD degree may seem like a solitary pursuit, mine certainly would not have been possible without the support of a number of mentors, colleagues, friends and family members. In this regard, I sincerely want to thank:

- Dr. Danie Uys, for stimulating my interest in Statistics in my undergraduate years, commitment to my research development, and friendship over the years;
- Prof. Daan Nel, for his wisdom and mentorship, encouragement, and deep understanding of mathematics;
- Prof. Tertius de Wet, for his mentorship over the years, on personal, work-related and subject matter levels;
- Prof. Sarel Steel, for his input and feedback on the research presented in this dissertation;
- Prof. Niel le Roux, for his mentorship, stimulating my interest in resampling methods, biplots, a geometric understanding of multivariate data, introducing me to R and statistical programming, and his feedback on the CPC biplots chapter;
- My friends at Mediclinic, for permission to use a subset of their data submitted to the Vermont Oxford Network (VON);
- Prof. Danie Brink and my colleagues at the Genetics Department, for giving me the necessary space and support to conduct my research;
- My parents, for always believing in me, their love, nurturing and encouragement, and stimulating my interest in books and learning;
- My wife, Hanli, for always believing in me, her love and encouragement, being by my side, putting up with all the late-night work, and making me a want to be a better person;
- My Lord and Saviour, Jesus Christ, for His provision, unconditional love and saving grace, and for creating the sea in which I find solace and rejuvenation.

SOLI DEO GLORIA

Contents

1	Introduction	1
1.1	Background	1
1.2	The Vermont Oxford Network data	3
1.3	Motivation for the study	6
1.4	Research approach	9
1.5	Dissertation outline	9
1.6	Notation	12
2	Principal component analysis	15
2.1	Introduction	15
2.2	Principal components in the population	16
2.3	Sample principal components	18
2.4	Inference on the eigenvectors and eigenvalues	19
2.5	Geometry of PCA	23
2.6	Standardisation of the variables	25
2.7	Number of principal components to retain	26
2.7.1	Subjective methods	26
2.7.2	Significance testing on the last $p - q$ components	27
2.7.3	Information in the last few components	30
2.8	Interpreting the eigenvectors	31
2.9	PCA as a variable selection technique	33
2.10	Application to the VON data	34
3	Common principal components	45
3.1	Introduction	45
3.2	The CPC model	46
3.3	Sample common principal components	47
3.4	Geometry of CPC	49
3.5	Simultaneous diagonalisation algorithms	50
3.6	Partial CPC	53
3.7	Flury's hierarchy	56

3.8	Inference for eigenvalues and common eigenvectors	57
3.8.1	Inference for the eigenvalues	58
3.8.2	Inference for the common eigenvectors	61
3.9	Interpreting the common eigenvectors	63
3.10	Other research related to the CPC model	64
3.11	Application to the VON data	66
3.11.1	Delivery mode	66
3.11.2	Regions	74
4	Identification of common eigenvectors	81
4.1	Introduction	81
4.2	Equality and proportionality	83
4.2.1	Testing for equality	85
4.2.2	Testing for proportionality	86
4.3	Identification of common eigenvectors	88
4.3.1	Chi-square test	91
4.3.2	Akaike Information Criterion (AIC)	92
4.3.3	Bootstrap hypothesis test (BootTest)	92
4.3.4	Random Vector Correlation (RVC)	93
4.3.5	Bootstrap Vector correlation Distribution (BVD)	94
4.3.6	Bootstrap Confidence Regions (BCR)	98
4.3.7	Ensemble method	100
4.3.8	Other methods	100
4.4	Simulation study	102
4.5	Application to known data sets	114
4.5.1	Bank notes data	115
4.5.2	Swiss heads data	121
4.5.3	Iris data	125
4.6	Application to the VON data	130
4.6.1	Delivery mode	131
4.6.2	Regions	134
5	Estimation of covariance matrices	137
5.1	Introduction	137
5.2	Accuracy of covariance matrix estimators	139
5.3	Estimating covariance matrices under the CPC model	140
5.4	CPC shrinkage estimator	141
5.5	Estimation of the shrinkage intensity parameter	142
5.5.1	Flury's ϕ method	142
5.5.2	Crossvalidation method	142
5.5.3	Schäfer and Strimmer method	143

CONTENTS

iii

5.6	Simulation study	145
5.6.1	Full CPC case	148
5.6.2	Half of eigenvectors common case	156
5.6.3	Few common eigenvectors case	163
5.6.4	Unrelated covariance matrices case	170
5.6.5	Effect of correlations among the variables	177
5.7	Application to the VON data	177
5.7.1	Delivery mode	179
5.7.2	Regions	181
6	CPC discriminant analysis	183
6.1	Introduction	183
6.2	Discriminant analysis under the CPC model	185
6.3	Regularized discriminant analysis	187
6.4	Shapes of the covariance matrix estimates	189
6.5	Simulation study	192
6.5.1	$p = 2$ variables case	192
6.5.2	$p = 5$ variables case	194
6.5.3	$p = 10$ variables case	197
6.6	Application to the VON data	199
6.6.1	Delivery mode	200
6.6.2	Regions	200
6.6.3	Mortality	201
7	CPC biplots	203
7.1	Introduction	203
7.2	PCA biplots	206
7.3	PC biplots for data with distinct groups	207
7.3.1	Pooled covariance matrix	207
7.3.2	Pooled data	208
7.3.3	Common eigenvectors	209
7.4	PC biplot measures of fit	209
7.4.1	Overall quality	210
7.4.2	Within-group quality	210
7.4.3	Between-group quality	212
7.4.4	Adequacy of the variables	213
7.4.5	Axis predictivities	214
7.4.6	Mean standard predictive errors	215
7.4.7	Sample predictivities	217
7.5	Comparison of PC biplots	217
7.5.1	Simulated data	219

7.5.2	Iris data	227
7.5.3	Bank notes data	230
7.6	Application to the VON data	234
7.6.1	Delivery mode	234
7.6.2	Regions	236
7.6.3	Mortality	238
8	Regression modelling of the VON data	241
8.1	Introduction	241
8.2	Regression models for data with groups	242
8.2.1	Multiple linear regression	242
8.2.2	Principal component regression	244
8.2.3	Common principal component regression	245
8.2.4	Partial least squares regression	246
8.3	Comparison of model fit	248
8.4	Application to the VON data	249
8.4.1	Mortality	251
8.4.2	Length of stay	257
8.5	Differences between PC and CPC regression	265
9	Summary	269
9.1	Results and new developments	269
9.2	Future research	274
Appendix A	Simulating CPC and CPC(q) data	277
A.1	Simulating data for a specified eigenvector structure	277
A.2	A method for simulating multivariate non-normal data	278
Appendix B	R code	281
B.1	The <code>cpc</code> package	281
B.1.1	Auxiliary functions	281
B.1.2	Simulation functions	284
B.1.3	Simultaneous diagonalisation algorithms	285
B.1.4	Identify common eigenvector functions	290
B.1.5	Covariance matrix estimation functions	309
B.1.6	Discriminant analysis function	312
B.1.7	Biplot functions	314
B.1.8	PLS regression function	322
B.2	VON data analysis script	324
B.2.1	Load the necessary R packages	324
B.2.2	Read in the VON data files	324

CONTENTS

v

B.2.3	Chapter 1	324
B.2.4	Chapter 2	325
B.2.5	Chapter 3	329
B.2.6	Chapter 4	340
B.2.7	Chapter 5	341
B.2.8	Chapter 6	343
B.2.9	Chapter 7	344
B.2.10	Chapter 8	346
Appendix C	Chapter 5: Eigenvalues	359
Appendix D	Chapter 6: Covariance matrices	365
References		371

Chapter 1

Introduction

1.1 Background

When comparing two or more populations it is often necessary to make an assumption about the variances of the populations. For the univariate case the choice is relatively simple: The population variances can either be equal or not. In the multivariate setting, the covariance matrices describe not only the variation of each of the variables, but also the covariances amongst the variables. Because of the increased complexity of this parameter of the population variation, there are a larger number of ways in which the covariance matrices of several groups can differ. For example, if the assumption of equal population covariance matrices seem untenable, the shapes and orientation of the multidimensional clouds of points they represent may still be the same and the assumption of proportional covariance matrices might be appropriate.

Even if the proportionality assumption seems questionable, there remains a number of possible similarities among the covariance matrices which have to be ruled out before we assume they are unequal. One of the options is the *common principal component (CPC)* model, proposed by Flury (1984). The CPC model assumes a shared set of eigenvectors in the covariance matrices of several multivariate populations, while allowing for individual sets of eigenvalues. It is thus less constrictive than assuming equality of the covariance matrices, but makes use of similarities in the covariance structures in a better way than the assumption of total heterogeneity.

Since its proposal, the CPC model was found to be useful in numerous applications, often in a biometrical context (for an example, see Klingenberg (1996)). In a meteorological context, Sengupta and Boyle (1998) employed the CPC model to compare the results from a number of atmospheric general

circulation models. Coffey et al. (2011) used a related technique known as common functional principal component analysis (CFPCA) to study human movement data in a functional form.

With regards to quantitative genetics research, the CPC model has also proved useful in modelling genetic covariance matrices and studying the morphology patterns of several groups of related organisms. See Ackermann and Cheverud (2000), Arnold and Phillips (1999), Phillips and Arnold (1999), Phillips et al. (2001), Cheverud and Marroig (2007), Stepan (1997), Stepan et al. (2002), and Waldmann and Andersson (2000)) in this regard.

In the case of this dissertation, interest in the CPC model was prompted by the analysis of a data set from the Vermont Oxford Network (VON) database, which is discussed in more detail in the next section. The observations in this data set may be grouped into a number of easily identifiable groups, based on qualitative characteristics. From an initial comparison of the covariance matrices of the groups it was seen that they share some similarities which make the CPC model appropriate.

Methods for inference on the parameters of the CPC model usually rely on maximum likelihood theory, based on the assumption of multivariate normality in the population distributions. These include the methodology proposed by Flury (1988) and Yuan and Bentler (1994). Robust methods have been proposed by Boente and Orellana (2001), Boik (2002) and Boente et al. (2009), but these still make use of some distributional assumptions. Additionally, as pointed out by Jolliffe (2002), the asymptotic theory results are only applicable to CPC analysis on covariance matrices, and are not necessarily valid for correlation matrices on which principal component analysis (PCA) and CPC analysis are often performed in practice.

Flury (1988) has shown that the similarities between the covariance structures of several groups may be summarised in a hierarchy of models, from equality of the covariance matrices on the one extreme, to complete heterogeneity on the other. An additional problem then presents itself in deciding which of the models will provide the best fit for the data. The methodology proposed by Flury (1988) for identifying the best fitting model also relies on the multivariate normality assumption. This assumption is often not valid for many real data sets, including the VON data analysed in this dissertation.

Flury (1988) suggested using bootstrap or jackknife techniques to estimate the error distributions of the common eigenvectors and eigenvalues for correlation matrices from several groups, which is the approach followed in this dissertation. The asymptotic distributions of these quantities are difficult to derive for the single group case, and are not yet known for the CPC analysis of several groups.

1.2 The Vermont Oxford Network data

The Vermont Oxford Network (VON) is a non-profit initiative with the purpose of improving the quality and safety of medical care for newborn infants (Vermont Oxford Network, 2009). This is done through research on data collected at over 900 neonatal intensive care unit (NICU) centres around the world, including 55 centres in Southern Africa.

According to the VON database rules, any infant with a birth weight of between 401 and 1500 grams, or whose gestational age was between 22 weeks 0 days and 29 weeks 6 days is eligible for entry into the VON database, regardless of where in the hospital the infant receives care. Furthermore, any infant with a birth weight of over 1500 grams, who was admitted to a NICU within the first 28 days of life without first having gone home, is eligible for inclusion. A NICU is defined as any location within the hospital where newborn infants receive continuous positive airway pressure (CPAP) or intermittent mandatory ventilation (IMV), (Vermont Oxford Network, 2009).

Data from the VON database for 2376 infants born in the period 1 January 2008 to 31 December 2008, and 3041 infants born in the period 1 January 2009 to 31 December 2009 were obtained with permission from the Mediclinic private hospital group. These infants were admitted during the aforementioned periods for observation and/or treatment to one of eighteen participating Mediclinic NICU centres located in South Africa and Namibia. The methodology developed in this dissertation is focused on investigating the characteristics of the 2009 cohort, with the data for the 2008 cohort only used as a test set to determine the predictive ability of the regression models in Chapter 8.

A number of perinatal input variables which under normal circumstances are readily available upon admission to a NICU were used for the analyses. The six numerical measures are Apgar scores (*AP1* and *AP5*, on a scale from 0–10) at one and five minutes after birth, respectively, temperature measured within one hour after birth (*ATEMP*, in °C), birth weight (*BWGT*, in gram), gestational age (*GESTAGE*, in days, converted to weeks) and birth head circumference (*BHEADCIR*, in cm). The qualitative variables, used to investigate differences between groups, are delivery mode (*Caesarean/Vaginal*), gender (*Male/Female*), maternal ethnicity (*Black/White/Asian/Other*), and hospital region (*South Africa/Namibia*). In the Southern African context, the *Other* maternal ethnicity group refers mainly to individuals from the mixed ancestry coloured population.

From Table 1.1 it can be seen that the two cohorts of 2008 and 2009 are comparable in terms of size and the variables of interest. A number of infants

died before discharge from the hospital (*Died*) and a number of infants were transferred to alternative hospitals before their final discharge (*Transferred*). These cases were excluded from the majority of the analyses presented in this dissertation.

The measurement scales of the numerical variables in the VON data set are not commensurate, with the birth weight variable dominating the rest in terms of absolute measurement units. This can be seen from inspection of the covariance matrix of the 2009 cohort:

	BWGT	AP1	AP5	GESTAGE	BHEADCIR	ATEMP
BWGT	679328.114	329.085	264.222	2438.469	2298.624	191.550
AP1	329.085	3.706	2.376	1.472	1.378	0.285
AP5	264.222	2.376	2.460	1.290	1.090	0.253
GESTAGE	2438.469	1.472	1.290	12.582	9.490	0.748
BHEADCIR	2298.624	1.378	1.090	9.490	10.809	0.708
ATEMP	191.550	0.285	0.253	0.748	0.708	0.566

In practice, the correlation rather than the covariance matrices are often used to perform PCA in a situation such as this. Rencher (1998) cautions against routinely following this route of analysis, for reasons of interpretability and using the principal component scores as input for further calculations. Interpretation of components accounting for a specific proportion of the overall observed variation is meaningful for covariance matrices but not for correlation matrices. Furthermore, if the principal components of the correlation matrix is transformed to the original variable scales, they will not be orthogonal anymore. It is therefore preferable to use the principal components of the covariance matrix for further applications such as regression modelling.

With regards to CPC analysis Flury (1988) noted that simply using the correlation matrices from several groups as input for the Flury-Gautschi (FG) algorithm to find the common eigenvectors may lead to estimates which are not maximum likelihood estimates.

It was therefore decided to scale the birth weight variable in the VON data by dividing the values by a thousand to obtain the birth weights in kilogram. The covariance matrix of the 2009 cohort then looks as follows:

	BWGT	AP1	AP5	GESTAGE	BHEADCIR	ATEMP
BWGT	0.679	0.329	0.264	2.438	2.299	0.192
AP1	0.329	3.706	2.376	1.472	1.378	0.285
AP5	0.264	2.376	2.460	1.290	1.090	0.253
GESTAGE	2.438	1.472	1.290	12.582	9.490	0.748
BHEADCIR	2.299	1.378	1.090	9.490	10.809	0.708
ATEMP	0.192	0.285	0.253	0.748	0.708	0.566

Table 1.1: Summary measures for the two Vermont Oxford Network (VON) cohorts.

	2008	2009
NICU centers	15	18
Number of infants (n)	2376	3041
Transferred	69 (2.9%)	111 (3.7%)
Died	91 (3.8%)	104 (3.4%)
Medians		
Apgar score (1 min)	8	8
Apgar score (5 mins)	9	9
Temperature (°C)	36.3	36.2
Birth weight (gram)	2350	2326
Gestational age (weeks)	35.9	35.3
Birth head circumference (cm)	33	33
Proportions		
Male / Female	54% / 46%	53% / 47%
Caesarean / Vaginal	83% / 17%	84% / 16%
South Africa / Namibia	97% / 3%	96% / 4%
Maternal ethnicity:		
<i>Black</i>	42.1%	48.0%
<i>White</i>	44.0%	38.3%
<i>Asian</i>	1.6%	2.3%
<i>Other</i>	12.3%	11.4%

For the analyses on the VON data presented throughout this dissertation, the birth weight variable measured in kilogram was used. The empirical marginal distributions of the six numerical variables in the VON 2009 cohort are shown in Figure 1.1. It seems that all of the marginal distributions deviate from normality.

The infants were grouped according to either delivery mode (*Caesarean* / *Vaginal*), hospital region (*South Africa* / *Namibia*) or mortality status (*Survived* / *Died*) to illustrate the utility of the methods developed in this dissertation for data with distinct groups.

In addition to a comparison of the covariance structures of the natural groupings in this data set, it is of interest to investigate how the perinatal variables may be used to predict the clinical outcomes of the infants, particularly mortality and length of stay (LOS). In neonatology, early prediction of neonatal mortality and length of hospital stay may help in decision making (Zernikow et al. (1999), Hintz et al. (2010)). It is of interest to health care providers and hospital administrators for both economic and organisational reasons. Previous studies (Hintz et al. (2010), Altman et al. (2009)) have found a close relationship between cost of neonatal care and LOS.

Parents also have a strong interest to know the anticipated date of discharge of their preterm child. The physician caring for the newborn needs to answer questions regarding neonatal LOS shortly after the infant's birth in order to counsel parents (Jijon and Jijon-Letort, 1995). Although similar studies to determine the factors influencing neonatal mortality and LOS have been done in the past, hospital policies and clinical practice probably differ enough between hospitals and countries to make a study in Southern Africa relevant (Altman et al., 2009).

Furthermore, good predictions of LOS can provide a benchmark for measuring and comparing quality of care between different neonatal units (Zernikow et al., 1999). This in turn can stimulate quality improvement initiatives in neonatal care (Hintz et al., 2010) as part of an increasing emphasis on quality control in modern medicine.

1.3 Motivation for the study

Previous research has been done on inference on common principal components in k populations, as well as selection of the most appropriate model in Flury's hierarchy of covariance matrices, with the assumption that the distributions of the populations are multivariate normal. This assumption is not valid for many real data sets. Current methodology to perform statistical inference on common eigenvectors and their associated eigenvalues is

1.3. MOTIVATION FOR THE STUDY

7

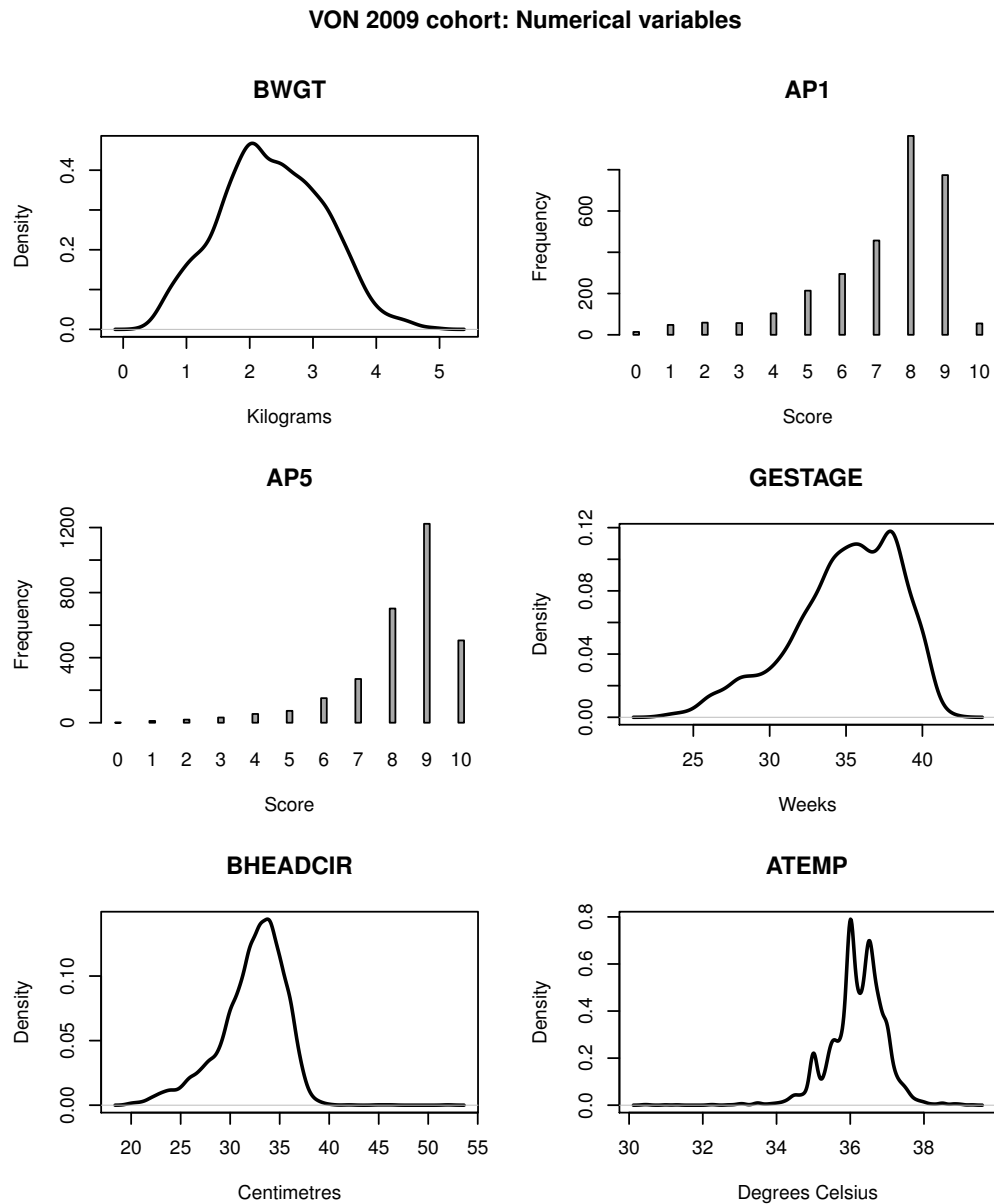


Figure 1.1: Empirical marginal distributions of the six numerical variables in the VON 2009 cohort. Density plots (smoothed histograms) and bar graphs are used to represent the continuous and discrete variables, respectively.

therefore not suitable. An important purpose of this study was to investigate the development of new statistical methodology to identify common principal components in k groups accurately for situations where the multivariate normality assumption is untenable.

Furthermore, the usefulness of the CPC model in providing estimates as input for other descriptive and predictive statistical methods has been hinted at by Flury (1988), but has not yet been explored in a comprehensive way. The second part of the study involved the application of the CPC model to improve a number of other statistical methods, namely estimation of covariance matrices, discriminant analysis, biplots and regression modelling. If appropriate, the CPC model can make use of information about commonalities in the covariance structures across the groups to provide improved estimates of the individual population covariance matrices. The impact of such improved covariance matrix estimates as input for the other statistical techniques (discriminant analysis, biplots and regression modelling) is yet to be investigated.

Even though the VON data is of clinical importance, this subset of data from Southern African NICUs has not been analysed thoroughly prior to this study. As discussed in Section 1.2, knowledge of the factors which influence infant mortality and LOS in NICUs, and the ability to reliably predict these outcomes at an early stage of hospitalisation can aid decision making for the stakeholders involved.

The data also have some features of statistical interest, such as clear natural groupings and non-normal multivariate distributions of these groups, which make it particularly suitable for illustration of the methods proposed in this dissertation. Another important purpose of the study is to determine whether the newly developed CPC analysis methods can provide a better understanding of the study population, compared to the results obtained from previous methods.

At the start of this research work, no software for conducting CPC analysis was publicly available in the R language and programming environment. Since then, two minor packages, *FGalgorithm* (Najarzadeh, 2013) and *cpca* (Ziyatdinov et al., 2014), have been added to the Comprehensive R Archive Network (CRAN). *FGalgorithm* contains an implementation of the Flury-Gautschi (FG) algorithm for the simultaneous diagonalisation of several covariance matrices (Flury and Gautschi, 1986), while *cpca* contains an implementation of the stepwise CPC algorithm proposed by Trendafilov (2010). The latest versions (from version 1.9-9 onwards) of the package *JADE* (Nordhausen et al., 2013) contains an implementation of a modified version of the FG algorithm.

An ancillary purpose of the study is to develop and test CPC analysis

software for the R programming environment, including functions for both the known methods and the new proposals. The software can be submitted to the CRAN to be made publicly available for use and modification by the wider scientific community.

1.4 Research approach

The approach taken in this study is to evaluate the available literature on the CPC model and its applications and outline the theory at the start of each chapter, followed by new developments made during the course of the research. The new proposals focus on statistical inference for the CPC model using methods which, compared to the known methods, make fewer assumptions about the distributions of the populations from which the data originated. In later chapters the CPC model is also applied in a novel way to a number of applications which have not been studied before in this context.

Monte Carlo simulation is employed to test and compare the newly proposed procedures to the known methods, using customised functions programmed in the statistical programming language R (R Development Core Team, 2011). The properties of the CPC model is studied under various covariance structure scenarios in large-scale simulation studies. For the application of the CPC model in other statistical methods, measures such as goodness of fit, error rates, efficiency and accuracy of the new methods were compared to that of the known (mostly parametric) methods.

Lastly, based on the theory and simulation results, the proposed methodology was applied to the VON data to illustrate the utility of the new methods. Where possible, the results from the new methods are compared to those from the known methods. Conclusions are drawn from the results and recommendations are made for future applications.

1.5 Dissertation outline

Following this introductory chapter, the theoretical basis of principal component analysis of a single group is explained in **Chapter 2**. A geometrical interpretation of PCA is given, as well as a number of methods to aid in deciding how many principal components to retain when PCA is used as a dimensionality reduction tool. Lastly, PCA is performed on the 2009 cohort of the VON data set.

The common principal component model is introduced in **Chapter 3**, again with an outline of the theory followed by a geometrical interpretation

of the model. A number of algorithms for the simultaneous diagonalisation of several covariance matrices are discussed. This is followed by an introduction to the partial CPC model, which is appropriate if only a subset of the eigenvectors are common to all of the groups. The theory and various methods for inference on the eigenvalues and common eigenvectors are given, followed by a discussion on the interpretation of the estimated common eigenvectors. After a number of references to other research related to the CPC model, the CPC model is applied to a number of the groups in the VON data.

Chapter 4 addresses the identification of common eigenvectors in several groups and the selection of the most appropriate covariance matrix model from Flury's hierarchy. After a short discussion on the properties of common eigenvectors and outlining a pragmatic approach to covariance matrix model selection, the likelihood ratio based tests for common eigenvectors are given. This is followed by two new non-parametric proposals, as well as proposed adaptations of two methods first mentioned by Klingenberg (1996) and Klingenberg and McIntyre (1998). The new non-parametric methods are shown to compare favourably with, and in most cases exceed the performance of the likelihood ratio tests proposed by Flury (1988) in a Monte Carlo simulation study. Results from an application of the proposed methods to a number of well-known data sets from Flury (1988) are given, followed by an application to the VON data to identify the number of common eigenvectors in the delivery mode and regional groups, respectively.

In **Chapter 5**, a new covariance matrix estimator is proposed under the CPC model. The CPC estimator suggested by Flury (1988) is given first, followed by the proposal of a James-Stein type of shrinkage estimator using the ordinary CPC estimator as the target matrix. Three methods for determining an appropriate value for the shrinkage intensity parameter is proposed. The new estimator is then compared to the ordinary CPC estimator and the unbiased covariance matrix estimator under various scenarios in a Monte Carlo simulation study, using a modified version of the Frobenius norm to judge the accuracy of the estimation. The proposed shrinkage estimator, which outperformed both the ordinary CPC estimator and the unbiased covariance matrix estimator, is also applied to the VON data to obtain estimates of the population covariance matrices.

The improved shrinkage estimator of the population covariance matrix under the CPC model is applied to discriminant analysis in **Chapter 6**. An outline of the idea behind CPC discriminant analysis is followed by a discussion of regularised discriminant analysis, which is a similar technique proposed by Friedman (1989). A heuristic explanation of the properties of the different covariance matrix estimators under a number of population covariance matrix scenarios is given. The results from a Monte Carlo simulation

study is discussed, where it is shown that the new covariance matrix shrinkage estimator outperforms the unbiased and pooled estimators under certain conditions.

The use of common eigenvectors in the construction of biplots for data with distinct groups is investigated in **Chapter 7**. After a brief introduction to principal component biplots, a number of biplot quality measures are evaluated. Due to the observation that the Type B orthogonality requirement for the calculation of axis predictivities is violated for CPC biplots, it is suggested to use the mean standard predictive error proposed by Rui Alves (2012) to quantify the goodness of fit of the variables in CPC biplots instead. Different orthogonal component biplots are then compared in terms of the quality measures on a number of artificial and real data sets, including the VON data.

The prediction of neonatal mortality and length of stay for the infants in the VON database is addressed in **Chapter 8**. Ways of fitting multiple linear regression models for data with distinct groups, including the concept of CPC regression, are discussed. Results for the different types of regression models to predict mortality and length of stay for NICU admissions in the VON data set are compared, and the chapter is concluded with some remarks on the differences between principal component (PC) and CPC regression.

Chapter 9 contains a summary of the main developments of this dissertation, together with some suggestions for future research.

Lastly, more details about the simulation methods, the most important R software developed for this dissertation, and eigenvalues and covariance matrices used in the simulation studies are given in appendices. An R script to replicate the analysis of the VON data presented in this dissertation is also given in Appendix B.

1.6 Notation

Below is a list of the notation and symbols used in this dissertation.

General

n_i	sample size of the i^{th} group
p	number of variables
k	number of groups
\mathbf{x}_i	stochastic vector in the i^{th} population
\mathbf{X}_i	$n \times p$ matrix of observations from the i^{th} group
Σ_i	$p \times p$ population covariance matrix of the i^{th} group
\mathbf{S}_i	$p \times p$ sample covariance matrix of the i^{th} group
$\ \mathbf{x}\ ^2$	inner product of vector \mathbf{x} with itself, i.e. $\mathbf{x}'\mathbf{x}$
$\mathbf{x} \sim N_p(\boldsymbol{\mu}, \Sigma)$	\mathbf{x} is distributed p -variate normal with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ

Principal components

\mathbf{E}_i	$p \times p$ population eigenvector matrix of the i^{th} group
Δ_i	$p \times p$ diagonal matrix of population eigenvalues (spectral matrix) of the i^{th} group
$\boldsymbol{\eta}_{ij}$	the j^{th} population eigenvector of the i^{th} group
δ_{ij}	the j^{th} population eigenvalue of the i^{th} group
\mathbf{E}_i	$p \times p$ sample eigenvector matrix of the i^{th} group
\mathbf{D}_i	$p \times p$ diagonal matrix of sample eigenvalues (spectral matrix) of the i^{th} group
\mathbf{e}_{ij}	the j^{th} sample eigenvector of the i^{th} group
d_{ij}	the j^{th} sample eigenvalue of the i^{th} group
y_{ij}	the j^{th} principal component of the i^{th} group
\mathbf{y}_{ij}	the vector of principal component scores for the j^{th} principal component of the i^{th} group

Common principal components

\mathbf{B}	$p \times p$ population common eigenvector matrix (modal matrix)
$\mathbf{\Lambda}_i$	$p \times p$ diagonal matrix of population eigenvalues (spectral matrix) for the i^{th} group under the CPC model
β_j	the j^{th} population common eigenvector
λ_{ij}	the j^{th} population eigenvalue for the i^{th} group under the CPC model
\mathbf{B}	$p \times p$ sample common eigenvector matrix (modal matrix)
\mathbf{L}_i	$p \times p$ diagonal matrix of sample eigenvalues (spectral matrix) for the i^{th} group under the CPC model
\mathbf{b}_j	the j^{th} sample common eigenvector
$\mathbf{b}_{(j)}$	the j^{th} row vector of the modal matrix, \mathbf{B}
l_{ij}	the j^{th} sample eigenvalue for the i^{th} group under the CPC model
z_j	the j^{th} common principal component
\mathbf{z}_{ij}	the vector of common principal component scores for the i^{th} group on the j^{th} common principal component

Other

Groups: $i = 1, \dots, k$.

Columns/variables: $j, h, u = 1, \dots, p$.

Observations: $m = 1, \dots, n$.

Matrix elements stacked in vectors:

$\text{vec}(\mathbf{S})$	a column vector containing the stacked columns of \mathbf{S} (i.e. the $\{s_{jh}\}_{j=1, \dots, p, h=1, \dots, p}$ elements)
$\text{vecs}(\mathbf{S})$	a column vector containing the stacked columns of the lower triangular part (including the diagonal) of \mathbf{S} (i.e. the $\{s_{jh}\}_{j=1, \dots, p, h \leq j, \dots, p}$ elements)

Chapter 2

Principal component analysis

2.1 Introduction

Principal component analysis (PCA) is a multivariate technique to construct uncorrelated components through linear combinations of the original variables. No assumptions are made about the multivariate distribution from which the data originated. The idea behind PCA was developed independently by Pearson (1901) and Hotelling (1933), but it was Hotelling who coined the name which is still widely used today. Rencher (2002, Chapter 12), Johnson and Wichern (2002, Chapter 8) and Anderson (2003, Chapter 11) provide good overviews, and Jolliffe (2002) a definitive book on the theory of PCA. Unless otherwise indicated, most of the material in this chapter is based on these four texts.

PCA entails the rotation of a cloud of multidimensional data points onto a set of orthogonal axes. The principal components of a data set are scores on orthogonal linear combinations of the original variables which describe the variation in the multidimensional data the best, in descending order of variance accounted for. In other words, the first principal component is the linear combination which accounts for the most of the observed variation in the data. The second principal component is the linear combination, orthogonal to the first, which accounts for the maximum of the remaining variation in the data (after the variation accounted for by the first component), and so forth. Because the set of principal component scores are uncorrelated, they can be useful as inputs for other multivariate techniques where multicollinearity poses difficulties.

For a data set in p -dimensional space (i.e. a data set with p variables), there will be p principal components needed to explain 100% of the observed variability in the data. However, unless the p -dimensional cloud of points

(as a representation of the data) is spherical, most of the variability will be accounted for by the first few principal components, especially if the original variables are highly correlated.

The analysis of principal components serves two main purposes. Firstly, inspection of which variables are “grouped together” (i.e. simultaneously have large coefficients) in each linear combination gives useful information about the data structure, as well as the correlations between the variables. This information can be used in deciding which variables to include for example in a regression model where high correlations between variables might lead to multicollinearity, inflating the error variance of the model. Alternatively, the correlated explanatory variables can be replaced with the uncorrelated principal components, leading to more stable estimates of the regression coefficients.

Secondly, because a large proportion of the observed variability in the data is often accounted for by the first few principal components, PCA can serve as a dimension reduction method by using only the first $q < p$ principal components (as an approximation of the variability represented by the p original variables) in subsequent analyses. In this application, PCA is not an end in itself, but rather an intermediate step to provide input for further analyses.

Sections 2.2 to 2.4 explain the theoretical foundation for PCA, together with methods for inference on eigenvalues and eigenvectors, which are important quantities used to calculate the principal components. A geometrical interpretation of PCA is given in Section 2.5. As the principal components calculated from covariance matrices are not scale invariant and depends on the measurement scales of the original variables, the variables can be standardised. This topic is discussed in Section 2.6.

When using PCA as a dimension reduction tool, a decision must be made on the number of principal components to retain. Subjective methods and some formal significance testing techniques for this purpose are given in Section 2.7.

Interpretation of the eigenvectors are discussed in Section 2.8, and the use of PCA as a variable selection method is briefly explained in Section 2.9.

Lastly, the PCA methods outlined in this chapter are applied to the VON 2009 cohort in Section 2.10.

2.2 Principal components in the population

Principal components are linear combinations of the p variables in the population. In the most basic form, PCA is a one-group technique with no

2.2. PRINCIPAL COMPONENTS IN THE POPULATION

17

grouping of variables into response and explanatory variable subsets.

Suppose there is a stochastic vector \mathbf{x} from a p -variate distribution with covariance matrix Σ . For the purposes of maximum likelihood estimation and the normal-theory based inference to follow in later sections, it is convenient to assume that \mathbf{x} originates from a multivariate normal distribution, i.e. $\mathbf{x} \sim N_p(\boldsymbol{\mu}, \Sigma)$.

Letting $y = \boldsymbol{\eta}'\mathbf{x}$ where $\boldsymbol{\eta}$ is a p -dimensional vector, y is a linear combination of the original p variables in \mathbf{x} and the variance of y is $\boldsymbol{\eta}'\Sigma\boldsymbol{\eta}$. To find the linear combination describing the first principal component of \mathbf{x} , the quantity

$$\delta_1 = \frac{\boldsymbol{\eta}'_1 \Sigma \boldsymbol{\eta}_1}{\boldsymbol{\eta}'_1 \boldsymbol{\eta}_1} \quad (2.1)$$

should be maximised, which is equivalent to the maximisation of $\boldsymbol{\eta}'_1 \Sigma \boldsymbol{\eta}_1$ if $\boldsymbol{\eta}_1$ is restricted to have unit length. Equation (2.1) can be written as

$$\delta_1 \boldsymbol{\eta}'_1 \boldsymbol{\eta}_1 = \boldsymbol{\eta}'_1 \Sigma \boldsymbol{\eta}_1 \quad (2.2)$$

which can be factorised as

$$\boldsymbol{\eta}'_1 (\Sigma \boldsymbol{\eta}_1 - \delta_1 \boldsymbol{\eta}_1) = 0. \quad (2.3)$$

Because $\boldsymbol{\eta}_1 = \mathbf{0}$ cannot be a solution to (2.1), (2.3) simplifies to

$$(\Sigma - \delta_1 \mathbf{I}) \boldsymbol{\eta}_1 = \mathbf{0}. \quad (2.4)$$

The maximum value for δ_1 in (2.4) is the first *eigenvalue* and the corresponding normalised vector $\boldsymbol{\eta}_1$ is the first *eigenvector* of Σ .

To obtain the rest of the eigenvectors and eigenvalues, the following procedure is performed $p-1$ times: At the j^{th} step, $\boldsymbol{\eta}_j$ is found as the normalised vector yielding a linear combination of the original p variables with maximum variance, subject to the constraint that it should be orthogonal to the first $j-1$ eigenvectors, $\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_{j-1}$.

As $\Sigma - \delta \mathbf{I}$ is singular,

$$|\Sigma - \delta \mathbf{I}| = 0 \quad (2.5)$$

will have p roots, $\delta_1 \geq \delta_2 \geq \dots \geq \delta_p$ (Anderson, 2003), of which the largest will be δ_1 as described in (2.4).

The j^{th} eigenvector, $\boldsymbol{\eta}_j$, is the normalised vector satisfying

$$(\Sigma - \delta_j \mathbf{I}) \boldsymbol{\eta}_j = \mathbf{0} \quad j = 1, \dots, p. \quad (2.6)$$

Thus, for the set of p eigenvalue-eigenvector pairs $(\delta_j, \boldsymbol{\eta}_j)$, the following two conditions should hold for a positive semi-definite matrix $\boldsymbol{\Sigma}$:

$$(1) \quad \delta_1 \geq \delta_2 \geq \dots \geq \delta_p \geq 0 \quad (2.7)$$

$$(2) \quad \boldsymbol{\eta}'_j \boldsymbol{\eta}_h = \begin{cases} 1, & j = h \\ 0, & j \neq h \end{cases} \quad (2.8)$$

Putting $\mathbf{E} = [\boldsymbol{\eta}_1 \quad \boldsymbol{\eta}_2 \quad \dots \quad \boldsymbol{\eta}_p]$,

$$\mathbf{E}'\boldsymbol{\Sigma}\mathbf{E} = \boldsymbol{\Delta} = \text{diag}(\delta_1, \delta_2, \dots, \delta_p). \quad (2.9)$$

The square eigenvector matrix \mathbf{E} therefore diagonalises the covariance matrix $\boldsymbol{\Sigma}$, and the spectral matrix $\boldsymbol{\Delta}$ contains the variances and covariances of the principal components of \mathbf{x} . Because the eigenvectors are orthogonal as described in (2.8), the off-diagonal elements of $\boldsymbol{\Delta}$ (i.e. the covariances of the principal components) will be equal to zero.

The p principal components are obtained by multiplying the original components with the transpose of the eigenvector matrix,

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}'_1 \mathbf{x} \\ \boldsymbol{\eta}'_2 \mathbf{x} \\ \vdots \\ \boldsymbol{\eta}'_p \mathbf{x} \end{bmatrix} \quad (2.10)$$

$$\Rightarrow \quad \mathbf{y} = \mathbf{E}'\mathbf{x}.$$

2.3 Sample principal components

Let $\mathbf{x}_1, \dots, \mathbf{x}_n$ be a simple random sample with regard to \mathbf{x} , and define the matrix,

$$\mathbf{X}_{n \times p} = \begin{bmatrix} \mathbf{x}'_1 \\ \mathbf{x}'_2 \\ \vdots \\ \mathbf{x}'_n \end{bmatrix}. \quad (2.11)$$

The unbiased estimator for the unknown covariance matrix $\boldsymbol{\Sigma}$ is

2.4. INFERENCE ON THE EIGENVECTORS AND EIGENVALUES 19

$$\mathbf{S} = \frac{1}{n-1} (\mathbf{X}'\mathbf{X} - n\bar{\mathbf{x}}\bar{\mathbf{x}}'), \quad (2.12)$$

where $\bar{\mathbf{x}}$ is a vector containing the column means of \mathbf{X} . Because the eigenvectors and eigenvalues are calculated from the covariance (or correlation) matrix, the value of $\bar{\mathbf{x}}$ is irrelevant here.

The first step in estimating the principal components of \mathbf{X} is to compute the spectral decomposition of \mathbf{S} . Denoting the p eigenvalue-eigenvector pairs of \mathbf{S} by $(d_1, \mathbf{e}_1), (d_2, \mathbf{e}_2), \dots, (d_p, \mathbf{e}_p)$ where $d_1 \geq d_2 \geq \dots \geq d_p \geq 0$, and letting $\mathbf{E} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_p]$, it follows that

$$\mathbf{E}'\mathbf{S}\mathbf{E} = \mathbf{D} = \text{diag}(d_1, d_2, \dots, d_p), \quad (2.13)$$

analogous to (2.9).

Again, because the eigenvectors are orthogonal, $\mathbf{e}_j'\mathbf{e}_h = 0$ for any $j \neq h$, the off-diagonal elements of \mathbf{D} (i.e. the covariances of the principal components) will be equal to zero. Furthermore, the eigenvectors are normalised so that $\mathbf{e}_j'\mathbf{e}_j = 1$ for any j .

The j^{th} sample principal component is the linear combination,

$$y_j = e_{j1}x_1 + e_{j2}x_2 + \dots + e_{jp}x_p, \quad j = 1, \dots, p, \quad (2.14)$$

where e_{jh} indicates the h^{th} loading in the j^{th} eigenvector, and x_h is the h^{th} element of the vector, \mathbf{x} .

The score for the m^{th} row of \mathbf{X} , \mathbf{x}_m , on the j^{th} principal component is thus obtained with

$$y_{jm} = \mathbf{e}_j'\mathbf{x}_m = e_{j1}x_{1m} + e_{j2}x_{2m} + \dots + e_{jp}x_{pm} \quad (2.15)$$

for $m = 1, \dots, n$. Letting \mathbf{E}_q contain the first $q \leq p$ eigenvectors of \mathbf{S} ,

$$\mathbf{Y}_q = \mathbf{X}\mathbf{E}_q \quad (2.16)$$

is the matrix containing the scores of \mathbf{X} on the first q principal components. The principal component scores can be used in further analyses, such as regression, MANOVA or biplots.

2.4 Inference on the eigenvectors and eigenvalues of Σ

Assume that $\mathbf{X}_{n \times p}$ is the data matrix of a large random sample from a $N_p(\boldsymbol{\mu}, \Sigma)$ distribution, and that the unknown eigenvalues of Σ are positive and well separated. The sample eigenvectors (i.e. the columns of $\mathbf{E} =$

$[\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_p]$) and eigenvalues (d_1, d_2, \dots, d_p) of \mathbf{S} , as defined in (2.12), has the following properties (Johnson and Wichern, 2002):

- (a) $\sqrt{(n-1)}(\mathbf{d} - \boldsymbol{\delta})$ is approximately $N_p(\mathbf{0}, 2\boldsymbol{\Delta}^2)$, where $\boldsymbol{\Delta}$ is the diagonal matrix of the eigenvalues $\delta_1, \delta_2, \dots, \delta_p$ of $\boldsymbol{\Sigma}$, and $\boldsymbol{\delta}$ and \mathbf{d} are the diagonals of $\boldsymbol{\Delta}$ and \mathbf{D} respectively. This means that $d_j \sim N(\delta_j, \frac{2\delta_j^2}{n-1})$.
- (b) $\sqrt{(n-1)}(\mathbf{e}_j - \boldsymbol{\eta}_j)$ is approximately $N_p(\mathbf{0}, 2\mathbf{H}_j)$, where the $\boldsymbol{\eta}_j$ are the eigenvectors of $\boldsymbol{\Sigma}$ and

$$\mathbf{H}_j = \delta_j \sum_{\substack{h=1 \\ h \neq j}}^p \frac{\delta_h}{(\delta_h - \delta_j)^2} \boldsymbol{\eta}_h \boldsymbol{\eta}_h'.$$

- (c) The eigenvalues of \mathbf{S} are distributed independently of the corresponding eigenvectors.

Inference on the eigenvalues

Using properties (a)–(c), the standard error of d_j is

$$s(d_j) = \sqrt{\frac{2}{n-1}} d_j, \quad (2.17)$$

and a large sample $100(1-\alpha)\%$ confidence interval for δ_j can be obtained with

$$\left[\frac{d_j}{1 + z_{\alpha/2} \sqrt{\frac{2}{n-1}}} ; \frac{d_j}{1 - z_{\alpha/2} \sqrt{\frac{2}{n-1}}} \right], \quad (2.18)$$

where $z_{\alpha/2}$ refers to the $100(1 - \frac{\alpha}{2})^{th}$ percentile of the standard normal distribution (Johnson and Wichern, 2002).

Inference on the eigenvectors

Anderson (1963) gave the large-sample estimator for the standard error of an eigenvector loading, e_{jh} (the h^{th} element of the j^{th} eigenvector), as

$$s(e_{jh}) = \sqrt{\frac{d_j}{n-1} \sum_{\substack{u=1 \\ u \neq j}}^p \frac{d_u}{(d_u - d_j)^2} e_{uh}^2}. \quad (2.19)$$

2.4. INFERENCE ON THE EIGENVECTORS AND EIGENVALUES 21

If the variances of any two of the principal components are nearly equal (leading to a small value for $(d_j - d_h)$ in (2.19)), some of the $s(e_{jh})$ will be fairly large and the estimation of the corresponding eigenvectors will not be very precise (Flury, 1988). This effect, due to near sphericity in the principal components concerned, holds true for multivariate normal as well as multivariate non-normal data, and can also be illustrated by computing bootstrap standard error estimates as done by Diaconis and Efron (1983) and Stauffer et al. (1985).

In addition to determining the variability of the eigenvector loadings, it can be of interest to test the hypothesis that $p-r$ (with $r < p$) loadings of the j^{th} eigenvector are equal to zero. Failure to reject the null hypothesis would mean that the corresponding variables may be redundant in the eigenvector, which should help to simplify the interpretation of the specific component (see Section 2.8).

Partitioning $\boldsymbol{\eta}_j$, the j^{th} eigenvector of $\boldsymbol{\Sigma}$, as

$$\boldsymbol{\eta}_j = \begin{bmatrix} \boldsymbol{\eta}_j^{(1)} \\ \boldsymbol{\eta}_j^{(2)} \end{bmatrix}, \quad (2.20)$$

and assuming that $\boldsymbol{\eta}_j^{(2)}$ contains the loadings of the $p-r$ possibly redundant variables, Flury (1988) showed that the hypothesis,

$$H_0 : \boldsymbol{\eta}_j^{(2)} = \mathbf{0}, \quad (2.21)$$

can be tested with the statistic,

$$T = (n-1) \mathbf{e}_j^{(2)'} \left[\sum_{\substack{h=1 \\ h \neq j}}^p \hat{\theta}_{jh} \mathbf{e}_h^{(2)} \mathbf{e}_h^{(2)'} \right]^{-1} \mathbf{e}_j^{(2)}, \quad (2.22)$$

where $\mathbf{e}_j^{(2)}$ is the sample estimator of $\boldsymbol{\eta}_j^{(2)}$ and the $\hat{\theta}_{jh}$ are defined as

$$\hat{\theta}_{jh} = \frac{d_j d_h}{(d_j - d_h)^2}. \quad (2.23)$$

Under hypothesis (2.21), the T statistic in (2.22) is distributed asymptotically chi-squared with $p-r$ degrees of freedom.

However, because PCA is often used as a dimension reduction method (see Section 2.7) with only $q < p$ of the components retained to represent the variability in the data, a more pertinent question may be whether the last $p-r$ variables are redundant in the q retained components simultaneously. Flury (1988) extended the test statistic in (2.22) to test hypothesis (2.21)

for all q eigenvectors simultaneously. Letting Q denote the indices of the q eigenvectors of interest, under the null hypothesis,

$$H_0(Q) : \boldsymbol{\eta}_j^{(2)} = \mathbf{0} \quad \text{for all } j \in Q, \quad (2.24)$$

the test statistic,

$$T = (n-1) \sum_{j \in Q} \mathbf{e}_j^{(2)'} \left[\sum_{h \notin Q}^p \hat{\theta}_{jh} \mathbf{e}_h^{(2)} \mathbf{e}_h^{(2)'} \right]^{-1} \mathbf{e}_j^{(2)}, \quad (2.25)$$

is distributed asymptotically chi-squared with $q(p-r)$ degrees of freedom.

It can be of interest to test the hypothesis that the j^{th} eigenvector is equal to a specified normalised vector, $\boldsymbol{\eta}_j^0$, i.e.

$$H_0 : \boldsymbol{\eta}_j = \boldsymbol{\eta}_j^0. \quad (2.26)$$

Anderson (1963) proposed the test statistic

$$X^2 = (n-1)(d_j \boldsymbol{\eta}_j^{0'} \mathbf{S}^{-1} \boldsymbol{\eta}_j^0 + d_j^{-1} \boldsymbol{\eta}_j^{0'} \mathbf{S} \boldsymbol{\eta}_j^0 - 2) \quad (2.27)$$

which, under hypothesis (2.26), is distributed asymptotically chi-squared with $p-1$ degrees of freedom.

Flury (1988) extended the test to more than one eigenvector, to test the more general hypothesis that q of the eigenvectors are equal to q predetermined vectors simultaneously, i.e.

$$H_0 : [\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_q] = [\boldsymbol{\eta}_1^0, \dots, \boldsymbol{\eta}_q^0] \quad (2.28)$$

where $[\boldsymbol{\eta}_1^0, \dots, \boldsymbol{\eta}_q^0]$ indicates a matrix of q orthonormal vectors. The statistic

$$X_q^2 = (n-1) \left[\frac{1}{4} \sum_{j=1}^{q-1} \sum_{h=j+1}^q \hat{\theta}_{jh}^{-1} (\mathbf{e}_h' \boldsymbol{\eta}_j^0 - \mathbf{e}_j' \boldsymbol{\eta}_h^0)^2 + \sum_{j=1}^q \sum_{h=q+1}^p \hat{\theta}_{jh}^{-1} (\mathbf{e}_h' \boldsymbol{\eta}_j^0)^2 \right] \quad (2.29)$$

is used, which under hypothesis (2.28) is distributed asymptotically chi-squared with $q[p-(q+1)/2]$ degrees of freedom and where the $\hat{\theta}_{jh}$ are defined as in (2.23).

An alternative to the parametric inference approach is to estimate the standard errors and confidence intervals of interest using bootstrap methods. Diaconis and Efron (1983) used bootstrap distributions to estimate the average error of the h^{th} loading of the j^{th} eigenvector as

$$s_{\text{boot}}(e_{jh}) = \frac{P_{84}^{jh} - P_{16}^{jh}}{2}, \quad (2.30)$$

where P_m^{jh} refers to the m^{th} percentile of the bootstrap distribution of the h^{th} loading of the j^{th} eigenvector.

Stauffer et al. (1985) estimated the proportion of the remaining variance accounted for by the j^{th} eigenvector (after subtracting the variance accounted for by the first $j - 1$ eigenvectors),

$$v_j = \frac{d_j}{\sum_{h=j}^p d_h}, \quad (2.31)$$

and used the bootstrap distribution of v_j to estimate its average error in a similar way as in (2.30). They used these average errors, together with standard normal quantiles, to test hypotheses regarding v_j .

The eigenvectors estimated through the spectral decomposition of \mathbf{S} are unique up to a factor of -1 . Thus when computing bootstrap distributions of the eigenvectors, special care must be taken to ensure that the signs of the eigenvector loadings stay consistent throughout all of the bootstrap replications. For loadings close to zero, there can be considerable uncertainty about the sign, and it may not even be preferable to keep the signs of such loadings constant. Therefore, to determine the correct factor (-1 or $+1$) to multiply the eigenvector with, inspection of the largest loadings is necessary. It seems reasonable to expect that the signs of the largest loadings (furthest away from zero) will remain constant over all of the bootstrap replications. If for any specific bootstrap replication it is found that the sign of the maximum loading of any eigenvector have changed, all the loadings of that specific eigenvector should be multiplied with -1 to ensure consistency in the signs.

2.5 Geometry of PCA

The row vectors in $\mathbf{X}_{n \times p}$ can be thought of as a cloud of points in p -dimensional space. If the variables in the stochastic vector $\mathbf{x} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ are correlated, the cloud of points will be ellipsoidal and the principal axes of \mathbf{X} (as defined by the principal components) will not be parallel to the original variables as represented by vectors in p -dimensional space.

PCA attempts to find the principal axes of \mathbf{X} in such a way that each subsequent axis

(a) is perpendicular to the previous axes, and

- (b) accounts for the maximum possible amount of the remaining variation in the data, after the variance accounted for by the previous axes.

The orthogonal matrix of eigenvectors, \mathbf{E} , is used to rotate the cloud of points so that the principal axes of \mathbf{X} are parallel to the vector representations of the principal components. Because \mathbf{E} is orthogonal, $\mathbf{E}'\mathbf{E} = \mathbf{I}$ which means that the distance of the centroid of \mathbf{X} to the origin remains unchanged by such a rotation. The centroid can be shifted to the origin by subtracting the column means of \mathbf{X} from each observation vector. Such a shift can also be done after the PCA rotation by subtracting the rotated column means from each of the rotated observation vectors.

The orthogonality of the principal components means that the columns of the rotated data matrix $\mathbf{Y} = \mathbf{X}\mathbf{E}$ will be completely uncorrelated. This fact can also be seen from examining the eigenvalue matrix,

$$\mathbf{D} = \mathbf{E}'\mathbf{S}\mathbf{E} = \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & d_p \end{bmatrix}, \quad (2.32)$$

of which the off-diagonal elements are equal to zero. \mathbf{D} is the covariance matrix of the rotated data, with the diagonal elements describing the variation of the cloud of points along each of the principal component axes. A small eigenvalue thus indicates that the data configuration is relatively “flat” (i.e. varies little) along the direction of the corresponding eigenvector. Equal eigenvalues indicate that the cloud of points is spherical in the corresponding dimensions.

If combinations of the variables (measured in commensurate scales) in the stochastic vector \mathbf{x} are highly correlated, the cloud of points representing \mathbf{X} can be relatively flat in a number of the principal component directions, and the corresponding eigenvalues will be relatively small. In such a case, the principal components corresponding to the small eigenvalues can be discarded to provide a lower-dimensional approximation of \mathbf{X} (see equation (2.16)).

Because \mathbf{S} will change with any changes in the measurement scales of the variables in \mathbf{X} , the principal components calculated from \mathbf{S} are not invariant to scale. With any scale change (other than a proportional change across all variables), the shape of the cloud of points will change, and therefore also its principal axes. If one of the variables is measured on a much larger scale than the rest, the cloud will have an elongated shape along the direction of this variable, and the direction of the first eigenvector will correspond closely to the direction of this “inflated” variable. Some care should thus be taken to

determine whether the measurement scales of the variables are commensurate before the principal components are calculated. If the measurement scales are incommensurable, the columns of \mathbf{X} can be standardised to solve the problem (see Section 2.6).

PCA is related to perpendicular regression in the sense that each eigenvector is found in such a way that the sum of squared perpendicular distances from the data points to the eigenvector is minimised, subject to the orthogonality constraint. The first eigenvector therefore corresponds to the perpendicular regression line that can be fitted to the data. In contrast, ordinary least squares regression minimises the sum of the squared distances from the data points to the regression line along the direction of the response variable.

The first principal component can be seen as a projection of the data onto a line of maximal variance. If the p -dimensional data are orthogonally projected onto the first eigenvector by multiplication,

$$\mathbf{y}_1 = \mathbf{X}\mathbf{e}_1, \quad (2.33)$$

the principal component scores $\mathbf{y}_{1m} = \mathbf{x}'_m \mathbf{e}_1$, $m = 1, \dots, n$ can be interpreted as a univariate ranking of the observations based on all p variables (Rencher, 1998).

2.6 Standardisation of the variables

It is not yet clear whether the covariance matrix or correlation matrix should be used to calculate the principal components of a data matrix (as the results from the two differ), but the majority of the literature advocates using the covariance matrix unless the variables are measured on widely different scales (Gower et al., 2011). Such incommensurability in the variables can lead to a situation where some of the variables dominate the principal component solution and seem to account for nearly all of the observed variation in the data (Rencher, 2002).

One advantage of using the covariance matrix is that the statistical theory for inference on the eigenvectors and eigenvalues are better developed and understood than for the eigenvectors and eigenvalues of the correlation matrix (Rencher, 1998). Statistical inference for population principal components based on eigenvector estimates from sample covariance matrices is also easier than when estimating the principal components from the eigenvectors of correlation matrices (Jolliffe, 2002).

However, measurements on incommensurable variables are often standardised by dividing each variable by its standard deviation (Johnson and Wichern, 2002). The standardised data matrix is given by

$$\mathbf{X}_s = \mathbf{X} \text{diag}(\mathbf{S})^{-\frac{1}{2}}, \quad (2.34)$$

where $\text{diag}(\mathbf{S})$ is the sample covariance matrix of \mathbf{X} , with the off-diagonal elements set to zero. The covariance matrix of \mathbf{X}_s will effectively be the correlation matrix of \mathbf{X} (and of \mathbf{X}_s), but it can be argued that the asymptotic theory for inference on the eigenvectors and eigenvalues is still applicable (Johnson and Wichern, 2002; Jolliffe, 2002). Another advantage is that the principal components of the standardised covariance matrix (or correlation matrix) are scale invariant, unlike the principal components of the unstandardised covariance matrix.

The data can also be centred by subtracting the column means, which will make the vector of the column means of \mathbf{X} equal to $\mathbf{0}$, putting the sample centroid at the origin in p -dimensional space. This would not effect the principal component solution itself as the calculation of the principal components does not depend on the location of the data. However, if it is important to distinguish between different groups in the data (as is done in later chapters), centring the data per group is not advisable as it will make the group centroids coincide.

2.7 Number of principal components to retain

When PCA is performed with the purpose of reducing the dimensionality of a data set, a decision should be made on the number of principal components to retain for subsequent analyses. Various methods have been devised to guide this decision, with a good summary of the most commonly used methods given by Rencher (1998).

2.7.1 Subjective methods

The simplest method is to select a threshold for the percentage of variance accounted for by the lower-dimensional approximation and retaining the minimum number of components for which the cumulative variance percentage exceeds this threshold. For most applications the threshold should be relatively high (say around 80–90%) so that little information would be lost by discarding the rest of the principal components.

A less subjective rule of thumb would be to discard the principal components for which the corresponding eigenvalues are lower than the average of the eigenvalues, \bar{d} . It would therefore be the components that account for less than the average variance of all the principal components. A number of researchers (Cattell and Jaspers, 1967; Browne, 1968; Linn, 1968) have found this method to perform reasonably well for situations with ≤ 30 variables with high correlations. In general, this method is conservative and seems to overestimate the true dimensionality of the data (Rencher, 1998).

Another very popular method is to inspect a scree plot of the eigenvalues. The eigenvalues are plotted in descending order of magnitude and a visual inspection is made to determine the point at which the plot “flattens out” (i.e. where the change in the negative slope of the eigenvalue trend stabilises). For correlated variables (measured on commensurable scales), the scree plot will flatten out at a relatively low number of components, with the last few eigenvalues exhibiting a near linear trend (for an example, see the eigenvalue scree plot for the forged Swiss bank notes described by Flury (1988) in Figure 2.1). With this method the components corresponding to the eigenvalues *before* the first one on the nearly straight line are retained. The variation in the components associated with the smallest eigenvalues are usually considered to be random noise and/or measurement error, and these components are discarded.

2.7.2 Significance testing on the last $p - q$ components

In addition to the aforementioned subjective methods to investigate the dimensions of \mathbf{X} , formal tests can be performed on the principal components to aid the decision on how many of the components to retain. However, the asymptotic theory of these tests is based on the assumption of multivariate normality in the population.

After inspection of the scree plot, the most obvious test would be to check whether the last $p - q$ eigenvalues are small and equal. If the hypothesis,

$$H_0 : \delta_{q+1} = \delta_{q+2} = \dots = \delta_p, \quad (2.35)$$

is true, it would imply that the corresponding components reflect noise and/or measurement error and can be discarded. To test hypothesis (2.35), Bartlett (1951) proposed the statistic

$$u = \left(n - \frac{2p + 11}{6} \right) \left[(p - q) \ln(\bar{d}) - \sum_{j=q+1}^p \ln(d_j) \right], \quad (2.36)$$

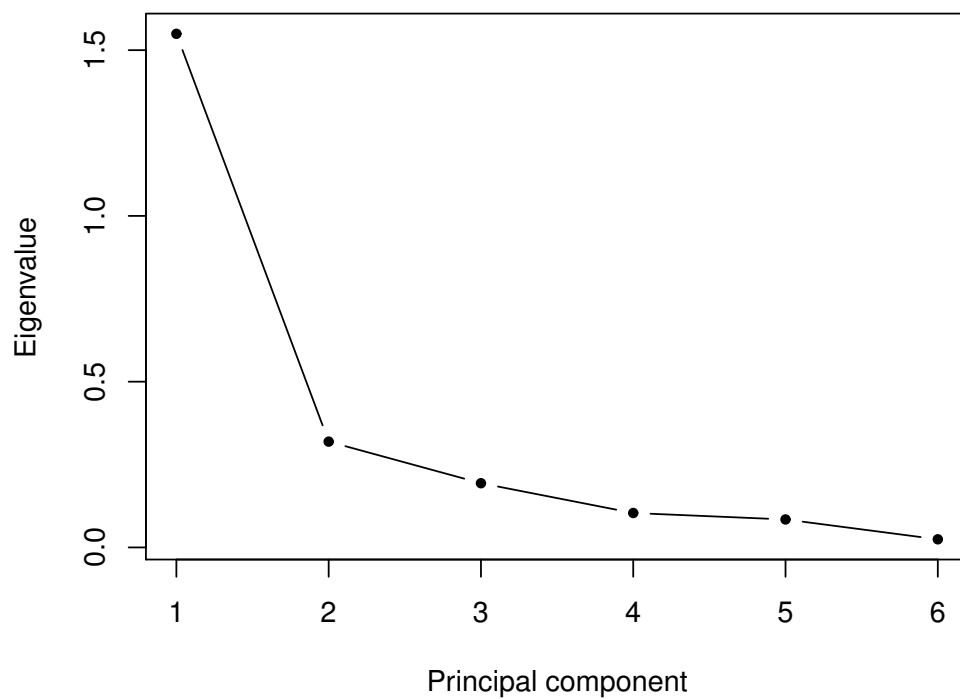
Forged Swiss bank notes: Scree plot of the eigenvalues

Figure 2.1: Scree plot of the eigenvalues of the 100 forged Swiss franc notes in the *Bank notes* data discussed by Flury (1988).

2.7. NUMBER OF PRINCIPAL COMPONENTS TO RETAIN

29

where

$$\bar{d} = \sum_{j=q+1}^p \frac{d_j}{p-q} \quad (2.37)$$

is the average of the last $p - q$ eigenvalues. Under hypothesis (2.35), the test statistic, u , in (2.36) is distributed approximately chi-squared with $\frac{1}{2}(p - q - 1)(p - q + 2)$ degrees of freedom.

Rencher (2002) suggested testing the hypotheses $H_{02} : \delta_{p-1} = \delta_p$, $H_{03} : \delta_{p-2} = \delta_{p-1} = \delta_p, \dots$ consecutively until the first test for which the hypothesis is rejected. At that point testing should stop and the last $p - q$ components discarded. Rencher noted that this test will often indicate a larger number of components to be retained than the three subjective methods mentioned in Section 2.7.1. For large p , Rencher's proposal can entail a large number of hypothesis tests, which will lead to inflation of the overall Type I error rate. A Bonferroni-type adjustment to the required significance level for a single test can be employed, and updated after each additional hypothesis test, to control the overall Type I error rate.

Flury (1988) tested the same hypothesis of sphericity in the last $p - q$ components with the log-likelihood ratio statistic

$$X_S^2 = (n - 1)(p - q) \log \frac{\frac{1}{p-q} \sum_{j=q+1}^p d_j}{(\prod_{j=q+1}^p d_j)^{\frac{1}{p-q}}} \quad (2.38)$$

which is distributed asymptotically chi-squared with $(p - q)(p - q + 1)/2 - 1$ degrees of freedom.

With regards to the standard errors of the eigenvalues in (2.17), Flury (1988) showed that the distribution of the sum of the last $p - q$ eigenvalues can be approximated with

$$N \left[\delta_{q+1} + \dots + \delta_p, \frac{2}{n-1} (\delta_{q+1}^2 + \dots + \delta_p^2) \right]. \quad (2.39)$$

If f is defined as the proportion of variance accounted for by the last $p - q$ principal components, i.e.

$$f = \frac{\sum_{j=q+1}^p \delta_j}{\sum_{j=1}^p \delta_j}, \quad (2.40)$$

a asymptotic one-sided $100(1 - \alpha)\%$ upper confidence limit for f can be obtained with

$$0 < f \leq \hat{f} + z_\alpha \sqrt{\frac{2}{n-1}} \times \frac{\sqrt{(\sum_{j=1}^q d_j)^2 \sum_{j=q+1}^p d_j^2 + (\sum_{j=q+1}^p d_j)^2 \sum_{j=1}^q d_j^2}}{[\text{tr}(\mathbf{S})]^2}}, \quad (2.41)$$

where \hat{f} is the sample estimator of f , and z_α the $100(1 - \alpha)^{th}$ percentile of the standard normal distribution. For a sufficiently small value of f , the last $p - q$ principal components can be discarded and the data approximated using only the scores for the first q components.

2.7.3 Information in the last few components

Assuming the variables in \mathbf{X} are measured on commensurable scales, if one or more of the eigenvalues are particularly small, it indicates (possibly unsuspected) correlations among the variables in \mathbf{X} . This information might be useful in prompting further investigation into the causes of the correlations in order to discard redundant variables and avoid multicollinearity in subsequent analyses.

Rencher (1998) suggested that the last few components can be used to detect outliers in the covariance structure of the data. Because the variances of the last few components are often relatively small, indicating that the data are nearly constant along the directions of the corresponding eigenvectors, observation vectors with significant deviations from the overall covariance structure may be clearly visible in a plot of the last two or more components.

Furthermore, the j^{th} principal component score of the m^{th} observation vector is given by $y_{jm} = \mathbf{e}_j' \mathbf{x}_m$ and it follows that

$$\mathbf{x}_m = y_{1m} \mathbf{e}_1 + y_{2m} \mathbf{e}_2 + \dots + y_{pm} \mathbf{e}_p, \quad m = 1, \dots, n, \quad (2.42)$$

because of the orthogonality of the eigenvectors. The last $p - q$ terms on the right-hand side of (2.42) can be interpreted as a “residual” for the fit of the first q components to \mathbf{x}_m , with the length of the residual vector given by

$$r_m^2 = y_{(p-q)m}^2 + y_{(p-q+1)m}^2 + \dots + y_{pm}^2. \quad (2.43)$$

An unusually large r_m^2 value will indicate that the m^{th} observation is not modelled well by the first q principal components and can be an outlier with regards to the overall covariance structure (Rencher, 1998).

2.8 Interpreting the eigenvectors

Inspection of the eigenvectors of the sample covariance matrix occasionally reveals complex relationships between variables which were not previously suspected. Such relationships would ordinarily not be revealed by a mere bivariate analysis of the variables. In this way PCA provides a more comprehensive view of the structure in the data.

As discussed in Section 2.6, \mathbf{S} is not scale invariant and the eigenvectors of \mathbf{S} will usually differ from the eigenvectors of the correlation matrix, \mathbf{R} . Therefore the interpretation of these two sets of eigenvectors will also differ and if the variables are incommensurable it is recommended to rather use the eigenvectors of \mathbf{R} . If PCA is performed using \mathbf{S} and one of the variables has a much larger variance than the others, the first eigenvector will be dominated by this variable.

In the case where all p variables in \mathbf{x} are uncorrelated, $\mathbf{E} = \mathbf{I}$ and the principal components will simply be the p original variables. For such uncorrelated variables the sample correlations will usually be small but not equal to zero, and each of the eigenvectors will be dominated by a single variable. The rank order of the eigenvectors will in this case correspond to the rank order of the variances of the original variables.

If all the loadings of the first eigenvector have the same sign, the associated principal component is a weighted average of the variables. According to the Perron-Frobenius theorem (Rencher 2002, p. 34), this will happen when all of the off-diagonal elements of \mathbf{S} or \mathbf{R} are positive. In the case where the measurements were made on physical objects, this first principal component will often be an indication of the overall *size* of the objects. Both positive and negative loadings in an eigenvector (contrasting the variables to one another) is often an indication of *shape*.

Where the p variables in \mathbf{x} include dimensional measurements on objects as well as other quantitative characteristics such as chemical properties, the eigenvectors associated with the largest eigenvalues will usually be dominated by the size and shape characteristics of the objects. The eigenvectors associated with the smaller eigenvalues can contain valuable information about the chemical characteristics, as the variances of these properties will generally be smaller than those of the size measurements.

A large absolute loading in any eigenvector means that the associated variable is highly correlated with the specific principal component (Krzanowski, 1979). With sufficient knowledge about the data under consideration, identification of the variables with large absolute loadings (relative to the rest of the loadings) in each of the principal components can aid the practitioner in labelling the different principal components as pertaining to properties such

as size, shape, chemical characteristics, etc. In this way, the interpretation of the principal components is similar to the description of the unobservable factors in factor analysis.

Even so, for very high dimensional data the interpretation of the eigenvectors by inspection of the loadings may still be problematic, as typically none of the loadings will be equal to zero. One solution is to rotate the individual eigenvectors further in order to find directions in which some of the loadings are equal to zero, to simplify interpretation. However, as pointed out by Rencher (2002), further rotation of the individual eigenvectors will cause them to not be mutually orthogonal anymore. The rotated solution will also not be optimal in the sense that the components successively account for the maximum of the remaining variance observed in the data.

Another proposal to aid with the interpretation of eigenvectors is to inspect the correlations between the original variables and the principal components. Variables showing high correlations with the first number of principal components are deemed to be important in accounting for the observed variation in the data. Rencher (2002) has shown that using this method to rank the variables in order of importance does not necessarily provide the same rank order as would be obtained by ranking the variables according to the absolute magnitude of their loadings in a specific eigenvector. Furthermore, this method only provides univariate information about the variables and is therefore not very useful for interpretation in the multivariate context.

Jolliffe et al. (2003) and Zou et al. (2006) developed a technique called *sparse principal component analysis (SPCA)* to ease the interpretation of principal components. SPCA shrinks many of the smallest eigenvector loadings to zero by imposing a constraint on the sum of the absolute values of the loadings. The approach taken by Jolliffe et al. (2003) is to maximise $\mathbf{e}_j'(\mathbf{X}'\mathbf{X})\mathbf{e}_j$ subject to

$$\sum_{h=1}^p |e_{jh}| \leq t \quad \text{and} \quad \mathbf{e}_j' \mathbf{e}_j = 1, \quad (2.44)$$

where t is some predetermined positive value. However, because this approach is computationally difficult and there is no clear guidance on how to select an appropriate value for t , Zou et al. (2006) proposed to find the SPCA solution by reformulating it as a ridge regression problem and obtaining the eigenvector matrix, \mathbf{E} , which minimises the criterion

$$\sum_{m=1}^n \|\mathbf{x}_m - \mathbf{A}\mathbf{E}'\mathbf{x}_m\|^2 + \theta \sum_{j=1}^p \|\mathbf{e}_j\|^2 + \sum_{j=1}^p \theta_{1,j} \|\mathbf{e}_j\|_1, \quad (2.45)$$

where \mathbf{A} is a $p \times p$ orthogonal matrix and

$$\| \mathbf{e}_j \|_1 = \sum_{h=1}^p |e_{jh}|, \quad (2.46)$$

subject to the constraint $\mathbf{A}'\mathbf{A} = \mathbf{I}_p$. The first penalisation factor, θ , is kept constant, while the second penalisation factor, $\theta_{1,j}$, is allowed to vary from component to component. The SPCA algorithm provided by Zou et al. (2006) initially sets \mathbf{A} equal to the eigenvectors of \mathbf{X} , whereafter \mathbf{A} and \mathbf{E} are updated iteratively until convergence. They also provide some guidelines for choosing appropriate values for the penalisation factors.

Shen and Huang (2008) proposed a simpler and computationally cheaper SPCA algorithm named *sparse PCA via regularised SVD (sPCA-rSVD)*. Their algorithm for the computation of the sparse principal components minimises a different objective function and proved to be considerably faster than the SPCA algorithm of Zou et al. (2006).

When many of the smaller eigenvector coefficients are shrunk to zero, the interpretation of the principal components is simplified, especially for data with a large number of variables. However, as pointed out by Shen and Huang (2008), the orthogonality property of the principal components is lost with most SPCA procedures.

2.9 PCA as a variable selection technique

PCA is often used as a dimension reduction tool, to approximate a data set with a reduced set of uncorrelated linear combinations of the original variables. The retained set of principal components can be used in further analyses and model building efforts (Rencher 1998, 2002).

However, as the interpretation of a statistical model with principal components as inputs is usually more complex than for a model constructed on the original variables, it might be desirable to rather use the original variables if interpretability of the model is a high priority. If the traditional variable selection techniques such as stepwise regression or all subset selection do not work well, it is possible to perform a manual variable selection in order to construct a good fitting model. Careful interpretation of the eigenvector loadings can aid in performing a manual variable selection for subsequent analyses. See Jolliffe (1972) and Jolliffe (1973) for a number of methods and their application to simulated and real data sets.

The first principal component accounts for the largest proportion of the variation observed in the data set. The variables with large loadings (relative to the rest of the loadings) in this eigenvector are those that contribute

the most to the variance accounted for by the first principal component. Geometrically, the directions of these variables correspond most closely to the direction of the first eigenvector. Therefore, if there is a subset of variables with large loadings in a specific eigenvector, these variables will be highly correlated. This means that the information contained in any one of these variables is also contained in the others. Including more than one of these variables in a regression model, for example, will increase the redundancy in the model and may give rise to multicollinearity.

In the manual variable selection process the variable which shows the highest correlation (or association, in the case of a categorical response variable) with the response of interest is selected from the aforementioned subset, with the rest of the variables from the subset being discarded. The same process is repeated for each subsequent eigenvector up to the point where all p variables have either been selected or discarded. The selected set of variables will be relatively uncorrelated and together account for a large proportion of the total variation in the data. These variables will also be those most correlated with the response of interest and their inclusion should lead to an adequate fit for the regression model.

In Chapter 8, use of the eigenvector loadings to perform variable selection before regression modelling will be demonstrated on the VON data.

2.10 Application to the VON data

PCA was performed on the set of 3041 infants (including deaths and transfers) in the VON 2009 cohort, using the covariance matrix of the six numerical variables (with birth weight in kilogram). The covariance matrix looks as follows:

$$\mathbf{S} = \begin{bmatrix} 0.679 & 0.329 & 0.264 & 2.438 & 2.299 & 0.192 \\ 0.329 & 3.706 & 2.376 & 1.472 & 1.378 & 0.285 \\ 0.264 & 2.376 & 2.460 & 1.290 & 1.090 & 0.253 \\ 2.438 & 1.472 & 1.290 & 12.582 & 9.490 & 0.748 \\ 2.299 & 1.378 & 1.090 & 9.490 & 10.809 & 0.708 \\ 0.192 & 0.285 & 0.253 & 0.748 & 0.708 & 0.566 \end{bmatrix}.$$

The total variance in the data is $\text{tr}(\mathbf{S}) = 30.803$. The eigenvectors and eigenvalues of \mathbf{S} are given in Table 2.1, together with the percentage of the total variance accounted for by each principal component. The first three components together account for more than 95% of the observed variation and should provide a sufficient approximation for the original data set in subsequent model fitting efforts.

Table 2.1: Eigenvectors and eigenvalues of the covariance matrix of the VON 2009 cohort ($n = 3041$). The percentage and cumulative percentage of the total variance accounted for by each principal component are given in the last two rows.

	e_1	e_2	e_3	e_4	e_5	e_6
BWGT	-0.15	0.02	0.03	-0.00	0.07	0.98
AP1	-0.12	-0.78	0.03	-0.61	0.06	-0.01
AP5	-0.10	-0.60	-0.03	0.77	-0.17	0.02
GESTAGE	-0.72	0.11	-0.68	-0.04	-0.02	-0.10
BHEADCIR	-0.66	0.11	0.73	0.02	-0.05	-0.12
ATEMP	-0.05	-0.05	0.01	0.17	0.98	-0.08
d_j	22.24	5.12	2.17	0.63	0.50	0.14
Variance (%)	72.2%	16.6%	7.0%	2.0%	1.6%	0.5%
Cum. variance (%)	72.2%	88.8%	95.9%	97.9%	99.5%	100.0%

The first eigenvector has high loadings (both negative) for the gestational age (*GESTAGE*) and head circumference at birth (*BHEADCIR*) variables, and the first principal component is thus an indication of size as both of these variables are positively correlated with the size of a newborn baby. The second eigenvector has large negative loadings for the two Apgar score variables (*AP1* and *AP5*) and the second principal component can be interpreted as a measure of the feasibility of life. Gestational age is contrasted to head circumference at birth in the third eigenvector, which means the third principal component gives an indication of whether the infant is small/large (or more specifically, has a small/large head) for the stage of development (i.e. the gestational age). The fourth principal component provides a contrast between the Apgar score at one minute after birth (*AP1*) and Apgar score at five minutes after birth (*AP5*) variables, as these two variables both have high loadings in the fourth eigenvector, but with opposite signs. The change in the feasibility of life of an infant within the first five minutes after birth is thus described by the fourth principal component. The fifth and sixth eigenvectors are dominated by the temperature (*ATEMP*) and birth weight (*BWGT*) variables, respectively.

For the purpose of fitting a linear regression model to predict length of stay (*LOS*) from the six numerical variables, manual variable selection can be performed by inspection of the eigenvectors. Pearson correlation coefficients (Snedecor and Cochran, 1989) for each of the numerical variables with *LOS* is given in the table below:

	Correlation with LOS
BWGT	-0.5983
AP1	-0.2047
AP5	-0.2042
GESTAGE	-0.6350
BHEADCIR	-0.6145
ATEMP	-0.1746

From the two variables with large absolute loadings in the first eigenvector, *GESTAGE* shows the strongest correlation with *LOS*. *GESTAGE* will therefore be retained and *BHEADCIR* discarded. The two Apgar score variables has large absolute loadings in the second eigenvector. *AP1* has a marginally stronger correlation with *LOS*, and *AP5* will therefore be discarded. The large absolute loadings for *GESTAGE* and *BHEADCIR* in the third eigenvector is ignored, as decisions on retention and rejection of these variables have already been made when the first eigenvector was considered. Continuation of this process with the remaining eigenvectors results in *ATEMP* and *BWGT* also being retained.

To perform manual variable selection for the purpose of fitting a logistic regression model to predict infant death before final discharge from the hospital (*DIED*), the subsets of variables with large absolute loadings in each eigenvector is inspected to determine which variable has the strongest association with the outcome of interest. As the normality assumption seems doubtful for most of the six numerical variables, Kruskal-Wallis tests (Hollander et al., 2014) were performed to test for differences between the surviving infants (*DIED* = 0) and infants who had died (*DIED* = 1). The test statistics and *p*-values for the Kruskal-Wallis tests are shown in the table below:

	Test statistic	<i>p</i> -value
BWGT	96.8624	7.43×10^{-23}
AP1	83.2134	7.37×10^{-20}
AP5	93.1361	4.88×10^{-22}
GESTAGE	83.8030	5.47×10^{-20}
BHEADCIR	76.7935	1.90×10^{-18}
ATEMP	14.1311	1.71×10^{-04}

Of the two variables with large absolute loadings in the first eigenvector, *GESTAGE* shows the strongest association with *DIED* and will therefore be retained while *BHEADCIR* is discarded. From the second eigenvector, *AP5* will be retained and *AP1* discarded. The *ATEMP* and *BWGT* variables will also be retained.

A scree plot of the eigenvalues is given in Figure 2.2. There seems to be a linear trend from the fourth to the sixth eigenvalue, indicating that the variation described by the last three principal components might be random error variation in the data.

The test statistic for a multivariate Shapiro-Wilk test (Jarek, 2012) of the six numeric variables in the VON 2009 cohort is $W = 0.9372$ ($p < 0.0001$), which means that it is highly unlikely that the population distribution is multivariate normal. This result is to be expected, as the sample size is very large and two of the variables ($AP1$ and $AP5$) are measured on an ordinal scale. The parametric methods for inference on the eigenvalues and eigenvectors can therefore not be applied to this data, and bootstrap samples to find standard errors for the estimates were used instead. A total of $r = 1000$ bootstrap samples were drawn from the original sample and confidence intervals were calculated from the percentiles of the bootstrap distributions of the eigenvalues and eigenvector loadings. The bootstrap distributions of the loadings for the first eigenvector is shown in Figure 2.3.

The 95% bootstrap percentile confidence intervals for the elements of the eigenvectors, eigenvalues, and variance accounted for by each principal component are given in Table 2.2. It is clear that there are a number of loadings (such as the those for $BWGT$, $BHEADCIR$ and $ATEMP$ in the fourth eigenvector) which may be equal to zero, as the 95% confidence intervals include the value of zero.

While the assumption of multivariate normality seems untenable for the VON data, it is interesting to compare the bootstrap standard errors for the eigenvector loadings and the eigenvalues with the parametric estimates (using equations 2.17 and 2.19). These standard errors are given in Table 2.3. The bootstrap standard errors in the lower half of Table 2.3 were calculated using the method from Diaconis and Efron (1983) as given in (2.30). It is noticeable that the bootstrap standard errors are in general slightly larger than the parametric standard errors. It thus seems that the assumption of multivariate normality in the VON population leads to underestimation of the standard errors of the eigenvalues and the eigenvector loadings.

In Table 2.4, parametric 95% confidence intervals for the eigenvalues are given for comparison with the bootstrap confidence intervals. The parametric confidence intervals were calculated with (2.18), and the bootstrap confidence limits are the 2.5th and 97.5th percentiles of the bootstrap replications of the eigenvalues. For all of the eigenvalues, the bootstrap confidence intervals are the same width or wider than the parametric confidence intervals. It thus seems that assuming multivariate normality in the VON population leads to confidence intervals that are too narrow.

Retaining the first three principal components as an approximation to the

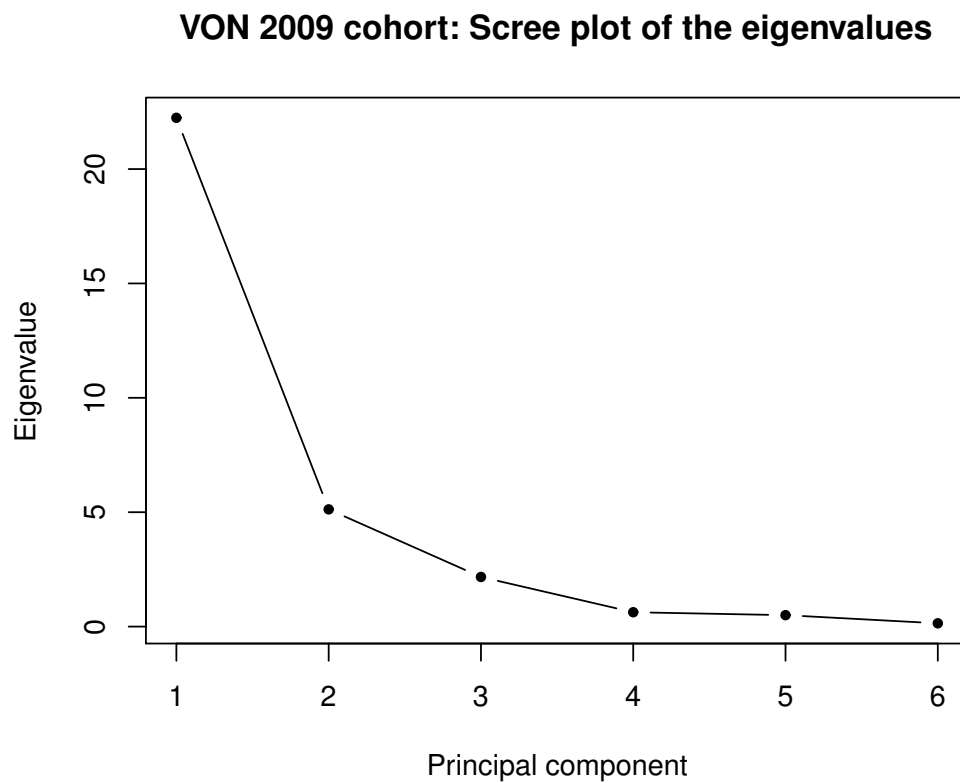


Figure 2.2: Scree plot of the eigenvalues of the covariance matrix of the VON 2009 cohort ($n = 3041$).

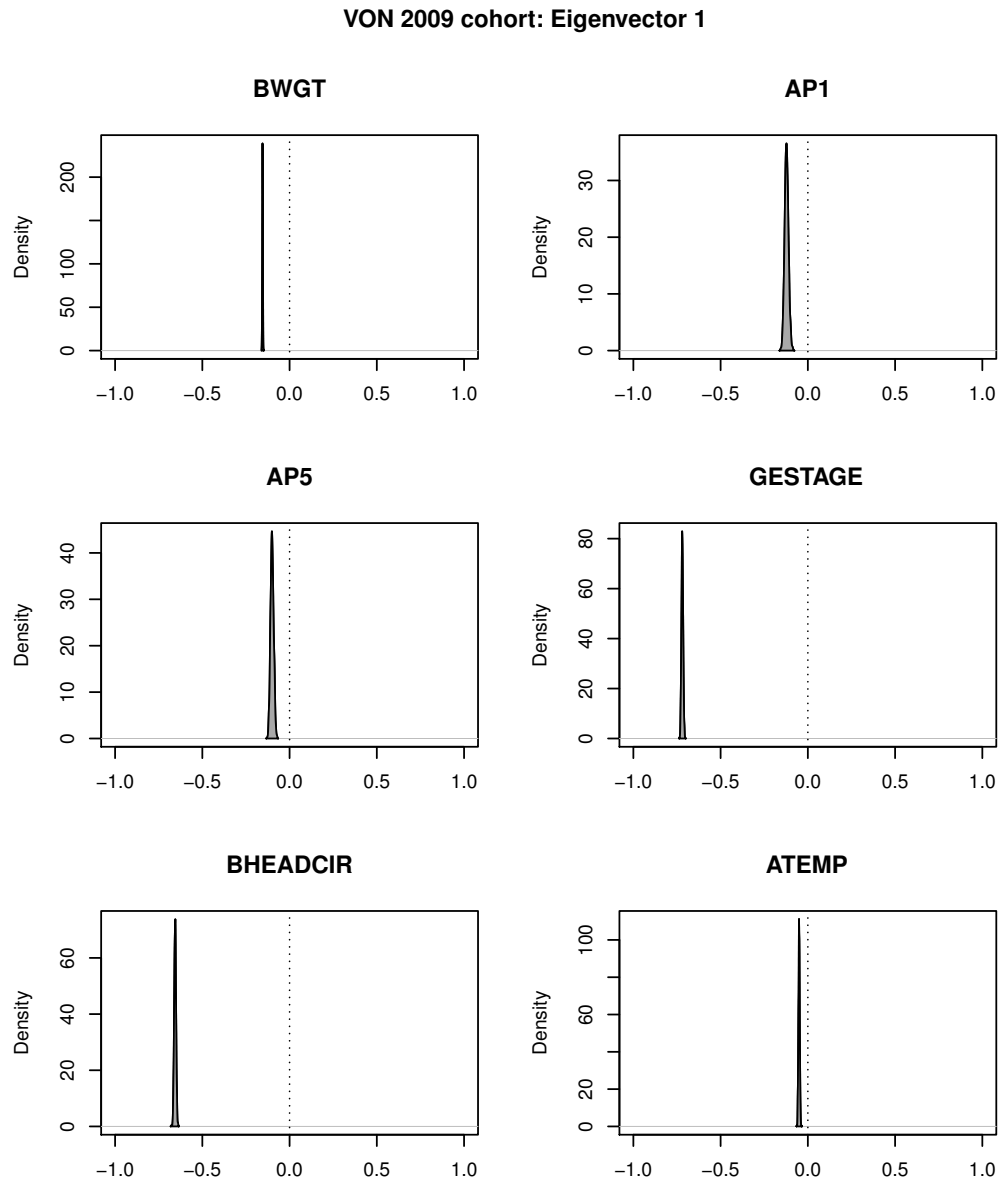


Figure 2.3: Bootstrap distributions ($r = 1000$ replications) of the loadings in the first eigenvector of the covariance matrix of the VON 2009 cohort.

Table 2.2: The 95% bootstrap confidence intervals for the eigenvectors and eigenvalues of the covariance matrix of the VON 2009 cohort ($n = 3041$). The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

			e_1	e_2	e_3	e_4	e_5	e_6
BWGT	LL	-0.158	0.016	0.010	-0.027	0.041	0.982	
	UL	-0.152	0.033	0.044	0.023	0.098	0.986	
AP1	LL	-0.143	-0.796	-0.007	-0.623	-0.042	-0.021	
	UL	-0.099	-0.769	0.071	-0.565	0.228	0.005	
AP5	LL	-0.118	-0.615	-0.067	0.696	-0.380	-0.001	
	UL	-0.083	-0.583	0.003	0.797	-0.026	0.032	
GESTAGE	LL	-0.729	0.075	-0.687	-0.059	-0.043	-0.106	
	UL	-0.710	0.149	-0.666	-0.017	-0.007	-0.085	
BHEADCIR	LL	-0.667	0.074	0.724	-0.004	-0.069	-0.135	
	UL	-0.646	0.147	0.743	0.048	-0.031	-0.110	
ATEMP	LL	-0.058	-0.060	-0.013	-0.003	0.892	-0.105	
	UL	-0.043	-0.030	0.034	0.442	0.995	-0.049	
<hr/>								
d_j	LL	20.9	4.7	2.0	0.6	0.5	0.1	
	UL	23.5	5.5	2.4	0.7	0.5	0.2	
Variance	LL	70.6%	15.4%	6.4%	1.8%	1.5%	0.4%	
	UL	73.6%	17.9%	7.8%	2.3%	1.7%	0.5%	
Cum. variance	LL	70.6%	88.1%	95.6%	97.8%	99.5%	100.0%	
	UL	73.6%	89.6%	96.2%	98.1%	99.6%	100.0%	

Table 2.3: Parametric and bootstrap standard errors of the eigenvector loadings and the eigenvalues of the covariance matrix of the VON 2009 cohort.

Eigenvector loadings: Parametric standard errors						
	PC1	PC2	PC3	PC4	PC5	PC6
BWGT	0.001	0.004	0.005	0.012	0.013	0.001
AP1	0.009	0.005	0.018	0.008	0.048	0.007
AP5	0.007	0.006	0.016	0.014	0.062	0.009
GESTAGE	0.004	0.016	0.005	0.010	0.009	0.004
BHEADCIR	0.005	0.017	0.005	0.011	0.009	0.004
ATEMP	0.003	0.006	0.011	0.078	0.014	0.013
Eigenvalues: Parametric standard errors						
d_j	0.121	0.058	0.038	0.020	0.018	0.010
Eigenvector loadings: Bootstrap standard errors						
	PC1	PC2	PC3	PC4	PC5	PC6
BWGT	0.002	0.005	0.009	0.012	0.015	0.001
AP1	0.010	0.007	0.021	0.011	0.058	0.007
AP5	0.009	0.008	0.018	0.019	0.077	0.009
GESTAGE	0.005	0.019	0.005	0.009	0.009	0.006
BHEADCIR	0.005	0.019	0.005	0.013	0.009	0.007
ATEMP	0.004	0.008	0.011	0.096	0.018	0.015
Eigenvalues: Bootstrap standard errors						
d_j	0.670	0.203	0.112	0.034	0.020	0.006

Table 2.4: 95% Parametric and bootstrap confidence intervals for the eigenvalues of the covariance matrix of the VON 2009 cohort. The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

Eigenvalues: Parametric 95% C.I.							
d_j	LL	21.2	4.9	2.1	0.6	0.5	0.1
	UL	23.4	5.4	2.3	0.7	0.5	0.2
Eigenvalues: Bootstrap 95% C.I.							
d_j	LL	20.9	4.7	2.0	0.6	0.5	0.1
	UL	23.5	5.5	2.4	0.7	0.5	0.2

full data set, the residuals for this “PCA model” was calculated according to (2.43). A box plot of the residuals (Figure 2.4) shows that there is a small number of outliers relative to the number of observations in the sample.

Lastly, Figure 2.5 shows a plot of the scores for the fifth and sixth principal components, with a small number of outliers indicated by their observation numbers.

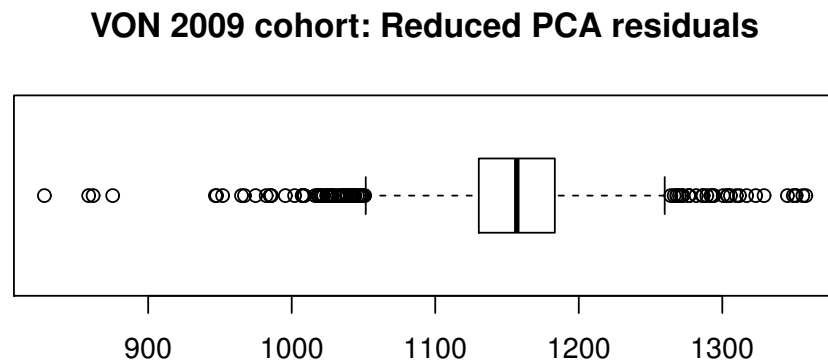


Figure 2.4: PCA “residuals” as per equation (2.43) for the VON 2009 cohort ($n = 3041$) when only the first three components (associated with the three largest eigenvalues) are retained. The whiskers extend up to 1.5 times the interquartile range on both sides of the box. Ignoring the thick tails of the distribution, a small number of outliers can be seen at the left- and rightmost ends of the plot.

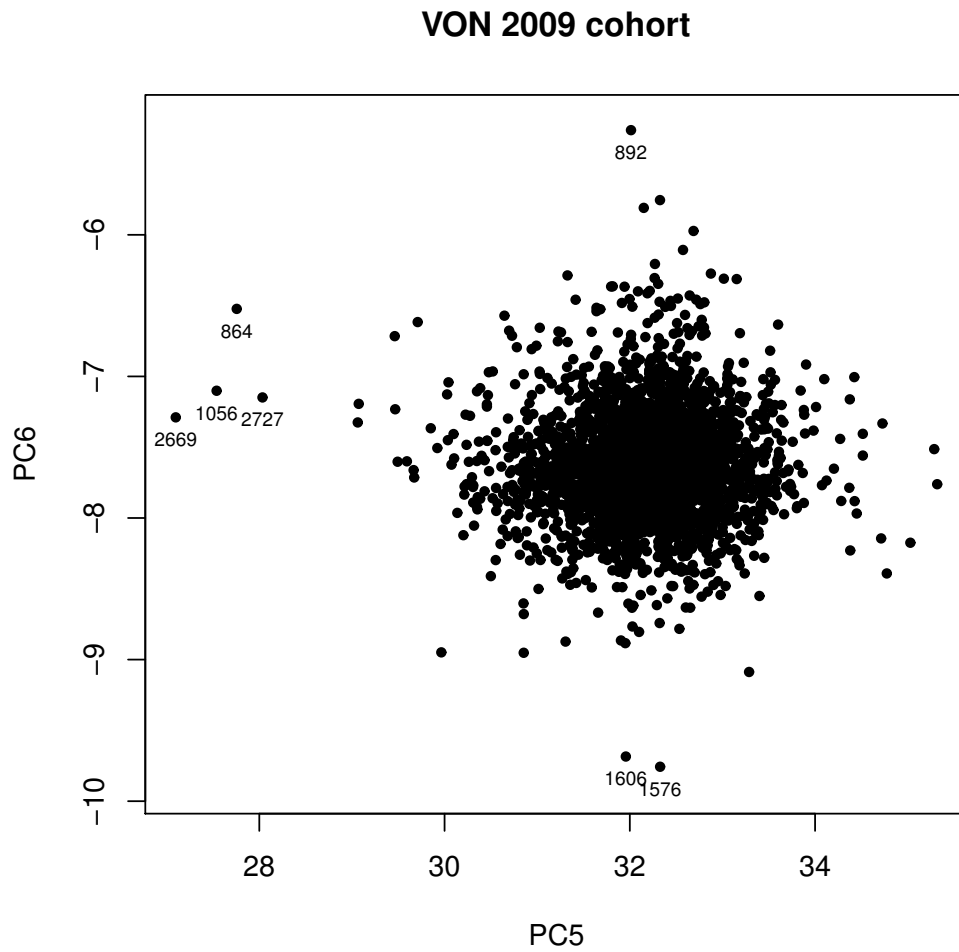


Figure 2.5: Scores for the fifth and sixth principal components of the VON 2009 cohort ($n = 3041$). A small number of outliers (with regards to the covariance structure) can be seen.

Chapter 3

Common principal components

3.1 Introduction

Common principal component (CPC) analysis extends the idea of PCA to more than one group. Flury (1984) developed the theory of CPC analysis in a definitive book on the subject (Flury, 1988). Much of the work in this chapter is based on the work of Flury (1988), updated with developments from more recent research.

For an informal explanation of the concept of common principal components, consider a situation where the same p variables were measured on k different natural groupings, for example *males* and *females*. If a PCA is performed on the data, it might be of interest to consider whether the principal components from the two populations differ, and if so, to what extent they differ.

The more pertinent question is whether the covariance structures of the k populations differ from each other. If the covariance structures are equal, the covariance matrices may simply be pooled before the PCA is carried out, because the eigenvectors and eigenvalues from the k groups would be equal. However, if the covariance matrices are not equal, pooling them would be inappropriate.

A comparison of k covariance matrices should therefore commence with a test of equality of the covariance matrices. In a univariate context, testing equality of variances is straightforward, with the process simply being a choice between homoscedasticity and heteroscedasticity. The multivariate situation is more complex, as there are a number of ways in which covariance matrices may differ from each other without being completely unrelated. These ways have been summarised in the literature as *Flury's hierarchy*, discussed in Section 3.7.

If homogeneity of the covariance matrices have been ruled out, the possibility still exists that these k multidimensional clouds of points share the same natural axes but that the variation along these axes differ between the populations. In this case, the CPC model will be appropriate, where it is assumed that the populations share a common eigenvector structure but with different sets of eigenvalues. The algebraic and geometrical properties of CPC will be discussed in Sections 3.2 to 3.4.

Known algorithms for the simultaneous diagonalisation of k square symmetric matrices, necessary for the estimation of common eigenvectors under the CPC model, are given in Section 3.5.

Even if the hypothesis of a common eigenvector structure is rejected, the populations might still have $q < p$ components in common. This assumption is known as the partial common principal component model, which is discussed in Section 3.6.

Section 3.8 outlines the statistical inference for the eigenvalues and common eigenvectors under the CPC model.

Interpretation of the common eigenvectors is briefly explained in Section 3.9, and Section 3.10 discusses other research on the CPC and related models.

In Section 3.11 the methods outlined in this chapter are applied to the delivery mode and regional groupings of the VON 2009 cohort.

3.2 The CPC model

Consider stochastic vectors $\mathbf{x}_i \sim N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, $i = 1, \dots, k$ and suppose that each of these vectors consists of measurements on the same set of p variables in each of k populations. Under the CPC model there exists a single orthogonal matrix $\mathbf{B} = [\boldsymbol{\beta}_1 \ \boldsymbol{\beta}_2 \ \dots \ \boldsymbol{\beta}_p]$ which diagonalises the $\boldsymbol{\Sigma}_i$ matrices simultaneously. The CPC hypothesis is

$$H_{\text{CPC}} : \boldsymbol{\Sigma}_i = \mathbf{B}\boldsymbol{\Lambda}_i\mathbf{B}', \quad i = 1, \dots, k, \quad (3.1)$$

where

$$\boldsymbol{\Lambda}_i = \begin{bmatrix} \lambda_{i1} & 0 & \dots & 0 \\ 0 & \lambda_{i2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \lambda_{ip} \end{bmatrix} \quad (3.2)$$

is a diagonal matrix with the eigenvalues of the i^{th} covariance matrix on the diagonal.

Under the CPC model the k groups thus share the same set of common eigenvectors (contained in the columns of the modal matrix \mathbf{B}), but with different sets of eigenvalues.

Because of the orthogonality of the common eigenvectors, the CPC model can also be written as

$$\Sigma_i = \lambda_{i1}\beta_1\beta_1' + \lambda_{i2}\beta_2\beta_2' + \dots + \lambda_{ip}\beta_p\beta_p' \quad (3.3)$$

for $i = 1, \dots, k$.

3.3 Sample common principal components

Consider data matrices $\mathbf{X}_1, \dots, \mathbf{X}_k$ containing n_1, \dots, n_k observations, respectively, on the same p variables from k multivariate normal populations. Let \mathbf{S}_i be the unbiased sample covariance estimator for Σ_i , i.e.

$$\mathbf{S}_i = \frac{1}{n_i - 1}(\mathbf{X}_i' \mathbf{X}_i - n_i \bar{\mathbf{x}}_i \bar{\mathbf{x}}_i'), \quad i = 1, \dots, k. \quad (3.4)$$

Under the CPC model, there exists a single square orthogonal matrix, \mathbf{B} , which diagonalises all k covariance matrices simultaneously, so that

$$\mathbf{B}' \mathbf{S}_i \mathbf{B} = \mathbf{L}_i, \quad i = 1, \dots, k, \quad (3.5)$$

where the elements of $\text{diag}(\mathbf{L}_i) = (l_{i1}, l_{i2}, \dots, l_{ip})$ are the sample eigenvalues for the i^{th} group.

Due to sampling variation, the estimated \mathbf{B} matrix will ordinarily not diagonalise the \mathbf{S}_i matrices perfectly, and therefore even if the CPC hypothesis is valid, the off-diagonal elements of the \mathbf{L}_i will generally be relatively small but not equal to zero.

To find the matrix $\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \dots \ \mathbf{b}_p]$ under hypothesis (3.1), the basic equation system (from Flury 1988) is

$$\mathbf{b}_j' \left(\sum_{i=1}^k (n_i - 1) \frac{l_{ij} - l_{ih}}{l_{ij} l_{ih}} \mathbf{S}_i \right) \mathbf{b}_h = 0 \quad (3.6)$$

for

$$i = 1, \dots, k; \quad j, h = 1, \dots, p; \quad j \neq h,$$

where

$$l_{ij} = \mathbf{b}_j' \mathbf{S}_i \mathbf{b}_j \quad (3.7)$$

is the j^{th} eigenvalue of the i^{th} group.

Equation (3.6) is solved under the usual orthogonality and normalisation constraints applicable to eigenvectors, i.e.

$$\mathbf{b}'_j \mathbf{b}_h = \begin{cases} 1, & j = h \\ 0, & j \neq h. \end{cases} \quad (3.8)$$

The *Flury-Gautschi* (FG) algorithm of Flury and Gautschi (1986) (discussed in Section 3.5) is an efficient method for solving (3.6) and thus estimating \mathbf{B} .

The j^{th} sample common principal component is the linear combination,

$$z_j = b_{j1}x_{i1} + b_{j2}x_{i2} + \cdots + b_{jp}x_{ip}, \quad j = 1, \dots, p, \quad (3.9)$$

where b_{jh} indicates the h^{th} loading in the j^{th} common eigenvector, and x_{ih} is the h^{th} element of the vector, \mathbf{x}_i , in the i^{th} population. The matrix of common principal component scores of the i^{th} group is thus obtained with

$$\mathbf{Z}_i = \mathbf{X}_i \mathbf{B}, \quad (3.10)$$

which has covariance matrix

$$\mathbf{L}_i = \mathbf{B}' \mathbf{S}_i \mathbf{B}. \quad (3.11)$$

The diagonal elements of \mathbf{L}_i are the estimated variances of the common principal components for the i^{th} group. The rank order of these variances need not be the same for all k groups, as the most important component (i.e. the component accounting for the largest proportion of variance) may differ between groups. There is no clear convention on the order of the common eigenvectors as in the single group PCA case, although Flury (1988) suggested ordering the columns of \mathbf{B} in such a way that the sums of the corresponding eigenvalues over all k groups are in decreasing order. Doing so provides some degree of consistency in the methodology and ensures that the common eigenvectors are sorted in order of their weighted average importance over all the groups.

The \mathbf{X}_i matrices can be centred by subtracting the column means for each \mathbf{X}_i individually, but this will not affect the CPC solution in any way as (3.6) does not incorporate any location estimators. However, centring of the \mathbf{X}_i will affect the CPC scores through (3.10). For applications where there is a focus on distinction between the groups, the \mathbf{X}_i matrices should not be centred as this will make the group centroids coincide.

The sample correlation matrix of the common components for the i^{th} group can be obtained with

$$\mathbf{R}_{L_i} = [\text{diag}(\mathbf{L}_i)]^{-\frac{1}{2}} \mathbf{L}_i [\text{diag}(\mathbf{L}_i)]^{-\frac{1}{2}}. \quad (3.12)$$

Under the CPC hypothesis, the \mathbf{R}_{L_i} are expected to be close to \mathbf{I}_p . Large off-diagonal elements in any of the \mathbf{R}_{L_i} will indicate that the hypothesis of common principal components may be untenable.

3.4 Geometry of CPC

The assumption of the CPC model is that the principal axes of the k multi-dimensional clouds of points are the same, without the restriction that these axes should account for the same proportion of variation in each of the populations. If they do account for the same proportion of variation in each of the k populations, the proportional covariance model is appropriate. The proportional model therefore also encompasses the CPC situation. If the stricter requirement that all of the components account for the same amount of variation in each of the k populations is justified, the model of equal covariance matrices is suitable.

Due to sampling variation the estimated eigenvectors of the k groups will usually not be identical, even if the CPC hypothesis is true. However, if the components are well defined (i.e. the eigenvalues are well separated), the angles between the eigenvectors which are common (i.e. the set of k individual eigenvectors, one from each of the k groups) should be relatively small. Krzanowski (1979) investigated the angles between eigenvector sets by using the fact that the cosine of the angle, θ , between two normalised vectors \mathbf{a} and \mathbf{b} in p -dimensional space is equal to the inner product of the two vectors, that is

$$\cos \theta = \mathbf{a}'\mathbf{b}. \quad (3.13)$$

This fact will be used in Chapter 4 where a new technique is proposed to identify common eigenvectors in k groups.

The quantity in (3.13) is also equal to the correlation between the values in \mathbf{a} and the values in \mathbf{b} . The correlation between the values in two orthogonal vectors therefore is $\cos(90^\circ) = 0$, and the correlation between the values in two parallel vectors is $\cos(0^\circ) = 1$.

Absolute values of the inner products of the pairs of eigenvectors which are common can be used to define a measure of similarity of the eigenvector matrices of two groups,

$$\text{similarity} = \text{tr} [\text{abs}(\mathbf{E}'_1 \mathbf{E}_2)], \quad (3.14)$$

which can range between 0 (heterogeneous eigenvector structures) and p (common eigenvector structures), (Krzanowski, 1979). \mathbf{E}_1 and \mathbf{E}_2 indicate the eigenvector matrices of \mathbf{S}_1 and \mathbf{S}_2 , respectively.

The common eigenvector matrix \mathbf{B} , selected from all $p \times p$ orthogonal matrices with normalised columns, is the matrix which best approximates the eigenvector structure of all k groups simultaneously. In this sense, \mathbf{B} is chosen to minimize

$$\sum_{i=1}^k \text{tr}(\mathbf{B}' \mathbf{E}_i), \quad (3.15)$$

given that the order of the common eigenvectors in the \mathbf{E}_i is identical for all k groups. The columns of \mathbf{B} may be regarded as the “average eigenvectors” over all the groups.

3.5 Simultaneous diagonalisation algorithms

The classical Jacobi iteration algorithm (Jacobi, 1846) is the oldest known method for the diagonalisation of a single symmetric matrix. It is a comparatively simple method which involves the systematic pre- and postmultiplication of a $p \times p$ symmetric matrix \mathbf{S} with a sequence of orthogonal matrices so as to annihilate all off-diagonal elements in \mathbf{S} . With each iteration of the procedure, the largest off-diagonal element in \mathbf{S} is turned to zero by the rotation of \mathbf{S} .

However, with the rotation at each iteration, some of the previously annihilated off-diagonal elements may again take non-zero values. The rotation process is therefore continued until the absolute values of all of the off-diagonal elements are smaller than some suitably small constant.

In a modification of the classical Jacobi algorithm, the *cyclical* Jacobi algorithm avoids the search for the largest off-diagonal element of \mathbf{S} by choosing the vector rotation pairs cyclically, for example in the order $(1, 2)$, $(1, 3)$, \dots , $(1, p)$, $(2, 3)$, \dots , $(2, p)$, $(p - 1, p)$, (Flury, 1988).

The *Flury-Gautschi* (FG) algorithm was proposed by Flury and Gautschi (1986) as an extension of the cyclical Jacobi procedure to two or more $p \times p$ symmetric matrices. As a measure of simultaneous deviation of the \mathbf{L}_i matrices from diagonality, they defined the measure

$$\phi(\mathbf{L}_1, \dots, \mathbf{L}_k; n_1, \dots, n_k) = \prod_{i=1}^k \left[\frac{\det(\text{diag} \mathbf{L}_i)}{\det(\mathbf{L}_i)} \right]^{n_i}, \quad (3.16)$$

which attains a minimum value of one when all k matrices are perfectly diagonal. The FG algorithm provides an estimate of the modal matrix \mathbf{B} in a way that (3.16) is minimised. Given that the CPC model is appropriate, the columns of \mathbf{B} are estimates of the common eigenvectors of the k groups and together will simultaneously rotate all of the \mathbf{S}_i matrices to nearly diagonal form.

To use the known parametric methods for inference on the common eigenvector loadings in \mathbf{B} , the assumption of multivariate normality in the populations is necessary. However, the FG algorithm does not depend on this assumption, which justifies its use in finding common eigenvectors of the covariance matrices of multivariate non-normal populations (see Flury, 1988, pp. 71 and 178–188).

For the special case where $k = 1$ the FG algorithm may also be used to obtain the eigenvectors of the single group.

The FORTRAN routines given by Flury (1988) for the FG algorithm to compute the modal matrix \mathbf{B} was translated to R and are given in Appendix B.

Krzanowski (2000) proposed a simple estimator for the matrix of common eigenvectors and the k sets of eigenvalues. Under CPC hypothesis (3.1),

$$\mathbf{B}'(\mathbf{\Sigma}_1 + \dots + \mathbf{\Sigma}_k)\mathbf{B} = \mathbf{\Lambda}_1 + \dots + \mathbf{\Lambda}_k. \quad (3.17)$$

Let $\mathbf{\Gamma} = \mathbf{\Sigma}_1 + \dots + \mathbf{\Sigma}_k$ and $\mathbf{\Lambda}_{ALL} = \mathbf{\Lambda}_1 + \dots + \mathbf{\Lambda}_k$. The spectral decomposition of $\mathbf{\Gamma}$ is given by

$$\mathbf{\Gamma} = \mathbf{B}\mathbf{\Lambda}_{ALL}\mathbf{B}', \quad (3.18)$$

with the columns of \mathbf{B} containing the eigenvectors of $\mathbf{\Gamma}$. Let $\mathbf{G} = \mathbf{S}_1 + \dots + \mathbf{S}_k$ be the unbiased sample estimator of $\mathbf{\Gamma}$. A simple estimator of the common eigenvector matrix, \mathbf{B} , is given by the spectral decomposition of \mathbf{G} , i.e.

$$\mathbf{G} = \mathbf{B}\mathbf{L}_{ALL}\mathbf{B}'. \quad (3.19)$$

When comparing the results from the simple estimator in (3.19) with the \mathbf{B} matrix as estimated with the FG algorithm, Krzanowski (2000) observed the common eigenvector loadings to be almost identical for the case where the CPC hypothesis is tenable.

Cardoso and Souloumiac (1996) gave, in closed form, the optimal Jacobi rotation angles to nearly diagonalise k symmetric matrices simultaneously. Their technique finds \mathbf{B} in a way that minimises the criterion

$$\sum_{i=1}^k \text{off}(\mathbf{B}'\mathbf{S}_i\mathbf{B}), \quad (3.20)$$

where

$$\text{off}(\mathbf{B}'\mathbf{S}_i\mathbf{B}) = \text{off}(\mathbf{L}_i) = \sum_{1 \leq j \neq h \leq p} l_{jh}^2 \quad (3.21)$$

is the sum of the squared off-diagonal elements of the nearly diagonalised matrix \mathbf{L}_i .

The algorithm for their method was implemented in the *rjd* function of the *JADE* package in R (Nordhausen et al., 2013), and will henceforth be referred to as the *JADE algorithm*.

Because the focus of the FG algorithm is on simultaneous diagonalisation, the common eigenvectors are found in an arbitrary order. It is therefore not always useful if the purpose of the CPC analysis is dimensionality reduction, as the common eigenvectors estimated by this method do not necessarily have the same rank order in all of the populations with regard to the amount of variation accounted for (Trendafilov, 2010).

Furthermore, the FG algorithm estimates the common eigenvectors (if they exist) simultaneously through an iterative procedure. If, for the purpose of dimensionality reduction, only the first $q < p$ common eigenvectors should be retained, it implies that the computation involved with finding the last $p - q$ eigenvectors is unnecessary.

Trendafilov (2010) proposed a stepwise CPC technique where the common eigenvectors are estimated sequentially, analogous to PCA for a single group. The method almost always ensures that the common eigenvectors are found in such a way that the rankings of the common eigenvectors with regard to the amount of variation accounted for are the same over all k groups. For data sets with a large number of correlated variables, the stepwise CPC procedure may be stopped at the point where $q < p$ common eigenvectors, which account for some minimum (for example, 90%) of variation within each of the k groups, are found. This may save valuable computing time and at the same time ensure that the variation of all the groups are represented sufficiently well in the q -dimensional approximation.

The stepwise CPC procedure is based on the standard power method (Golub and Van Loan, 1996) and estimates the j^{th} common eigenvector by minimising the criterion,

$$\sum_{i=1}^k (n_i - 1) \log(\mathbf{b}_j' \mathbf{S}_i \mathbf{b}_j), \quad (3.22)$$

subject to the usual orthogonality constraints, $\mathbf{b}_j' \mathbf{b}_j = 1$ and $\mathbf{b}_j' \mathbf{B}_{[j-1]} = \mathbf{0}$, where the columns of $\mathbf{B}_{[j-1]}$ are the first $j - 1$ common eigenvectors.

Trendafilov (2010) have further shown mathematically and also by numerical examples that when $l_{i1} \geq l_{i2} \geq \dots \geq l_{ip}$ for the k groups simultaneously, the stepwise CPC solution coincides with the solution given by the FG algorithm. If this is not the case, the stepwise CPC solution will still ensure that the subsequent eigenvalues are decreasing (or at least not increasing too much) within each of the groups. Stepwise CPC is thus better suited for dimensionality reduction than the FG algorithm, and the proportion of variation accounted for by the first $q < p$ stepwise CPC eigenvectors is usually equal to or greater than that of the first q common eigenvectors estimated with the FG algorithm.

On the other hand, the stepwise CPC algorithm does not attempt to minimise the criteria in (3.16) or (3.20) and thus generally performs worse than the FG and JADE algorithms in the simultaneous diagonalisation of symmetric matrices.

The stepwise CPC algorithm supplied by Trendafilov (2010) was implemented in the R function `stepwisecpc` which is given in Appendix B.

Two new algorithms for the estimation of common eigenvectors have recently been developed by Browne and McNicholas (2014b) and Browne and McNicholas (2014a), namely the *accelerated line search (ALS)* and *majorisation-minimisation (MM)* algorithms. They applied these new algorithms in the mixture model-based clustering context, showing that it surpasses the FG algorithm in speed, specifically in higher-dimensional ($p \geq 20$) situations. For the $k = 5$, $p = 100$ case considered in the simulation study reported in Browne and McNicholas (2014a), the computational time of the FG algorithm became prohibitively large. However, the FG algorithm seems slightly superior to the ALS and MM algorithms in terms of the convergence criterion used.

3.6 Partial CPC

Suppose that only q of the eigenvectors are common to all k population covariance matrices. This is the case, discussed by Flury (1987), where

$$\Sigma_i = \mathbf{B}_i \mathbf{\Lambda}_i \mathbf{B}_i', \quad (3.23)$$

with

$$\mathbf{B}_i = \begin{bmatrix} \boldsymbol{\beta}_1 & \boldsymbol{\beta}_2 & \cdots & \boldsymbol{\beta}_q; & \boldsymbol{\beta}_{q+1}^{(i)} & \boldsymbol{\beta}_{q+2}^{(i)} & \cdots & \boldsymbol{\beta}_p^{(i)} \end{bmatrix}. \quad (3.24)$$

For the sake of simplicity we may assume that the q common eigenvectors are those associated with the largest eigenvalues across all k populations simultaneously, although it need not necessarily be the case. Either way, the common eigenvectors are grouped in the first part of the \mathbf{B}_i matrices, followed by the $p - q$ eigenvectors (orthogonal to the first q common eigenvectors) unique to each population covariance matrix.

Equations (3.23) and (3.24) describe the situation where there are *partial* common principal components in the k populations. It will henceforth be referred to as the CPC(q) model to indicate that q of the eigenvectors are common.

The appeal of the partial CPC model is in the situation where the last few eigenvectors in each of the population covariance matrices may account for random noise or a negligible amount of variation. If these last $p - q$ eigenvectors differ across the population covariance matrices but the first q eigenvectors are common, CPC hypothesis (3.1) may be rejected in favour of the unrelated covariance matrices model, even though the last $p - q$ non-common components will possibly be discarded after performing the CPC analysis. In this scenario, the partial CPC model is more appropriate than assuming the covariance matrices are completely unrelated.

Each of the population covariance matrices will have its own set of eigenvalues, contained in the diagonals of the $\mathbf{\Lambda}_i$ matrices. The common eigenvectors need not have the same rank order in terms of proportion of variation accounted for in each population. For example, the first common eigenvector may correspond to the first eigenvector of the first population covariance matrix (i.e. $\boldsymbol{\eta}_{11}$), the third eigenvector of the second population covariance matrix ($\boldsymbol{\eta}_{23}$) and the last eigenvector of the third population covariance matrix ($\boldsymbol{\eta}_{3p}$). As in the full CPC situation, one of the challenges is to determine which of the eigenvectors in the population covariance matrices are common (a question which will be addressed in Chapter 4).

To be consistent with the sorting of the eigenvectors in PCA, the eigenvectors in the \mathbf{B}_i may be sorted as follows: First the common eigenvectors are sorted in the order of the sums of the associated eigenvalues across all the groups, so that $\lambda_{11} + \dots + \lambda_{k1} \geq \lambda_{12} + \dots + \lambda_{k2} \geq \dots \geq \lambda_{1p} + \dots + \lambda_{kp}$, where λ_{ij} indicates the j^{th} eigenvalue of the i^{th} covariance matrix. Thereafter the non-common eigenvectors are sorted in order of their importance in the i^{th} group.

In terms of Flury's hierarchy as will be described in Section 3.7, there is a range of partial CPC models between the full CPC and unrelated covariance matrices levels. After full CPC, the models CPC($p - 2$), CPC($p - 3$) down to CPC(1) follow, with each subsequent model entailing another relaxation of the constraints on the k covariance matrices. However, each step down in the

hierarchy also involves the estimation of an additional number of parameters.

Note that a $\text{CPC}(p-1)$ model is not possible, because if the first $p-1$ eigenvectors are common, the last eigenvector would also be common due to the orthogonality constraint. $\text{CPC}(p-1)$ will therefore imply the full CPC model. The partial CPC model thus requires $p \geq 3$ dimensions.

Estimation of partial CPC parameters

Estimation of the eigenvectors under the $\text{CPC}(q)$ model is more complex than for the full CPC model. The likelihood equations are given by Flury (1988). Let

$$\lambda_{ij} = \begin{cases} \boldsymbol{\beta}_j' \mathbf{S}_i \boldsymbol{\beta}_j & j = 1, \dots, q \\ \boldsymbol{\beta}_j^{(i)'} \mathbf{S}_i \boldsymbol{\beta}_j^{(i)} & j = q+1, \dots, p \end{cases} \quad (3.25)$$

for $i = 1, \dots, k$, with the Lagrange multiplier,

$$\delta_{uj}^{(i)} = \frac{(n_i - 1) \boldsymbol{\beta}_u' \mathbf{S}_i \boldsymbol{\beta}_j^{(i)}}{\lambda_{ij}} \quad 1 \leq u \leq q, \quad (3.26)$$

the non-common eigenvectors of each covariance matrix, $\{\boldsymbol{\beta}_j^{(i)}\}$, should satisfy the equation,

$$\boldsymbol{\beta}_j^{(i)'} \mathbf{S}_i \boldsymbol{\beta}_h^{(i)} = 0 \quad j \neq h. \quad (3.27)$$

The equations

$$\boldsymbol{\beta}_u' \left(\sum_{i=1}^k \frac{\lambda_{iu} - \lambda_{ih}}{\lambda_{iu} \lambda_{ih}} (n_i - 1) \mathbf{S}_i \right) \boldsymbol{\beta}_h = 0, \quad (3.28)$$

$$1 \leq u; h \leq q; u \neq h,$$

and

$$\left(\frac{1}{\lambda_{rj}} - \frac{1}{\lambda_{ru}} \right) (n_r - 1) \boldsymbol{\beta}_u' \mathbf{S}_r \boldsymbol{\beta}_j^{(r)} = \boldsymbol{\beta}_j^{(r)'} \left[\sum_{\substack{i=1 \\ i \neq r}}^k \left(\frac{(n_i - 1) \mathbf{S}_i \boldsymbol{\beta}_u}{\lambda_{iu}} - \sum_{h=q+1}^p \delta_{uh}^{(i)} \boldsymbol{\beta}_h^{(i)} \right) \right],$$

$$i, r = 1, \dots, k; 1 \leq u \leq q < j \leq p, \quad (3.29)$$

should be solved under the orthogonality constraints

$$\beta'_j \beta_h = \begin{cases} 0 & \text{if } j \neq h \\ 1 & \text{if } j = h, \text{ for } 1 \leq j, h \leq q \end{cases} \quad (3.30)$$

$$\beta_j^{(i)'} \beta_h^{(i)} = \begin{cases} 0 & \text{if } j \neq h \\ 1 & \text{if } j = h, \text{ for } q < j, h \leq p; i = 1, \dots, k \end{cases} \quad (3.31)$$

and

$$\beta_j' \beta_h^{(i)} = 0, \text{ for } i = 1, \dots, k; 1 \leq j \leq q < h \leq p. \quad (3.32)$$

The constraints in (3.30), (3.31) and (3.32) mean that the common eigenvectors are normalised and mutually orthogonal, the non-common eigenvectors are normalised and mutually orthogonal within each group, and the non-common eigenvectors unique to each covariance matrix should be orthogonal to the common eigenvectors.

However, given that the estimates of the common eigenvectors under the full CPC model (as can be estimated with the FG algorithm) are available, an approximate partial CPC solution can be obtained with the following algorithm given by Flury (1988): Partitioning $\mathbf{B} = (\mathbf{B}_1 : \mathbf{B}_2)$ so that \mathbf{B}_1 contains the q common eigenvectors, orthogonal $(p - q) \times (p - q)$ matrices \mathbf{Q}_i are found so that

$$\mathbf{Q}_i' \mathbf{B}_2' \mathbf{S}_i \mathbf{B}_2 \mathbf{Q}_i \quad (3.33)$$

is diagonal. The approximate estimates of the non-common eigenvectors specific to the i^{th} group are given by

$$\mathbf{B}_2^{(i)} = \mathbf{B}_2 \mathbf{Q}_i, \quad (3.34)$$

for $i = 1, \dots, k$.

3.7 Flury's hierarchy

When the covariance matrices of k populations are compared in practice, an overall test of equality such as Box's M test (Box, 1949) is often performed. If the null hypothesis of equality is rejected, the covariance matrices are generally assumed to be completely unrelated in subsequent analyses.

However, there are a number of ways in which covariance matrices can differ between the two extremes of equality and total heterogeneity. Flury (1988) organised the levels of similarity between k covariance matrices in a helpful hierarchy (see Table 3.1). Although the common space model (Level

3.8. INFERENCE FOR EIGENVALUES AND COMMON EIGENVECTORS 57

4^*) is part of Flury's hierarchy, it was not studied and will not be mentioned any further in this dissertation.

For large p and k , assuming complete heterogeneity of the covariance matrices when the proportional or CPC model may be more appropriate will substantially increase the number of parameters to be estimated. Thus for relatively small data sets the degrees of freedom available for error estimation may be decreased considerably, leading to an overall decrease in the precision of the analysis.

The higher levels in Flury's hierarchy encompasses all of the lower levels. For example, equal covariance matrices (Level 1) are also proportional (Level 2, with a proportionality constant of $\rho = 1$) and have common eigenvectors (Level 3). Moving to lower levels in the hierarchy, the restrictions on the covariance matrices are relaxed but the number of parameters that need to be estimated increases. Good practice would entail using the highest level model which provides an adequate fit for the data. Finding the appropriate model to work with is the subject of Chapter 4.

A sub-hierarchy of models within the partial CPC level, from $\text{CPC}(p-2)$, $\text{CPC}(p-3)$ down to $\text{CPC}(1)$, may also be included. Due to the orthogonality constraint, $\text{CPC}(p-1)$ is not possible as commonality of $p-1$ principal components would mean that the p^{th} component would also be common, thus implying the full CPC model.

3.8 Inference for the eigenvalues and common eigenvectors

The results for inference on the eigenvectors and eigenvalues under the CPC and partial CPC models are discussed here to illustrate the methodology

Table 3.1: Flury's hierarchy of similarities between k covariance matrices (Flury, 1988).

Level	Model	Number of parameters
1	Equality	$\frac{1}{2}p(p-1) + p$
2	Proportionality	$\frac{1}{2}p(p-1) + p + k - 1$
3	CPC	$\frac{1}{2}p(p-1) + kp$
4	Partial CPC (or $\text{CPC}(q)$)	$\frac{1}{2}p(p-1) + kp + \frac{1}{2}(k-1)(p-q)(p-q-1)$
4^*	<i>Common space</i> ($\text{CS}(q)$)	$\frac{1}{2}p(p-1) + kp + \frac{1}{2}(k-1)(p-q)(p-q-1) + \frac{1}{2}(k-1)q(q-1)$
5	Heterogeneity	$k \left[\frac{1}{2}p(p-1) + p \right]$

available to the researcher. Although use of the FG algorithm to estimate common eigenvectors and eigenvalues of covariance matrices of multivariate non-normal populations under the CPC model is justified, the parametric methods for inference on the common eigenvectors and eigenvalues depend on the assumption that the populations are distributed multivariate normal.

All of the results in Sections 3.8.1 and 3.8.2 are from Flury (1988), unless otherwise indicated.

3.8.1 Inference for the eigenvalues

For a sufficiently large sample from the i^{th} population, the standard error of the j^{th} sample eigenvalue of the i^{th} covariance matrix under the CPC model, $l_{ij} \sim N(\lambda_{ij}, 2\frac{\lambda_{ij}^2}{n_i-1})$, is

$$s(l_{ij}) = \sqrt{\frac{2}{n_i-1}} l_{ij}. \quad (3.35)$$

As in (2.18), an approximate $(1-\alpha)100\%$ large sample confidence interval for λ_{ij} can be obtained with

$$\left[\frac{l_{ij}}{1 + z_{\alpha/2} \sqrt{\frac{2}{n_i-1}}} ; \frac{l_{ij}}{1 - z_{\alpha/2} \sqrt{\frac{2}{n_i-1}}} \right]. \quad (3.36)$$

As in the case of single group PCA, a more relevant problem is the estimation of the amount of variation accounted for in the i^{th} group by the last $p-q$ components together. For the purpose of dimensionality reduction, the last $p-q$ components of the i^{th} group may be discarded if

$$f_i = \frac{\sum_{j=q+1}^p \lambda_j}{\text{tr}(\Sigma_i)} \quad (3.37)$$

is sufficiently small. Letting $f_0 \in (0, 1)$, the hypothesis to test in this case is

$$H_0 : f_i \leq f_0, \quad (3.38)$$

for $i = 1, \dots, k$ simultaneously. The test statistics (one for each of the k populations) are given by

$$z_i(f_0) = \sqrt{\frac{n_i-1}{2}} \frac{(1-f_0) \sum_{j=q+1}^p l_{ij} - f_0 \sum_{j=1}^q l_{ij}}{\left[f_0^2 \sum_{j=1}^q l_{ij}^2 + (1-f_0)^2 \sum_{j=q+1}^p l_{ij}^2 \right]^{\frac{1}{2}}}, \quad (3.39)$$

3.8. INFERENCE FOR EIGENVALUES AND COMMON EIGENVECTORS 59

and hypothesis (3.38) is rejected if

$$\max z_i(f_0) > z_\gamma, \quad 1 \leq i \leq k,$$

where $\gamma = 1 - (1 - \alpha)^{\frac{1}{k}}$, and z_γ refers to the upper γ^{th} quantile of the standard normal distribution.

An approximate one-sided large-sample $(1 - \alpha)100\%$ confidence interval for the f_i can be constructed with

$$0 < f_i \leq \hat{f}_i + z_\alpha \sqrt{\frac{2}{n_i - 1} \frac{\left[\left(\sum_{j=1}^q l_{ij} \right)^2 \sum_{j=q+1}^p l_{ij}^2 + \left(\sum_{j=q+1}^p l_{ij} \right)^2 \sum_{j=1}^q l_{ij}^2 \right]^{\frac{1}{2}}}{[\text{tr}(\mathbf{S}_i)]^2}}. \quad (3.40)$$

If the true dimensionality of the \mathbf{S}_i is $q < p$, the variation accounted for by the last $p - q$ components may be due to random noise. In such a case, there may be sphericity in the last $p - q$ components. Flury (1986) developed a likelihood ratio test for equality of the last $p - q$ eigenvalues in the k groups simultaneously, testing the hypothesis

$$H_0 : \lambda_{i(q+1)} = \dots = \lambda_{ip} \quad (3.41)$$

for $i = 1, \dots, k$ simultaneously. With $\tilde{l}_{ij}, j = 1, \dots, p$, denoting the maximum likelihood estimators of the eigenvalues of the i^{th} covariance matrix under hypothesis (3.41), and

$$\tilde{l}_i^* = \frac{\sum_{j=q+1}^p \tilde{l}_{ij}}{p - q}, \quad i = 1, \dots, k, \quad (3.42)$$

the log-likelihood ratio statistic for hypothesis (3.41) is

$$X_S^2 = \sum_{i=1}^k (n_i - 1) \ln \frac{\left(\tilde{l}_i^* \right)^{p-q} \prod_{j=1}^q \tilde{l}_{ij}}{\prod_{j=1}^p l_{ij}}. \quad (3.43)$$

Under hypothesis (3.41), the test statistic in (3.43) is distributed asymptotically chi-squared with $(p - q - 1)(p - q + 2k)/2$ degrees of freedom (Flury, 1986).

Letting

$$l_i^* = \frac{\sum_{j=q+1}^p l_{ij}}{p - q} \quad (3.44)$$

be the mean of the last $p - q$ eigenvalues of the i^{th} group, an approximate log-likelihood ratio test statistic for hypothesis (3.41) can be obtained with

$$X_S^2(\text{approx}) = \sum_{i=1}^k (n_i - 1) \ln \frac{(l_i^*)^{p-q}}{\prod_{j=q+1}^p l_{ij}}. \quad (3.45)$$

Under hypothesis (3.41), $X_S^2(\text{approx})$ is distributed approximately chi-squared with $(p - q - 1)(p - q + 2k)/2$ degrees of freedom. However, Flury (1986) noted that, because $X_S^2(\text{approx})$ will always be larger or equal to the exact log-likelihood ratio test statistic in (3.43), it may be used to confirm the non-rejection of hypothesis (3.41) but not necessarily to reject it.

A different view was taken by Yuan and Bentler (1994) who suggested that, even if the last $p - q$ population eigenvalues are equal, the sample eigenvalues will generally not be equal but rather exhibit a linear trend due to sampling variation and measurement error. Therefore the null hypothesis of equality will almost always be rejected and they propose to rather test the hypothesis of a linear trend in the last $p - q$ eigenvalues, i.e.

$$H_0 : \lambda_{ij} = \alpha_i + \beta_i x_{ij}, \quad j = q + 1, \dots, p. \quad (3.46)$$

Letting \tilde{l}_{ij} , $\tilde{\alpha}_i$ and $\tilde{\beta}_i$ indicate the maximum likelihood estimators under linear trend hypothesis (3.46), they derived a likelihood ratio test statistic,

$$\chi_L^2 = \sum_{i=1}^k (n_i - 1) \left[\ln \frac{\prod_{j=1}^q \tilde{l}_{ij} \prod_{t=1}^{p-q} (\tilde{\alpha}_i + \tilde{\beta}_i x_{it})}{\prod_{j=1}^p \hat{l}_{ij}} + \sum_{t=1}^{p-q} \frac{\tilde{l}_{i(q+t)}}{(\tilde{\alpha}_i + \tilde{\beta}_i x_{it})} - (p - q) \right], \quad (3.47)$$

for $t = 1, \dots, p - q$. Under the linear trend hypothesis, the test statistic in (3.47) is distributed asymptotically chi-squared with $k(p - q - 2)$ degrees of freedom.

For partial CPC models, inference for the k sets of eigenvalues remain relatively uncomplicated. Given that

$$l_{ij} = \begin{cases} \mathbf{b}_j' \mathbf{S}_i \mathbf{b}_j & j = 1, \dots, q \\ \mathbf{b}_j^{(i)'} \mathbf{S}_i \mathbf{b}_j^{(i)} & j = q + 1, \dots, p \end{cases}, \quad (3.48)$$

the pk eigenvalues are distributed independently of each other and independently of the eigenvector matrices, \mathbf{B}_i , as

$$l_{ij} \sim N(\lambda_{ij}, \frac{2\lambda_{ij}^2}{n_i - 1}). \quad (3.49)$$

3.8. INFERENCE FOR EIGENVALUES AND COMMON EIGENVECTORS 61

The standard errors of the l_{ij} will therefore be given as in (3.35), and all the tests on the eigenvalues mentioned in this section will also be applicable to the partial CPC setting.

Because the aforementioned tests for the eigenvalues are based on the assumption that the population distributions are multivariate normal, they may not be applicable to data originating from non-normal multivariate distributions. Bootstrap or jackknife estimates of standard errors of the eigenvalues may be helpful to do inference in that case.

3.8.2 Inference for the common eigenvectors

Under the assumption of multivariate normal distributions in the populations and letting β_{jh} and b_{jh} denote the h^{th} loading of the j^{th} common eigenvector for the population and the sample, respectively,

$$b_{jh} \sim N(\beta_{jh}, \frac{v_{jh}}{n-1}), \quad (3.50)$$

with v_{jh} being the $(h, h)^{th}$ element of the matrix

$$\sum_{\substack{j=1 \\ j \neq h}}^p \theta_{jh} \beta_j \beta_j' \quad (3.51)$$

where

$$\theta_{jh} = \left[\frac{1}{n-1} \sum_{i=1}^k (n_i - 1) \frac{(\lambda_{ij} - \lambda_{ih})^2}{\lambda_{ij} \lambda_{ih}} \right]^{-1}. \quad (3.52)$$

The sample estimator of the harmonic mean in (3.52) is (Flury, 1988)

$$\hat{\theta}_{jh} = \left[\frac{1}{n-1} \sum_{i=1}^k (n_i - 1) \frac{(l_{ij} - l_{ih})^2}{l_{ij} l_{ih}} \right]^{-1}, \quad (3.53)$$

and the asymptotic standard error for b_{jh} is given by

$$s(b_{jh}) = \sqrt{\frac{1}{n-1} \sum_{\substack{u=1 \\ u \neq j}}^p \hat{\theta}_{uj} b_{uh}^2}. \quad (3.54)$$

From (3.53) and (3.54) it can be seen that, if the CPCs are not well defined (i.e. the eigenvalues are not well separated) in at least one of the k populations, the $\hat{\theta}_{jh}$ values will be large and the common eigenvector loadings will have large standard errors.

Flury (1988) gave a test for the hypothesis that q of the common eigenvectors are simultaneously equal to a set of predetermined orthonormal vectors $(\beta_1^0, \dots, \beta_q^0)$,

$$H_q : (\beta_1, \dots, \beta_q) = (\beta_1^0, \dots, \beta_q^0), \quad (3.55)$$

for $q < p$. Under hypothesis (3.55), the test statistic

$$\chi_q^2 = (n-1) \left[\frac{1}{4} \sum_{j=1}^{q-1} \sum_{h=j+1}^q \frac{(\mathbf{b}'_h \beta_j^0 - \mathbf{b}'_j \beta_h^0)^2}{\hat{\theta}_{jh}} + \sum_{j=1}^q \sum_{h=q+1}^p \frac{(\mathbf{b}'_h \beta_j^0)^2}{\hat{\theta}_{jh}} \right] \quad (3.56)$$

is distributed asymptotically chi-squared with $q[p - (q+1)/2]$ degrees of freedom.

Additionally, it may be of interest to test whether $p - q$ variables are redundant in $j \in J$ of the common eigenvectors simultaneously, where J indicates a set of $m \leq q$ distinct integers between 1 and p . Partitioning the j^{th} common eigenvector as

$$\mathbf{b}_j = \begin{bmatrix} \mathbf{b}_j^{(1)} \\ \mathbf{b}_j^{(2)} \end{bmatrix}, \quad (3.57)$$

where $\mathbf{b}_j^{(2)}$ contains the loadings for the $p - q$ possibly redundant variables, the hypothesis

$$H_J(q) : \beta_j^{(2)} = \mathbf{0} \quad \text{for all } j \in J \text{ simultaneously} \quad (3.58)$$

can be tested with the statistic

$$T_J = (n-1) \sum_{j \in J} \mathbf{b}_j^{(2)'} \left(\sum_{h \notin J} \hat{\theta}_{jh} \mathbf{b}_h^{(2)} \mathbf{b}_h^{(2)'} \right)^{-1} \mathbf{b}_j^{(2)}. \quad (3.59)$$

Under hypothesis (3.58), the test statistic in (3.59) is distributed asymptotically chi-squared with $m(p - q)$ degrees of freedom.

However, as pointed out by Jolliffe (2002), the aforementioned asymptotic theory results are only applicable to CPC analysis on covariance matrices, and not to CPC analysis on correlation matrices. The asymptotic results also depend on multivariate normality in the populations, which may not be a workable assumption for many real data sets. For data from non-normal multivariate populations, bootstrap distributions of the common eigenvector loadings may be used to do inference on the eigenvectors.

In addition to the aforementioned problems, theory for inference on the common eigenvectors under the partial CPC model is not yet known.

3.9 Interpreting the common eigenvectors

The common eigenvectors (and non-common eigenvectors under the partial CPC model) may be interpreted in the same way as the eigenvectors in the single group PCA case. One of the advantages of CPC analysis, however, is that the common eigenvectors are estimated by combining the information from all of the groups, and because of the larger combined sample the estimators are usually more precise than if the groups were to be analysed separately. As noted by Flury (1986), the common eigenvectors should be interpreted only if they are well defined (i.e. if there is not sphericity in the components, causing the standard errors of the common eigenvector loadings to be large).

The common eigenvectors provide a “consensus view” over the k groups of the relationships among the variables. Simultaneous large loadings for a subset of variables in a single eigenvector indicate strong correlation between the variables.

The existence of common eigenvectors shows that the sources of variation are the same for the different groups, but the relative importance of these sources may vary between the groups. Comparing the proportion of variation accounted for by each of the common eigenvectors within each group can provide valuable information on how the sources of variation differ in relative importance across the groups.

Another use of CPC analysis is in longitudinal studies where the same set of variables are measured on the same individuals at different points in time. The different time points can be treated as “groups”, and it may be of interest to investigate how the relative importance of the sources of variation change over time (Jolliffe, 2002). In this context the CPC model for dependent random vectors (Neuenschwander and Flury, 2000) might be more appropriate though, as the CPC model as discussed in this dissertation is based on the assumption that the populations are independent of each other.

In a similar context, CPC analysis of different groups of the same species or biological type may be useful in the study of evolution or morphology. Changes in the relative importance of the common eigenvectors over time can help to understand how the effects of different sources of variation change across developmental stages.

Despite the advantages CPC analysis brought to the analysis of biological data, Houle et al. (2002) issued a word of caution to biologists attempting to infer causal relationships from the results of CPC analyses. They noted that there is a distinct difference about the way structure and causal relationships are understood in real biological data and the way it is viewed in CPC anal-

ysis, and found that a single change in a non-orthogonal causal factor can sometimes lead to the conclusion that the covariance matrices of two very similar groups are completely unrelated.

3.10 Other research related to the CPC model

Several researchers have been working on the theory and application of the CPC model and derivatives thereof since its introduction by Flury in the 1980's.

Hills (1982), Klingenberg (1996) and Bartoletti et al. (1999) extended the theoretical basis of the multivariate allometric growth model introduced by Jolicoeur (1963). The allometric growth model may be seen as a special case of the CPC model (Tarpey, 2000). Klingenberg and Froese (1991) tested the assumption of a common growth pattern in the larvae of 17 marine fish species, and employed bootstrap methods to assess the accuracy of the allometric growth model.

A graphical procedure for comparing the eigenvectors of several covariance matrices was proposed by Keramidas et al. (1987).

Schott (1988) proposed an approximate test of the partial CPC model for two groups. Schott (1991a) extended this test to more than two groups, and also provided extensions to correlation based analyses and robust PCA.

Krzanowski (1990) proposed two methods for between group analysis using the CPC model, and compared them using test score data from Venezuelan students attending ten different British colleges.

The use of the CPC model in the Behrens-Fisher problem was explored by Nel and Pienaar (1998). Flury and Neuenschwander (1995) considered the same problem, but for the more specific case of testing whether a subset of the p means are equal.

Flury and Neuenschwander (1995) and Neuenschwander and Flury (2000) investigated the situation where the assumption of the CPC model that the k groups are independent is not valid, as in longitudinal studies where the goal is to compare sets of repeated measurements on the same group of individuals. In such cases the sets of measurements taken at the different time points are not independent of each other.

Schott (1998) developed a method for the estimation of correlation matrices under the CPC model.

Boik (2002) proposed a broader and more flexible spectral model for the simultaneous eigenstructure of several covariance matrices and derived the asymptotic distributions of the proposed estimators. He also gave likelihood

tests for the model, and noted that most of the related common eigenstructure models may be seen as special cases of this spectral model.

Boente and Orellana (2001) proposed two approaches for robust estimation under the CPC model, one based on projection-pursuit, and another based on replacing the unbiased sample covariance matrices with robust versions. Additional results for the projection-pursuit method were given by Boente et al. (2002) and Boente et al. (2006), with Boente et al. (2008) extending the idea further to weighting of the projection-pursuit estimators. Boente et al. (2010a) discussed the asymptotic behaviour of the general projection-pursuit estimators under the CPC model.

Boente et al. (2002) also discussed the application of the projection-pursuit influence functions to detect outliers. Boente et al. (2010b) improved on this idea, showing that improved detection of outliers (for small samples in particular) is possible using cutoff values computed with a cross-validation approach under the CPC model. Gu and Fung (2001) also discussed the detection of influential observations in CPC analyses and illustrated their approach with a numerical example.

Boente et al. (2009) derived robust log-likelihood tests for the CPC model versus proportional and heterogeneous covariance matrices, respectively. They also compared the results of these newly proposed robust tests to the classical tests in a small simulation study.

Boente et al. (2010c) presented the covariance matrix estimators and their asymptotic distributions under the functional CPC model.

Use of the CPC model have also been described in a number of applied research settings. Reyment (1997) illustrated the use of the CPC model in three examples from paleontology, while Steppan (1997) employed the CPC model in the study of macro-evolutionary patterns.

In the field of atmospheric science, Sengupta and Boyle (1998) found the CPC model useful in the comparison of members of an ensemble of forecasts from a general circulation model.

A number of authors have used the CPC model in the comparison of genetic variance-covariance matrices (otherwise known as G matrices). Phillips and Arnold (1999) provided a helpful framework for such comparisons, and described a randomisation adaptation to the CPC analysis which they employed to deal with the dependence in the G matrices. Phillips et al. (2001) used the same approach to study changes in the G matrices of the *Drosophila melanogaster* fruit fly due to inbreeding.

In botanical research, Waldmann and Andersson (2000) employed the CPC model and the approach from Phillips and Arnold (1999) to study differences in the G matrices of two types of pincushion flowers, *Scabiosa columbaria* and *Scabiosa canescens*. They also produced biplots of their

data, but these were not strictly CPC biplots. The topic of CPC biplots will be discussed in Chapter 7.

However, Mezey and Houle (2003) found that G matrices will only have common principal components under some very specific circumstances. They concluded that common eigenvectors will exist only if the populations have modules in common and that the apparent effectiveness of the approach outlined by Phillips and Arnold (1999) is due to the dependence and order of the tests in Flury's hierarchy. They found the CPC model to be a useful tool in their context, but advised caution in the interpretation of the results.

Berner (2011) investigated the reliability of biological size correction based on the CPC model. Berner (2011) found the generally used CPC-based approaches to be inappropriate, and advocated their abandonment in favour of univariate general linear models with the first common eigenvector (of a subset of traits of the biological organism) as a size metric input.

3.11 Application to the VON data

CPC analysis was performed on the delivery mode and regional groupings in the VON 2009 cohort, and the results are given in the following two sections. Infants who died or were transferred to other hospitals were not excluded in these analyses.

3.11.1 Delivery mode

The VON 2009 cohort contains observations on $n_1 = 2549$ infants delivered by Caesarean section. The other $n_2 = 492$ infants were delivered by normal vertex (vaginal) delivery. Sample covariance matrices for the two delivery mode groups are given below:

- Caesarean ($n_1 = 2549$)

$$\mathbf{S}_1 = \begin{bmatrix} 0.664 & 0.334 & 0.261 & 2.270 & 2.225 & 0.173 \\ 0.334 & 3.404 & 2.121 & 1.450 & 1.386 & 0.195 \\ 0.261 & 2.121 & 2.256 & 1.249 & 1.070 & 0.167 \\ 2.270 & 1.450 & 1.249 & 11.383 & 8.755 & 0.609 \\ 2.225 & 1.386 & 1.070 & 8.755 & 10.348 & 0.601 \\ 0.173 & 0.195 & 0.167 & 0.609 & 0.601 & 0.493 \end{bmatrix}$$

- Vaginal ($n_2 = 492$)

$$\mathbf{S}_2 = \begin{bmatrix} 0.737 & 0.350 & 0.329 & 3.207 & 2.700 & 0.287 \\ 0.350 & 5.195 & 3.611 & 1.818 & 1.308 & 0.749 \\ 0.329 & 3.611 & 3.428 & 1.743 & 1.167 & 0.695 \\ 3.207 & 1.818 & 1.743 & 18.227 & 13.399 & 1.471 \\ 2.700 & 1.308 & 1.167 & 13.399 & 13.216 & 1.264 \\ 0.287 & 0.749 & 0.695 & 1.471 & 1.264 & 0.948 \end{bmatrix}$$

The eigenvectors for the two separate covariance matrices are given in Table 3.2 together with the common eigenvectors estimated with the FG algorithm (under the assumption that the two groups indeed have p eigenvectors in common). The eigenvalues of the two groups under the CPC hypothesis are given at the bottom of the table, together with the variance and cumulative variance accounted for by each common eigenvector. Cumulatively, the first three common eigenvectors account for more than 95% of the variance observed in each of the groups.

The angles between the eigenvectors of the first group (\mathbf{e}_{1j}) and the eigenvectors of the second group (\mathbf{e}_{2j}) are

	\mathbf{e}_{21}	\mathbf{e}_{22}	\mathbf{e}_{23}	\mathbf{e}_{24}	\mathbf{e}_{25}	\mathbf{e}_{26}
\mathbf{e}_{11}	4.3°	88.7°	86.4°	88.3°	89.5°	89.0°
\mathbf{e}_{12}	88.5°	4.9°	89.1°	85.6°	88.7°	89.5°
\mathbf{e}_{13}	86.4°	88.9°	4.4°	88.5°	89.3°	88.5°
\mathbf{e}_{14}	89.7°	89.1°	89.9°	62.7°	27.3°	88.6°
\mathbf{e}_{15}	88.6°	85.5°	88.4°	28.2°	63.0°	84.1°
\mathbf{e}_{16}	88.9°	90.0°	88.3°	85.5°	86.1°	6.3°

From an inspection of the angles between the eigenvectors, it seems that the fourth eigenvector of the *Caesarean* group is similar to the fifth eigenvector of the *Vaginal* group, and the fifth eigenvector of the *Caesarean* group is similar to the fourth eigenvector of the *Vaginal* group. The rest of the eigenvectors with similar loadings in the two groups also have the same rankings within the groups.

The value of the overall measure of similarity between the two separate sets of eigenvectors (according to equation 3.14) is

$$\text{tr} [\text{abs}(\mathbf{E}'_1 \mathbf{E}_2)] = 4.896.$$

The similarity score (compared to the maximum of $p = 6$) is deceptive in this case, as it compares the first eigenvectors of the two groups, the second eigenvectors of the two groups, and so forth. This means that the fourth

Table 3.2: Eigenvectors of the delivery mode covariance matrices separately, and estimated common eigenvectors under the CPC hypothesis. Eigenvalues of the two groups under the CPC model are given at the bottom of the table, together with the variance and cumulative variance accounted for by each common eigenvector.

Separate eigenvectors						
	e_{i1}	e_{i2}	e_{i3}	e_{i4}	e_{i5}	e_{i6}
Caesarean						
BWGT	-0.16	0.03	-0.03	-0.01	0.10	0.98
AP1	-0.13	-0.78	-0.04	-0.61	0.01	-0.01
AP5	-0.10	-0.60	0.03	0.79	-0.07	0.02
GESTAGE	-0.71	0.11	0.69	-0.04	-0.03	-0.09
BHEADCIR	-0.67	0.12	-0.72	0.03	-0.05	-0.13
ATEMP	-0.04	-0.03	-0.01	0.06	0.99	-0.11
Vaginal						
BWGT	-0.14	0.02	0.01	0.00	-0.01	0.99
AP1	-0.10	-0.78	0.06	-0.33	0.52	0.00
AP5	-0.09	-0.60	-0.01	0.25	-0.75	-0.01
GESTAGE	-0.75	0.08	-0.64	-0.03	0.03	-0.10
BHEADCIR	-0.62	0.13	0.76	-0.05	-0.05	-0.10
ATEMP	-0.07	-0.11	0.05	0.91	0.40	-0.00
Common eigenvectors						
	b_1	b_2	b_3	b_4	b_5	b_6
BWGT	0.15	-0.02	0.03	-0.01	0.06	0.99
AP1	0.13	0.78	0.05	-0.61	-0.03	-0.01
AP5	0.10	0.60	-0.03	0.79	-0.04	0.01
GESTAGE	0.72	-0.10	-0.68	-0.04	-0.03	-0.09
BHEADCIR	0.66	-0.13	0.73	0.03	-0.05	-0.13
ATEMP	0.05	0.04	0.01	0.01	1.00	-0.07
Caesarean						
l_{1j}	20.63	4.59	2.10	0.63	0.45	0.14
Variance	72.3%	16.1%	7.4%	2.2%	1.6%	0.5%
Cum. variance	72.3%	88.3%	95.7%	97.9%	99.5%	100.0%
Vaginal						
l_{2j}	30.43	7.67	2.15	0.59	0.76	0.14
Variance	72.9%	18.4%	5.2%	1.4%	1.8%	0.3%
Cum. variance	72.9%	91.3%	96.4%	97.8%	99.7%	100.0%

eigenvector of the *Caesarean* group is compared to the fourth eigenvector of the *Vaginal* group, and also the fifth eigenvector is compared with the fifth. If the fourth and fifth eigenvectors of the *Vaginal* group is switched around, the similarity score increases to 5.754. The directions of variation in the two groups thus appear to be similar, even though their order of importance differ. In terms of the common eigenvectors, the fourth eigenvector is a contrast of the Apgar scores at one minute and five minutes (thus an indication of the change in the feasibility of life) and is a more important source of variation in the *Caesarean* group than the fifth common eigenvector (which is dominated by the temperature variable). For the *Vaginal* group, differences in temperature (fifth common eigenvector) appears to be a slightly more important source of variation than the change in the feasibility of life (fourth common eigenvector).

Estimated eigenvectors for the delivery mode groups using the FG, JADE and stepwise CPC algorithms, respectively, are given for comparison in Table 3.3. It is clear that the common eigenvector loadings estimated with the three different algorithms do not differ much.

Bootstrap percentiles (1000 replications) were used to calculate 95% confidence intervals for the common eigenvectors loadings and the eigenvalues. These confidence intervals are reported in Table 3.4.

Even though the numerical variables in the VON data do not have a multivariate normal distribution, the standard errors of the eigenvalues and eigenvector loadings under the CPC model were estimated using the parametric methods discussed in this chapter (equations 3.35 and 3.54), in order to compare it with the bootstrap estimates. The parametric and bootstrap standard errors are shown in the top and bottom halves of Table 3.5, respectively. The bootstrap standard errors were calculated using the method from Diaconis and Efron (1983) given in (2.30). In almost all cases, the bootstrap standard errors are larger than the parametric standard errors. This is particularly noticeable in the standard errors for the eigenvalues, where the bootstrap standard errors for the first three eigenvalues (of both the *Caesarean* and *Vaginal* groups) are much larger than the parametric estimates.

Parametric and bootstrap confidence intervals for the eigenvalues of the covariance matrices of the delivery mode groups are given in Table 3.6. The parametric confidence intervals were estimated using (3.36), and the bootstrap confidence limits were obtained as the 2.5th and 97.5th percentiles of the bootstrap replications of the eigenvalues. The bootstrap confidence intervals are in all cases wider than the parametric confidence intervals, which implies that the assumption of multivariate normality in the VON population leads to underestimation of the variability of the eigenvalues.

Table 3.3: Common eigenvectors of the delivery group covariance matrices, estimated with the FG, JADE and stepwise CPC algorithms, respectively. Below each set of eigenvectors, the percentage variance accounted for by each eigenvector in each of the delivery mode groups are given.

	b_1	b_2	b_3	b_4	b_5	b_6
Flury-Gautschi						
BWGT	0.15	-0.02	0.03	-0.01	0.06	0.99
AP1	0.13	0.78	0.05	-0.61	-0.03	-0.01
AP5	0.10	0.60	-0.03	0.79	-0.04	0.01
GESTAGE	0.72	-0.10	-0.68	-0.04	-0.03	-0.09
BHEADCIR	0.66	-0.13	0.73	0.03	-0.05	-0.13
ATEMP	0.05	0.04	0.01	0.01	1.00	-0.07
<i>Caesarean</i>	72.3%	16.1%	7.4%	2.2%	1.6%	0.5%
<i>Vaginal</i>	72.9%	18.4%	5.2%	1.4%	1.8%	0.3%
JADE						
BWGT	0.15	-0.02	0.03	-0.01	0.02	0.99
AP1	0.11	0.78	0.06	-0.57	-0.22	-0.01
AP5	0.10	0.60	-0.02	0.78	0.13	0.00
GESTAGE	0.74	-0.09	-0.66	-0.03	-0.04	-0.09
BHEADCIR	0.64	-0.13	0.75	0.04	-0.04	-0.12
ATEMP	0.06	0.08	0.02	-0.24	0.96	-0.03
<i>Caesarean</i>	72.1%	16.0%	7.5%	2.1%	1.7%	0.5%
<i>Vaginal</i>	73.1%	18.4%	5.0%	1.3%	1.8%	0.3%
Stepwise CPC						
BWGT	-0.16	0.02	0.03	-0.00	0.08	0.98
AP1	-0.13	-0.78	0.04	-0.60	0.07	-0.01
AP5	-0.10	-0.60	-0.03	0.78	-0.17	0.01
GESTAGE	-0.72	0.11	-0.68	-0.04	-0.02	-0.09
BHEADCIR	-0.66	0.12	0.73	0.03	-0.05	-0.13
ATEMP	-0.05	-0.04	0.01	0.18	0.98	-0.08
<i>Caesarean</i>	72.3%	16.1%	7.4%	2.2%	1.6%	0.5%
<i>Vaginal</i>	72.9%	18.4%	5.2%	1.5%	1.7%	0.4%

Table 3.4: 95% bootstrap confidence intervals for the estimated common eigenvectors loadings, eigenvalues, percentage variance and cumulative percentage variance accounted for by each of the common eigenvectors in the delivery mode groups of the VON 2009 cohort. The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

Common eigenvectors							
		b_1	b_2	b_3	b_4	b_5	b_6
BWGT	LL	0.151	-0.034	0.015	-0.038	0.035	0.982
	UL	0.158	-0.015	0.050	0.011	0.093	0.987
AP1	LL	0.104	0.769	0.007	-0.621	-0.124	-0.022
	UL	0.146	0.796	0.087	-0.584	0.083	0.006
AP5	LL	0.085	0.581	-0.063	0.773	-0.183	-0.004
	UL	0.122	0.614	0.009	0.804	0.089	0.029
GESTAGE	LL	0.709	-0.142	-0.690	-0.058	-0.045	-0.103
	UL	0.727	-0.068	-0.669	-0.015	-0.011	-0.081
BHEADCIR	LL	0.646	-0.161	0.719	0.012	-0.066	-0.140
	UL	0.667	-0.088	0.740	0.056	-0.027	-0.114
ATEMP	LL	0.042	0.025	-0.010	-0.145	0.973	-0.100
	UL	0.055	0.052	0.036	0.198	0.997	-0.043
Eigenvalues and percentages under the CPC model							
Caesarean							
l_{1j}	LL	19.4	4.2	1.9	0.6	0.4	0.1
	UL	21.9	5.0	2.4	0.7	0.5	0.2
Variance	LL	70.7	14.8	6.6	1.9	1.5	0.5
	UL	73.9	17.3	8.2	2.5	1.7	0.5
Cum. variance	LL	70.7	87.4	95.4	97.8	99.5	100.0
	UL	73.9	89.2	96.0	98.1	99.5	100.0
Vaginal							
l_{2j}	LL	25.7	6.4	1.8	0.4	0.6	0.1
	UL	34.8	9.1	2.5	0.7	1.0	0.2
Variance	LL	68.8	15.3	4.2	1.1	1.4	0.3
	UL	76.4	21.9	6.2	1.8	2.4	0.4
Cum. variance	LL	68.8	89.9	95.8	97.3	99.6	100.0
	UL	76.4	92.4	97.0	98.3	99.7	100.0

Table 3.5: Parametric and bootstrap standard errors of the eigenvector loadings and the eigenvalues of the delivery mode groups (VON 2009 cohort) under the CPC model. These values were calculated using the eigenvectors and eigenvalues obtained from the FG algorithm.

Eigenvector loadings: Parametric standard errors						
	CPC1	CPC2	CPC3	CPC4	CPC5	CPC6
BWGT	0.001	0.004	0.005	0.012	0.013	0.001
AP1	0.009	0.005	0.018	0.006	0.035	0.007
AP5	0.007	0.006	0.016	0.005	0.045	0.009
GESTAGE	0.004	0.016	0.005	0.010	0.008	0.004
BHEADCIR	0.005	0.016	0.005	0.011	0.009	0.004
ATEMP	0.003	0.006	0.011	0.057	0.001	0.013
Eigenvalues: Parametric standard errors						
<i>Caesarean</i>						
l_{1j}	0.127	0.060	0.041	0.022	0.019	0.011
<i>Vaginal</i>						
l_{2j}	0.352	0.177	0.094	0.049	0.056	0.024
Eigenvector loadings: Bootstrap standard errors						
	CPC1	CPC2	CPC3	CPC4	CPC5	CPC6
BWGT	0.002	0.005	0.009	0.012	0.014	0.001
AP1	0.011	0.007	0.019	0.009	0.048	0.007
AP5	0.009	0.009	0.019	0.008	0.064	0.009
GESTAGE	0.005	0.018	0.005	0.012	0.009	0.006
BHEADCIR	0.005	0.018	0.005	0.012	0.009	0.007
ATEMP	0.003	0.007	0.012	0.079	0.003	0.014
Eigenvalues: Bootstrap standard errors						
<i>Caesarean</i>						
l_{1j}	0.620	0.196	0.115	0.039	0.016	0.006
<i>Vaginal</i>						
l_{2j}	2.227	0.672	0.181	0.076	0.108	0.011

Table 3.6: 95% Parametric and bootstrap confidence intervals for the eigenvalues of the covariance matrices of the delivery mode groups (VON 2009 cohort) under the CPC model. The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

Eigenvalues: Parametric 95% C.I.								
<i>Caesarean</i>								
l_{1j}	LL	19.6	4.3	2.0	0.6	0.4	0.1	
	UL	21.8	4.9	2.2	0.7	0.5	0.1	
<i>Vaginal</i>								
l_{2j}	LL	27.0	6.8	1.9	0.5	0.7	0.1	
	UL	34.8	8.8	2.5	0.7	0.9	0.2	
Eigenvalues: Bootstrap 95% C.I.								
<i>Caesarean</i>								
l_{1j}	LL	19.4	4.2	1.9	0.6	0.4	0.1	
	UL	21.9	5.0	2.4	0.7	0.5	0.2	
<i>Vaginal</i>								
l_{2j}	LL	25.7	6.4	1.8	0.4	0.6	0.1	
	UL	34.8	9.1	2.5	0.7	1.0	0.2	

3.11.2 Regions

A total of $n_1 = 2921$ infants in the VON 2009 cohort were born in hospitals in South Africa, with the remaining $n_2 = 120$ infants born in a Namibian hospital. Sample covariance matrices for the two regional groups (including deaths and transfers) are given below:

- South Africa ($n_1 = 2921$)

$$\mathbf{S}_1 = \begin{bmatrix} 0.672 & 0.316 & 0.258 & 2.414 & 2.251 & 0.196 \\ 0.316 & 3.673 & 2.383 & 1.392 & 1.307 & 0.295 \\ 0.258 & 2.383 & 2.470 & 1.236 & 1.054 & 0.269 \\ 2.414 & 1.392 & 1.236 & 12.460 & 9.311 & 0.783 \\ 2.251 & 1.307 & 1.054 & 9.311 & 10.578 & 0.719 \\ 0.196 & 0.295 & 0.269 & 0.783 & 0.719 & 0.563 \end{bmatrix}$$

- Namibia ($n_2 = 120$)

$$\mathbf{S}_2 = \begin{bmatrix} 0.866 & 0.617 & 0.453 & 3.017 & 3.393 & 0.037 \\ 0.617 & 4.479 & 2.299 & 3.312 & 2.877 & -0.076 \\ 0.453 & 2.299 & 2.184 & 2.751 & 2.212 & -0.041 \\ 3.017 & 3.312 & 2.751 & 15.365 & 13.449 & -0.311 \\ 3.393 & 2.877 & 2.212 & 13.449 & 15.702 & 0.048 \\ 0.037 & -0.076 & -0.041 & -0.311 & 0.048 & 0.498 \end{bmatrix}$$

The eigenvectors for the two separate covariance matrices are given in Table 3.7 together with the common eigenvectors estimated with the FG algorithm (under the assumption that the two groups indeed have p eigenvectors in common). The eigenvalues of the two groups under the CPC hypothesis are given at the bottom of the table, together with the variance and cumulative variance accounted for by each common eigenvector. Cumulatively, the first three common eigenvectors account for about 96% of the variance observed in each of the groups. For dimension reduction purposes, the first three common principal components should therefore provide a sufficient approximation of the six original variables.

The angles between the eigenvectors of the *South Africa* group (\mathbf{e}_{1j}) and the eigenvectors of the *Namibia* group (\mathbf{e}_{2j}) are

	\mathbf{e}_{21}	\mathbf{e}_{22}	\mathbf{e}_{23}	\mathbf{e}_{24}	\mathbf{e}_{25}	\mathbf{e}_{26}
\mathbf{e}_{11}	5.9°	86.8°	87.0°	90.0°	86.1°	89.9°
\mathbf{e}_{12}	86.4°	14.6°	79.4°	80.8°	88.8°	89.2°
\mathbf{e}_{13}	87.1°	78.9°	16.5°	85.7°	80.2°	85.3°
\mathbf{e}_{14}	89.0°	81.6°	84.8°	10.8°	86.0°	88.9°
\mathbf{e}_{15}	86.5°	87.8°	80.2°	86.5°	11.4°	88.0°
\mathbf{e}_{16}	90.0°	88.4°	85.1°	89.0°	88.9°	5.3°

3.11. APPLICATION TO THE VON DATA

75

Table 3.7: Eigenvectors of the regional covariance matrices separately, and estimated common eigenvectors under the CPC hypothesis. Eigenvalues of the two groups under the CPC model are given at the bottom of the table, together with the variance and cumulative variance accounted for by each common eigenvector.

Separate eigenvectors						
	e_{i1}	e_{i2}	e_{i3}	e_{i4}	e_{i5}	e_{i6}
South Africa						
BWGT	-0.16	0.02	0.02	-0.00	0.07	0.98
AP1	-0.12	-0.78	0.02	-0.61	0.08	-0.01
AP5	-0.10	-0.60	-0.03	0.77	-0.19	0.02
GESTAGE	-0.72	0.12	-0.67	-0.04	-0.03	-0.10
BHEADCIR	-0.65	0.10	0.74	0.02	-0.05	-0.12
ATEMP	-0.05	-0.05	0.00	0.20	0.97	-0.08
Namibia						
BWGT	-0.15	0.07	0.11	-0.03	0.09	0.98
AP1	-0.17	-0.83	0.23	-0.48	0.06	-0.02
AP5	-0.13	-0.48	-0.03	0.86	-0.11	0.05
GESTAGE	-0.68	0.01	-0.71	-0.10	0.13	-0.04
BHEADCIR	-0.68	0.28	0.63	0.06	-0.13	-0.18
ATEMP	0.01	0.02	0.15	0.15	0.97	-0.11
Common eigenvectors						
	b_1	b_2	b_3	b_4	b_5	b_6
BWGT	0.16	-0.02	0.03	-0.00	0.08	0.98
AP1	0.12	0.78	0.03	-0.60	0.08	-0.01
AP5	0.10	0.60	-0.03	0.77	-0.20	0.02
GESTAGE	0.72	-0.11	-0.68	-0.04	-0.03	-0.09
BHEADCIR	0.66	-0.11	0.73	0.02	-0.05	-0.12
ATEMP	0.05	0.05	0.01	0.21	0.97	-0.08
South Africa						
l_{1j}	21.86	5.14	2.16	0.62	0.49	0.14
Variance	71.9%	16.9%	7.1%	2.0%	1.6%	0.5%
Cum. variance	71.9%	88.8%	95.9%	97.9%	99.5%	100.0%
Namibia						
l_{2j}	30.70	4.65	2.18	0.82	0.61	0.14
Variance	78.5%	11.9%	5.6%	2.1%	1.6%	0.3%
Cum. variance	78.5%	90.4%	96.0%	98.1%	99.7%	100.0%

and the similarity score of the two sets of eigenvectors is

$$\text{tr} [\text{abs}(\mathbf{E}'_1 \mathbf{E}_2)] = 5.879.$$

Clearly the directions of variation for the two regional groups are similar, as the similarity score is very close to the maximum possible value of $p = 6$. It also appears that the relative importance of the sources of variation are the same for the two groups, as the percentages of variance accounted for by each of the common eigenvectors decreases monotonically in both groups.

95% Confidence intervals were calculated from the percentiles of bootstrap distributions (1000 replications) of the common eigenvectors and the eigenvalues. The bootstrap confidence intervals are reported in Table 3.8.

Estimated standard errors of the eigenvector loadings and the eigenvalues of the regional groups, under the CPC model, are shown in Table 3.9. The parametric estimates were obtained using (3.35) and (3.54), and the bootstrap estimates were obtained using (2.30). As in the case of the delivery mode groups, the parametric standard errors are in general smaller than the bootstrap standard errors.

Lastly, parametric and bootstrap 95% confidence intervals for the eigenvalues of the covariance matrices of *South Africa* and *Namibia* are given in Table 3.10. In general the bootstrap confidence intervals are slightly wider than the parametric confidence intervals.

Table 3.8: 95% bootstrap confidence intervals for the estimated common eigenvectors, eigenvalues, percentage variance and cumulative percentage variance accounted for by each of the common eigenvectors in the regional groups of the VON 2009 cohort. The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

Common eigenvectors			\mathbf{b}_1	\mathbf{b}_2	\mathbf{b}_3	\mathbf{b}_4	\mathbf{b}_5	\mathbf{b}_6
BWGT	LL		0.152	-0.033	0.010	-0.037	0.038	0.981
	UL		0.158	-0.015	0.046	0.022	0.103	0.986
AP1	LL		0.104	0.769	-0.013	-0.621	-0.062	-0.022
	UL		0.144	0.795	0.066	-0.519	0.233	0.003
AP5	LL		0.085	0.584	-0.064	0.610	-0.389	0.000
	UL		0.120	0.615	0.009	0.795	-0.002	0.034
GESTAGE	LL		0.710	-0.151	-0.688	-0.055	-0.043	-0.106
	UL		0.729	-0.076	-0.664	-0.010	-0.006	-0.084
BHEADCIR	LL		0.644	-0.151	0.723	-0.007	-0.066	-0.137
	UL		0.667	-0.075	0.744	0.047	-0.026	-0.110
ATEMP	LL		0.043	0.031	-0.015	-0.036	0.791	-0.110
	UL		0.058	0.061	0.032	0.450	0.994	-0.052
Eigenvalues and percentages under the CPC model								
South Africa								
l_{1j}	LL		20.5	4.8	2.0	0.5	0.5	0.1
	UL		23.2	5.5	2.4	0.7	0.6	0.2
Variance	LL		70.2%	15.6%	6.4%	1.7%	1.5%	0.4%
	UL		73.4%	18.1%	7.9%	2.3%	1.8%	0.5%
Cum. variance	LL		70.2%	87.9%	95.6%	97.7%	99.5%	100.0%
	UL		73.4%	89.6%	96.2%	98.1%	99.6%	100.0%
Namibia								
l_{2j}	LL		22.6	3.2	1.6	0.6	0.4	0.1
	UL		37.8	6.2	2.9	1.1	0.8	0.2
Variance	LL		73.3%	8.7%	3.9%	1.4%	1.1%	0.2%
	UL		82.6%	15.2%	8.0%	3.0%	2.1%	0.5%
Cum. variance	LL		73.3%	87.4%	94.9%	97.5%	99.5%	100.0%
	UL		82.6%	92.5%	97.0%	98.6%	99.8%	100.0%

Table 3.9: Parametric and bootstrap standard errors of the eigenvector loadings and the eigenvalues of the regional groups (VON 2009 cohort) under the CPC model. These values were calculated using the eigenvectors and eigenvalues obtained from the FG algorithm.

Eigenvector loadings: Parametric standard errors						
	CPC1	CPC2	CPC3	CPC4	CPC5	CPC6
BWGT	0.001	0.004	0.005	0.012	0.014	0.001
AP1	0.009	0.005	0.018	0.009	0.047	0.007
AP5	0.007	0.006	0.016	0.016	0.059	0.009
GESTAGE	0.004	0.016	0.005	0.010	0.008	0.004
BHEADCIR	0.005	0.017	0.005	0.011	0.009	0.004
ATEMP	0.003	0.006	0.011	0.075	0.016	0.014
Eigenvalues: Parametric standard errors						
<i>South Africa</i>						
l_{1j}	0.122	0.059	0.039	0.021	0.018	0.010
<i>Namibia</i>						
l_{2j}	0.718	0.280	0.191	0.118	0.101	0.048
Eigenvector loadings: Bootstrap standard errors						
	CPC1	CPC2	CPC3	CPC4	CPC5	CPC6
BWGT	0.002	0.005	0.009	0.014	0.016	0.001
AP1	0.010	0.007	0.019	0.014	0.064	0.007
AP5	0.008	0.008	0.018	0.024	0.082	0.009
GESTAGE	0.005	0.018	0.005	0.010	0.010	0.005
BHEADCIR	0.005	0.018	0.005	0.013	0.010	0.006
ATEMP	0.004	0.007	0.012	0.101	0.025	0.015
Eigenvalues: Bootstrap standard errors						
<i>South Africa</i>						
l_{1j}	0.640	0.204	0.113	0.036	0.022	0.006
<i>Namibia</i>						
l_{2j}	3.786	0.730	0.316	0.152	0.097	0.024

Table 3.10: 95% Parametric and bootstrap confidence intervals for the eigenvalues of the covariance matrices of the regional groups (VON 2009 cohort) under the CPC model. The “LL” and “UL” indicate the lower and upper confidence limits, respectively.

Eigenvalues: Parametric 95% C.I.							
<i>South Africa</i>							
l_{1j}	LL	20.8	4.9	2.1	0.6	0.5	0.1
	UL	23.0	5.4	2.3	0.6	0.5	0.2
<i>Namibia</i>							
l_{2j}	LL	24.5	3.7	1.7	0.7	0.5	0.1
	UL	41.2	6.2	2.9	1.1	0.8	0.2
Eigenvalues: Bootstrap 95% C.I.							
<i>South Africa</i>							
l_{1j}	LL	20.5	4.8	2.0	0.5	0.5	0.1
	UL	23.2	5.5	2.4	0.7	0.6	0.2
<i>Namibia</i>							
l_{2j}	LL	22.6	3.2	1.6	0.6	0.4	0.1
	UL	37.8	6.2	2.9	1.1	0.8	0.2

Chapter 4

Identification of common eigenvectors

4.1 Introduction

As noted by Box and Draper (1987), “all models are wrong, but some are useful”. When modelling the covariance matrices of several groups, the first and most important step is to determine which of the models in Flury’s hierarchy introduced in Chapter 3 fits the data (and study population) the best.

In keeping with the principle of parsimony, it is important to select the model with the fewest parameters to estimate (i.e. the highest level in Flury’s hierarchy, where equal covariance matrices is the “highest” possible level and unrelated covariance matrices is the “lowest” possible level) which still provides an adequate fit for the data. For the analysis of small samples, a reduction in the number of parameters to estimate can improve the precision of estimation at the expense of a negligible increase in bias. It would thus be preferable to assume that the population covariance matrices are equal or proportional, if one of these two models indeed provides a good fit for the observed data. Some tests for these first two levels in Flury’s hierarchy are discussed in Section 4.2.

If the CPC model is appropriate, it should provide more precise estimates of the covariance matrices than when assuming that the population covariance matrices are unrelated, especially for smaller samples (Airoldi and Flury, 1988). On the other hand, the CPC model should also provide less biased estimates of the covariance matrices than when incorrectly assuming that the population covariance matrices are equal.

Flury (1988) proposed two methods for the identification of the most ap-

82 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

propriate model in the covariance matrix hierarchy, both based on maximum likelihood theory which assumes multivariate normality in the populations. The multivariate normality assumption is untenable for many real data sets, including the subset of VON data studied in this dissertation.

The purpose of this chapter is to introduce two new non-parametric model selection methods, based on bootstrap distributions, for identifying the most appropriate CPC or partial CPC model in Flury's hierarchy for two groups. These methods have the advantage that they can be used for multivariate normal data as well as multivariate non-normal data. The performance of the proposed model selection methods is compared to that of the two known parametric methods of Flury (1988) and modified versions of two non-parametric methods proposed by Klingenberg (1996) and Klingenberg and McIntyre (1998).

All of the methods depend to a considerable extent on knowledge of which combinations of eigenvectors from the k groups are most likely to be common. Identification of the best model for the covariance structure of k groups will therefore generally involve two distinct steps: (1) Finding the combinations of eigenvectors from the k groups which are most likely to be common, and (2) applying the chosen statistic, criteria or test to determine the most appropriate model.

The first step will be discussed in Section 4.3. In Sections 4.3.1 and 4.3.2 the parametric methods proposed by Flury (1988) to identify common eigenvectors in several populations will be described, followed by modified versions of the two non-parametric solutions by Klingenberg (1996) and Klingenberg and McIntyre (1998) in Sections 4.3.3 and 4.3.4. Two new non-parametric alternatives and an ensemble test will be proposed in Sections 4.3.5 to 4.3.7. Other related techniques are briefly mentioned in Section 4.3.8.

The known and newly proposed methods were compared in a Monte Carlo simulation study, of which the results are presented in Section 4.4. This is followed in Section 4.5 by a comparison of the methods to identify the number of common eigenvectors in three well known data sets with distinct groups.

Lastly, in Section 4.6 the proposed methods to identify common eigenvectors are applied to the delivery mode and regional groupings in the VON 2009 cohort.

To improve readability, the eigenvectors of the sample covariance matrix of a group is referred to as the "eigenvectors of the group". The term "population eigenvectors" is used as shorthand to refer to the eigenvectors of the population covariance matrix. The term *rank order* of eigenvectors is used to refer to the order of the eigenvectors when sorted according to the associated eigenvalues per group. For example, if the common eigenvectors in two population covariance matrices have *opposite* rank orders, it means

that the common eigenvector associated with the largest eigenvalue of the one covariance matrix is associated with the smallest eigenvalue of the other covariance matrix.

At the time of writing, a shortened version of the work in this chapter has been accepted for publication in Pepler et al. (2014).

4.2 Tests for equality and proportionality of covariance matrices

To select the appropriate model in Flury's hierarchy, the first step should be to sequentially test for equality or proportionality of the k population covariance matrices. If both hypotheses are rejected, the question of how many (if any) of the eigenvectors are common to all k covariance matrices should be investigated.

The log-likelihood ratio test statistic (i.e. $-2 \ln(\text{likelihood ratio})$) for homogeneity of the covariance matrices (Level 1 in Table 3.1) versus k unrelated covariance matrices is

$$X^2_{\text{total}} = \sum_{i=1}^k n_i \ln \frac{\det(\mathbf{S}_p)}{\det(\mathbf{S}_i)}, \quad (4.1)$$

where \mathbf{S}_p is the pooled sample covariance matrix,

$$\mathbf{S}_p = \frac{\sum_{i=1}^k (n_i - 1) \mathbf{S}_i}{\sum_{i=1}^k (n_i - 1)}. \quad (4.2)$$

Flury (1988) has shown that the statistic in (4.1) may be decomposed into *partial chi-squared* statistics as

$$\begin{aligned} X^2_{\text{total}} = & X^2(\text{inequality of proportionality constants} \mid \text{proportionality}) \\ & + X^2(\text{deviation from proportionality} \mid \text{CPC}) \\ & + X^2(\text{non-equality of the last } p - q \text{ eigenvectors} \mid \text{CPC}(q)) \\ & + X^2(\text{non-equality of the first } q \text{ eigenvectors}). \end{aligned} \quad (4.3)$$

The last partial X^2 statistic in (4.3) may also be decomposed further as a hierarchy of partial CPC models from $\text{CPC}(p - 2)$ down to $\text{CPC}(1)$. The general hierarchy of partial chi-squared statistics with the associated degrees of freedom is given in Table 4.1.

84 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

Table 4.1: Decomposition of the X^2_{total} log-likelihood ratio statistic into partial X^2 statistics (Flury, 1988).

Higher model	Lower model	Degrees of freedom
Homogeneity	Proportionality	$k - 1$
Proportionality	CPC	$(p - 1)(k - 1)$
CPC	CPC(q)	$\frac{1}{2}(k - 1)(p - q)(p - q - 1)$
CPC(q)	Heterogeneity	$\frac{1}{2}(k - 1)(2pq - q^2 - q)$

The log-likelihood ratio test statistic for any higher model against any lower model in the hierarchy is given by

$$X^2(\text{higher}|\text{lower}) = \sum_{i=1}^k n_i \ln \frac{\det(\mathbf{S}_{i(\text{higher})})}{\det(\mathbf{S}_{i(\text{lower})})}, \quad (4.4)$$

where $\mathbf{S}_{i(\text{higher})}$ and $\mathbf{S}_{i(\text{lower})}$ refer to the sample estimators of the population covariance matrix for the i^{th} group under the higher and lower models in the hierarchy, respectively.

Flury (1988) cautioned against the use of these partial X^2 statistics for formal hypothesis testing, as they may not be independent of each other. He suggested dividing each X^2 statistic by its associated degrees of freedom and rather comparing the relative sizes of these values for the different models. The $\frac{X^2}{df}$ value closest to one indicates the most appropriate model for the data. Note that no X^2 statistic can be calculated for the unrelated covariance matrices model, which is a major disadvantage of this method.

The second method suggested (and preferred) by Flury is the comparison of Akaike Information Criterion (AIC) (Akaike, 1974) values for the different models. This method does not constitute a formal hypothesis test but is a model selection technique penalising the number of parameters in the model, usually leading to a parsimonious, good-fitting model.

Suppose there are r possible models in Flury's hierarchy which may fit the sample data, with the r^{th} model being that of unrelated covariance matrices. Let $m_1 < m_2 < \dots < m_r$ indicate the number of parameters to be estimated and $L_1 \leq L_2 \leq \dots \leq L_r$ the maxima of the likelihood functions for each of the r models. The AIC for model M , where $M = 1, \dots, r$, is given by

$$AIC(M) = -2 \ln \frac{L_M}{L_r} + 2(m_M - m_1). \quad (4.5)$$

Letting $\mathbf{S}_{i(M)}$ indicate the estimator of the covariance matrix of the i^{th} group under model M , the AIC can be calculated as

$$\begin{aligned} \text{AIC}(M) = \sum_{i=1}^k n_i \left\{ \text{tr}(\mathbf{S}_{i(M)}^{-1} \mathbf{S}_i) + \ln[\det(\mathbf{S}_{i(M)})] - p - \ln[\det(\mathbf{S}_i)] \right\} \\ + 2(m_M - m_1). \end{aligned} \quad (4.6)$$

After calculating the AIC values for all r models, the model with the minimum AIC value is considered to be the most appropriate for the data.

Estimation of the covariance matrices under the CPC and partial CPC models will be discussed in Chapter 5. The likelihood equations and an algorithm to estimate the covariance matrices under the proportional model are given in Section 4.2.2.

The Chi-square and AIC methods make use of maximum likelihood estimation for which the assumption of multivariate normality is necessary. This assumption is not valid for many real data sets.

4.2.1 Testing for equality

To determine whether the population covariances matrices of k groups are equal, the hypothesis

$$H_0 : \Sigma_1 = \Sigma_2 = \dots = \Sigma_k \quad (4.7)$$

should be tested. Box's M test (Box 1949, 1950) is a well known multivariate test for the hypothesis in (4.7). If we can assume that $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_k$ are the covariance matrices of independent samples from k multivariate normal populations, the test statistic

$$U = -2(1 - c_1) \ln M, \quad (4.8)$$

where

$$c_1 = \left[\sum_{i=1}^k \frac{1}{n_i - 1} - \frac{1}{\sum_{i=1}^k n_i - 1} \right] \left[\frac{2p^2 + 3p - 1}{6(p + 1)(k - 1)} \right] \quad (4.9)$$

and

$$M = \frac{\prod_{i=1}^k |\mathbf{S}_i|^{\frac{n_i - 1}{2}}}{|\mathbf{S}_p|^{\sum_{i=1}^k \frac{n_i - 1}{2}}}, \quad (4.10)$$

is asymptotically chi-squared with $\frac{1}{2}(k - 1)p(p + 1)$ degrees of freedom.

However, Box's M test is known to be sensitive to some forms of non-normality, particularly deviations in kurtosis (Rencher, 2002).

The log-likelihood ratio test statistic in (4.4) can also be used to test for equality of the k covariance matrices against the alternative of heterogeneity. The test statistic for equality will be as given in (4.1). Under the null hypothesis of equal covariance matrices, the test statistic in (4.1) is distributed asymptotically chi-squared with $(k - 1) \left[\frac{1}{2}p(p - 1) + p \right]$ degrees of freedom.

Like Box's M test, Flury's log-likelihood ratio test for equality of covariance matrices is also based on the multivariate normality assumption, limiting its usefulness for analysing non-normal data.

4.2.2 Testing for proportionality

The log-likelihood ratio test statistic in (4.4) may be modified to test for proportional covariance structures in k groups, against the alternative of heterogeneity. The test statistic is

$$X^2(\text{proportionality}|\text{heterogeneity}) = \sum_{i=1}^k n_i \ln \frac{\det(\mathbf{S}_{i(\text{PROP})})}{\det(\mathbf{S}_i)}, \quad (4.11)$$

where $\mathbf{S}_{i(\text{PROP})}$ is the covariance matrix estimator for the i^{th} group under the assumption of proportional covariance structures. Under the null hypothesis, the test statistic in (4.11) is distributed asymptotically chi-squared with $(k - 1) \left[\frac{1}{2}p(p - 1) + p \right] - k + 1$ degrees of freedom.

To obtain the maximum likelihood estimates for $\mathbf{S}_{i(\text{PROP})}$, let $n = n_1 + \dots + n_k$ and indicate proportionality constants for the covariance matrices as $\rho_i, i = 1, \dots, k$ with the constraint $\rho_1 = 1$. The likelihood equations for the proportional covariance matrix model are (Flury, 1988)

$$\rho_i = \frac{1}{p} \sum_{j=1}^p \frac{\beta_j' \mathbf{S}_i \beta_j}{\lambda_j}, \quad i = 2, \dots, k, \quad (4.12)$$

$$\lambda_j = \frac{1}{n} \sum_{i=1}^k \frac{n_i}{\rho_i} \beta_j' \mathbf{S}_i \beta_j, \quad j = 1, \dots, p, \quad (4.13)$$

and

$$\left(\frac{1}{\lambda_j} - \frac{1}{\lambda_h} \right) \beta_h \left(\sum_{i=1}^k \frac{n_i}{\rho_i} \mathbf{S}_i \right) \beta_j = 0, \quad j \neq h. \quad (4.14)$$

To solve likelihood equations (4.12), (4.13) and (4.14) under the usual eigenvector orthogonality constraints,

$$\beta'_j \beta_h = \begin{cases} 0 & \text{if } j \neq h \\ 1 & \text{if } j = h, \end{cases} \quad (4.15)$$

Flury (1988) gave the following iterative algorithm: Let $r_i = \frac{n_i}{n}$ and initialise the vector of proportionality constants, $\boldsymbol{\rho}^{(0)} = (\rho_1, \dots, \rho_k)'$, to the values $\rho_i = 1, i = 1, \dots, k$. At the t^{th} iteration of the algorithm:

- Step 1: Put

$$\begin{aligned} \mathbf{S} &\leftarrow \sum_{i=1}^k \frac{r_i \mathbf{S}_i}{\rho_i} \\ \mathbf{b}_1, \dots, \mathbf{b}_p &\leftarrow \text{eigenvectors of } \mathbf{S} \\ a_{ij} &\leftarrow \mathbf{b}'_j \mathbf{S}_i \mathbf{b}_j, \quad i = 1, \dots, k, \quad j = 1, \dots, p. \end{aligned}$$

- Step 2: Put

$$l_j \leftarrow \sum_{i=1}^k \frac{r_i a_{ij}}{\rho_i}, \quad j = 1, \dots, p.$$

- Step 3: Put

$$\rho_i \leftarrow \frac{1}{p} \sum_{j=1}^p \frac{a_{ij}}{l_j}, \quad i = 2, \dots, k.$$

- Step 4: Put $\boldsymbol{\rho}^{(t)} \leftarrow (1, \rho_2, \dots, \rho_k)'$. Repeat steps 1 to 4 until $\|\boldsymbol{\rho}^{(t)} - \boldsymbol{\rho}^{(t-1)}\| < \epsilon$ for a predetermined small positive value, ϵ , and a predetermined vector norm.

For some of the examples in Flury (1988), the convergence criterion was chosen as $\epsilon = 10^{-4}$, and the absolute value of the largest element was used as the vector norm. The values of $\boldsymbol{\rho}$, l_j and \mathbf{b}_j after the last iteration of the algorithm are the maximum likelihood estimates to solve equations (4.12), (4.13) and (4.14).

Finally, the estimates of the covariance matrices under the proportional model is obtained with

$$\mathbf{S}_{i(\text{PROP})} = \rho_i \mathbf{B}' \mathbf{L} \mathbf{B}, \quad (4.16)$$

where $\mathbf{B} = [\mathbf{b}_1 \dots \mathbf{b}_p]$ and $\mathbf{L} = \text{diag}(l_1, \dots, l_p)$ contain the maximum likelihood estimates of the common eigenvectors and eigenvalues under the proportional model, respectively.

4.3 Methods for the identification of common eigenvectors

If the possibilities that the covariance matrices may be equal or proportional have been ruled out, the next step is to investigate whether any of the eigenvectors of the population covariance matrices are equal.

It is important to determine which of the eigenvector combinations from k covariance matrices are potentially common, because the rank orders of common eigenvectors may differ across the covariance matrices. For k groups with measurements on p variables, there are p^k possible eigenvector combinations. Even for very moderate values of p and k , the number of eigenvector combinations to consider can be large, leading to multiple testing concerns in model selection methods based on hypothesis testing.

One way to avoid testing all p^k eigenvector combinations is to inspect the angles between the eigenvectors per combination, and picking only the p most likely common eigenvector combinations to test for equality. Suppose standard PCA is performed on two groups separately, yielding two sets of eigenvectors and eigenvalues. If two normalised population eigenvectors, say $\boldsymbol{\eta}_{11}$ and $\boldsymbol{\eta}_{21}$, are truly common, the sample estimate of the angle between them, $\hat{\theta}$, should be small (or close to 180° , depending on the signs of the eigenvector loadings) and the value of $\mathbf{e}'_{11}\mathbf{e}_{21}$ should be close to one in absolute value. Due to sample variation the angles between sample eigenvectors which are common in the populations will usually not be exactly equal to zero, but one can expect that they should be consistently small over a large number of samples. As shown in (3.13), the inner product of two normalised p -dimensional vectors is equal to the cosine of the angle between them (Krzanowski, 1979). This fact (illustrated in Figure 4.1) can be used to find the most likely combinations of common eigenvectors from all p^k possible combinations of the $k \times p$ eigenvectors (one eigenvector from each group).

Suppose \mathbf{a} and \mathbf{b} are two stochastic vectors in the p -variate real space (\mathbb{R}^p). The Pearson correlation between \mathbf{a} and \mathbf{b} is defined as

$$\text{Cor}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}'\mathbf{b}}{\sqrt{(\mathbf{a}'\mathbf{a})(\mathbf{b}'\mathbf{b})}}. \quad (4.17)$$

Therefore, if \mathbf{a} and \mathbf{b} are normalised, (4.17) can be simplified to $\text{Cor}(\mathbf{a}, \mathbf{b}) = \mathbf{a}'\mathbf{b}$. The inner product of two normalised vectors \mathbf{a} and \mathbf{b} (for example, two eigenvectors) will from here on be referred to as the *vector correlation* between \mathbf{a} and \mathbf{b} . For computational simplicity, the absolute values of these vector correlations can be used.

Inspection of the vector correlations between all pairwise combinations

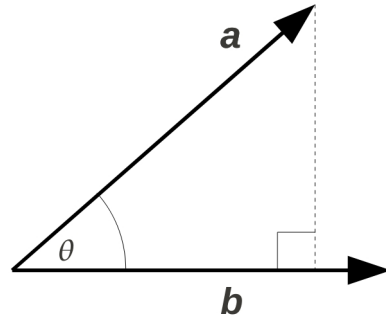


Figure 4.1: *Vector correlation* (inner product) between two vectors. The cosine of the angle θ between the vectors in p -dimensional space is equal to the inner product of the two normalised vectors.

of the eigenvectors from k covariance matrices should provide valuable information about which of the eigenvector pairs are most likely to be common. Such information may be presented in the form of a scree plot as shown in Figure 4.2. The assumption that the correspondingly ranked eigenvectors from each group would be most likely to be common is not always correct, as shown in this simulated example for $k = 2$ groups with common eigenvectors in the population covariance matrices. A clear break between the vector correlations of the common and non-common eigenvector pairs can be seen. This type of pattern tends to become clearer with an increase in the sample sizes. For more than two groups, a summary measure such as the arithmetic mean of the pairwise vector correlations per eigenvector combination can be used to identify the p most likely common eigenvector combinations.

Model selection can proceed after the p most likely common eigenvector combinations have been identified. This may lead to better fitting covariance matrix models than when simply assuming the rank orders of the common eigenvectors to be the same for both groups, and that the first eigenvectors (associated with the largest eigenvalues) per group are the most likely to be common.

However, an investigation of the possibility that the eigenvectors from several groups are common should ideally take the variability of the loadings of the sample eigenvectors into account. In some cases large vector correlation between sample eigenvectors can be due to sampling variability rather than actual commonness in the populations, especially for the last few eigenvectors of which the standard errors of the loadings are often large.

Formal hypothesis tests for the CPC and partial CPC models have been proposed, of which some are robust such as those by Schott (1991a), Boente et al. (2009) and Hallin et al. (2010). However, the majority of the proposed

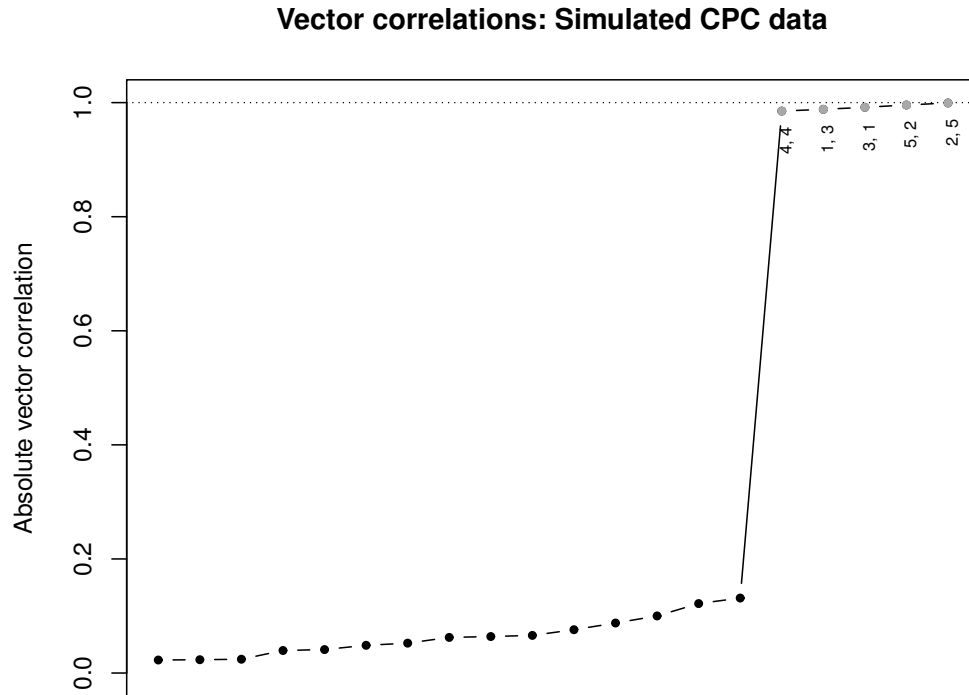


Figure 4.2: Scree plot of the largest eigenvector correlations of simulated CPC data for two groups ($n_1 = n_2 = 200$) with $p = 5$ variables and five common eigenvectors. The vertical numbering under the dots indicate the rank orders of the eigenvectors, for example “j, h” indicates that it is the vector correlation of the j^{th} eigenvector of the first group with the h^{th} eigenvector of the second group.

robust tests involve replacing the maximum likelihood estimators of the covariance matrices with robust versions, instead of providing a test which is intrinsically free from parametric assumptions.

Rublík (2009) also proposed a test, which does not depend on the multivariate normality assumption, for the hypothesis of partial common principal components. No algorithmic computer implementation or application of this method could be found to date. As the proposed test was designed exclusively for the partial CPC hypothesis, and cannot be used to test for any of the other levels in Flury's hierarchy of covariance matrices, it was excluded from the study presented in this chapter.

In the sections that follow, Flury's (1988) parametric methods based on maximum likelihood theory will be given, followed by modifications to the non-parametric methods proposed by Klingenberg (1996) and Klingenberg and McIntyre (1998) and two new non-parametric alternatives. An ensemble method is also proposed, combining the outcomes from different model selection techniques.

4.3.1 Chi-square test

The basic form of the log-likelihood ratio statistic in (4.4) may be modified to test for the CPC and CPC(q) models, respectively, against the unrelated covariance matrices model. For testing the fit of the CPC model against the alternative of unrelated covariance matrices, the log-likelihood ratio test statistic is

$$X^2(\text{CPC}|\text{heterogeneity}) = \sum_{i=1}^k n_i \ln \frac{\det(\mathbf{S}_{i(\text{CPC})})}{\det(\mathbf{S}_i)}, \quad (4.18)$$

where $\mathbf{S}_{i(\text{CPC})}$ is the covariance matrix estimator for the i^{th} group under the assumption of common eigenvectors. Estimation of the $\mathbf{S}_{i(\text{CPC})}$ matrices is postponed until Chapter 5. Under the null hypothesis of p common eigenvectors and with the assumption that the populations are multivariate normal, the test statistic in (4.18) is distributed asymptotically chi-squared with $\frac{1}{2}p(p-1)(k-1)$ degrees of freedom.

For testing the partial CPC hypothesis that $q < p$ of the eigenvectors in k covariance matrices are common, the log-likelihood ratio test statistic is

$$X^2(\text{CPC}(q)|\text{heterogeneity}) = \sum_{i=1}^k n_i \ln \frac{\det(\mathbf{S}_{i(\text{CPC}(q))})}{\det(\mathbf{S}_i)}, \quad (4.19)$$

92 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

where $\mathbf{S}_{i(CPC(q))}$ is the covariance matrix estimator for the i^{th} group under the assumption of q common eigenvectors. Estimation of the $\mathbf{S}_{i(CPC(q))}$ matrices is also discussed in Chapter 5. With the normality assumption and under the hypothesis of q common eigenvectors, test statistic (4.19) is distributed asymptotically chi-squared with $\frac{1}{2}q(k-1)(2p-q-1)$ degrees of freedom.

Schott (1991b) proposed a test to determine whether a specific eigenvector of a correlation (or standardised covariance) matrix is equal to a predetermined vector, i.e.

$$H_0 : \boldsymbol{\eta}_j = \boldsymbol{\eta}_j^0, \quad j = 1, \dots, p, \quad (4.20)$$

and proved that the test statistic is asymptotically chi-squared distributed under the usual multivariate normality assumption. A different approach in testing for commonness of the eigenvectors of $k \geq 2$ groups may be to compare the common eigenvector candidates pairwise using this test. However, an adjustment will be needed for multiple testing to control the family-wise Type I error rate.

4.3.2 Akaike Information Criterion (AIC)

The AIC statistics as defined in (4.5) may be used to determine whether the CPC model or a partial CPC model will provide the best fit for the data from k populations. After using the likelihood ratios to calculate the AIC values for all possible models in Flury's hierarchy, the model with the minimum AIC value is considered to be the most appropriate for the data.

The number of common eigenvectors may be inferred from the selected model. If the CPC model has the minimum AIC value, there are p common eigenvectors in the populations. The $CPC(q)$ model will indicate q common eigenvectors, while the unrelated covariance matrices model will imply that none of the population eigenvectors are common.

4.3.3 Bootstrap hypothesis test (BootTest)

Klingenberg and Froese (1991) referred to Flury's work (1988) on testing for common principal components in several groups, and suggested that bootstrap sampling (Efron and Tibshirani, 1993) can be helpful when the assumption of multivariate normality is doubtful. They used bootstrap replications to calculate standard errors for the estimated loadings of the eigenvectors associated with the largest eigenvalue from each covariance matrix, as well as standard errors of the percentage of variance accounted for by the first principal components.

Klingenberg (1996) and Klingenberg and McIntyre (1998) developed this idea further, proposing two tests for the hypothesis of common eigenvectors in two groups.

The first method tests the hypothesis

$$H_0 : \boldsymbol{\eta}_{1j} = \boldsymbol{\eta}_{2h}, \quad j, h = 1, \dots, p, \quad (4.21)$$

against the alternative that the two population eigenvectors are not common. To perform the test, the angle between the sample estimates of the potentially common eigenvectors is compared to the distribution of angles between bootstrap replications of the same eigenvectors under the null hypothesis of commonness.

To find the bootstrap distribution under the null hypothesis, the data from the two groups are rotated separately, each with its own eigenvector matrix \mathbf{E}_i , and thereafter both groups are rotated further by multiplication with the estimated set of common eigenvectors. For the implementation of BootTest in this study, the common eigenvectors were estimated using the Flury-Gautschi algorithm (Flury and Gautschi, 1986). The twice rotated data matrix for the i^{th} group is given by

$$\mathbf{X}_i^* = \mathbf{X}_i \mathbf{E}_i \mathbf{B}', \quad i = 1, 2, \quad (4.22)$$

where $\mathbf{X}_i : n_i \times p$ is the original data matrix for this group and \mathbf{B} is the common eigenvector matrix.

Bootstrap samples are taken from the \mathbf{X}_i^* matrices to calculate bootstrap replications of the angle between \mathbf{e}_{1j} and \mathbf{e}_{2h} under hypothesis (4.21). To improve computational efficiency, the present study compared the absolute vector correlations between eigenvectors, instead of the angles. For a nominal significance level of α , if the sample eigenvector correlation $\mathbf{e}_{1j}' \mathbf{e}_{2h}$ exceeds the $100(1 - \alpha)^{th}$ percentile of its bootstrap distribution under the null hypothesis, the null hypothesis is rejected for the pair of eigenvectors under consideration and it is concluded that they are not common.

To control the overall Type I error risk associated with the testing of hypothesis (4.21) for each of the potentially common eigenvectors pairs, a Bonferroni-type adjustment to the nominal significance level per test is made.

4.3.4 Random Vector Correlation (RVC)

The second method proposed by Klingenberg and McIntyre (1998) tests the non-specific null hypothesis,

$$H_0 : \boldsymbol{\eta}_{1j} \neq \boldsymbol{\eta}_{2h}, \quad j, h = 1, \dots, p, \quad (4.23)$$

by comparing the angle between the sample eigenvectors to the distribution of angles between 100000 pairs of randomly generated p -dimensional vectors on the unit sphere. A computationally more efficient way is to compare the vector correlations instead of the angles between the vectors, similar to the BootTest method.

An example of the distribution of the absolute vector correlations in $p = 5$ dimensions under hypothesis (4.23) is shown in Figure 4.3.

If the absolute vector correlation between the sample eigenvector pair exceeds the $100(1 - \alpha)^{th}$ percentile of the random absolute vector correlation distribution, the hypothesis of non-commonness is rejected and the two population eigenvectors are considered to be common.

Rather than generating the random vector correlation distribution anew for each instance of the RVC test, the sample vector correlations may be compared to quantiles from a precalculated distribution, further improving computational efficiency.

As for the BootTest method, the inflation of the overall Type I error risk is controlled by employing a Bonferroni-type adjustment to the nominal significance level for each eigenvector test.

4.3.5 Bootstrap Vector correlation Distribution (BVD)

We propose a new non-parametric method for the identification of the most appropriate CPC (or partial CPC) model in Flury's hierarchy, based on bootstrap distributions of eigenvector correlations.

Not all common eigenvectors can be identified easily from an inspection of the sample vector correlations as those of the simulated data used in Figure 4.2. The problem is therefore to determine how close to one the absolute vector correlation should be before the corresponding population eigenvectors can be considered common.

Taking bootstrap samples from the original data, bootstrap distributions of the vector correlations between all eigenvector pairs can be obtained. Figure 4.4 (simulated CPC data) shows that the peaks of bootstrap distributions of the common eigenvector pairs are all close to one, while the distributions of the non-common eigenvector pairs are either more or less uniform or has peaks closer to zero.

The following simple procedure can be used to decide whether any two specific eigenvectors are common: Letting D be the median minus the 2.5th percentile of the vector correlation bootstrap distribution (see Figure 4.5), the associated eigenvectors are considered to be common if

- (a) the median > 0.71 , and

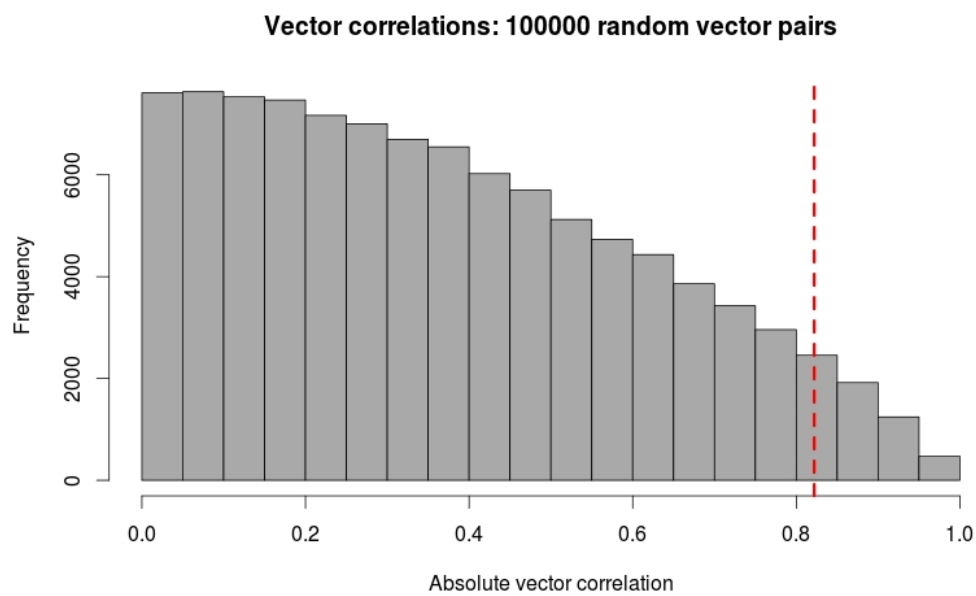


Figure 4.3: Distribution of the absolute vector correlations for 100000 pairs of unit length vectors in $p = 5$ dimensions under the null hypothesis of non-commonness, as used in the *Random Vector Correlation (RVC)* method modified from Klingenberg and McIntyre (1998). The dashed vertical line indicates the 95th percentile of the distribution.

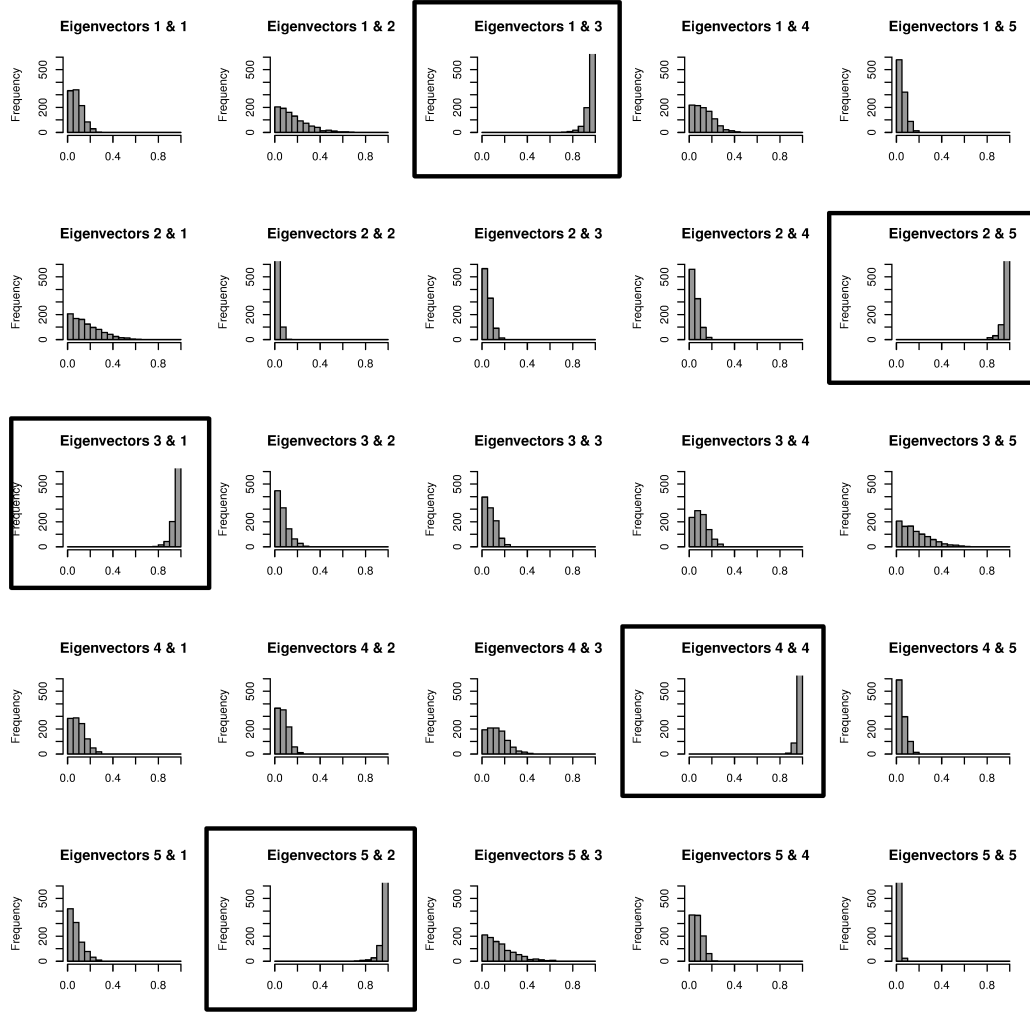


Figure 4.4: Bootstrap distributions of the absolute vector correlations of all pairwise eigenvector combinations for simulated CPC data from $k = 2$ groups ($n_1 = n_2 = 200$) with $p = 5$ variables and $q = 5$ common eigenvectors. The distributions of the truly common eigenvector pairs are indicated by the rectangular frames.

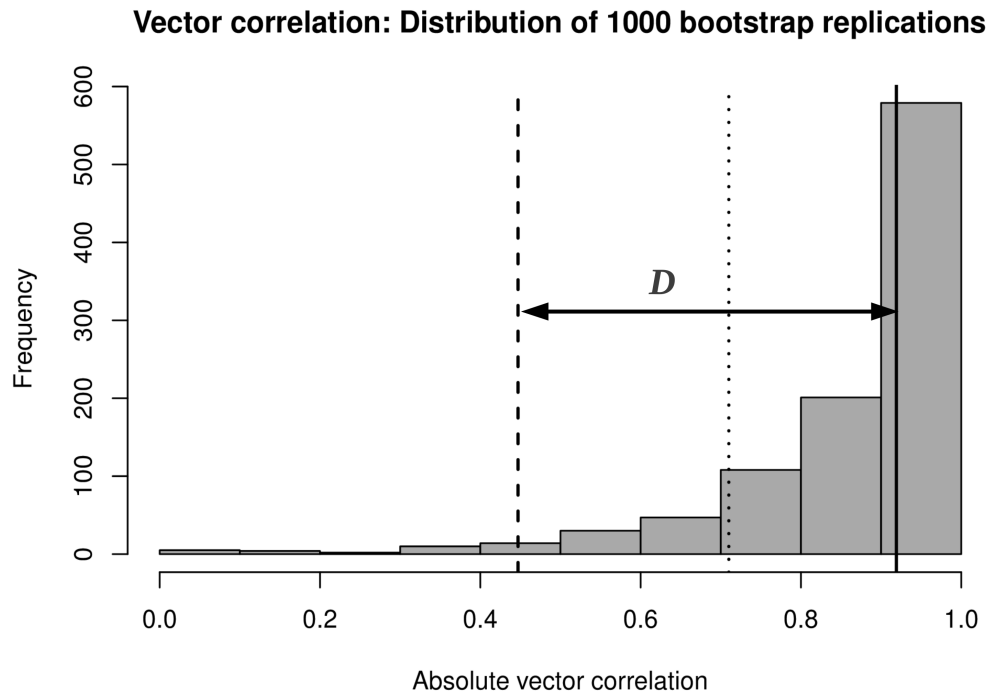


Figure 4.5: Calculation of the value of D for the BVD method from the bootstrap distribution of the absolute value vector correlation between two eigenvectors. The dashed and solid lines indicate the 2.5th percentile and median of the bootstrap distribution, respectively, and the dotted line indicates the value 0.71 (i.e. the vector correlation corresponding to a 45° angle between the eigenvectors under consideration).

(b) the median $+ D \geq 1$.

The first condition ensures that, for a specific eigenvector combination, the angles between the two eigenvectors should be smaller than $\cos^{-1}(0.71) = 45^\circ$ in at least 50% of the bootstrap samples. This implies that there is more evidence for commonness of the two eigenvectors than evidence for orthogonality.

The second condition is an attempt to account for the sampling variation of the eigenvector estimates. If the directions of the eigenvectors can be estimated with small errors (as in the case of eigenvectors associated with the larger eigenvalues), the variability of the bootstrap vector correlation estimators will be smaller. For precisely estimated non-common eigenvectors with an angle of less than 45° between them, the second condition will ensure that they are not considered to be common.

As this method does not rely on any assumptions about the population distributions from which the data originated, it may be used to analyse both multivariate normal and non-normal data.

4.3.6 Bootstrap Confidence Regions (BCR)

Our second proposed non-parametric method for the selection of the most appropriate CPC model from Flury's hierarchy makes use of bootstrap confidence regions. Bootstrap replications of a sample eigenvector pair are used to estimate a p -dimensional confidence region for each of the eigenvectors, after which the confidence regions of potentially common eigenvectors are compared to see whether they overlap. An overlap is considered an indication that the two population eigenvectors are common (see Figure 4.6).

For a nominal level of $\alpha \in (0; 1)$, the algorithm for estimating the $100(1 - \alpha)\%$ bootstrap confidence region limits for two groups in p dimensions and checking for overlap is as follows:

1. Take r bootstrap samples from the two original data matrices.
2. Calculate r bootstrap replications of the eigenvectors of the two groups.
3. For each potentially common eigenvector combination:
 - (a) Per group, calculate the absolute vector correlations between the original sample eigenvector and each of its bootstrap replications.
 - (b) Per group, discard the $100(\alpha)\%$ bootstrap replications showing the smallest vector correlations with the original sample eigenvector.

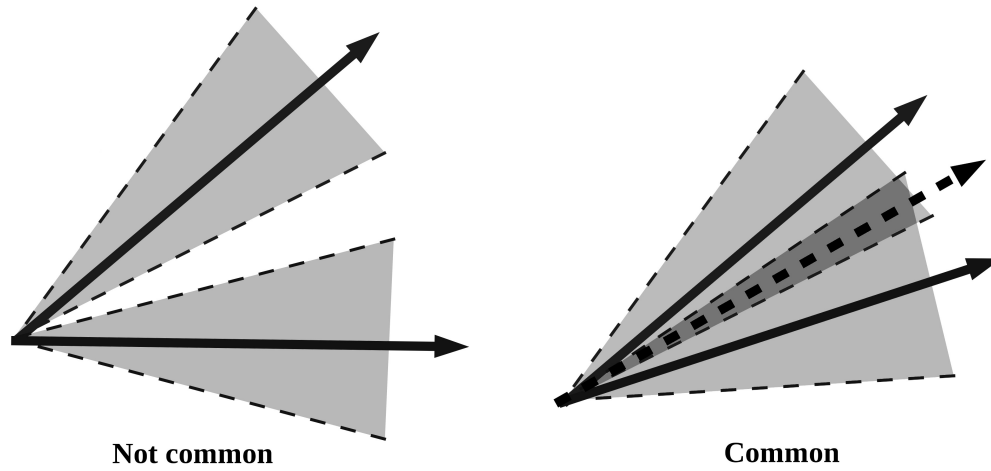


Figure 4.6: Illustration of the *Bootstrap Confidence Region (BCR)* method to identify common eigenvectors in $k = 2$ groups in two dimensions. The shaded areas indicate the 95% confidence regions around each eigenvector, and an overlap between the confidence regions is considered to indicate commonness of the population vectors. The dashed arrow on the right indicates a possibly common population eigenvector.

- (c) From the remaining bootstrap replications of the first group, find the vector with the largest vector correlation with the original sample eigenvector of the second group. Indicate this vector correlation with $r_{\max}^{(1)}$.
- (d) From the remaining bootstrap replications of the second group, find the vector with the smallest vector correlation with the original sample eigenvector of the second group. Indicate this vector correlation with $r_{\min}^{(2)}$.
- (e) If $r_{\max}^{(1)} \geq r_{\min}^{(2)}$, the two eigenvector confidence regions overlap and the associated population eigenvectors are considered to be common.

The confidence region for each eigenvector is thus estimated by discarding the $100(\alpha)\%$ bootstrap replications which lie “furthest away” from the sample eigenvector in the p -variate space. Of the remaining bootstrap replications, those which lie the furthest away from the sample eigenvector (in all directions in the p -variate space) define the boundaries of the $100(1 - \alpha)\%$ bootstrap confidence region.

The BCR method takes the sampling error of the eigenvector loadings into account when judging the evidence in favour of commonness. It should

therefore tend to indicate commonness less readily for the eigenvectors associated with the largest eigenvalues in each group than for the eigenvectors associated with the smallest eigenvalues, as the loadings of the eigenvectors associated with the largest eigenvalues will generally have smaller estimation errors.

Once the p most likely eigenvector pairs from the two groups have been inspected as outlined above, the BCR method selects the model from Flury's hierarchy with the appropriate number of common eigenvectors.

4.3.7 Ensemble method

Following the suggestion of Walsh and Lynch (2013) on using an ensemble of tests to identify similarities in the covariance structures of several populations, an ensemble method was constructed from the model selection methods described in Sections 4.3.2 to 4.3.6. The Chi-square test method was excluded because of its inability to select the unrelated covariance matrices model, even if appropriate.

For each of the potentially common eigenvector pairs from the two groups, the majority vote on commonness from the AIC, RVC, BootTest, BVD and BCR methods are determined. If at least three of the five methods indicate that the eigenvector under consideration is common in the two population covariance matrices, the Ensemble method will indicate it as common.

Once the p most likely eigenvector pairs from the two groups have been evaluated for commonness in this way, the Ensemble method selects the model from Flury's hierarchy with the appropriate number of common eigenvectors.

4.3.8 Other methods

In one of the first attempts to compare the principal components of several groups, Krzanowski (1979) inspected the angles between the eigenvectors from different groups, rather than fitting a common eigenvector model and testing the model fit.

Krzanowski (1984) proposed that the common eigenvector hypothesis may be assessed informally by comparing the eigenvectors of the pooled covariance matrix, \mathbf{S}_p , with the eigenvectors of

$$\mathbf{T} = \sum_{i=1}^k \mathbf{S}_i. \quad (4.24)$$

A high degree of agreement between the two sets of eigenvectors would

show that the common eigenvector hypothesis may be reasonable. However, if the sample sizes of all k groups are equal, $\mathbf{S}_p = \frac{1}{k}\mathbf{T}$, and the eigenvector loadings will necessarily agree, making this method inappropriate in such a case.

Keramidas et al. (1987) proposed a graphical procedure for the comparison of eigenvectors from several groups, based on the maximisation of the sum of squared cosines between an estimated common eigenvector and each of the associated eigenvectors from k sample covariance matrices. Letting \mathbf{b}_j be the estimator of the j^{th} common eigenvector, the distances

$$\delta_{ij}^2 = \min [(\mathbf{e}_{ij} - \mathbf{b}_j)'(\mathbf{e}_{ij} - \mathbf{b}_j), (\mathbf{e}_{ij} + \mathbf{b}_j)'(\mathbf{e}_{ij} + \mathbf{b}_j)] \quad (4.25)$$

are calculated. Under the null hypothesis that the k samples originated from populations of which the covariance matrices have the same eigenvectors, the δ_{ij}^2 will have an approximate gamma distribution. Gamma Q-Q plots of the δ_{ij}^2 provides an informal method to assess the reasonableness of the null hypothesis. However, equality of the eigenvectors does not rule out the possibility of equality or proportionality of the population covariance matrices. Secondly, as noted by Jolliffe (2002), this method will in practice only be useful when the number of groups is large, making it easier to graphically detect deviations from the null hypothesised gamma distribution.

Boente and Orellana (2004) proposed a robust test of the equality of the covariance matrices against a proportional model by replacing the maximum likelihood covariance estimators in the log-likelihood ratio test statistics with robust versions. Boente et al. (2009) extended this idea to testing the CPC model against total heterogeneity, and also used the same method to construct a robust test for assessing the hypothesis of proportionality against the CPC model.

Walsh and Lynch (2013) found the power of Flury's methods based on the log-likelihood ratio statistics to be too weak in small samples, tending to indicate common eigenvector structures where there is none. On the other hand, in larger samples the hypothesis of common eigenvectors was rejected too easily. They concluded that the differences in the performance of different model selection methods are likely due to differences in power.

Hallin et al. (2010) modified Flury's likelihood ratio test statistics to arrive at a pseudo-Gaussian test which is robust to deviations from multivariate normality and homokurticity in the k populations. In their modification, the asymptotic chi-squared distribution of the test statistic under the CPC null hypothesis is preserved, and the test remains valid for heterokurtic elliptical populations. For multivariate normal populations, the modified likelihood ratio test is equivalent to the original test statistic proposed by Flury (1988).

However, the proposed pseudo-Gaussian test does not make any provision for testing partial CPC models, limiting its usefulness in the present study.

4.4 Simulation study

A Monte Carlo simulation study was carried out to compare the performance of the proposed non-parametric methods against the known methods. The first two levels in Flury's hierarchy (equality and proportionality) were not considered for the purpose of this simulation study, and the data were simulated in a way that these two scenarios were excluded.

Only $k = 2$ groups were used throughout, with equally sized samples of $n_i = 50, 100, 200, 500$ and 1000 observations on $p = 5, 10$ and 20 variables simulated from populations with multivariate normal, multivariate chi-squared (with two degrees of freedom) and multivariate t (with one degree of freedom) distributions. More details about the way in which the multivariate chi-squared distributions were simulated are given in Appendix A. In the five variables case, data were simulated for populations with CPC, CPC(3), CPC(1) and unrelated covariance structures. For the populations with ten variables, data were simulated for CPC, CPC(5), CPC(1) and unrelated covariance structures. In the twenty variables scenarios, data were simulated for CPC, CPC(10), CPC(2) and unrelated covariance structures.

For the two samples from each simulation run, Flury's Chi-square statistics and AIC measures were calculated for every model in the hierarchy before the best fitting model could be selected for each of these two methods. The same simulated samples were used to select a model using the BootTest, RVC, BVD, BCR and Ensemble methods, respectively. A total of $r = 1000$ bootstrap replications were used throughout for the BootTest, BVD and BCR methods. The null distributions for the RVC method were estimated using 100000 random vector pairs.

The overall Type I error risk for the multiple hypothesis tests performed by the BootTest and RVC methods were controlled at a level of 5%.

In all cases and for all of the methods, if $p - 1$ of the eigenvector pairs were identified as common, the p^{th} pair were also considered to be common, due to the orthogonality constraint on the common eigenvectors. The simulation results were adjusted accordingly.

The relative separation of the eigenvalues were varied, from the worst case scenario where there was only a 10% (first group) or 20% (second group) difference between subsequent eigenvalues (poor separation), to 40% or 50% (moderate separation) and 80% or 90% (good separation) differences between the subsequent eigenvalues per group. Two types of eigenvalue patterns were

considered: One where the rank orders of the common eigenvectors were identical, and one where the common eigenvectors had exactly opposite rank orders in the two groups.

A total of 100 simulation runs were performed for each of the 1080 different (number of variables, sample size, covariance structure, data distribution, eigenvalue separation, eigenvalue pattern) scenarios, giving a total of 108000 simulation runs. Because of the large number of simulations, fitting linear models to the simulation results and performing ANOVA to determine which of the effects (and interactions) are significant did not prove useful, as almost all of the higher order interactions were statistically significant. Considering the main effects only, the summarised results for $p = 5, 10$ and 20 are given in Tables 4.2, 4.3 and 4.4, respectively. The Chi-square percentages were calculated excluding the runs with unrelated covariance structures, as Chi-square statistics cannot be calculated for these.

With the exception of Flury's Chi-square, all of the methods performed best with multivariate normal data and worst with the data simulated from multivariate t distributions with one degree of freedom (Figure 4.7). Flury's Chi-square performed about equally poor with all three types of distributions. BVD performed the best overall, and also showed the greatest accuracy in each of the three distribution types separately. The non-parametric methods seem relatively robust in analysis of data from the simulated chi-squared distributions, but fared poorly with the t distributions.

For the $p = 5, 10$ and 20 scenarios considered, there were respectively 5, 10 and 20 possible models in Flury's hierarchy to select from, including the model of unrelated covariance matrices. A completely random selection from these models should thus yield long run accuracies of 20%, 10% and 5% for the $p = 5, 10$ and 20 scenarios, respectively. These benchmark accuracies are indicated with broken horizontal lines in Figure 4.8.

Increased separation of the eigenvalues per group translated into improved performance for the majority of the methods (Figure 4.8). Flury's AIC and Chi-square methods were the exceptions in this regard, showing slight deterioration in performance in the $p = 10, 20$ and $p = 5, 10$ cases, respectively. With poor separation between the eigenvalues, BCR fared worse than the benchmark (i.e. random selection from the models in Flury's hierarchy) for $p = 10, 20$ variables.

For populations with unrelated covariance matrices ($q = 0$), BVD and Flury's AIC showed the greatest accuracy (Figure 4.9). RVC also performed well for no or few common eigenvectors in the $p = 5, 10$ cases. Most methods showed a dip in performance for situations where about half of the eigenvectors were common to the two populations, but fared better again in the full CPC scenarios. By combining the strengths of the different methods,

Table 4.2: Simulation results for $p = 5$ variables. The given values are the percentage of simulation runs (combined over the other factors) for which each of the methods identified the correct model for the covariance matrices of the two groups. Standard errors are given in brackets.

	AIC	Chi ²	BootTest	RVC	BVD	BCR	Ensemble
Overall	39.9 (0.26)	31.0 (0.28)	37.8 (0.26)	45.1 (0.26)	46.7 (0.26)	39.6 (0.26)	46.2 (0.26)
Covariance structure							
Unrelated	67.8 (0.49)	–	17.1 (0.40)	87.6 (0.35)	90.2 (0.31)	20.0 (0.42)	59.6 (0.52)
CPC(1)	30.2 (0.48)	43.1 (0.52)	16.0 (0.39)	44.9 (0.52)	43.6 (0.52)	18.9 (0.41)	41.8 (0.52)
CPC(3)	19.3 (0.42)	18.4 (0.41)	28.4 (0.48)	22.7 (0.44)	20.4 (0.43)	27.1 (0.47)	27.6 (0.47)
CPC(5)	42.2 (0.52)	31.3 (0.49)	89.6 (0.32)	25.1 (0.46)	32.7 (0.49)	92.2 (0.28)	55.8 (0.52)
Eigenvalue separation							
Poor	27.2 (0.41)	23.1 (0.44)	26.7 (0.40)	27.7 (0.41)	30.0 (0.42)	27.0 (0.41)	28.2 (0.41)
Good	43.5 (0.45)	36.4 (0.51)	37.5 (0.44)	47.0 (0.46)	49.0 (0.46)	39.8 (0.45)	49.0 (0.46)
Excellent	49.0 (0.46)	33.3 (0.50)	49.2 (0.46)	60.5 (0.45)	61.2 (0.44)	51.8 (0.46)	61.3 (0.44)
Eigenvalue pattern							
Same	39.0 (0.36)	30.5 (0.40)	38.4 (0.36)	45.2 (0.37)	47.2 (0.37)	40.0 (0.37)	45.9 (0.37)
Opposite	40.8 (0.37)	31.4 (0.40)	37.1 (0.36)	44.9 (0.37)	46.2 (0.37)	39.1 (0.36)	46.4 (0.37)
Sample size							
$n = 50$	33.1 (0.55)	27.0 (0.60)	26.1 (0.52)	30.0 (0.54)	33.9 (0.56)	25.6 (0.51)	32.5 (0.55)
$n = 100$	34.2 (0.56)	30.7 (0.63)	26.4 (0.52)	32.2 (0.55)	36.1 (0.57)	29.4 (0.54)	35.0 (0.56)
$n = 200$	43.1 (0.58)	28.1 (0.61)	33.1 (0.55)	47.2 (0.59)	44.4 (0.59)	35.3 (0.56)	46.1 (0.59)
$n = 500$	43.3 (0.58)	34.8 (0.65)	46.1 (0.59)	53.3 (0.59)	56.4 (0.58)	49.4 (0.59)	54.2 (0.59)
$n = 1000$	45.8 (0.59)	34.1 (0.64)	57.2 (0.58)	62.5 (0.57)	62.8 (0.57)	58.1 (0.58)	63.1 (0.57)
Data							
Normal	51.5 (0.46)	32.4 (0.49)	49.3 (0.46)	58.2 (0.45)	62.5 (0.44)	51.5 (0.46)	59.3 (0.45)
Chi-square	43.5 (0.45)	34.2 (0.50)	39.5 (0.45)	52.0 (0.46)	51.0 (0.46)	42.7 (0.45)	52.5 (0.46)
Multivariate t	24.7 (0.39)	26.2 (0.46)	24.5 (0.39)	25.0 (0.40)	26.7 (0.40)	24.5 (0.39)	26.7 (0.40)

4.4. SIMULATION STUDY

105

Table 4.3: Simulation results for $p = 10$ variables. The given values are the percentage of simulation runs (combined over the other factors) for which each of the methods identified the correct model for the covariance matrices of the two groups. Standard errors are given in brackets.

	AIC	Chi ²	BootTest	RVC	BVD	BCR	Ensemble
Overall	39.9 (0.26)	18.7 (0.24)	25.3 (0.23)	44.8 (0.26)	48.2 (0.26)	29.0 (0.24)	45.6 (0.26)
Covariance structure							
Unrelated	80.9 (0.41)	–	14.7 (0.37)	74.0 (0.46)	94.9 (0.23)	22.2 (0.44)	62.0 (0.51)
CPC(1)	27.8 (0.47)	29.1 (0.48)	14.2 (0.37)	51.8 (0.53)	44.4 (0.52)	20.2 (0.42)	50.4 (0.53)
CPC(5)	22.7 (0.44)	10.0 (0.32)	19.8 (0.42)	22.7 (0.44)	26.0 (0.46)	20.0 (0.42)	30.0 (0.48)
CPC(10)	28.2 (0.47)	17.1 (0.40)	52.4 (0.53)	30.7 (0.49)	27.6 (0.47)	53.6 (0.53)	40.0 (0.52)
Eigenvector separation							
Poor	28.2 (0.41)	13.8 (0.36)	8.0 (0.25)	27.8 (0.41)	28.3 (0.41)	7.2 (0.24)	19.3 (0.36)
Good	46.3 (0.46)	23.6 (0.45)	26.2 (0.40)	45.3 (0.45)	51.0 (0.46)	33.2 (0.43)	52.7 (0.46)
Excellent	45.2 (0.45)	18.9 (0.41)	41.7 (0.45)	61.2 (0.44)	65.3 (0.43)	46.7 (0.46)	64.8 (0.44)
Eigenvalue pattern							
Same	39.6 (0.36)	17.9 (0.33)	25.1 (0.32)	45.2 (0.37)	47.7 (0.37)	29.1 (0.34)	45.7 (0.37)
Opposite	40.2 (0.37)	19.6 (0.34)	25.4 (0.32)	44.3 (0.37)	48.8 (0.37)	28.9 (0.34)	45.6 (0.37)
Sample size							
$n = 50$	33.9 (0.56)	17.4 (0.52)	11.1 (0.37)	30.6 (0.54)	32.8 (0.55)	10.0 (0.35)	26.4 (0.52)
$n = 100$	37.5 (0.57)	14.1 (0.47)	14.2 (0.41)	37.2 (0.57)	39.4 (0.58)	14.7 (0.42)	36.4 (0.57)
$n = 200$	38.1 (0.57)	15.9 (0.50)	17.5 (0.45)	44.7 (0.59)	47.8 (0.59)	25.0 (0.51)	44.2 (0.59)
$n = 500$	44.4 (0.59)	24.8 (0.59)	34.4 (0.56)	53.3 (0.59)	58.9 (0.58)	45.3 (0.59)	58.9 (0.58)
$n = 1000$	45.6 (0.59)	21.5 (0.56)	49.2 (0.59)	58.1 (0.58)	62.2 (0.57)	50.0 (0.59)	62.2 (0.57)
Data							
Normal	54.5 (0.45)	20.2 (0.42)	36.8 (0.44)	55.5 (0.45)	62.2 (0.44)	43.2 (0.45)	57.0 (0.45)
Chi-square	40.0 (0.45)	19.1 (0.41)	32.7 (0.43)	53.8 (0.46)	57.2 (0.45)	39.7 (0.45)	54.7 (0.45)
Multivariate t	25.2 (0.40)	16.9 (0.39)	6.3 (0.22)	25.0 (0.40)	25.3 (0.40)	4.2 (0.18)	25.2 (0.40)

Table 4.4: Simulation results for $p = 20$ variables. The given values are the percentage of simulation runs (combined over the other factors) for which each of the methods identified the correct model for the covariance matrices of the two groups. Standard errors are given in brackets.

	AIC	Chi ²	BootTest	RVC	BVD	BCR	Ensemble
Overall	34.8 (0.25)	9.1 (0.18)	21.3 (0.22)	22.2 (0.22)	45.7 (0.26)	26.8 (0.23)	38.7 (0.26)
Covariance structure							
Unrelated	88.4 (0.34)	–	14.2 (0.37)	23.8 (0.45)	95.3 (0.22)	24.0 (0.45)	50.4 (0.53)
CPC(2)	9.8 (0.31)	9.1 (0.30)	12.9 (0.35)	18.4 (0.41)	35.8 (0.51)	22.2 (0.44)	37.6 (0.51)
CPC(10)	13.1 (0.36)	3.8 (0.20)	20.7 (0.43)	7.6 (0.28)	26.0 (0.46)	24.0 (0.45)	30.0 (0.48)
CPC(20)	27.8 (0.47)	14.4 (0.37)	37.6 (0.51)	38.9 (0.51)	25.8 (0.46)	37.1 (0.51)	36.7 (0.51)
Eigenvalue separation							
Poor	33.3 (0.43)	8.4 (0.29)	5.7 (0.21)	17.5 (0.35)	25.5 (0.40)	3.3 (0.16)	14.8 (0.32)
Good	39.5 (0.45)	9.3 (0.31)	21.5 (0.38)	24.5 (0.39)	47.2 (0.46)	30.8 (0.42)	43.5 (0.45)
Excellent	31.5 (0.42)	9.6 (0.31)	36.8 (0.44)	24.5 (0.39)	64.5 (0.44)	46.3 (0.46)	57.7 (0.45)
Eigenvalue pattern							
Same	33.9 (0.35)	9.2 (0.25)	21.1 (0.30)	22.4 (0.31)	45.6 (0.37)	26.2 (0.33)	38.3 (0.36)
Opposite	35.7 (0.36)	9.0 (0.25)	21.6 (0.31)	21.9 (0.31)	45.9 (0.37)	27.4 (0.33)	39.0 (0.36)
Sample size							
$n = 50$	32.8 (0.55)	5.6 (0.31)	5.6 (0.27)	20.0 (0.47)	26.9 (0.52)	5.0 (0.26)	16.9 (0.44)
$n = 100$	30.8 (0.54)	5.6 (0.31)	10.0 (0.35)	18.9 (0.46)	38.3 (0.57)	10.8 (0.37)	25.6 (0.51)
$n = 200$	33.3 (0.56)	8.9 (0.39)	15.0 (0.42)	22.2 (0.49)	46.7 (0.59)	27.2 (0.52)	38.9 (0.57)
$n = 500$	39.4 (0.58)	12.6 (0.45)	30.3 (0.54)	23.1 (0.50)	57.2 (0.58)	43.6 (0.58)	51.1 (0.59)
$n = 1000$	37.5 (0.57)	13.0 (0.46)	45.8 (0.59)	26.7 (0.52)	59.4 (0.58)	47.5 (0.59)	60.8 (0.58)
Data							
Normal	43.7 (0.45)	6.2 (0.25)	32.2 (0.43)	25.2 (0.40)	57.3 (0.45)	41.5 (0.45)	49.8 (0.46)
Chi-square	35.7 (0.44)	7.1 (0.27)	27.8 (0.41)	23.8 (0.39)	54.3 (0.45)	37.7 (0.44)	49.0 (0.46)
Multivariate t	25.0 (0.40)	14.0 (0.37)	4.0 (0.18)	17.5 (0.35)	25.5 (0.40)	1.3 (0.10)	17.2 (0.34)

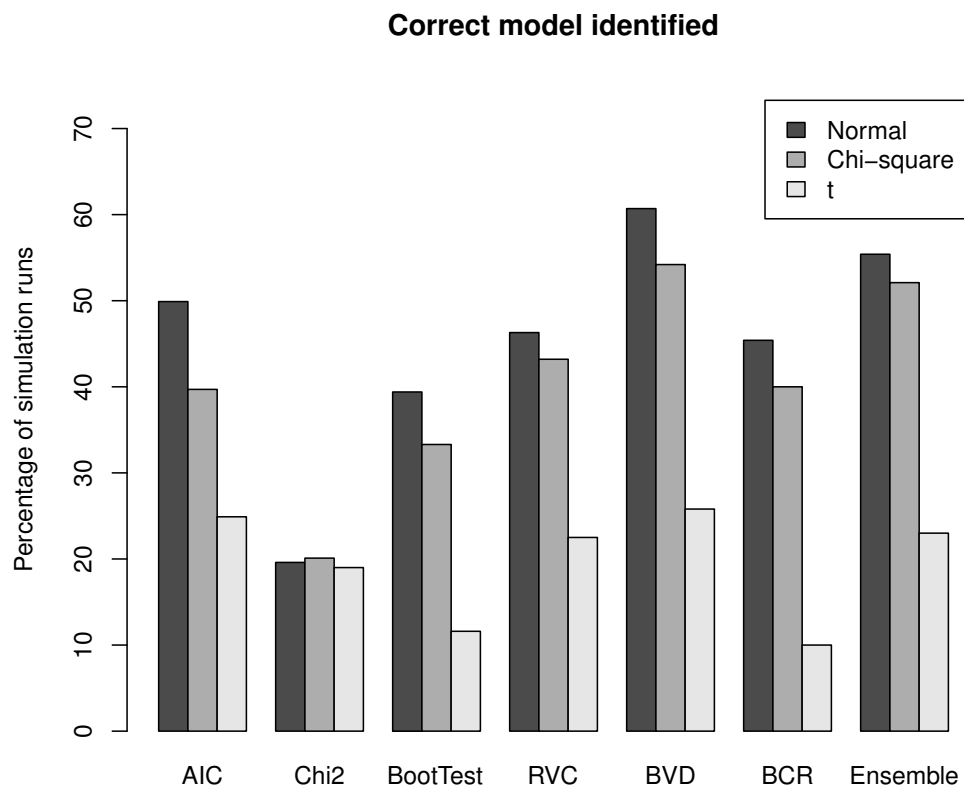


Figure 4.7: Overall percentage of simulation runs for which each of the methods identified the correct covariance structure model, per type of population distribution.

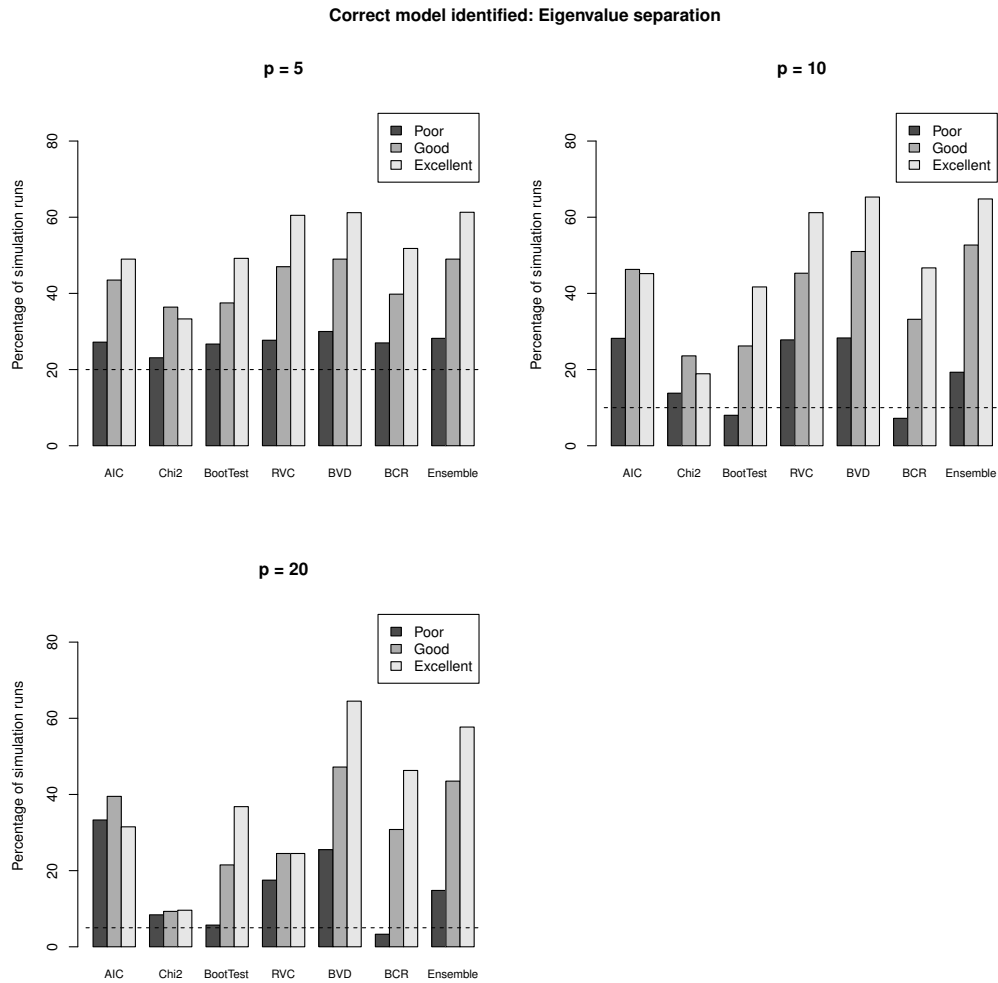


Figure 4.8: Percentage of simulation runs for which each of the methods identified the correct covariance structure model. The benchmark accuracies (long run frequency of selecting the correct model completely at random) are indicated with the broken horizontal lines.

4.4. SIMULATION STUDY

109

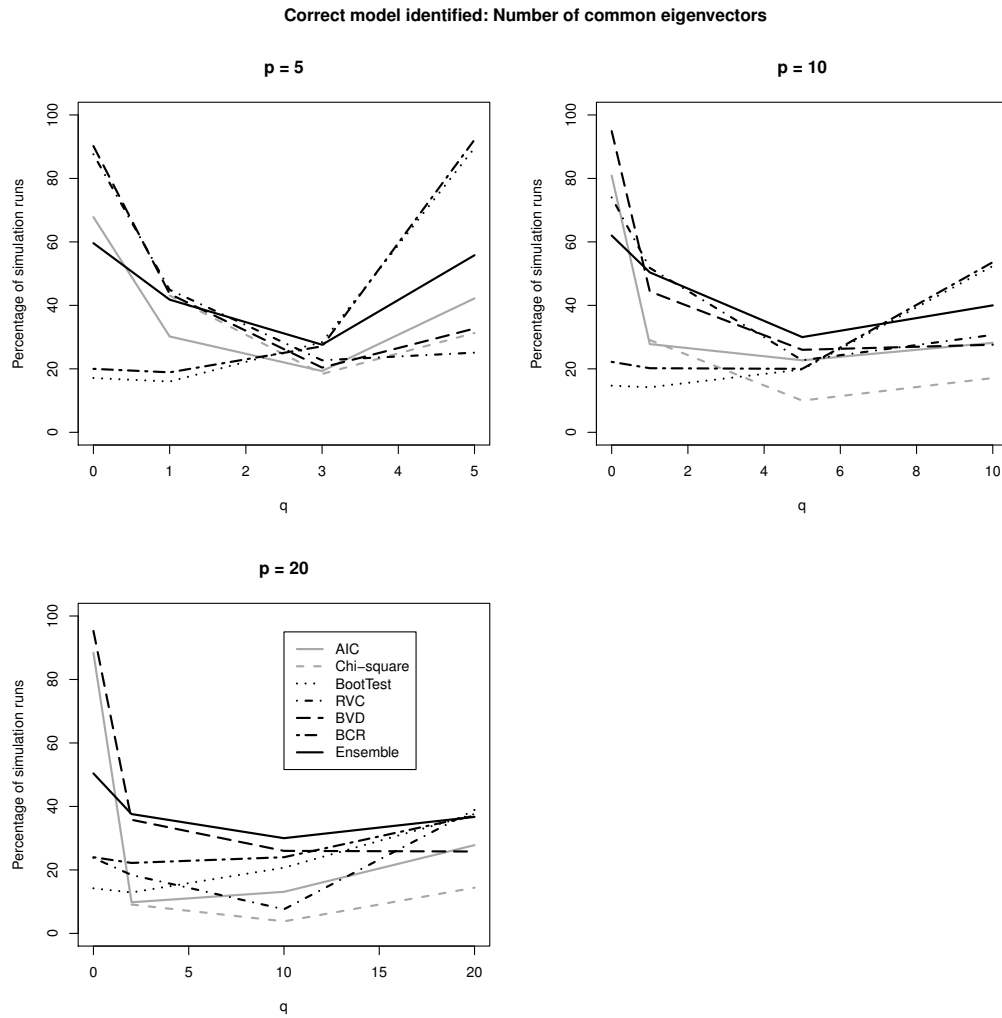


Figure 4.9: Overall percentage of simulation runs for which each of the methods identified the correct covariance structure model, for the different values of q (number of common eigenvectors).

110 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

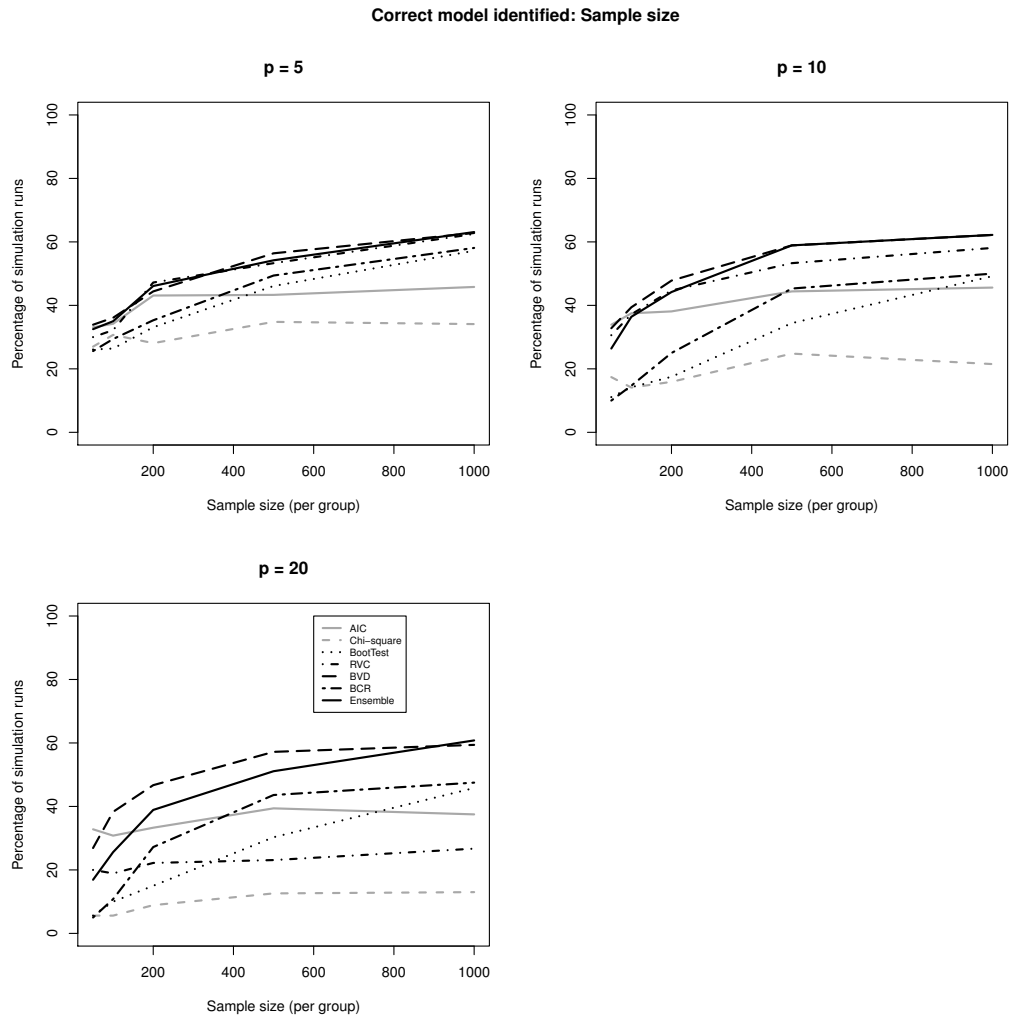


Figure 4.10: Overall percentage of simulation runs for which each of the methods identified the correct covariance structure model, for the different values of n_i (sample size per group).

Ensemble showed the most consistent performance over the range of common eigenvector scenarios. Flury's Chi-square showed uncharacteristic accuracy in the $p = 5, 10$ scenarios for the CPC(1) covariance structures.

With the exception of Flury's AIC and Chi-square, the accuracy of all the methods consistently improves with an increase in sample size (Figure 4.10). This is to be expected for the methods based on bootstrap distributions, as larger samples will usually be more representative of the populations they originate from. For larger samples ($n_i = 200, 500, 1000$), BVD and Ensemble were the clear winners. RVC also fared relatively well in the $p = 5, 10$ cases, but its performance was poor in the $p = 20$ scenarios and did not improve much with an increase in sample size.

The mean number of common eigenvectors identified for each of the p by q scenarios is shown in Table 4.5. BootTest and BCR generally indicate more common eigenvectors and a higher model in Flury's hierarchy than the other methods. This tendency translates into better performance for these methods in the full CPC scenarios. At the other end of the spectrum, BVD is the most conservative and performs best when there are zero or only a small number of common eigenvectors in the two population covariance matrices.

None of the methods were seriously affected by a reversal in the rank order of the common eigenvectors in the second group, compared to the first (see the results for *Eigenvalue pattern* in Tables 4.2, 4.3 and 4.4). For the *Opposite* eigenvalue pattern, the common eigenvectors associated with the largest eigenvalues in the first group were associated with the smallest eigenvalues in the second group, and vice versa.

The following observations can be made from an inspection of a more detailed breakdown of the simulation results:

- *Flury's AIC*. For unrelated covariance matrices, Flury's AIC performs well when the eigenvalues per group are well separated. If the eigenvalues are poorly separated, Flury's AIC still fares well with data from multivariate t distributions with unrelated covariance structures. It performs poorly for multivariate t distributions with common eigenvectors though, and its performance also deteriorates for data from normal and chi-squared type distributions when the dimensionality becomes larger ($p = 20$). This method performs particularly poor with smaller samples ($n_i = 50, 100, 200$) from chi-squared or normal distributions with unrelated covariance structures and poorly separated eigenvalues.
- *Flury's Chi-square*. This method performs at its best for larger samples ($n_i = 500, 1000$) from chi-squared or normal distributions with

Table 4.5: Mean number of common eigenvectors identified for each of the p by q scenarios. Standard errors are given in brackets.

p	q	AIC	Chi ²	BootTest	RVC	BVD	BCR	Ensemble
5	0	1.0 (0.018)	–	3.4 (0.020)	0.2 (0.005)	0.1 (0.004)	3.3 (0.021)	1.0 (0.017)
	1	1.4 (0.018)	2.6 (0.018)	3.9 (0.016)	0.5 (0.006)	0.6 (0.007)	3.9 (0.017)	1.6 (0.016)
	3	2.2 (0.021)	2.8 (0.016)	4.4 (0.010)	1.1 (0.014)	1.6 (0.019)	4.4 (0.010)	2.7 (0.020)
	5	2.6 (0.023)	3.0 (0.016)	4.8 (0.007)	1.8 (0.022)	2.1 (0.023)	4.8 (0.006)	3.2 (0.023)
10	0	1.2 (0.032)	–	5.0 (0.027)	0.3 (0.007)	0.1 (0.002)	4.7 (0.031)	1.1 (0.022)
	1	1.6 (0.031)	5.4 (0.038)	5.4 (0.024)	0.8 (0.008)	0.4 (0.005)	5.3 (0.028)	1.7 (0.020)
	5	3.7 (0.036)	5.0 (0.033)	7.0 (0.017)	2.9 (0.023)	1.8 (0.023)	7.1 (0.016)	4.2 (0.027)
	10	5.3 (0.044)	5.3 (0.032)	8.5 (0.018)	5.0 (0.042)	3.7 (0.045)	8.7 (0.016)	6.4 (0.040)
20	0	1.5 (0.051)	–	7.7 (0.040)	1.0 (0.009)	0.0 (0.002)	7.6 (0.054)	1.4 (0.027)
	2	2.1 (0.055)	12.7 (0.078)	8.9 (0.036)	2.9 (0.017)	0.9 (0.010)	8.7 (0.046)	3.2 (0.027)
	10	6.6 (0.068)	10.1 (0.067)	12.2 (0.024)	8.9 (0.046)	3.4 (0.046)	12.6 (0.021)	9.2 (0.039)
	20	12.0 (0.091)	11.5 (0.064)	15.9 (0.042)	13.6 (0.072)	6.8 (0.090)	16.6 (0.033)	14.4 (0.065)

CPC(1) covariance structures and well separated eigenvalues, and relatively poorly in all other situations. Because of its inability to select the unrelated covariance matrices model, it is not really an equal competitor to the other methods.

- *BootTest*. This method performs well for large samples ($n_i = 500, 1000$) from multivariate normal and chi-squared distributions with a moderate to large number of common eigenvectors and well separated eigenvalues. Under these same conditions, it also performs well for smaller sample sizes when the number of common eigenvectors is large. For a smaller number of common eigenvectors, it performs well as long as data consist of large samples ($n_i = 500, 1000$) from normal or chi-squared distributions with well separated eigenvalues per group. BootTest had notably poor performance for small samples from populations with few or no common eigenvectors.
- *RVC*. For larger samples ($n_i = 200, 500, 1000$) from normal or chi-squared distributions with a larger number of common eigenvectors and well separated eigenvalues, RVC performed at its best. With samples from populations with unrelated covariance structures, this method still fares good as long as the number of dimensions is small ($p = 5, 10$). Its performance deteriorates rapidly as p increases and it therefore seems unsuitable for the analysis of higher dimensional data.
- *BVD*. This method performs well when the groups have unrelated covariance structures. It also clearly excels in situations with large samples ($n_i = 500, 1000$) from normal or chi-squared distributed populations with well separated eigenvalues. The accuracy of BVD deteriorated for samples from populations with multivariate t distributions and/or poorly separated eigenvalues, if such populations had at least some of their eigenvectors in common.
- *BCR*. This method performs best for large samples ($n_i = 500, 1000$) from multivariate normal or chi-squared distributions with well separated eigenvalues. Compared to the other methods, it also performs relatively well when the populations have a moderate to large number of eigenvectors in common.
- *Ensemble*. The Ensemble method performs well for samples from populations with normal or chi-squared distributions and well separated eigenvalues. It also fares reasonably well for data from multivariate t distributions with unrelated covariance structures, probably due to the effect of the BVD method as one of its components.

114 *CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS*

In summary, the BVD and Ensemble methods seem promising, with the results from the simulation study indicating that these outperform the two parametric methods suggested by Flury and the modified non-parametric methods proposed by Klingenberg in the majority of the scenarios considered. Because it is based on bootstrap distributions, the BVD method is inherently free from assumptions about the distributions of the populations from which the data originated. Such assumptions are problematic in many real data sets (see Klingenberg and Froese, 1991; Goodnight and Schwartz, 1997; Phillips and Arnold, 1999; Waldmann and Andersson, 2000; Hallin et al., 2010), and the proposed BVD method thus offers an alternative to the known parametric methods in these situations. The BVD method also outperforms the parametric methods even when the assumption of multivariate normality is valid.

The results of the simulation study support the conclusions made by Flury (1988) about the use of the Chi-square method for model selection in this context. It has the worst performance of the methods considered in this study, and its use is not recommended.

The Ensemble method combines the strengths of the AIC, BootTest, RVC, BVD and BCR methods, and it never performed poorly relative to the other methods considered here. In some cases it even outperformed the BVD method, and the use of the Ensemble method to identify the most appropriate CPC model for the covariance structures of two groups is therefore also recommended.

Future research may aim to extend the proposed non-parametric methods to more than two groups and compare the performance of the methods in this context. It may also be of interest to investigate the adjustment of the penalisation constant of the AIC method to improve its performance for the use of covariance matrix model selection as considered in this dissertation.

4.5 Application to known data sets

The methodology developed in this chapter was applied to three well known data sets from the literature, of which the results are presented in the following sections. The reason for including so many cases here is that it illustrates the new methodology on these known data sets, and therefore enables comparison of the findings with the conclusions made by other researchers, particularly by Flury (1988).

4.5.1 Bank notes data

To compare the known and newly proposed methods for identification of common eigenvectors in two groups, consider the Swiss bank notes data described in Flury (1988). The following six variables were measured (in mm) on sets of *Genuine* ($n_1 = 100$) and *Forged* ($n_2 = 100$) Swiss 1000-franc bank notes:

- Length of the bank note (LENGTH),
- Width, measured on left side (LEFT),
- Width, measured on right side (RIGHT),
- Width of lower margin (BOTTOM),
- Width of upper margin (TOP), and
- Length of the diagonal (DIAG).

The unbiased sample covariance matrices of the two groups look as follows:

- *Genuine* ($n_1 = 100$):

$$\mathbf{S}_1 = \begin{bmatrix} 0.150 & 0.058 & 0.057 & 0.057 & 0.014 & 0.005 \\ 0.058 & 0.133 & 0.086 & 0.057 & 0.049 & -0.043 \\ 0.057 & 0.086 & 0.126 & 0.058 & 0.031 & -0.024 \\ 0.057 & 0.057 & 0.058 & 0.413 & -0.263 & -0.000 \\ 0.014 & 0.049 & 0.031 & -0.263 & 0.421 & -0.075 \\ 0.005 & -0.043 & -0.024 & -0.000 & -0.075 & 0.200 \end{bmatrix}$$

- *Forged* ($n_2 = 100$):

$$\mathbf{S}_2 = \begin{bmatrix} 0.124 & 0.032 & 0.024 & -0.101 & 0.019 & 0.012 \\ 0.032 & 0.065 & 0.047 & -0.024 & -0.012 & -0.005 \\ 0.024 & 0.047 & 0.089 & -0.019 & 0.000 & 0.034 \\ -0.101 & -0.024 & -0.019 & 1.281 & -0.490 & 0.238 \\ 0.019 & -0.012 & 0.000 & -0.490 & 0.404 & -0.022 \\ 0.012 & -0.005 & 0.034 & 0.238 & -0.022 & 0.311 \end{bmatrix}$$

116 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

Using the multivariate Shapiro-Wilk test, the hypothesis of multivariate normality is rejected for the *Genuine* group ($W = 0.9348$, $p < 0.0001$) and the *Forged* group ($W = 0.9109$, $p < 0.0001$).

With the use of Box's M test, the hypothesis of equal covariance matrices is rejected ($p < 0.0001$). However, the results from this test may be invalid due to the lack of multivariate normality in the populations. Likelihood ratio tests, such as Box's M test, is known to be sensitive to deviations from multivariate normality (Rencher, 2002).

As an informal check for proportionality of the population covariance matrices, the elements of \mathbf{S}_1 were divided by the corresponding elements of \mathbf{S}_2 :

$$\begin{bmatrix} 1.21 & 1.84 & 2.39 & -0.57 & 0.74 & 0.47 \\ 1.84 & 2.04 & 1.84 & -2.36 & -4.12 & 8.53 \\ 2.39 & 1.84 & 1.42 & -3.13 & 231.60 & -0.70 \\ -0.57 & -2.36 & -3.13 & 0.32 & 0.54 & -0.00 \\ 0.74 & -4.12 & 231.60 & 0.54 & 1.04 & 3.41 \\ 0.47 & 8.53 & -0.70 & -0.00 & 3.41 & 0.64 \end{bmatrix}$$

Because there is a large range of $\frac{s_{1jh}}{s_{2jh}}$ values, proportionality of the population covariance matrices seems unlikely.

The absolute vector correlations of all pairwise combinations of eigenvectors from the two groups were calculated. The largest six of these correlations are shown in Table 4.6 and displayed on the scree plot in Figure 4.11. From this informal assessment, it seems that there may be two or three common eigenvectors.

The eigenvectors of the covariance matrices of the *Genuine* and *Forged* groups are given in Table 4.7, together with the estimated common eigenvectors under the assumption of full CPC, and the percentage variance ac-

Table 4.6: Six largest absolute vector correlations between all pairwise combinations of eigenvectors from the two bank notes groups.

Eigenvectors		
<i>Genuine</i>	<i>Forged</i>	Correlation
6	6	0.982
2	4	0.977
1	1	0.917
3	2	0.817
4	5	0.800
5	3	0.680

Swiss bank notes: Vector correlations for the permutations

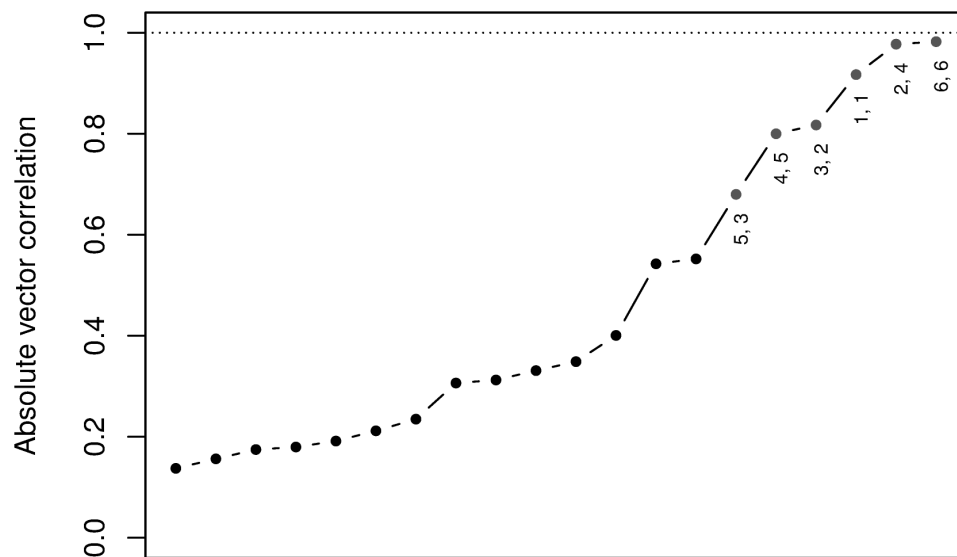


Figure 4.11: Sample vector correlation scree plot for the two bank notes groups.

118 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

counted for by each common eigenvector in each of the groups. A visual inspection shows that the $(\mathbf{e}_{16}, \mathbf{e}_{26})$ and $(\mathbf{e}_{12}, \mathbf{e}_{24})$ eigenvector pairs are nearly collinear.

The results for Flury's AIC and Chi-square methods are reported in Table 4.8. The smallest AIC measure is obtained for the CPC(2) model, indicating that the model with two common eigenvectors fits the data best. The partial $\frac{\chi^2}{df}$ value is closest to one for the CPC(1) model, and the Chi-square method therefore indicates the model with only one common eigenvector.

Results for the Ensemble test (and the constituent methods) to identify common eigenvectors in the bank note groups are shown in Table 4.9. Common eigenvector \mathbf{b}_6 is indicated as common by all of the methods, while \mathbf{b}_2 is considered common by all but the BVD method. The BCR and BootTest methods indicate four and five common eigenvectors, respectively. Due to the orthogonality of the eigenvectors, if the first five are common, the last one should also be common. Considered on its own, the outcome from the BootTest method will thus indicate the full CPC model with six common eigenvectors. The Ensemble method (majority vote from Flury's AIC, BVD, BCR, RVC and BootTest) indicates the CPC(2) model as the most appropriate for the bank note covariance matrices.

Common eigenvector \mathbf{b}_6 contrasts the widths as measured on the left and right sides of the note, thus giving an indication of how parallel the longer sides of the note are. The forged notes show less variation (1.1%) than the genuine notes (3.0%) in this regard. Common eigenvector \mathbf{b}_2 is a weighted combination of the first five variables, contrasted to the sixth (DIAG). It may be interpreted as describing the variation in the size of the note in relation to its shape, as the DIAG variable is related to the aspect ratio (and thus the shape) of the note. It is interesting to note that the genuine notes also displayed greater variation (25.1%) in this aspect, compared to the forged notes (4.6%). One possible explanation for these conclusions might be that the forged notes were printed on the same machine over a short time, leading to greater consistency in the measurements of these notes, while the genuine notes might have been printed over a longer time period on different machines.

Another interesting observation is that the non-common first eigenvectors (\mathbf{e}_{11} and \mathbf{e}_{21}) of both groups involve a contrast between the bottom and top margins of the notes. The variation of the forged notes in this regard is noticeably greater (68.1%) than the genuine notes (47.7%), showing that, while the paper sizes of the forged notes were more consistent, the location of the printed image within the paper frame accounted for more of the observed variation.

Table 4.7: Eigenvectors of the two bank note covariance matrices, and estimated common eigenvectors (using the FG algorithm) under the CPC hypothesis. Eigenvalues of the two groups under the CPC model are given at the bottom of the table, together with the percentage variance accounted for by each common eigenvector.

Separate eigenvectors						
	e_{i1}	e_{i2}	e_{i3}	e_{i4}	e_{i5}	e_{i6}
Genuine						
LENGTH	0.06	-0.38	-0.47	0.79	0.11	-0.01
LEFT	0.01	-0.51	-0.10	-0.24	-0.36	0.74
RIGHT	0.04	-0.45	-0.20	-0.28	-0.48	-0.67
BOTTOM	0.70	-0.36	0.11	-0.24	0.56	-0.05
TOP	-0.71	-0.36	-0.07	-0.24	0.55	-0.06
DIAG	0.11	0.36	-0.84	-0.35	0.12	0.07
d_{1j}	0.69	0.36	0.19	0.09	0.08	0.04
Variance	47.7%	24.9%	12.9%	6.0%	5.6%	2.9%
Forged						
LENGTH	-0.07	0.09	-0.53	-0.29	0.77	0.15
LEFT	-0.01	-0.01	-0.36	-0.42	-0.26	-0.79
RIGHT	-0.01	0.13	-0.40	-0.41	-0.56	0.58
BOTTOM	0.90	0.05	0.24	-0.34	0.10	0.02
TOP	-0.39	0.49	0.56	-0.53	0.10	-0.01
DIAG	0.18	0.85	-0.23	0.41	-0.06	-0.11
d_{2j}	1.55	0.32	0.19	0.10	0.08	0.02
Variance	68.1%	14.0%	8.5%	4.6%	3.7%	1.1%
Common eigenvectors						
	b_1	b_2	b_3	b_4	b_5	b_6
LENGTH	0.02	0.33	-0.45	-0.34	-0.74	0.12
LEFT	0.04	0.48	-0.19	-0.21	0.30	-0.77
RIGHT	0.04	0.44	-0.31	-0.16	0.56	0.61
BOTTOM	0.81	0.32	0.15	0.46	-0.12	0.03
TOP	-0.58	0.46	0.03	0.66	-0.13	0.01
DIAG	0.09	-0.40	-0.80	0.41	0.11	-0.12
Genuine						
l_{1j}	0.67	0.36	0.18	0.10	0.09	0.04
Variance	46.6%	25.1%	12.8%	6.7%	5.9%	3.0%
Forged						
l_{2j}	1.47	0.10	0.26	0.33	0.08	0.02
Variance	64.4%	4.6%	11.5%	14.6%	3.7%	1.1%

120 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

Table 4.8: Flury's AIC and Chi-square statistics for the bank notes data ($k = 2, p = 6$).

Model	χ^2	df	$\frac{\chi^2}{\text{df}}$	AIC
Equality	0.18	1	0.18	125.94
Proportionality	78.53	5	15.71	127.76
CPC	0.04	1	0.04	59.23
CPC(4)	5.61	2	2.81	61.19
CPC(3)	37.75	3	12.58	59.58
CPC(2)	1.51	4	0.38	27.83
CPC(1)	2.32	5	0.46	34.32
Heterogeneity	—	—	—	42.00

Table 4.9: Results from Ensemble test to identify common eigenvectors in the covariance matrices of the two bank note groups. A “Yes” indicates that the specific eigenvector in \mathbf{B} is considered to be common by the method applied.

	Common eigenvector					
	\mathbf{b}_6	\mathbf{b}_2	\mathbf{b}_1	\mathbf{b}_3	\mathbf{b}_5	\mathbf{b}_4
Flury AIC	Yes	Yes	No	No	No	No
BVD	Yes	No	No	No	No	No
BCR	Yes	Yes	No	Yes	Yes	No
RVC	Yes	Yes	Yes	No	No	No
BootTest	Yes	Yes	Yes	Yes	Yes	No
Ensemble	Yes	Yes	No	No	No	No

4.5.2 Swiss heads data

Consider the Swiss heads data discussed by Flury (1988). The data set consists of the following six measurements (in mm) taken on the heads of *Male* ($n_1 = 200$) and *Female* ($n_2 = 59$) soldiers in the Swiss army:

- Minimum frontal breadth (MFB),
- Breadth of angulus mandibulae (BAM),
- True facial height (TFH),
- Length from glabella to apex nasi (LGAN),
- Length from tracion to nasion (LTN), and
- Length from tracion to gnathion (LTG).

The unbiased sample covariance matrices of the two groups are given below.

- *Male* ($n_1 = 200$):

$$\mathbf{S}_1 = \begin{bmatrix} 26.901 & 12.623 & 5.383 & 2.931 & 8.177 & 12.107 \\ 12.623 & 27.252 & 2.880 & 2.058 & 7.126 & 11.441 \\ 5.383 & 2.880 & 35.230 & 10.369 & 6.027 & 7.972 \\ 2.931 & 2.058 & 10.369 & 17.845 & 2.919 & 4.994 \\ 8.177 & 7.126 & 6.027 & 2.919 & 15.370 & 14.521 \\ 12.107 & 11.441 & 7.972 & 4.994 & 14.521 & 31.837 \end{bmatrix}$$

- *Female* ($n_2 = 59$):

$$\mathbf{S}_2 = \begin{bmatrix} 63.203 & 13.156 & 4.393 & -16.120 & 0.044 & 0.470 \\ 13.156 & 35.887 & -0.690 & -1.753 & 8.348 & 5.003 \\ 4.393 & -0.690 & 47.808 & 5.727 & 9.572 & 5.004 \\ -16.120 & -1.753 & 5.727 & 19.393 & 6.716 & 3.845 \\ 0.044 & 8.348 & 9.572 & 6.716 & 26.063 & 12.890 \\ 0.470 & 5.003 & 5.004 & 3.845 & 12.890 & 37.199 \end{bmatrix}$$

With the multivariate Shapiro-Wilk test, the hypothesis of multivariate normality is rejected for both the *Male* ($W = 0.9698$, $p = 0.0003$) and *Female* ($W = 0.9199$, $p = 0.0008$) groups.

As before, due to a lack of an appropriate robust test, Box's M test was used to test the hypothesis of equal covariance matrices. The null hypothesis

122 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

was rejected ($p < 0.0001$), indicating that the population covariance matrices are probably not equal.

The elements of \mathbf{S}_1 were divided by the corresponding elements of \mathbf{S}_2 , producing the values shown below. Proportionality of the population covariance matrices seems unlikely.

$$\begin{bmatrix} 0.43 & 0.96 & 1.23 & -0.18 & 185.99 & 25.73 \\ 0.96 & 0.76 & -4.17 & -1.17 & 0.85 & 2.29 \\ 1.23 & -4.17 & 0.74 & 1.81 & 0.63 & 1.59 \\ -0.18 & -1.17 & 1.81 & 0.92 & 0.43 & 1.30 \\ 185.99 & 0.85 & 0.63 & 0.43 & 0.59 & 1.13 \\ 25.73 & 2.29 & 1.59 & 1.30 & 1.13 & 0.86 \end{bmatrix}.$$

The absolute vector correlations of all pairwise combinations of eigenvectors from the two groups were calculated. The largest six of these correlations are shown in Table 4.10 and displayed on the scree plot in Figure 4.12. There is not a clear indication of the number of common eigenvectors, as the six largest vector correlations exhibit a linear trend and none of them are very close to the value of one. Note that the sixth eigenvector pair (\mathbf{e}_{14} , \mathbf{e}_{24}) in Table 4.10 cannot be common, as the fourth eigenvector of the *Female* group is more highly correlated with the third eigenvector of the *Male* group (fifth row in the table).

The eigenvectors of the covariance matrices of the *Male* and *Female* groups are given in Table 4.11, together with the estimated common eigenvectors under the assumption of full CPC, and the percentage variance accounted for by each common eigenvector in each of the groups.

The results for Flury's AIC and Chi-square methods to identify the appropriate model in Flury's hierarchy are given in Table 4.12. Although the AIC values for the CPC, CPC(4) and CPC(3) models are close together,

Table 4.10: Six largest absolute vector correlations between all pairwise combinations of eigenvectors from the *Male* and *Female* Swiss head groups.

Eigenvectors		
<i>Male</i>	<i>Female</i>	Correlation
5	6	0.843
6	5	0.810
2	3	0.769
1	2	0.739
3	4	0.693
4	4	0.607

Swiss heads data: Vector correlations for the permutations

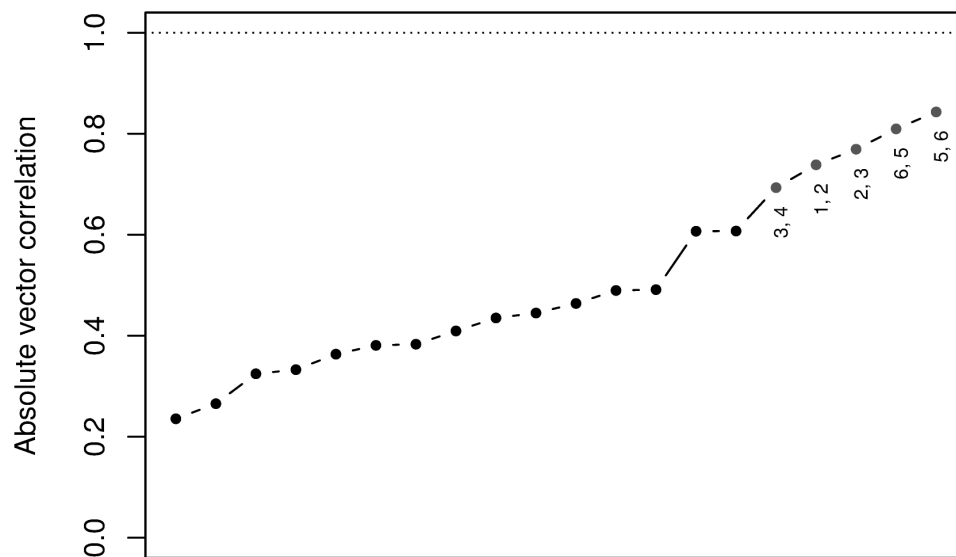


Figure 4.12: Sample vector correlation scree plot for the two Swiss head groups.

124 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

Table 4.11: Eigenvectors of the *Male* and *Female* covariance matrices, and estimated common eigenvectors (using the FG algorithm) under the CPC hypothesis. Eigenvalues of the two groups under the CPC model are given at the bottom of the table, together with the percentage variance accounted for by each common eigenvector.

Separate eigenvectors						
	e_{i1}	e_{i2}	e_{i3}	e_{i4}	e_{i5}	e_{i6}
Male						
MFB	-0.44	0.26	0.42	0.73	-0.13	0.07
BAM	-0.41	0.37	0.50	-0.66	0.10	-0.00
TFH	-0.40	-0.80	0.19	-0.00	0.40	0.06
LGAN	-0.21	-0.35	0.04	-0.17	-0.90	-0.06
LTN	-0.35	0.07	-0.31	0.04	0.09	-0.88
LTG	-0.56	0.16	-0.66	-0.06	0.02	0.47
d_{1j}	66.33	34.42	19.63	14.33	12.96	6.77
Variance	42.9%	22.3%	12.7%	9.3%	8.4%	4.4%
Female						
MFB	0.89	-0.16	0.14	0.21	-0.24	0.25
BAM	0.35	0.14	-0.48	-0.71	0.34	0.06
TFH	0.13	0.65	0.68	-0.15	0.26	-0.05
LGAN	-0.25	0.27	-0.04	-0.14	-0.36	0.85
LTN	0.07	0.46	-0.24	-0.08	-0.71	-0.45
LTG	0.08	0.49	-0.47	0.63	0.35	0.09
d_{2j}	73.51	59.57	41.97	27.99	15.56	10.95
Variance	32.0%	26.0%	18.3%	12.2%	6.8%	4.8%
Common eigenvectors						
	b_1	b_2	b_3	b_4	b_5	b_6
MFB	0.44	-0.26	-0.77	0.23	0.30	-0.01
BAM	0.42	-0.38	0.07	-0.81	-0.13	-0.04
TFH	0.40	0.82	-0.21	-0.14	-0.30	-0.07
LGAN	0.17	0.29	0.29	-0.17	0.88	0.04
LTN	0.36	-0.05	0.22	0.21	-0.12	0.87
LTG	0.56	-0.16	0.47	0.45	-0.11	-0.48
Male						
l_{1j}	66.25	34.33	16.83	16.93	13.27	6.81
Variance	42.9%	22.2%	10.9%	11.0%	8.6%	4.4%
Female						
l_{2j}	62.73	49.62	60.61	26.46	13.10	17.04
Variance	27.3%	21.6%	26.4%	11.5%	5.7%	7.4%

it indicates the CPC(4) model as the most appropriate for the Swiss heads data. The Chi-square method obtains a value closest to one for the CPC(3) model, which implies three common eigenvectors.

Table 4.13 shows the results for the Ensemble test (and the constituent methods) to identify common eigenvectors in the *Male* and *Female* groups. The AIC, BCR and BootTest methods concur on the CPC(4) model, indicating four common eigenvectors. The BVD and RVC methods do not indicate any common eigenvectors, implying that the covariance matrices of the two groups are completely unrelated. The Ensemble test therefore indicates four common eigenvectors and selects the CPC(4) model as the most appropriate for the Swiss head groups.

From the estimated loadings for the first common eigenvector, \mathbf{b}_1 , in Table 4.11 it appears that the first common principal component can be interpreted as a *size* measure, because the loadings for the six physical measurements all have the same sign. Note that variation in this *Size* component is relatively less for females (27.3%) than for males (42.9%), even though the eigenvalues for the two groups are similar. This result shows that the female soldiers in the sample exhibited relatively more variation in other head/facial features (for example, shape) than their male counterparts.

4.5.3 Iris data

Another well known multivariate data set is the Iris data first described by Anderson (1935). Flury (1988) concluded that the *Versicolor* and *Virginica* groups had one common eigenvector, so the analysis below was restricted to these two groups only. The four variables measured (in cm) on each iris flower were:

Table 4.12: Flury's AIC and Chi-square statistics for the Swiss heads data ($k = 2$, $p = 6$).

Model	χ^2	df	$\frac{\chi^2}{df}$	AIC
Equality	42.15	1	42.15	89.78
Proportionality	25.81	5	5.16	49.63
CPC	4.80	1	4.80	33.82
CPC(4)	1.14	2	0.57	31.03
CPC(3)	2.73	3	0.91	33.89
CPC(2)	5.48	4	1.37	37.15
CPC(1)	7.68	5	1.54	39.68
Heterogeneity	—	—	—	42.00

126 CHAPTER 4. IDENTIFICATION OF COMMON EIGENVECTORS

Table 4.13: Results from Ensemble test to identify common eigenvectors in the covariance matrices of the *Male* and *Female* groups. A “Yes” indicates that the specific eigenvector in \mathbf{B} is considered to be common by the method applied.

	Common eigenvector					
	\mathbf{b}_5	\mathbf{b}_6	\mathbf{b}_2	\mathbf{b}_1	\mathbf{b}_3	\mathbf{b}_4
Flury AIC	Yes	Yes	Yes	Yes	No	No
BVD	No	No	No	No	No	No
BCR	Yes	Yes	Yes	Yes	No	No
RVC	No	No	No	No	No	No
BootTest	Yes	Yes	Yes	Yes	No	No
Ensemble	Yes	Yes	Yes	Yes	No	No

- Sepal length (SLENGTH),
- Sepal width (SWIDTH),
- Petal length (PLENGTH), and
- Petal width (PWIDTH).

The unbiased sample covariance matrices of the *Versicolor* and *Virginica* groups are as follows:

- *Versicolor* ($n_1 = 50$):

$$\mathbf{S}_1 = \begin{bmatrix} 0.266 & 0.085 & 0.183 & 0.056 \\ 0.085 & 0.098 & 0.083 & 0.041 \\ 0.183 & 0.083 & 0.221 & 0.073 \\ 0.056 & 0.041 & 0.073 & 0.039 \end{bmatrix}$$

- *Virginica* ($n_2 = 50$):

$$\mathbf{S}_2 = \begin{bmatrix} 0.404 & 0.094 & 0.303 & 0.049 \\ 0.094 & 0.104 & 0.071 & 0.048 \\ 0.303 & 0.071 & 0.305 & 0.049 \\ 0.049 & 0.048 & 0.049 & 0.075 \end{bmatrix}$$

Using multivariate Shapiro-Wilk tests, the hypothesis of multivariate normality is rejected for both *Versicolor* ($W = 0.9304$, $p = 0.0057$) and *Virginica* ($W = 0.9341$, $p = 0.0080$) at a 5% significance level.

Despite the lack of normality, Box's M test was used to test whether the two population covariances may be equal, and the equality hypothesis was rejected ($p = 0.0001$). From a division of the elements of \mathbf{S}_1 by the corresponding elements of \mathbf{S}_2 , it does not seem as if the two population covariance matrices are proportional:

$$\begin{bmatrix} 0.66 & 0.91 & 0.60 & 1.14 \\ 0.91 & 0.95 & 1.16 & 0.87 \\ 0.60 & 1.16 & 0.72 & 1.50 \\ 1.14 & 0.87 & 1.50 & 0.52 \end{bmatrix}.$$

The largest absolute vector correlations between all pairs of eigenvectors from the two iris groups are given in Table 4.14 and shown on the scree plot in Figure 4.13. From the graph, it seems that Flury's conclusion of one common eigenvector may be correct.

The eigenvectors of the covariance matrices of the *Versicolor* and *Virginica* groups are given in Table 4.15, together with the estimated common eigenvectors under the assumption of full CPC, and the percentage variance accounted for by each common eigenvector in each of the groups. The absolute loadings of the $(\mathbf{e}_{11}, \mathbf{e}_{21})$ eigenvector pair look similar, and the loadings of the $(\mathbf{e}_{14}, \mathbf{e}_{24})$ eigenvector pair are also not very different.

Flury's AIC and Chi-square statistics for all possible covariance structure models applicable to the two iris groups are given in Table 4.16. The lowest AIC value is attained by the CPC(1) model with a single common eigenvector. The Chi-square method showed a slightly better fit for the CPC(1) model than the full CPC model with four common eigenvectors.

Results for the Ensemble test (and the constituent methods) to identify common eigenvectors in the two iris groups are reported in Table 4.17. Common eigenvector \mathbf{b}_1 is indicated as common by all of the methods, and is the only common eigenvector identified by Flury's AIC and the RVC method.

Table 4.14: Four largest absolute vector correlations between all pairwise combinations of eigenvectors from the two iris groups, *Versicolor* ($n_1 = 50$) and *Virginica* ($n_2 = 50$).

Eigenvectors		
<i>Versicolor</i>	<i>Virginica</i>	Correlation
1	1	0.989
4	4	0.876
3	3	0.815
2	2	0.685

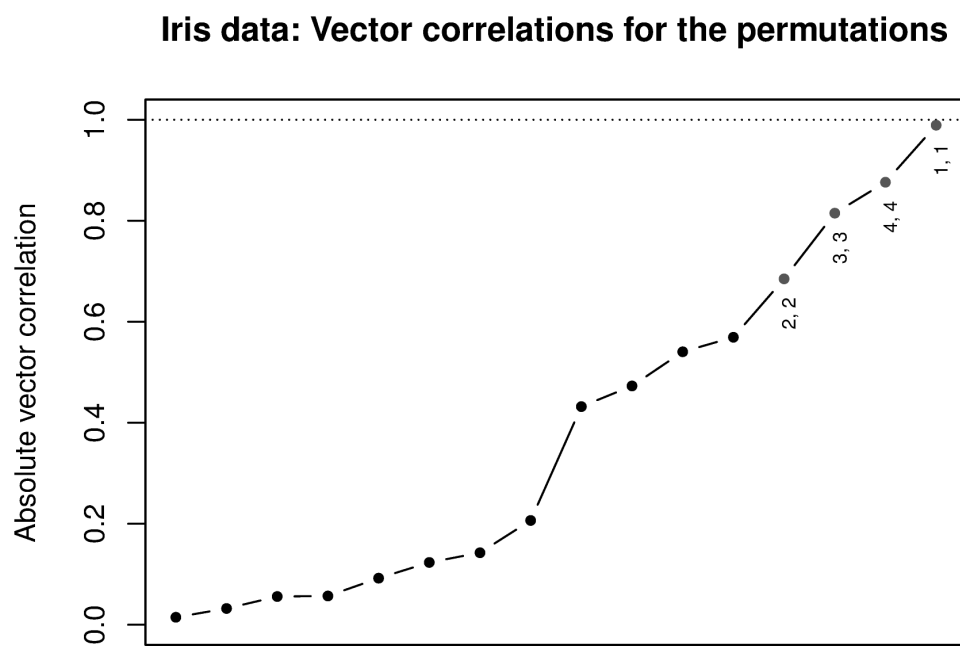


Figure 4.13: Sample vector correlation scree plot for the two iris groups, *Versicolor* and *Virginica*.

Table 4.15: Eigenvectors of the *Versicolor* and *Virginica* sample covariance matrices, and estimated common eigenvectors (using the FG algorithm) under the CPC hypothesis. Eigenvalues of the two groups under the CPC model are given at the bottom of the table, together with the percentage variance accounted for by each common eigenvector.

Separate eigenvectors				
	e_{i1}	e_{i2}	e_{i3}	e_{i4}
Versicolor				
SLENGTH	0.69	0.67	0.27	0.10
SWIDTH	0.31	-0.57	0.73	-0.23
PLENGTH	0.62	-0.34	-0.63	-0.32
PWIDTH	0.21	-0.34	-0.06	0.92
d_{1j}	0.49	0.07	0.05	0.01
Variance	78.1%	11.6%	8.8%	1.6%
Virginica				
SLENGTH	-0.74	-0.17	-0.53	0.37
SWIDTH	-0.20	0.75	-0.33	-0.54
PLENGTH	-0.63	-0.17	0.65	-0.39
PWIDTH	-0.12	0.62	0.43	0.65
d_{2j}	0.70	0.11	0.05	0.03
Variance	78.3%	12.0%	5.9%	3.9%
Common eigenvectors				
	b_1	b_2	b_3	b_4
SLENGTH	0.72	-0.29	-0.62	0.13
SWIDTH	0.25	0.90	-0.19	-0.29
PLENGTH	0.62	-0.12	0.72	-0.29
PWIDTH	0.18	0.30	0.26	0.90
Versicolor				
l_{1j}	0.49	0.07	0.06	0.01
Variance	77.8%	10.7%	10.0%	1.6%
Virginica				
l_{2j}	0.69	0.10	0.05	0.05
Variance	77.8%	11.2%	5.8%	5.2%

Table 4.16: Flury's AIC and Chi-square statistics for the two iris groups ($k = 2, p = 4$).

Model	χ^2	df	$\frac{\chi^2}{df}$	AIC
Equality	9.12	1	9.12	36.64
Proportionality	14.07	3	4.69	29.53
CPC	1.04	1	1.04	21.46
CPC(2)	9.30	2	4.65	22.42
CPC(1)	3.11	3	1.04	17.11
Heterogeneity	—	—	—	20.00

The BVD, BCR and BootTest methods indicate at least three common eigenvectors, which, due to the orthogonality constraint, implies the full CPC model with four common eigenvectors. The Ensemble method therefore also indicates the full CPC model as most appropriate for the covariance matrices of the *Versicolor* and *Virginica* populations.

The first common principal component, for which there is the most certainty regarding commonness, appears to be weighted combination of the four original variables. It can thus be interpreted as providing a description of the *size* of the iris flower, accounting for close to 80% of the variation observed in each of the *Versicolor* and *Virginica* samples.

Table 4.17: Results from Ensemble test to identify common eigenvectors in the covariance matrices of the two iris groups. A “Yes” indicates that the specific eigenvector in \mathbf{B} is considered to be common by the method applied.

	Common eigenvector			
	\mathbf{b}_1	\mathbf{b}_4	\mathbf{b}_3	\mathbf{b}_2
Flury AIC	Yes	No	No	No
BVD	Yes	Yes	Yes	No
BCR	Yes	Yes	Yes	Yes
RVC	Yes	No	No	No
BootTest	Yes	Yes	Yes	Yes
Ensemble	Yes	Yes	Yes	No

4.6 Application to the VON data

The methods to identify common eigenvectors in two groups were applied first to the delivery mode groups (*Caesarean* and *Vaginal*), and secondly to the

regional groups (*South Africa* and *Namibia*) in the VON 2009 cohort. Results from these analyses are given and discussed in the following two sections. Infants who died before final discharge and those who were transferred to alternative NICUs were not excluded from these analyses.

4.6.1 Delivery mode

The data for the two delivery mode groups were tested for multivariate normality of the populations using the multivariate Shapiro-Wilk test. The null hypothesis was strongly rejected for both the *Caesarean* ($W = 0.9223$, $p < 0.0001$) and the *Vaginal* groups ($W = 0.8897$, $p < 0.0001$).

For lack of a robust alternative, Box's M test was used to test for equality of the two population covariance matrices. The null hypothesis was rejected at a 5% significance level ($p < 0.0001$). Using the unbiased sample covariance matrices as given in Section 3.11.1, the elements of \mathbf{S}_1 were divided by the corresponding elements of \mathbf{S}_2 as an informal check for proportionality of the covariance matrices. The values obtained (see below) does not support the proportionality hypothesis, although most of values are smaller than one, indicating that the variation in the *Vaginal* group is generally larger than in the *Caesarean* group.

$$\begin{bmatrix} 0.90 & 0.95 & 0.79 & 0.71 & 0.82 & 0.60 \\ 0.95 & 0.66 & 0.59 & 0.80 & 1.06 & 0.26 \\ 0.79 & 0.59 & 0.66 & 0.72 & 0.92 & 0.24 \\ 0.71 & 0.80 & 0.72 & 0.62 & 0.65 & 0.41 \\ 0.82 & 1.06 & 0.92 & 0.65 & 0.78 & 0.48 \\ 0.60 & 0.26 & 0.24 & 0.41 & 0.48 & 0.52 \end{bmatrix}$$

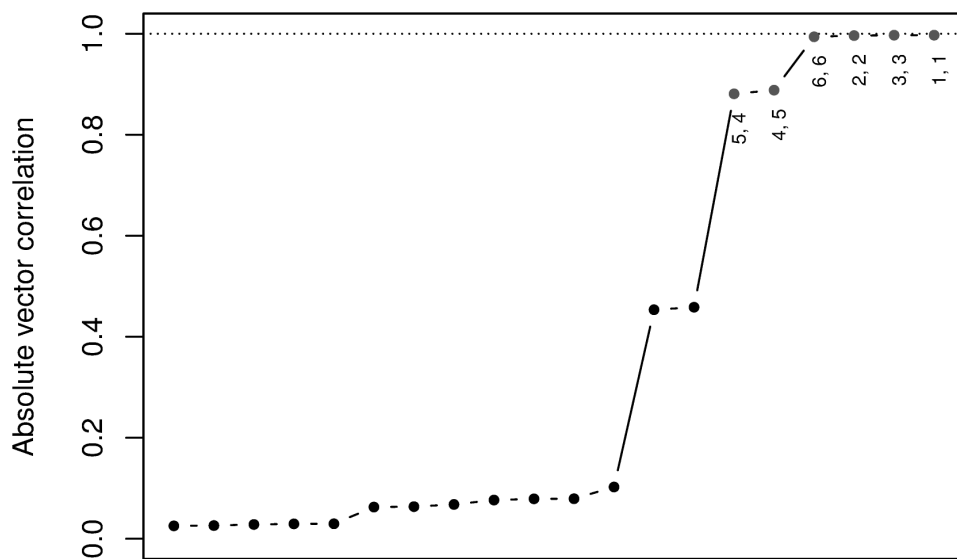
Vector correlations were calculated for all pairs of eigenvectors from the two delivery mode groups. The six eigenvector pairs with the largest correlations are shown in Table 4.18 and Figure 4.14. It seems that the CPC(4) or full CPC model may be the most appropriate for the covariance matrices, as there are clear breaks between the fourth and fifth largest, and the sixth and seventh largest vector correlations, respectively, in Figure 4.14.

The results for Flury's AIC and Chi-square methods to identify common eigenvectors in the two delivery mode groups are given in Table 4.19. While the Chi-square method indicates the CPC(2) model, the minimum AIC value is obtained for the completely unrelated covariance matrices model.

Results for the Ensemble test (and the constituent methods) to identify common eigenvectors in the two delivery mode groups are reported in Table 4.20. The BCR and BootTest methods both indicate the full CPC model, while BVD and RVC indicate the CPC(4) model. The Ensemble method thus

Table 4.18: Six largest absolute vector correlations between all pairwise combinations of eigenvectors from the delivery mode groups.

Eigenvectors		Correlation
<i>Caesarean</i>	<i>Vaginal</i>	
1	1	0.997
3	3	0.997
2	2	0.996
6	6	0.994
4	5	0.888
5	4	0.881

Delivery mode: Vector correlations for the permutationsFigure 4.14: Sample vector correlation scree plot for the *Caesarean* and *Vaginal* groups.

4.6. APPLICATION TO THE VON DATA

133

Table 4.19: Flury's AIC and Chi-square statistics for the two delivery mode groups ($k = 2$, $p = 6$).

Model	χ^2	df	$\frac{\chi^2}{df}$	AIC
Equality	82.40	1	82.40	258.64
Proportionality	70.03	5	14.01	178.25
CPC	7.78	1	7.78	118.22
CPC(4)	14.84	2	7.42	112.44
CPC(3)	22.34	3	7.45	101.59
CPC(2)	1.67	4	0.42	85.26
CPC(1)	59.59	5	11.92	91.59
Heterogeneity	—	—	—	42.00

identifies \mathbf{b}_1 , \mathbf{b}_2 and \mathbf{b}_3 as common and selects the CPC(3) model as the most appropriate for the delivery mode groups.

The first three common principal components together account for more than 95% of the variation observed in each of the delivery mode groups (see Table 3.2), and should provide a sufficient approximation for the full p -dimensional data set.

Table 4.20: Results from Ensemble test to identify common eigenvectors in the covariance matrices of the two delivery mode groups. A “Yes” indicates that the specific eigenvector in \mathbf{B} is considered to be common by the method applied.

	Common eigenvector					
	\mathbf{b}_1	\mathbf{b}_3	\mathbf{b}_2	\mathbf{b}_6	\mathbf{b}_4	\mathbf{b}_5
Flury AIC	No	No	No	No	No	No
BVD	Yes	Yes	Yes	Yes	No	No
BCR	Yes	Yes	Yes	No	Yes	Yes
RVC	Yes	Yes	Yes	Yes	No	No
BootTest	Yes	Yes	Yes	No	Yes	Yes
Ensemble	Yes	Yes	Yes	No	No	No

4.6.2 Regions

The populations for the two regional groups are also not multivariate normally distributed, as the null hypothesis for both *South Africa* ($W = 0.935$, $p < 0.0001$) and *Namibia* ($W = 0.792$, $p < 0.0001$) were rejected at a 5% significance level.

Box's M test was again used to test for equality of the two population covariance matrices, and the null hypothesis was rejected at a 5% significance level ($p < 0.0001$). Using the unbiased sample covariance matrices as given in Section 3.11.2, the elements of \mathbf{S}_1 were divided by the corresponding elements of \mathbf{S}_2 as an informal check for proportionality of the covariance matrices. The wide range of values obtained (see below) gives evidence against the proportionality hypothesis.

$$\begin{bmatrix} 0.78 & 0.51 & 0.57 & 0.80 & 0.66 & 5.26 \\ 0.51 & 0.82 & 1.04 & 0.42 & 0.45 & -3.91 \\ 0.57 & 1.04 & 1.13 & 0.45 & 0.48 & -6.52 \\ 0.80 & 0.42 & 0.45 & 0.81 & 0.69 & -2.52 \\ 0.66 & 0.45 & 0.48 & 0.69 & 0.67 & 14.90 \\ 5.26 & -3.91 & -6.52 & -2.52 & 14.90 & 1.13 \end{bmatrix}$$

Vector correlations were calculated for all pairs of eigenvectors from the two regions. The six eigenvector pairs with the highest correlations are shown in Table 4.21 and Figure 4.15. It seems that there are six common eigenvectors as there is a clear break between the sixth and seventh largest vector correlations in Figure 4.15.

The results for Flury's AIC and Chi-square methods to identify common eigenvectors in the covariance matrices of the two regions are given in Table 4.22. The Chi-square method favours the proportional covariance matrices model, while the AIC method indicates the unrelated covariance matrices and thus no common eigenvectors.

To conclude this chapter, results for the Ensemble test (and the constituent methods) to identify common eigenvectors in the two regional groups are shown in Table 4.23. While the AIC method finds no common eigenvectors, the results from the BVD, BCR, RVC and BootTest methods all point to six common eigenvectors. The Ensemble test thus indicates the full CPC model assuming six common eigenvectors in the populations of the two regions.

Table 4.22: Flury’s AIC and Chi-square statistics for *South Africa* and *Namibia* ($k = 2, p = 6$).

Model	χ^2	df	$\frac{\chi^2}{df}$	AIC
Equality	5.99	1	5.99	85.77
Proportionality	10.09	5	2.02	81.78
CPC	2.06	1	2.06	81.69
CPC(4)	5.27	2	2.63	81.63
CPC(3)	12.87	3	4.29	80.37
CPC(2)	34.37	4	8.59	73.50
CPC(1)	15.13	5	3.03	47.13
Heterogeneity	–	–	–	42.00

Table 4.23: Results from Ensemble test to identify common eigenvectors in the covariance matrices of *South Africa* and *Namibia*. A “Yes” indicates that the specific eigenvector in \mathbf{B} is considered to be common by the method applied.

	Common eigenvector					
	\mathbf{b}_1	\mathbf{b}_3	\mathbf{b}_2	\mathbf{b}_6	\mathbf{b}_4	\mathbf{b}_5
Flury AIC	No	No	No	No	No	No
BVD	Yes	Yes	Yes	Yes	Yes	Yes
BCR	Yes	Yes	Yes	Yes	Yes	Yes
RVC	Yes	Yes	Yes	Yes	Yes	Yes
BootTest	Yes	Yes	Yes	Yes	Yes	Yes
Ensemble	Yes	Yes	Yes	Yes	Yes	Yes

Chapter 5

Improved estimation of covariance matrices

5.1 Introduction

Accurate estimation of population covariance matrices is important as these matrices are used as input for many statistical techniques. Because it is well known that both the maximum likelihood covariance matrix estimator,

$$\mathbf{S}_{\text{ML}} = \frac{1}{n}(\mathbf{X}'\mathbf{X} - n\bar{\mathbf{x}}\bar{\mathbf{x}}'), \quad (5.1)$$

and the unbiased sample covariance matrix estimator,

$$\mathbf{S} = \frac{1}{n-1}(\mathbf{X}'\mathbf{X} - n\bar{\mathbf{x}}\bar{\mathbf{x}}'), \quad (5.2)$$

are very sensitive to outliers (Huber, 2004; Flury, 1988), various robust estimators have been proposed. These include M -estimators (Maronna, 1976) and other proposals discussed in Gnanadesikan and Kettenring (1972), Devlin et al. (1975), Huber (1977) and Mosteller and Tukey (1977).

Empirical Bayes approaches have also been suggested, usually involving a James-Stein type of linear shrinkage estimator. The unbiased sample covariance matrix is shrunk towards a predetermined target matrix, chosen for its desirable properties in the context of application. For recent discussions of some of these shrinkage estimators, see Daniels and Kass (2001), Ledoit and Wolf (2004), and Schäfer and Strimmer (2005).

A related approach is the regularisation of sample covariance matrices to find well-conditioned estimates. This work is particularly relevant in a high-dimensional, sparse data setting. For some solutions using a regularisation approach, see Hastie et al. (2009) and Bien and Tibshirani (2011).

In practice, when samples are obtained from a number of multivariate populations, the assumption of a common covariance matrix for the populations is often made in order to calculate a pooled sample covariance matrix to be used in subsequent analyses. Instead of assuming equal population covariance matrices, the less restrictive assumption of common (or partially common) eigenvectors in the population covariance matrices can be made. Information regarding the assumed common eigenvector structure in the populations can be used to find improved estimators for the population covariance matrices which are less biased than the pooled covariance matrix estimator. In the CPC situation, estimating the covariance matrices under the CPC model from small samples will also lead to more stable estimators of the population covariance matrices than when using the unbiased sample covariance matrix estimator.

For an example where this idea may be useful, consider the situation where only a small sample is available from a multivariate population of patients with a rare disease. The elements of the unbiased sample covariance matrix estimator may be unstable due to the few degrees of freedom available to estimate all of the variances and covariances. However, if a larger sample of measurements on the same set of variables is available from a population of healthy people, an assumed CPC structure in the diseased and healthy populations can be used to obtain a more accurate covariance matrix estimate for the diseased population.

This chapter presents new ways of employing the CPC and partial CPC models to provide improved estimators of population covariance matrices. To quantify the improvement in estimation accuracy, a modified version of the Frobenius norm for symmetric matrices is proposed in Section 5.2.

The CPC estimator proposed by Flury (1988) is presented in Section 5.3. A new Stein-type shrinkage estimator for the population covariance matrices is introduced in Section 5.4, using the CPC estimator as the target matrix. Three ways of finding the optimal value for the shrinkage intensity parameter are proposed in Section 5.5.

The performance of the proposed covariance matrix estimators were compared to the unbiased and pooled estimators in a Monte Carlo simulation study, of which the results are reported in Section 5.6. Lastly, in Section 5.7 the CPC shrinkage estimator is applied to obtain covariance matrix estimates for the regional and delivery mode groupings in the VON 2009 cohort.

5.2 Accuracy of covariance matrix estimators

For a population with known covariance matrix, Σ , the accuracy of a covariance matrix estimator, \mathbf{S} , can be measured using the Frobenius matrix norm. With σ_{jh} and s_{jh} indicating the $(j, h)^{th}$ elements of Σ and \mathbf{S} , respectively, the Frobenius norm for the difference between Σ and \mathbf{S} is defined as (Golub and Van Loan, 1996)

$$\begin{aligned} \|\mathbf{S} - \Sigma\|_F &= \sqrt{\text{vec}'(\mathbf{S} - \Sigma)\text{vec}(\mathbf{S} - \Sigma)} \\ &= \sqrt{\sum_{j=1}^p \sum_{h=1}^p (s_{jh} - \sigma_{jh})^2}, \end{aligned} \quad (5.3)$$

with $\text{vec}(\mathbf{A})$ indicating a column vector containing the stacked columns of matrix \mathbf{A} .

Due to the symmetric nature of covariance matrices, deviations in the covariances (off-diagonal elements) are given twice the weight of deviations in the variances (diagonal elements) using the Frobenius norm as defined in (5.3). To allow for an equal weighting of all $p(p+1)/2$ covariance matrix parameters, a modified version of the Frobenius norm,

$$\begin{aligned} \|\mathbf{S} - \Sigma\|_{F^*} &= \sqrt{\text{vecs}'(\mathbf{S} - \Sigma)\text{vecs}(\mathbf{S} - \Sigma)} \\ &= \sqrt{\sum_{j=1}^p \sum_{h \leq j}^p (s_{jh} - \sigma_{jh})^2}, \end{aligned} \quad (5.4)$$

is proposed to measure deviations from a population covariance matrix in a comparison of the accuracy of different covariance matrix estimators. The notation $\text{vecs}(\mathbf{A})$ indicate a column vector containing the stacked columns of the lower triangular part (including the diagonal) of matrix \mathbf{A} .

With \mathbf{S}_i indicating the unbiased sample estimator of the i^{th} population covariance matrix Σ_i , the modified Frobenius norm of $(\mathbf{S}_i - \Sigma_i)$ will in this chapter be used as the benchmark against which to compare alternative estimators for Σ_i . The improvement in the modified Frobenius norm for each of the proposed estimators, relative to the unbiased sample estimator, are reported in the simulation study in Section 5.6.

5.3 Estimating covariance matrices under the CPC model

Suppose $\mathbf{X}_i, i = 1, \dots, k$ are matrices containing observations on the same p variables from k distinct populations. Let Σ_i and \mathbf{S}_i indicate the population and unbiased sample covariance matrices of the i^{th} group, respectively.

Under the assumption of common eigenvectors in the population covariance matrices, and indicating the population common eigenvector matrix as \mathbf{B} , the $\Lambda_i = \mathbf{B}'\Sigma_i\mathbf{B}$ matrices should be diagonal. However, due to sampling variation, the off-diagonal elements of the sample estimators, $\mathbf{L}_i = \mathbf{B}'\mathbf{S}_i\mathbf{B}$, will generally be small but not equal to zero.

The information in all k samples can be utilised to determine the common eigenvector matrix, \mathbf{B} , of which the elements will generally have smaller standard errors than the corresponding elements of the estimators of the individual eigenvector matrices, \mathbf{E}_i (Flury, 1988). Assuming that the lack of diagonality in the \mathbf{L}_i matrices is due to sampling error exclusively, and assuming that the CPC model provides an accurate description of the populations, a CPC estimator for the Σ_i can be obtained by shrinking the off-diagonal elements of the \mathbf{L}_i to zero before multiplying it with the common eigenvector matrix, \mathbf{B} , i.e.

$$\mathbf{S}_{i(\text{CPC})} = \mathbf{B}\mathbf{L}_i^0\mathbf{B}', \quad i = 1, \dots, k, \quad (5.5)$$

where

$$\mathbf{L}_i^0 = \text{diag}(\mathbf{B}'\mathbf{S}_i\mathbf{B}) \quad (5.6)$$

is a diagonalised matrix with the eigenvalues of the i^{th} group (under the CPC model) on the diagonal (Flury, 1988).

The extension to the partial CPC case is straightforward. Letting \mathbf{B}_i be the matrix of common and non-common eigenvectors of the i^{th} covariance matrix, the sample estimators of the principal component covariance matrices are given by $\mathbf{L}_i = \mathbf{B}_i'\mathbf{S}_i\mathbf{B}_i$, and the CPC estimator for Σ_i in the partial CPC scenario is

$$\mathbf{S}_{i(\text{CPC}(q))} = \mathbf{B}_i\mathbf{L}_i^0\mathbf{B}_i', \quad i = 1, \dots, k, \quad (5.7)$$

where the

$$\mathbf{L}_i^0 = \text{diag}(\mathbf{B}_i'\mathbf{S}_i\mathbf{B}_i) \quad (5.8)$$

are the diagonalised sample eigenvalue matrices, analogous to (5.6), (Flury, 1988).

5.4 CPC shrinkage estimator

If the CPC (or partial CPC) model is not a true representation of the population covariance matrices, shrinking the off-diagonal elements of the \mathbf{L}_i matrices to zero as in (5.5) and (5.7) may be too severe, discarding information about where the CPC model does not provide a good fit.

Under the equal population covariance matrices assumption, Hastie et al. (2009) proposed a Stein-type weighted estimator,

$$\mathbf{S}_{i(\text{pooled})}^* = \alpha_i \mathbf{S}_i + (1 - \alpha_i) \mathbf{S}_p, \quad i = 1, \dots, k, \quad (5.9)$$

for the \mathbf{S}_i matrices, where α_i is the shrinkage intensity parameter for the i^{th} group and \mathbf{S}_p is the pooled covariance matrix as in (4.2). They suggested estimating α_i by crossvalidation. Under the common eigenvectors assumption, the pooled covariance matrix in (5.9) can be replaced with the CPC estimators from (5.5) or (5.7), i.e.

$$\mathbf{S}_{i(\text{CPC})}^* = \alpha_i \mathbf{S}_i + (1 - \alpha_i) \mathbf{S}_{i(\text{CPC})}, \quad i = 1, \dots, k, \quad (5.10)$$

or

$$\mathbf{S}_{i(\text{CPC}(q))}^* = \alpha_i \mathbf{S}_i + (1 - \alpha_i) \mathbf{S}_{i(\text{CPC}(q))}, \quad i = 1, \dots, k. \quad (5.11)$$

Because $\mathbf{S}_i = \mathbf{B} \mathbf{L}_i \mathbf{B}'$ and $\mathbf{S}_{i(\text{CPC})} = \mathbf{B} \mathbf{L}_i^0 \mathbf{B}'$ as defined in (5.5), (5.10) can be written as

$$\begin{aligned} \mathbf{S}_{i(\text{CPC})}^* &= \alpha_i \mathbf{B} \mathbf{L}_i \mathbf{B}' + (1 - \alpha_i) \mathbf{B} \mathbf{L}_i^0 \mathbf{B}' \\ &= \mathbf{B} [\alpha_i \mathbf{L}_i + (1 - \alpha_i) \mathbf{L}_i^0] \mathbf{B}' \\ &= \mathbf{B} [\alpha_i (\mathbf{L}_i - \mathbf{L}_i^0) + \mathbf{L}_i^0] \mathbf{B}'. \end{aligned} \quad (5.12)$$

The term $\alpha_i (\mathbf{L}_i - \mathbf{L}_i^0)$ in (5.12) thus performs a linear shrinkage of the off-diagonal elements of \mathbf{L}_i , according to the size of the shrinkage coefficient, α_i .

An appropriate value for α_i can be obtained using crossvalidation or by testing possible values on a validation data set, as suggested by Hastie et al. (2009). As these validation methods involve the estimation of covariance matrices for subsets of the p -dimensional data, this approach needs sufficiently large samples and may be unreliable (and possibly unfeasible) for small n_i .

A related approach is to select α_i to minimise the error rate for a specific application, such as the misclassification rate in linear discriminant analysis.

In this case the estimates of Σ_i will depend partly on the (possibly ill-chosen) application, which can lead to inaccurate estimates of the population covariance matrices.

5.5 Estimation of the shrinkage intensity parameter

In the following sections, three methods for estimation of the shrinkage intensity parameter in (5.10) or (5.11) are proposed.

5.5.1 Flury's ϕ method

Flury (1988) defined a measure of deviation from diagonality for a positive definite matrix \mathbf{F} as

$$\phi(\mathbf{F}) = \frac{\det(\text{diag}(\mathbf{F}))}{\det(\mathbf{F})}. \quad (5.13)$$

Letting $\mathbf{F} = \mathbf{L}_i$, the ϕ measure in (5.13) can be used to judge how well the common eigenvector matrix, \mathbf{B} , diagonalises the i^{th} sample covariance matrix. Thus $\phi(\mathbf{L}_i) \geq 1$, with $\prod_{i=1}^k \phi(\mathbf{L}_i) = 1$ if the CPC model fits the sample data perfectly.

Noting that $0 < \frac{1}{\phi(\mathbf{L}_i)} \leq 1$, subtracting the inverse of $\phi(\mathbf{L}_i)$ from one yields the following estimator for α_i :

$$\begin{aligned} \hat{\alpha}_i &= 1 - \frac{\det(\mathbf{L}_i)}{\det[\text{diag}(\mathbf{L}_i)]} \\ &= 1 - \frac{\det(\mathbf{L}_i)}{\det(\mathbf{L}_i^0)}. \end{aligned} \quad (5.14)$$

5.5.2 Crossvalidation method

In the context of regularised discriminant analysis, to find the optimal value for the shrinkage parameter α_i in (5.9), Friedman (1989) and Hastie et al. (2009) proposed using either crossvalidation or calculating the value of α_i which minimises a predetermined error measure on a validation data set. The crossvalidation idea can be applied to estimate the optimal value for α_i in (5.10) in the following way.

After calculating the common eigenvector matrix, \mathbf{B} , from the grouped sample data, the data for the i^{th} group are randomly divided R times into

5.5. ESTIMATION OF THE SHRINKAGE INTENSITY PARAMETER 143

a 70% training set and a 30% test set. Such a 70 : 30 division, as used in Sections 5.6 and 5.7 of this chapter, allows for estimation of the shrinkage parameter on about two-thirds of the data, and leaves the remaining third for testing purposes. However, the proportions for the testing and training sets can be altered according to the data under consideration.

Let $\mathbf{S}_{i(\text{TRAIN})}^{(r)}$ and $\mathbf{S}_{i(\text{TEST})}^{(r)}$ indicate the unbiased sample covariance matrices of the r^{th} training and test sets, respectively. Let $\mathbf{S}_{i(\text{CPC})}^{(r)}$ indicate the covariance matrix of the i^{th} group under the CPC assumption, calculated from the r^{th} training set, i.e.

$$\mathbf{S}_{i(\text{CPC})}^{(r)} = \mathbf{B} \text{diag}(\mathbf{B}' \mathbf{S}_{i(\text{TRAIN})}^{(r)} \mathbf{B}) \mathbf{B}', \quad r = 1, \dots, R. \quad (5.15)$$

For each of the R replications, the value of $\alpha_i^{(r)} \in [0; 1]$ which minimises

$$\left\| \left[\alpha_i^{(r)} \mathbf{S}_{i(\text{TRAIN})}^{(r)} + (1 - \alpha_i^{(r)}) \mathbf{S}_{i(\text{CPC})}^{(r)} \right] - \mathbf{S}_{i(\text{TEST})}^{(r)} \right\|_{F^*} \quad (5.16)$$

is calculated. The optimal value for α_i in (5.10), calculated with this crossvalidation method, is the mean of the $\alpha_i^{(r)}$ values,

$$\hat{\alpha}_i = \frac{\sum_{r=1}^R \alpha_i^{(r)}}{R}. \quad (5.17)$$

For small samples where $0.3 \times n_i < p$, the covariance matrices of the test sets will be singular. The effect of singularity in the $\mathbf{S}_{i(\text{TEST})}^{(r)}$ matrices on the estimation of α_i have not been studied for the purpose of this dissertation, and is a topic which may be explored in future research. To avoid singularity in the $\mathbf{S}_{i(\text{TEST})}^{(r)}$ matrices, the crossvalidation method should not be applied to any group for which $0.3n_i < p$.

Under the partial CPC assumption, the common eigenvector matrix in (5.15) can be replaced by \mathbf{B}_i , the matrix of common and non-common eigenvectors of the covariance matrix of the i^{th} group.

5.5.3 Schäfer and Strimmer method

Schäfer and Strimmer (2005) noted that neither the unbiased sample covariance matrix estimator, \mathbf{S} , nor the maximum likelihood estimator, $\mathbf{S}_{ML} = \frac{n-1}{n} \mathbf{S}$, performs well in a sparse data setting where $p \gg n$. To estimate an unknown population covariance matrix, $\mathbf{\Sigma}$, they proposed the linear shrinkage estimator,

$$\mathbf{S}^* = \lambda \mathbf{T} + (1 - \lambda) \mathbf{S}, \quad (5.18)$$

where \mathbf{S} denotes the unbiased sample covariance matrix estimator and \mathbf{T} some predetermined target matrix. Because $E(\mathbf{S}) = \mathbf{\Sigma}$, the Ledoit-Wolf lemma (Ledoit and Wolf, 2003) can be used to find the optimal value for the shrinkage intensity parameter, λ , as

$$\lambda^* = \frac{\sum_{j=1}^p \sum_{h=1}^p \text{Var}(s_{jh}) - \text{Cov}(t_{jh}, s_{jh})}{\sum_{j=1}^p \sum_{h=1}^p E[(t_{jh} - s_{jh})^2]}, \quad (5.19)$$

where s_{jh} and t_{jh} indicate the $(j, h)^{th}$ elements of \mathbf{S} and \mathbf{T} , respectively. An advantage of this approach is that it does not require computationally expensive procedures such as bootstrap or crossvalidation to estimate the shrinkage intensity parameter.

Schäfer and Strimmer (2005) gave a number of suitable forms for the target matrix \mathbf{T} . In the CPC context, the CPC covariance matrix estimator in (5.5) can serve as the target matrix for the i^{th} group. In this case, (5.18) becomes

$$\begin{aligned} \mathbf{S}_i^* &= \lambda_i \mathbf{S}_{i(\text{CPC})} + (1 - \lambda_i) \mathbf{S}_i \\ &= (1 - \lambda_i) \mathbf{S}_i + \lambda_i \mathbf{S}_{i(\text{CPC})}, \end{aligned} \quad (5.20)$$

which means that λ_i is analogous to $1 - \alpha_i$ as defined in (5.10). From (5.12) it can be observed that (5.20) implies $\mathbf{T} = \mathbf{L}_i^0$, and that the target matrix corresponds to the diagonal target with unequal variances as specified by Schäfer and Strimmer (2005). The elements of the target matrix for the i^{th} group, \mathbf{T}_i , are thus defined as

$$t_{ijh} = \begin{cases} l_{ijh} & \text{if } j = h \\ 0 & \text{if } j \neq h. \end{cases} \quad (5.21)$$

Replacing the parameters in (5.19) with their sample counterparts, Schäfer and Strimmer (2005) showed that the optimal value for the shrinkage parameter for the target as defined in (5.21) can be calculated with

$$\hat{\lambda}_i^* = \frac{\sum_{j \neq h} \widehat{\text{Var}}(l_{ijh})}{\sum_{j \neq h} l_{ijh}^2}. \quad (5.22)$$

From (5.20) and (5.22) it can be observed that, the larger the variances of the off-diagonal elements of \mathbf{L}_i are in comparison to their estimated values, the more weight will be assigned to the CPC estimator, $\mathbf{S}_{i(\text{CPC})}$. This makes sense, as relatively large variances for the off-diagonal l_{ijh} values mean that any observed deviation of these estimates from zero can probably be ascribed mainly to sampling variation. In that case the assumption of CPC (or partial

CPC) is appropriate and the CPC estimator for Σ_i will be more accurate than the unbiased estimator.

Because a closed form solution for the variances of the off-diagonal elements of the \mathbf{L}_i matrices is not available, we propose using bootstrap procedures to estimate these. Plugging the bootstrap estimates into (5.22) will yield estimates of the optimal values for the shrinkage intensity parameters, $\lambda_i = 1 - \alpha_i$. To avoid assignment of a negative weight to the unbiased estimator in (5.20), $\hat{\lambda}_i^*$ is constrained (by truncation) to fall in the interval $[0; 1]$. The optimal value for α_i in (5.10) is estimated as

$$\hat{\alpha}_i = 1 - \hat{\lambda}_i^*. \quad (5.23)$$

However, the original computational efficiency advantage of the Schäfer and Strimmer method is eroded by the need to calculate bootstrap estimates for the variances of the off-diagonal l_{ijh} values. The Schäfer and Strimmer method as implemented in this dissertation was found to be no faster than the crossvalidation method in finding an appropriate value for the shrinkage intensity parameter.

It can be observed that in the case of perfect CPC in sample data, the common eigenvector matrix \mathbf{B} will diagonalise the \mathbf{S}_i matrices perfectly. In that case, $\mathbf{L}_i^0 = \mathbf{L}_i$ and $\mathbf{S}_{i(\text{CPC})} = \mathbf{S}_i$, which implies that the values for $\alpha_i, i = 1, \dots, k$, in (5.10) will be undefined. From a computational point of view, (5.14), (5.17) and (5.23) will still provide satisfactory values for the shrinkage parameters, α_i .

It may be tempting to always assume the full CPC model and use the CPC estimate, $\mathbf{S}_{i(\text{CPC})}$, in (5.10), before trying to infer from the estimated value of α_i how many of the eigenvectors in \mathbf{B} are truly common. However, the previous observation means that as \mathbf{S}_i approaches $\mathbf{S}_{i(\text{CPC})}$, the values of the α_i will become ill-defined and such inference about the number of common eigenvectors is therefore not advisable.

To improve the readability of the rest of this chapter, the Schäfer and Strimmer method will from here on be referred to as the “Schäfer method”.

5.6 Simulation study

A Monte Carlo simulation study was performed to compare the CPC estimator and the new CPC shrinkage estimators to the unbiased and pooled covariance matrix estimators. The modified Frobenius norm in (5.4) was used as the error measure to compare the estimation accuracy of the different covariance matrix estimators. In all cases the common eigenvector

matrices were estimated with the FG algorithm. The following names are used to denote the covariance matrix estimators in this simulation study:

- **Unbiased.** The unbiased sample covariance matrix estimator as in (5.2).
- **CPC.** The estimator of the covariance matrix under the appropriate CPC or partial CPC model, as in (5.5) or (5.7).
- **Flury phi.** The CPC shrinkage estimator in (5.11), with the shrinkage intensity parameter calculated according to (5.14).
- **CPC crossvalid.** The CPC shrinkage estimator in (5.11), using the \mathbf{B}_i matrices containing the common and non-common eigenvectors of the covariance matrix of the i^{th} group to calculate the shrinkage intensity parameter by the crossvalidation method according to (5.17).
- **Full CPC crossvalid.** The CPC shrinkage estimator in (5.10), using the common eigenvector matrix under the assumption of full CPC, \mathbf{B} , to calculate the shrinkage intensity parameter by the crossvalidation method according to (5.17).
- **Schäfer.** The CPC shrinkage estimator in (5.11), with the shrinkage intensity parameter as in (5.23). For each simulation run, the sample estimate of the common (or partially common) eigenvector matrix was kept constant while calculating bootstrap estimates of the variances in (5.22) from 1000 bootstrap replications.
- **Pooled.** The pooled sample covariance matrix estimator in (4.2).

The strength of covariance matrix estimation using the CPC model lies in combining information regarding the common eigenvectors from several groups, because the elements of the common eigenvector estimators have smaller standard errors than the elements of the individually (per group) eigenvector estimators (Flury, 1988). Therefore the greatest improvement in estimation error is expected when the eigenvector structure estimated from a large sample from one group is used to estimate the covariance matrix for another group with relatively few observations. For this simulation study, only $k = 2$ groups were considered, with the sample size of the first group chosen to be relatively large ($n_1 = 200, 500, 1000$) and the sample size of the second group to be relatively small ($n_2 = 30, 50, 100, 200$). All of the results reported in the following sections are for the estimation of the population covariance matrix of the second group (smaller sample).

The number of variables per group was fixed at $p = 5, 10$ or 20 , and the number of common eigenvectors varied according to the value of p : For $p = 5$, $q = 0, 1, 3$ or 5 common eigenvectors were considered; for $p = 10$, $q = 0, 1, 5$ or 10 common eigenvectors were used; for $p = 20$, $q = 0, 2, 10$ or 20 common eigenvectors were used. These choices allow for the evaluation of the estimators in the cases where respectively, no, few, half, or all of the population eigenvectors are common (see the table below).

Case	p	q
Full CPC	5	5
	10	10
	20	20
(About) Half of eigenvectors common	5	3
	10	5
	20	10
Few common eigenvectors	5	1
	10	1
	20	2
No common eigenvectors	5	0
	10	0
	20	0

From (3.53) and (3.54) it is known that the standard errors of the common eigenvector loadings depend on how well each of the associated eigenvalues are separated from the rest of the eigenvalues within each group. If a subset of common eigenvectors is not well defined in at least one of the population covariance matrices, the standard errors of the common eigenvector loadings will tend to be large (Flury, 1988), causing the CPC covariance matrix estimators to perform poorly. The choice of population eigenvalues for the simulation study is thus also important, and the following three eigenvalue patterns were considered:

- **Same pattern.** The common eigenvectors have the same rank order in both groups, when ordered according to the size of the eigenvalues per group.
- **Similar pattern.** The largest eigenvalues in both groups are associated with the same subset of common eigenvectors, and the smallest eigenvalues in both groups are associated with the same subset of common eigenvectors.

- **Opposite pattern.** The largest eigenvalues in the first group and smallest eigenvalues in the second group are associated with the same subset of common eigenvectors, and vice versa.

In addition to the three eigenvalue patterns considered, the relative separation between subsequent eigenvalues per group were varied, from *Poor* (10% separation between eigenvalues in the first group; 20% separation between eigenvalues in the second group), to *Good* (first group: 40% separation; second group: 50% separation), to *Excellent* (first group: 80% separation; second group: 90% separation). The vectors of population eigenvalues for each of the eigenvalue patterns and degrees of separation are given in Appendix C.

Data were simulated from multivariate normal, multivariate chi-squared with two degrees of freedom (see Appendix A for further details), and multivariate t with one degree of freedom distributions for each of the population covariance structure scenarios. Using 100 simulation runs per $(p \times q \times n_1 \times n_2 \times \text{Eigenvalue separation} \times \text{Eigenvalue pattern} \times \text{Multivariate distribution type})$ factor combination led to a total of 291600 simulation runs. The $(p = 10, n_2 = 30)$ and $(p = 20, n_2 = 30, 50)$ combinations were excluded to avoid using singular covariance matrices in the estimation of the shrinkage parameters for the *CPC crossvalid* and *Full CPC crossvalid* estimators.

Because of the large number of simulations, fitting linear models to the simulation results and performing ANOVA to determine which of the effects (and interactions) are significant did not prove useful, as almost all of the higher order interactions were statistically significant. It was therefore decided to consider only the main effects in the sections which follow.

5.6.1 Full CPC case

The case where all p eigenvectors are common to both groups is considered first. For each simulation run, the modified Frobenius measure was calculated for each of the estimators. Indicating the minimum and maximum of these modified Frobenius measure values for the r^{th} simulation run with $F_{\min}^{(r)}$ and $F_{\max}^{(r)}$, respectively, the Frobenius value for the each estimator, $F_{\text{estimator}}^{(r)}$, was standardised with

$$F_{\text{estimator}}^{\star(r)} = \frac{F_{\text{estimator}}^{(r)} - F_{\min}^{(r)}}{F_{\max}^{(r)} - F_{\min}^{(r)}} \quad (5.24)$$

to fall in the interval $[0; 1]$. This enabled comparison of the standardised modified Frobenius values across the simulation runs. The mean and median

standardised modified Frobenius values for each of the estimators in the full CPC case are shown in Table 5.1. Overall, the *Schäfer* estimator performed the best in the full CPC case, slightly outperforming the *CPC crossvalid* and *Full CPC crossvalid* estimators. Note that, in the full CPC case, the *CPC crossvalid* and *Full CPC crossvalid* estimators are the same, as both make use of the full common eigenvector matrix, \mathbf{B} , to estimate the optimal value for the shrinkage intensity parameter.

Figure 5.1 shows the distribution of the ratio improvement in the modified Frobenius measure for each of the covariance matrix estimators, compared to the *Unbiased* estimator, i.e.

$$\frac{F_{\text{Unbiased}}^{(r)} - F_{\text{estimator}}^{(r)}}{F_{\text{Unbiased}}^{(r)}}. \quad (5.25)$$

The graph was constructed using kernel density estimation with a Gaussian kernel function (Silverman, 1986). Because of its very poor performance, the *Pooled* estimator is not included on this graph. The bulk of the distributions of *Full CPC crossvalid* and *Schäfer* are positive, and it can be seen that these two methods offer the greatest improvement compared to *Unbiased*.

Wilcoxon signed-rank tests were used to determine whether the ratio improvement in the modified Frobenius measure for each of the covariance matrix estimators, compared to *Unbiased*, is equal zero. Two-sided p -values for these tests are reported in Table 5.2, together with a point estimate and 95% interval estimate for the improvement. *CPC* performed about 16% worse than *Unbiased*, and *Pooled* performed about 151% worse than *Unbiased*. *Schäfer* offers an improvement of 5.4%, and the crossvalidation estimators an improvement of 4.7%, compared to *Unbiased*.

The effects of the sample sizes from the first and second populations, respectively, on each of the estimators (compared to *Unbiased*) are shown in Figure 5.2. The performance of *CPC crossvalid*, *Full CPC crossvalid* and *Schäfer*, with regards to the estimation of the covariance matrix of the second population, all improve with an increase in the sample size from the first population. *Flury phi* showed a decrease in accuracy (compared to *Unbiased*) with an increase in the size of the sample from the first population. As the size of the sample from the second population was increased, all of the estimators showed a gradual decrease in improvement over *Unbiased*. This is to be expected, for as the sample from the second population increases in size, the accuracy of the unbiased covariance matrix estimator should improve. For large samples from the second population, covariance matrix estimators under the CPC model will thus not offer a large improvement when compared to the unbiased estimator.

The distribution of the ratio improvement of each estimator, compared to *Unbiased*, for the different types of multivariate distributions are shown in Figure 5.3. The largest improvement in the accuracy of estimation for the CPC estimators is clearly when the populations have multivariate t distributions. For multivariate t distributed populations, *CPC*, *CPC crossvalid*, *Full CPC crossvalid* and *Schäfer* fared about equally well.

Changes in the eigenvalue patterns (*Same*, *Similar* or *Opposite* patterns) did not seem to have a great effect on the accuracy of estimation, as can be seen in Figure 5.4. *CPC*, *CPC crossvalid*, *Full CPC crossvalid* and *Schäfer* perform only slightly better when the eigenvalue pattern is the *Same*, compared to an *Opposite* ranking of the eigenvectors in the two population covariance matrices.

As the separation between the eigenvalues increases, the distributions of the ratio improvement in the modified Frobenius measure for the CPC estimators gradually move to the right in Figure 5.5. This means that an increase in the eigenvalue separation per group leads to greater improvement in the estimation accuracy of the CPC estimators, compared to *Unbiased*.

Wilcoxon signed-rank tests were also used to compare *Full CPC crossvalid* to the other estimators. The results for these two-sided tests are shown in Table 5.3. In the full CPC case, *Full CPC crossvalid* performs 4.7% better than *Unbiased*, 14.9% better than *CPC*, 4.1% better than *Flury phi*, and 46.9% better than *Pooled*. *Full CPC crossvalid* fared marginally worse (0.5%) than *Schäfer*, but (as expected) there was no significant difference between *Full CPC crossvalid* and *CPC crossvalid*.

Table 5.1: Mean and median standardised modified Frobenius values for the different covariance matrix estimators in the full CPC case.

	Mean	Median
Unbiased	0.269	0.036
CPC	0.372	0.284
Flury phi	0.239	0.041
CPC crossvalid	0.193	0.028
Full CPC crossvalid	0.192	0.028
Schäfer	0.192	0.020
Pooled	0.792	1.000

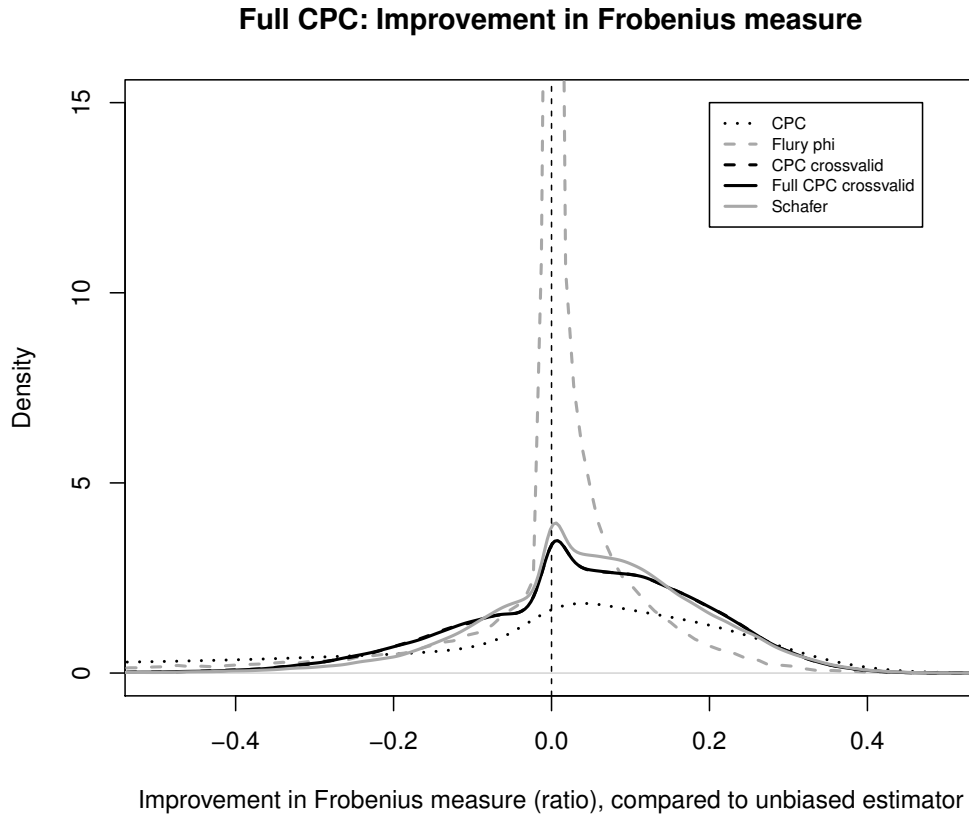


Figure 5.1: Ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

Table 5.2: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the covariance matrix estimators, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

	Estimate	95% LCL	95% UCL	<i>p</i> -value
CPC	-0.157	-0.162	-0.152	< 0.0001
Flury phi	0.005	0.005	0.005	< 0.0001
CPC crossvalid	0.047	0.046	0.048	< 0.0001
Full CPC crossvalid	0.047	0.046	0.048	< 0.0001
Schäfer	0.054	0.053	0.055	< 0.0001
Pooled	-1.512	-1.530	-1.494	< 0.0001

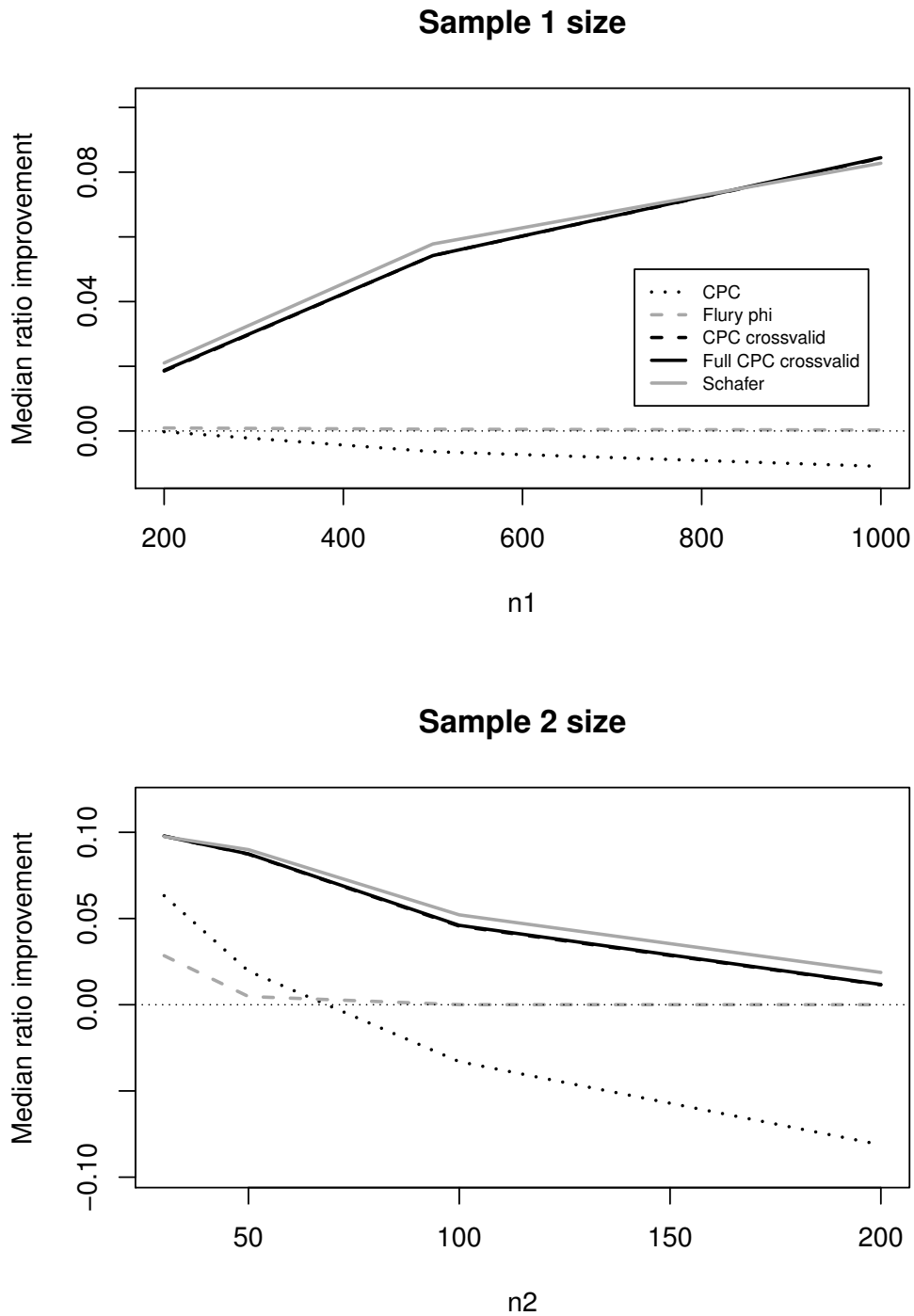
Full CPC: Effect of sample sizes

Figure 5.2: Effect of sample size on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

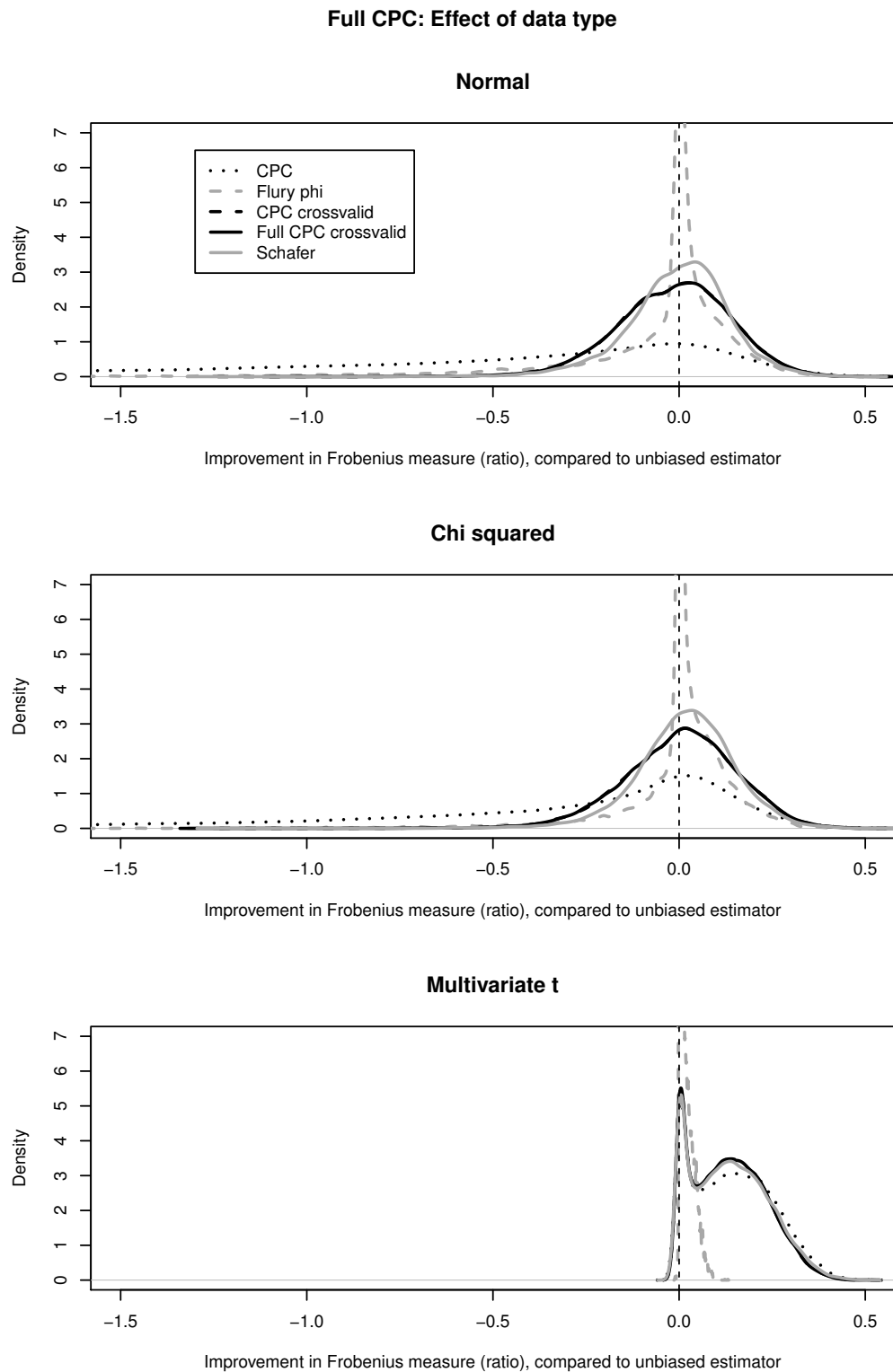


Figure 5.3: Effect of multivariate distribution type on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

Full CPC: Effect of eigenvalue pattern

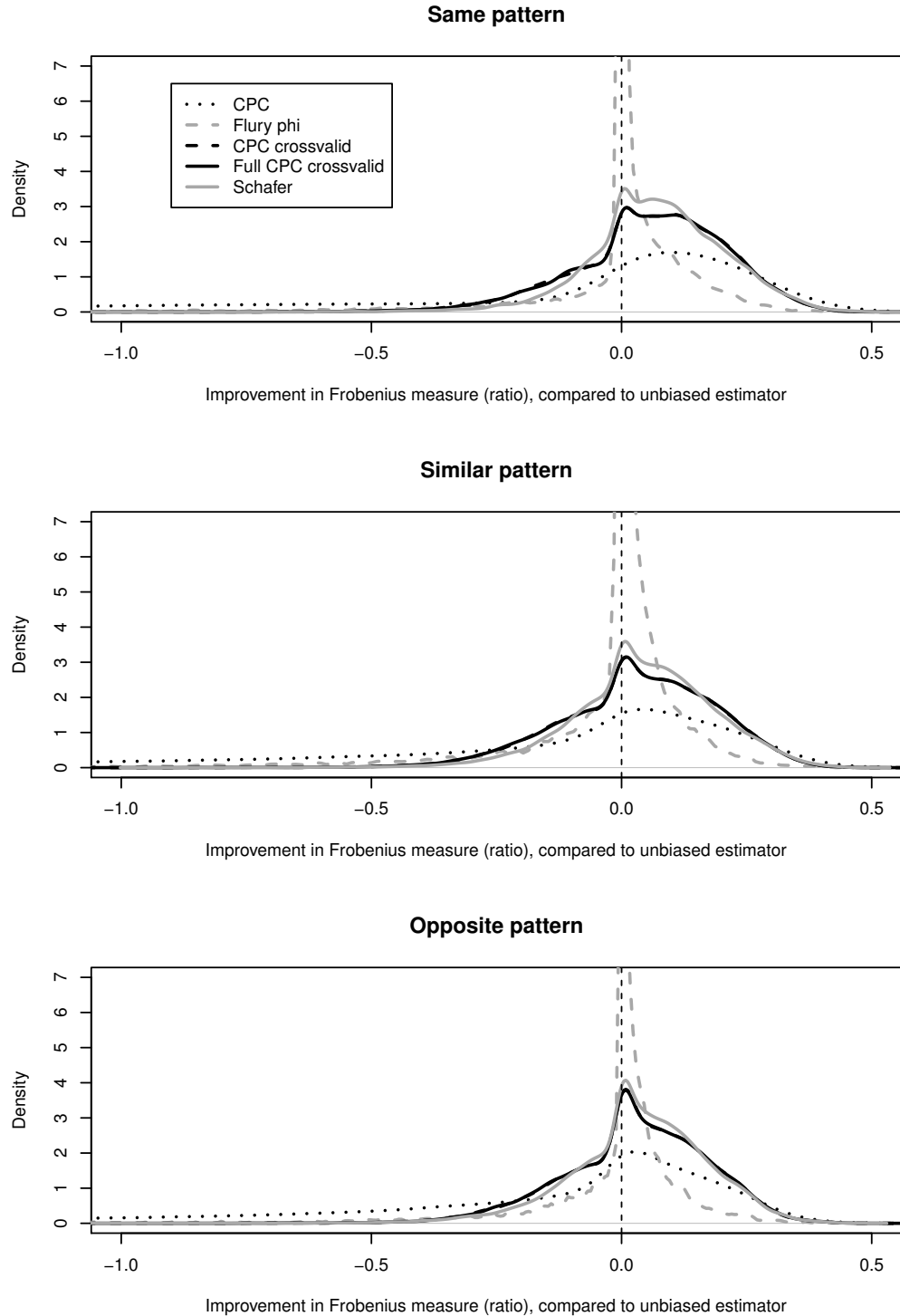


Figure 5.4: Effect of eigenvalue pattern on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

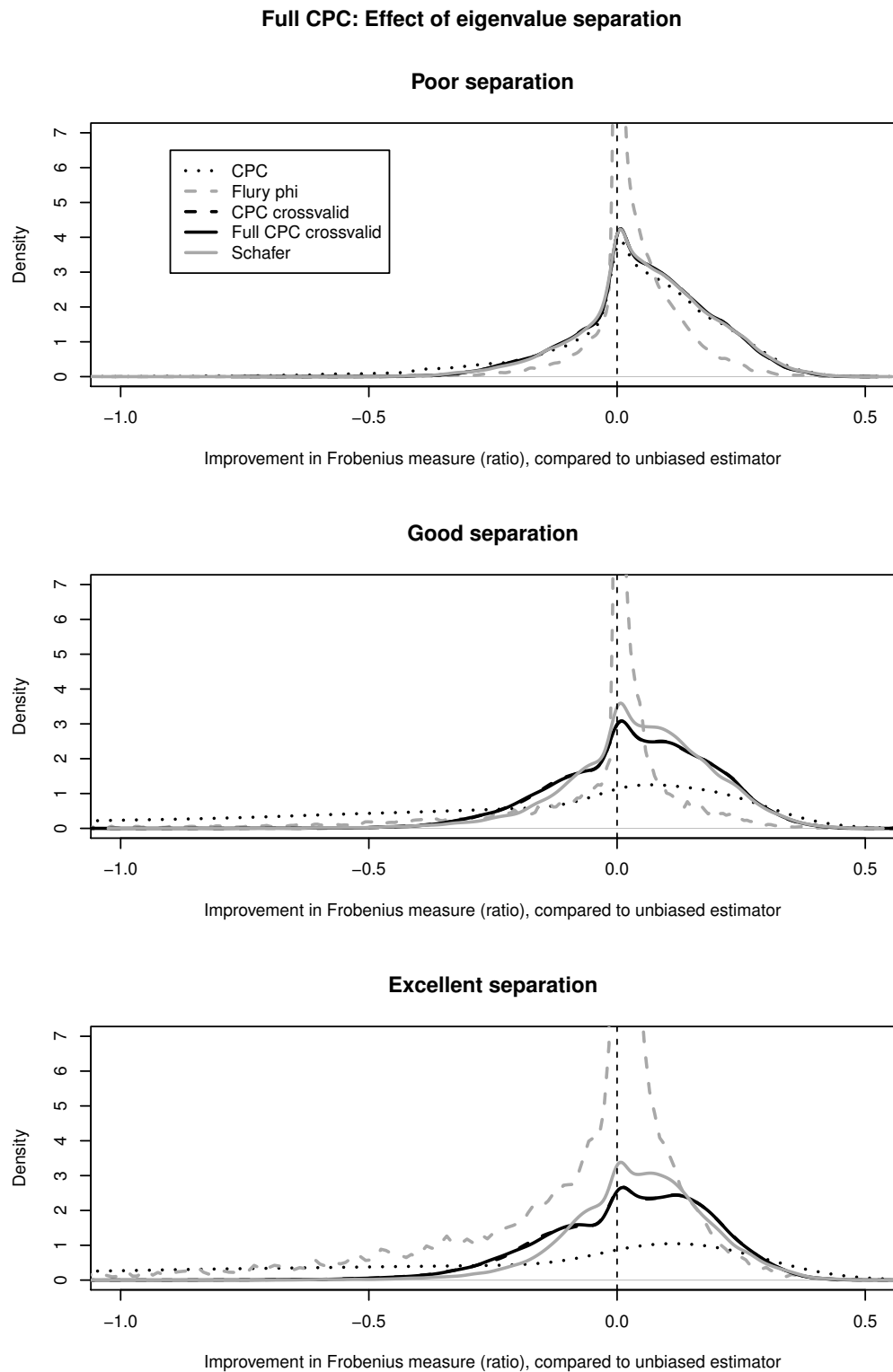


Figure 5.5: Effect of eigenvalue separation on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the full CPC case.

Table 5.3: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the full CPC crossvalidation estimator compared to each of the other covariance matrix estimators, in the full CPC case.

	Estimate	95% LCL	95% UCL	<i>p</i> -value
Full CPC cross vs. Unbiased	0.047	0.046	0.048	< 0.0001
Full CPC cross vs. CPC	0.149	0.147	0.151	< 0.0001
Full CPC cross vs. Phi	0.041	0.040	0.042	< 0.0001
Full CPC cross vs. CPC cross	-0.000	-0.000	0.000	0.3616
Full CPC cross vs. Schäfer	-0.005	-0.006	-0.005	< 0.0001
Full CPC cross vs. Pooled	0.469	0.466	0.473	< 0.0001

5.6.2 Half of eigenvectors common case

For the simulation runs where about half of the eigenvectors were common to both populations ($\text{CPC}(q)$ where $q \approx \frac{1}{2}p$), the mean and median standardised modified Frobenius values for the different covariance matrix estimators are shown in Table 5.4. *Full CPC crossvalid* clearly performs the best in this case, followed by *CPC crossvalid* and *Schäfer*. *Unbiased* still fares better than *CPC* and *Pooled*.

The ratio improvement in the modified Frobenius measure of each estimator, compared to *Unbiased*, can be seen in Figure 5.6. The distribution of *Full CPC crossvalid* lies the furthest to the right, showing that it offers the greatest improvement over *Unbiased* in this case.

Table 5.5 reports the results of two-sided Wilcoxon signed-rank tests of the modified Frobenius measure for each of the estimators compared to *Unbiased*. *CPC* (-7.3%) and *Pooled* (-151%) performs significantly worse than *Unbiased*. In the case where half of the eigenvectors are common, *Full CPC crossvalid* improves on the accuracy of *Unbiased* by 4.7% ($p < 0.0001$).

The effects of the sample sizes on the estimators are shown in Figure 5.7. The same trends can be seen as in the full CPC case: As the size of the sample from the first population increases, the performance of the CPC estimators (for the covariance matrix of the second population) improves compared to *Unbiased*. With an increase in size of the sample from the second population, the CPC estimators gradually lose their advantage over the unbiased covariance matrix estimator.

As in the full CPC case, the greatest improvement in covariance matrix estimation is obtained in the case where the populations have multivariate t distributions (see Figure 5.8). However, the performance of *CPC*, *CPC crossvalid* and *Schäfer* is decidedly worse with multivariate t data in this case (compared to the full CPC case), while *Full CPC crossvalid* is not

affected significantly.

As in the full CPC case, the rank orders of the common eigenvectors per group did not affect *CPC* and the shrinkage covariance matrix estimators seriously when half of the eigenvectors were common (see Figure 5.9). *CPC* and all of the shrinkage estimators only had a slight advantage in the *Same* pattern case, when compared to the *Similar* and *Opposite* patterns.

Increased separation between the eigenvalues per group led to slight improvements for the *CPC* and shrinkage covariance matrix estimators, compared to *Unbiased*, as can be seen in Figure 5.10. The profiles of the different estimators compared to each other remained unchanged, however, with *Full CPC crossvalid* performing the best in all three eigenvalue separation scenarios.

The results of two-sided Wilcoxon signed-rank tests of the ratio improvement in the modified Frobenius measure of *Full CPC crossvalid*, compared to each of the other covariance matrix estimators, are reported in Table 5.6. *Full CPC crossvalid* fared significantly better than both *CPC crossvalid* (2.3%, $p < 0.0001$) and *Schäfer* (2.1%, $p < 0.0001$) when half of the eigenvectors are common to both groups. This is a surprising result, as *Full CPC crossvalid* is calculated under the assumption of full CPC, while the latter two methods use the more appropriate \mathbf{B}_2 matrix to estimate Σ_2 . It seems that the added structure afforded by the full common eigenvector matrix, \mathbf{B} , outweighs the reduced bias of the \mathbf{B}_2 matrix, leading to more accurate estimates of the population covariance matrices in general. In the full CPC situation, *Full CPC crossvalid*, *CPC crossvalid* and *Schäfer* perform equally well, while in the partial CPC case, *Full CPC crossvalid* outperforms the other two estimators. An advantage of the *Full CPC crossvalid* estimator is thus that it may be used in situations where common eigenvectors are suspected, but the exact number of common eigenvectors is unknown.

Table 5.4: Mean and median standardised modified Frobenius values for the different covariance matrix estimators in the case when half of the eigenvectors are common.

	Mean	Median
Unbiased	0.271	0.036
CPC	0.337	0.197
Flury phi	0.253	0.054
CPC crossvalid	0.231	0.060
Full CPC crossvalid	0.194	0.027
Schäfer	0.233	0.056
Pooled	0.789	1.000

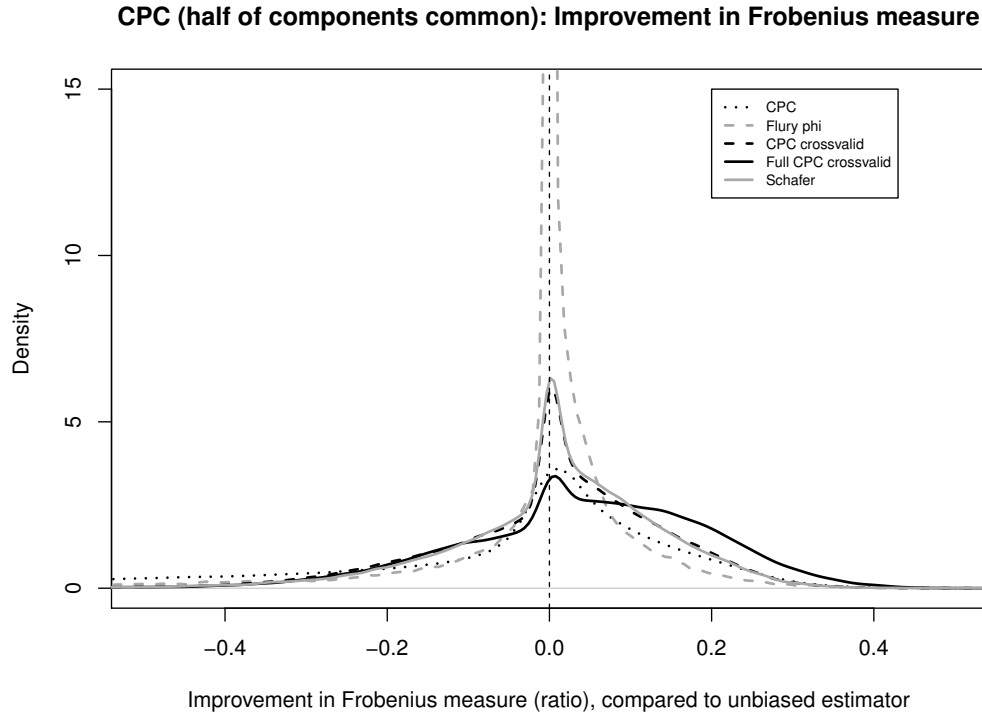


Figure 5.6: Ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

Table 5.5: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the covariance matrix estimators, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

	Estimate	95% LCL	95% UCL	p -value
CPC	-0.073	-0.076	-0.070	< 0.0001
Flury phi	0.002	0.002	0.003	< 0.0001
CPC crossvalid	0.013	0.012	0.014	< 0.0001
Full CPC crossvalid	0.047	0.046	0.048	< 0.0001
Schäfer	0.017	0.016	0.018	< 0.0001
Pooled	-1.516	-1.534	-1.498	< 0.0001

CPC (half of components common): Effect of sample sizes

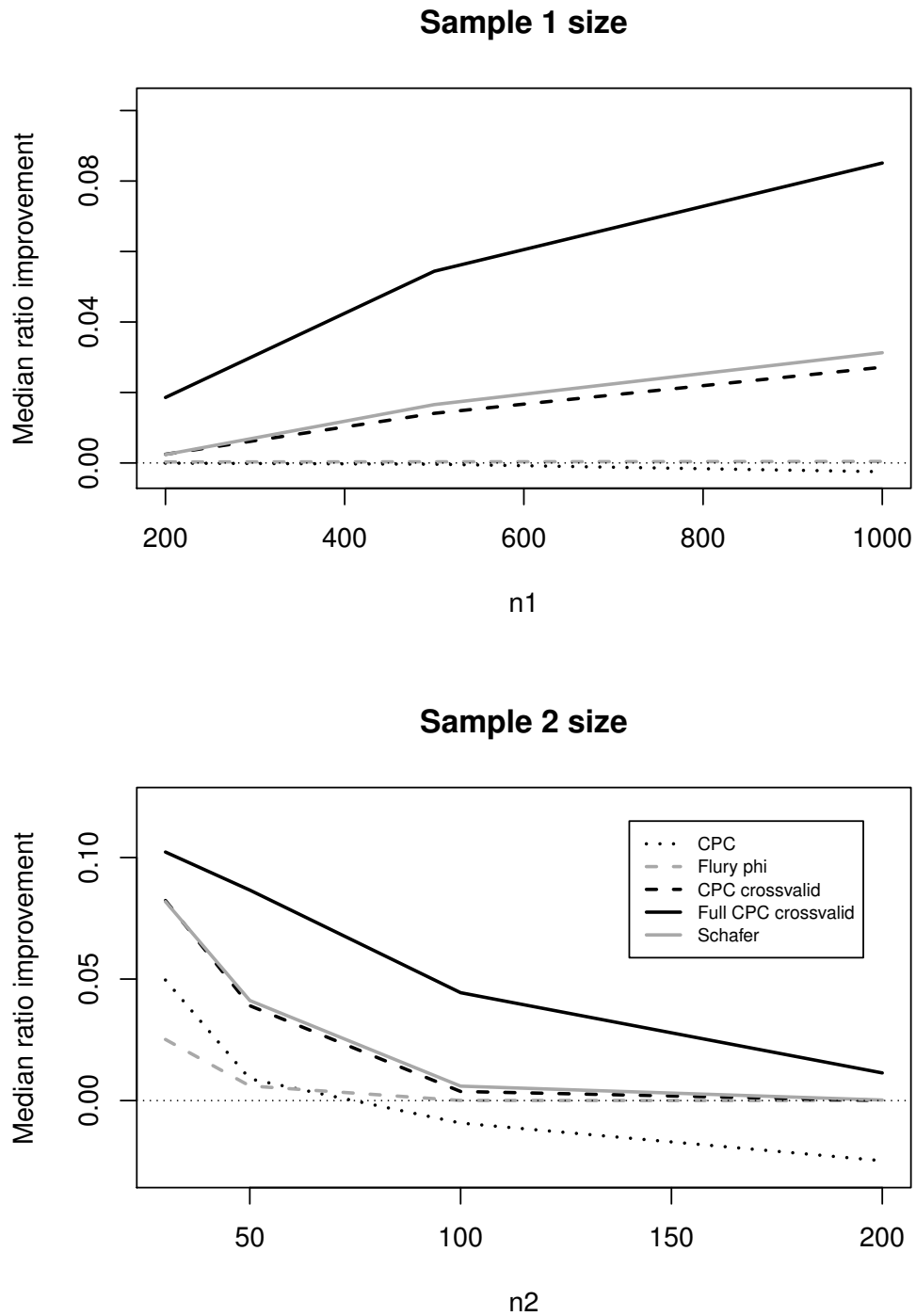


Figure 5.7: Effect of sample size on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

CPC (half of components common): Effect of data type

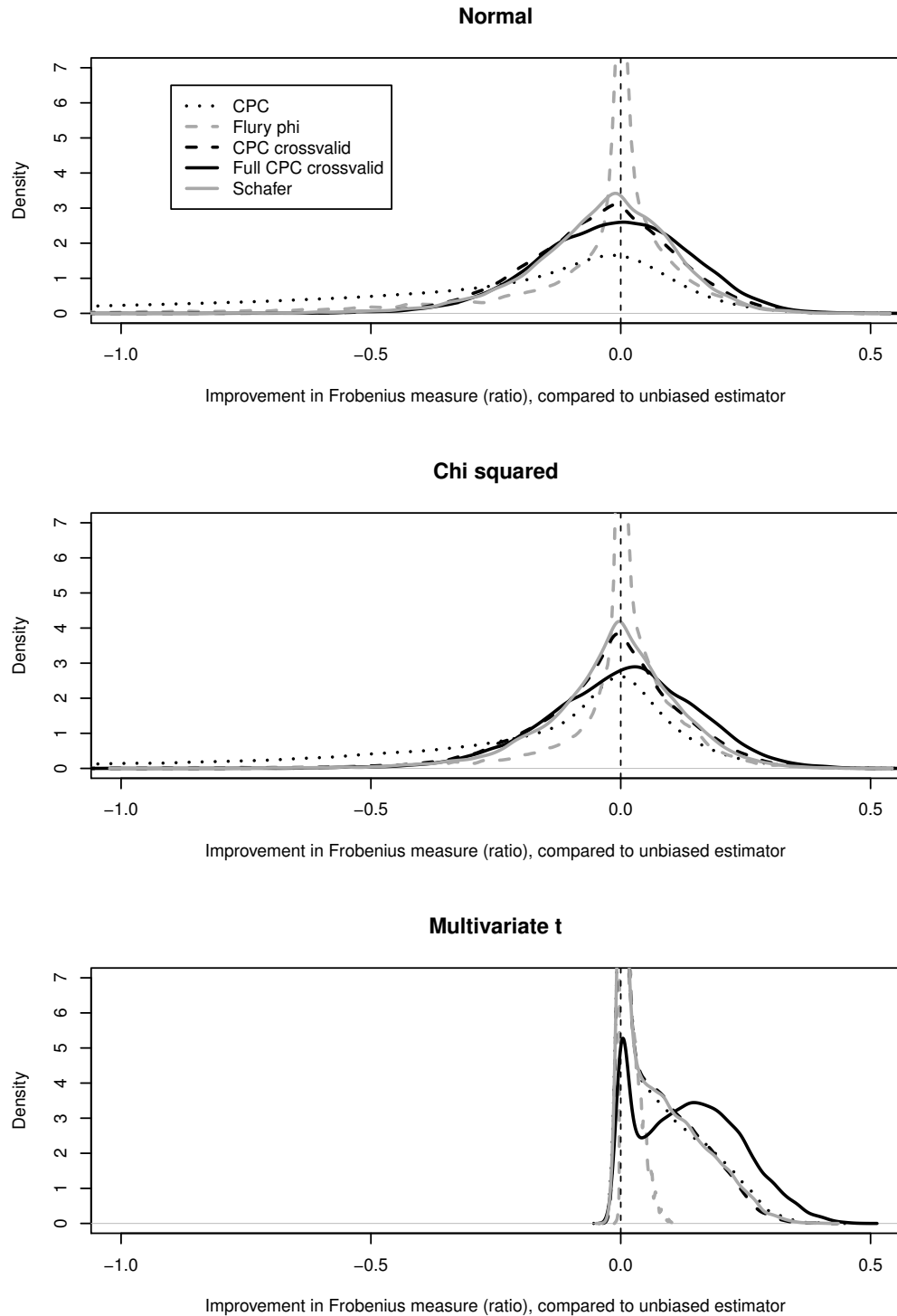


Figure 5.8: Effect of multivariate distribution type on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

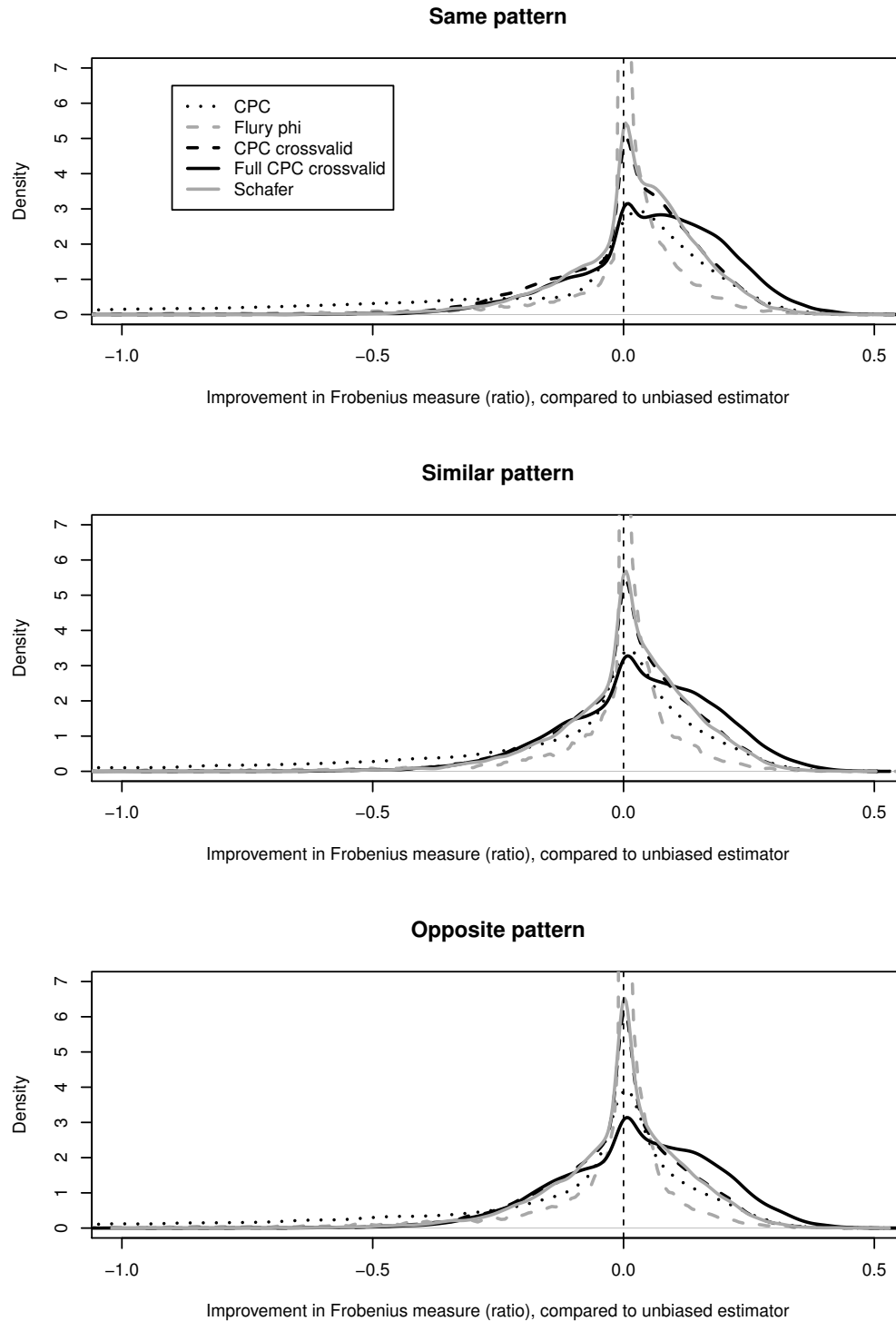
CPC (half of components common): Effect of eigenvalue pattern

Figure 5.9: Effect of eigenvalue pattern on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

CPC (half of components common): Effect of eigenvalue separation

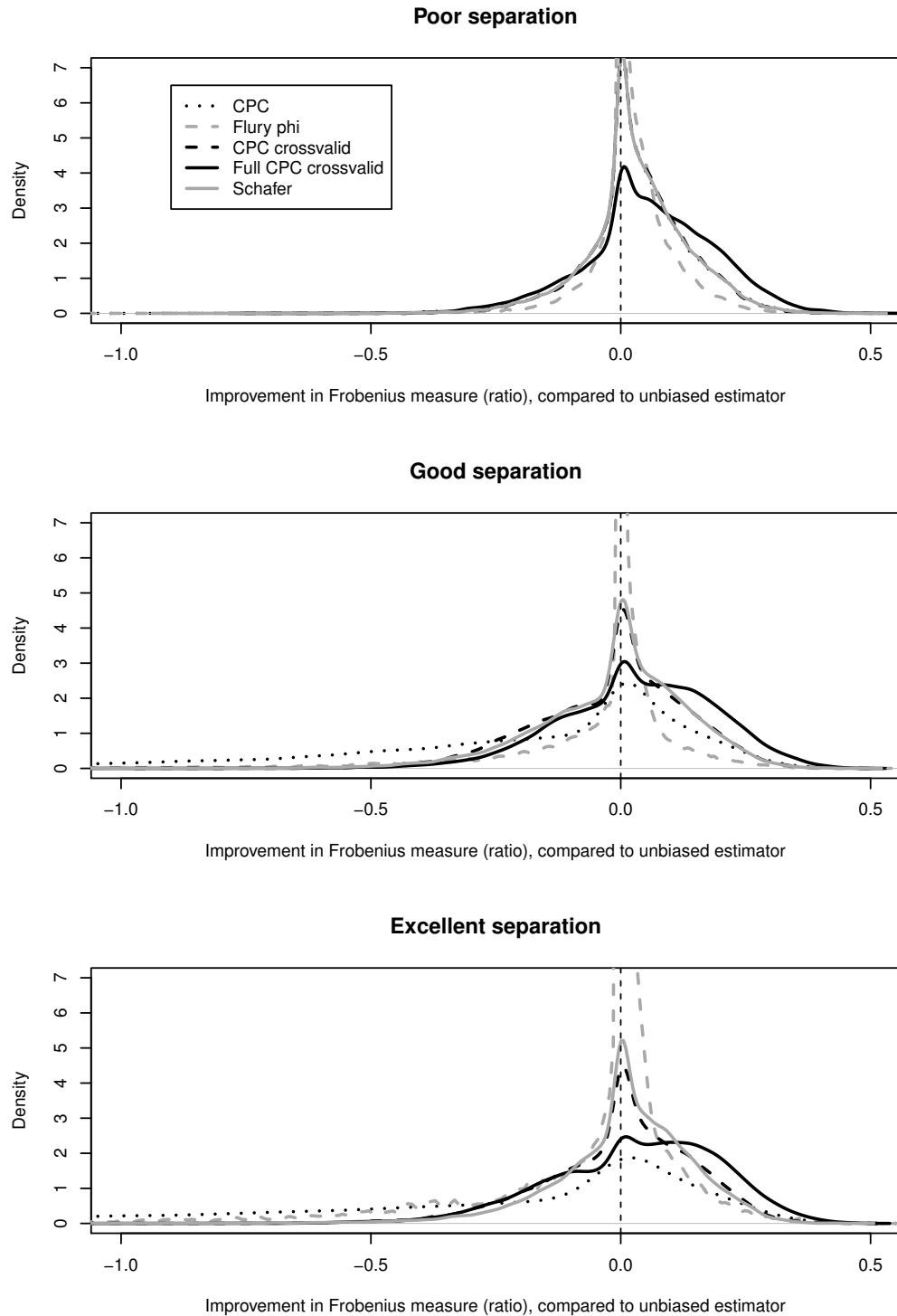


Figure 5.10: Effect of eigenvalue separation on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the case when half of the eigenvectors are common.

Table 5.6: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the full CPC crossvalidation estimator, compared to each of the other covariance matrix estimators, in the case when half of the eigenvectors are common.

	Estimate	95% LCL	95% UCL	<i>p</i> -value
Full CPC cross vs. Unbiased	0.047	0.046	0.048	< 0.0001
Full CPC cross vs. CPC	0.110	0.108	0.111	< 0.0001
Full CPC cross vs. Phi	0.049	0.048	0.050	< 0.0001
Full CPC cross vs. CPC cross	0.023	0.023	0.024	< 0.0001
Full CPC cross vs. Schäfer	0.021	0.021	0.022	< 0.0001
Full CPC cross vs. Pooled	0.466	0.462	0.470	< 0.0001

5.6.3 Few common eigenvectors case

The mean and median standardised modified Frobenius values for the covariance matrix estimators when few of the eigenvectors are common to both groups ($CPC(q)$, where q is small) are shown in Table 5.7. In this case, *Full CPC crossvalid* still clearly outperforms the other estimators, including *CPC crossvalid* and *Schäfer*. This result can also be seen in Figure 5.11, which shows the ratio improvement in the modified Frobenius measure of each of the covariance matrix estimators compared to *Unbiased*.

From Table 5.8 it can be seen that *CPC crossvalid* does not fare better than *Unbiased* when only few of the eigenvectors are common ($p = 0.6983$), and *CPC* performs significantly worse than *Unbiased* (-5.2%, $p < 0.0001$). *Flury phi* and *Schäfer* perform significantly better than *Unbiased*, but the difference in estimation accuracy is very small (0.1%) and therefore of little practical significance.

The effects of the sample sizes on the covariance matrix estimators are shown in Figure 5.12. Only the *Full CPC crossvalid* estimator showed a large increase in accuracy compared to *Unbiased* when the size of the sample from the first population was increased.

Figure 5.13 shows that the largest improvement over *Unbiased* is obtained by *Full CPC crossvalid* when the populations have multivariate t distributions (i.e. for thick-tailed marginal distributions).

Changes in the eigenvalue patterns and separation between the eigenvalues per group has little effect in the case when only a few of the eigenvectors are common, as can be seen in Figures 5.14 and 5.15, respectively. These two figures show the same trends that have been observed in the full CPC case and the case where half of the eigenvectors are common to both population covariance matrices.

Two-sided Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of *Full CPC crossvalid* compared to each of the other covariance matrix estimators were performed, with the results shown in Table 5.9. *Full CPC crossvalid* performs significantly better than any of the other covariance matrix estimators, offering an improvement of 3.9% ($p < 0.0001$) over *Unbiased* when only few of the eigenvectors are common.

Table 5.7: Mean and median standardised modified Frobenius values for the different covariance matrix estimators in the few common eigenvectors case.

	Mean	Median
Unbiased	0.262	0.032
CPC	0.318	0.163
Flury phi	0.256	0.056
CPC crossvalid	0.245	0.067
Full CPC crossvalid	0.196	0.032
Schäfer	0.248	0.065
Pooled	0.794	1.000

Table 5.8: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the covariance matrix estimators, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

	Estimate	95% LCL	95% UCL	p -value
CPC	-0.052	-0.054	-0.050	< 0.0001
Flury phi	0.001	0.000	0.001	< 0.0001
CPC crossvalid	-0.000	-0.000	0.000	0.6983
Full CPC crossvalid	0.039	0.038	0.040	< 0.0001
Schäfer	0.001	0.000	0.001	< 0.0001
Pooled	-1.534	-1.552	-1.516	< 0.0001

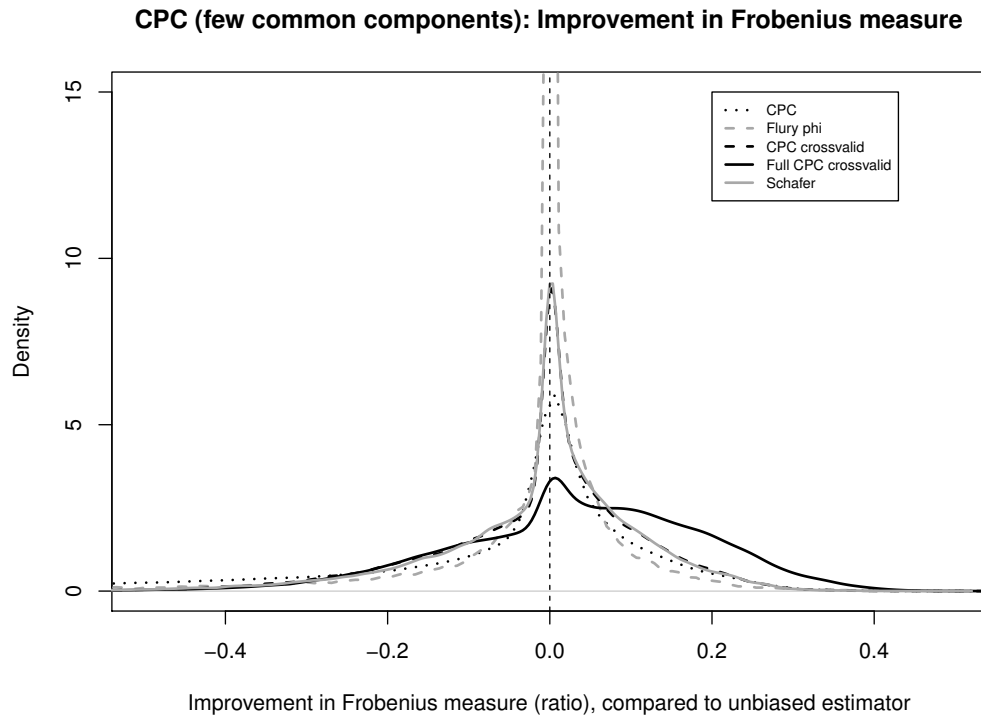


Figure 5.11: Ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

Table 5.9: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the full CPC crossvalidation estimator, compared to each of the other covariance matrix estimators, for the few common eigenvectors case.

	Estimate	95% LCL	95% UCL	p -value
Full CPC cross vs. Unbiased	0.039	0.038	0.040	< 0.0001
Full CPC cross vs. CPC	0.101	0.100	0.103	< 0.0001
Full CPC cross vs. Phi	0.049	0.048	0.050	< 0.0001
Full CPC cross vs. CPC cross	0.039	0.038	0.040	< 0.0001
Full CPC cross vs. Schäfer	0.037	0.036	0.038	< 0.0001
Full CPC cross vs. Pooled	0.468	0.464	0.471	< 0.0001

CPC (few common components): Effect of sample sizes

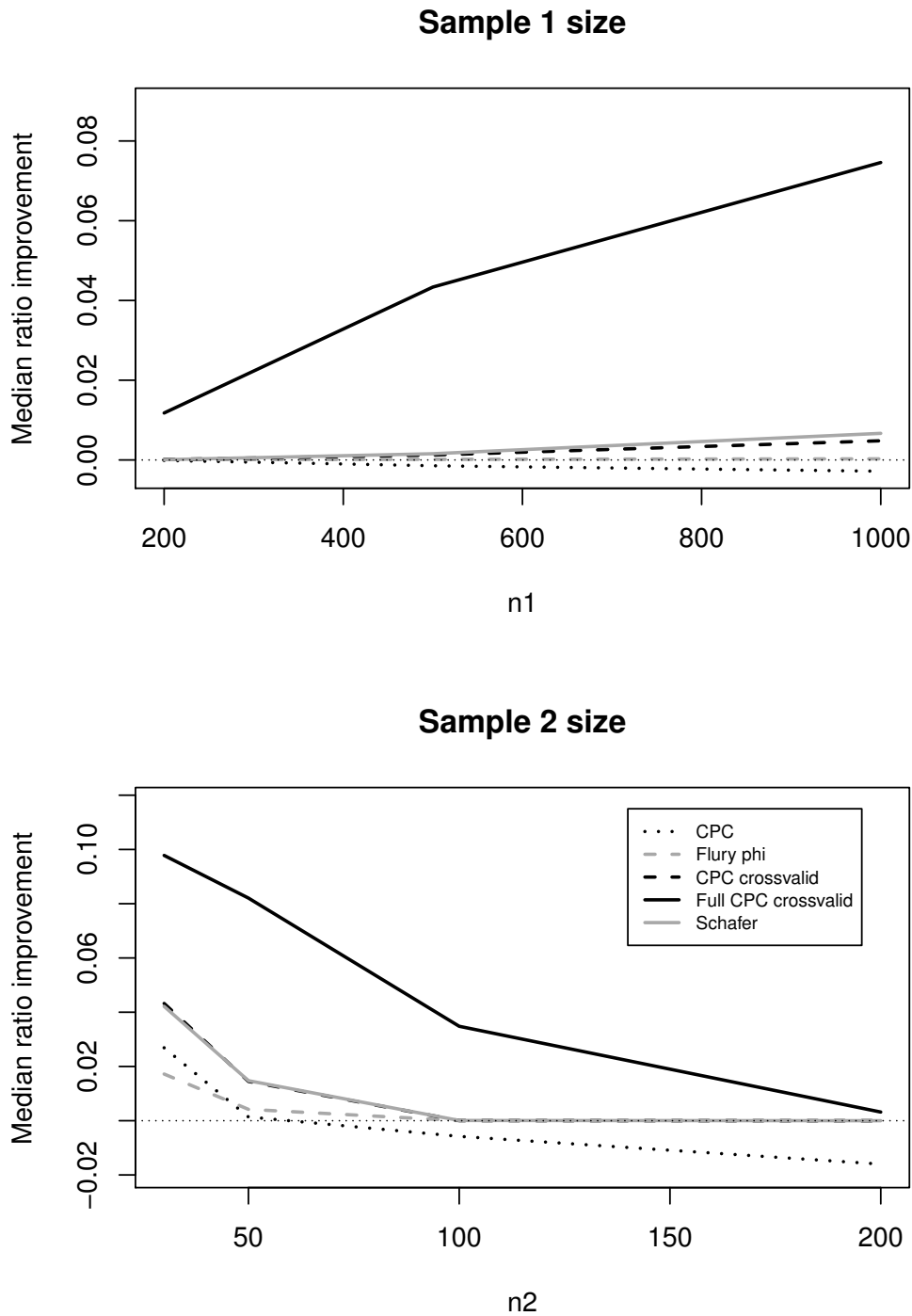


Figure 5.12: Effect of sample size on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

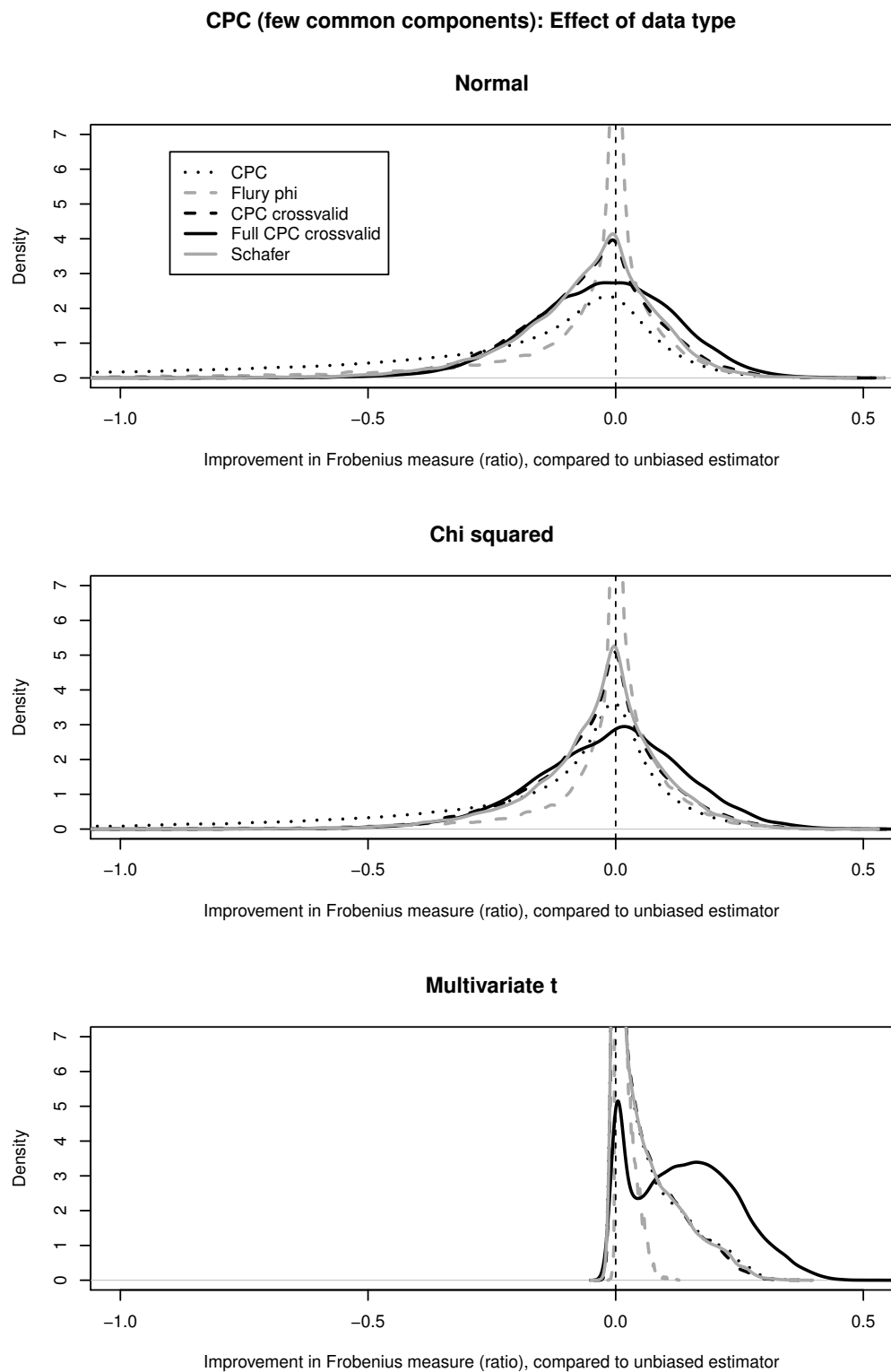


Figure 5.13: Effect of multivariate distribution type on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

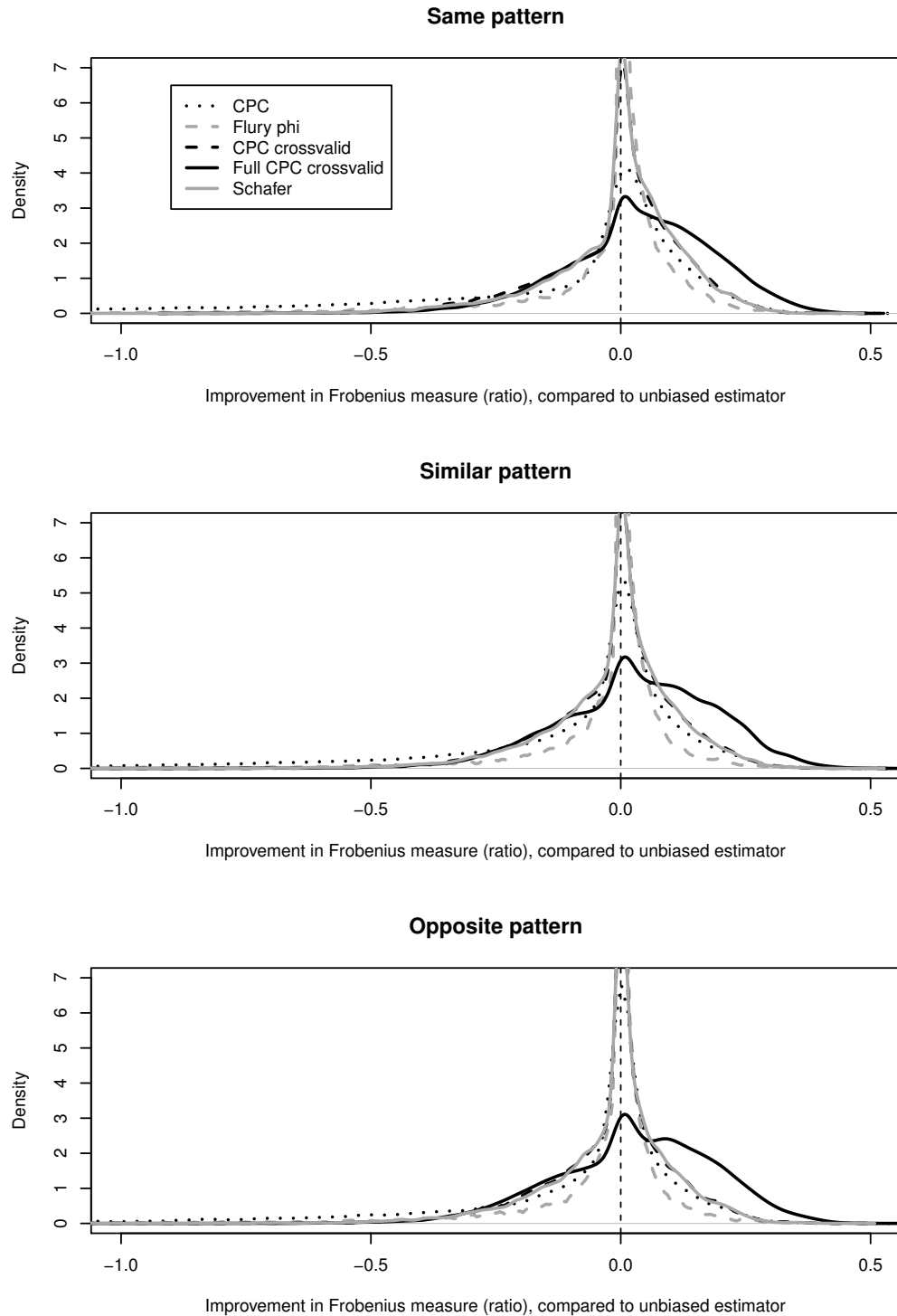
CPC (few common components): Effect of eigenvalue pattern

Figure 5.14: Effect of eigenvalue pattern on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

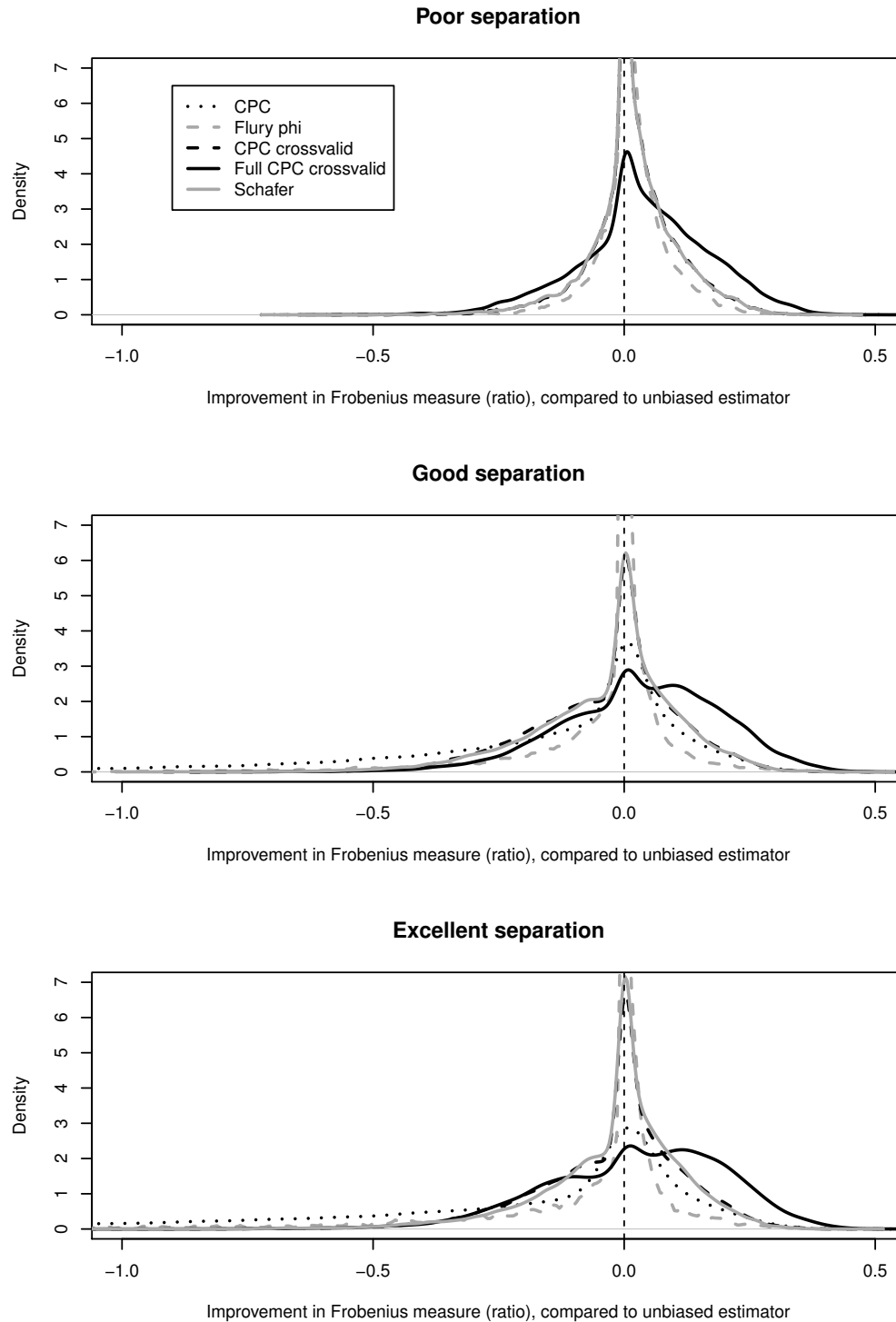
CPC (few common components): Effect of eigenvalue separation

Figure 5.15: Effect of eigenvalue separation on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the few common eigenvectors case.

5.6.4 Unrelated covariance matrices case

When the two population covariance matrices are unrelated, only *Unbiased* computes the covariance matrices under the correct assumption. From the mean standardised modified Frobenius values reported in Table 5.10, however, it appears that *Full CPC crossvalid* still outperforms all of the other estimators (including *Unbiased*) in this case. The good performance of *Full CPC crossvalid* can also be seen in Figure 5.16, where the bulk of the distribution of *Full CPC crossvalid* lies to the right of zero.

Results of two-sided Wilcoxon signed-rank tests of the ratio improvement in the modified Frobenius measure of each of the covariance matrix estimators, compared to *Unbiased*, are reported in Table 5.11. Clearly none but *Full CPC crossvalid* is more accurate than *Unbiased* in this case. *Full CPC crossvalid* offers an improvement of 3.5% ($p < 0.0001$) over *Unbiased* when the population covariance matrices are unrelated.

Figure 5.17 shows the effects of sample size on the covariance matrix estimators when the two population covariance matrices are unrelated. In this case, it seems that only the *Full CPC crossvalid* estimator offers any improvement on the accuracy of the unbiased covariance matrix estimator. The same trends can be observed for *Full CPC crossvalid* as in the CPC cases: Increasing the size of the sample from the first population improves its accuracy in the estimation of the covariance matrix of the second population, while an increase in the sample size from the second population decreases its accuracy to the point where it is equivalent to that of the unbiased estimator.

For unrelated population covariance matrices, the largest improvement in the modified Frobenius measure is obtained by *Full CPC crossvalid* when the populations have multivariate t distributions (see Figure 5.18). Figures 5.19 and 5.20 show that changes in the eigenvalue patterns and separation between the eigenvalues per group did not affect the covariance matrix estimators much, as was also seen in the full CPC and partial CPC cases considered.

Full CPC crossvalid outperforms both *CPC crossvalid* (5%, $p < 0.0001$) and *Schäfer* (5.2%, $p < 0.0001$) in the unrelated covariance matrices case, as can be seen from Table 5.12 where the results of two-sided Wilcoxon signed-rank tests of the ratio improvement in the modified Frobenius measure of *Full CPC crossvalid*, compared to each of the other covariance matrix estimators, are reported.

Table 5.10: Mean and median standardised modified Frobenius values for the different covariance matrix estimators in the unrelated covariance matrices case.

	Mean	Median
Unbiased	0.259	0.027
CPC	0.294	0.104
Flury phi	0.263	0.054
CPC crossvalid	0.261	0.065
Full CPC crossvalid	0.195	0.029
Schäfer	0.264	0.066
Pooled	0.798	1.000

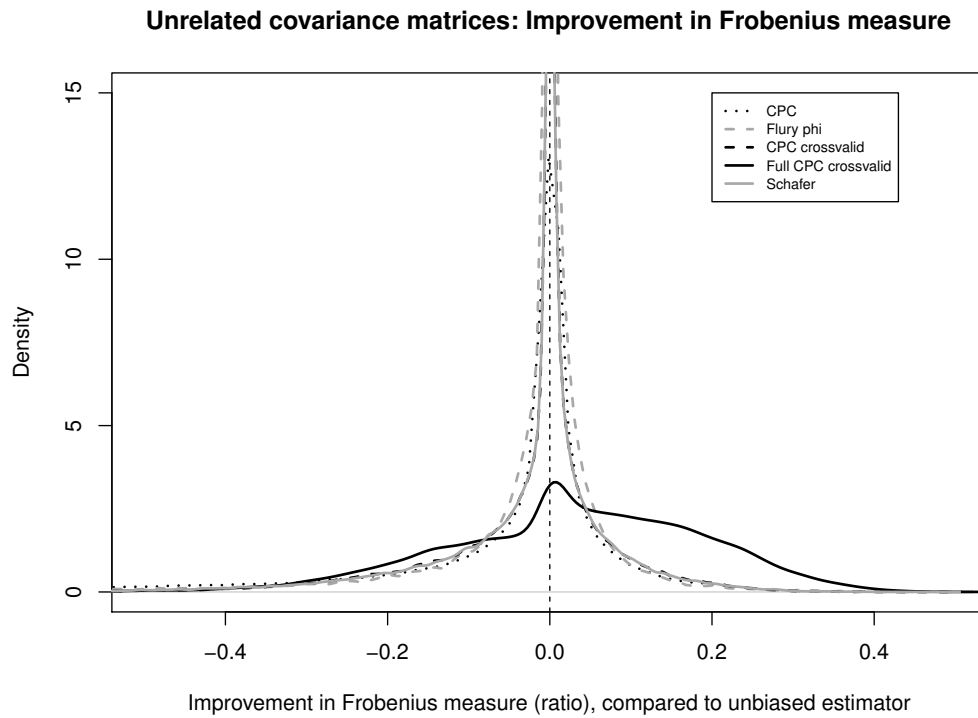


Figure 5.16: Ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

Table 5.11: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the covariance matrix estimators, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

	Estimate	95% LCL	95% UCL	p -value
CPC	-0.01873	-0.01959	-0.01791	< 0.0001
Flury phi	-0.00016	-0.00031	-0.00009	< 0.0001
CPC crossvalid	-0.00617	-0.00668	-0.00568	< 0.0001
Full CPC crossvalid	0.03495	0.03377	0.03614	< 0.0001
Schäfer	-0.00643	-0.00694	-0.00595	< 0.0001
Pooled	-1.51827	-1.53658	-1.50026	< 0.0001

Table 5.12: Wilcoxon signed-rank tests for the ratio improvement in the modified Frobenius measure of the full CPC crossvalidation estimator, compared to each of the other covariance matrix estimators, for the unrelated covariance matrices case.

	Estimate	95% LCL	95% UCL	p -value
Full CPC cross vs. Unbiased	0.035	0.034	0.036	< 0.0001
Full CPC cross vs. CPC	0.081	0.080	0.082	< 0.0001
Full CPC cross vs. Phi	0.049	0.047	0.050	< 0.0001
Full CPC cross vs. CPC cross	0.050	0.049	0.051	< 0.0001
Full CPC cross vs. Schäfer	0.052	0.051	0.053	< 0.0001
Full CPC cross vs. Pooled	0.466	0.462	0.469	< 0.0001

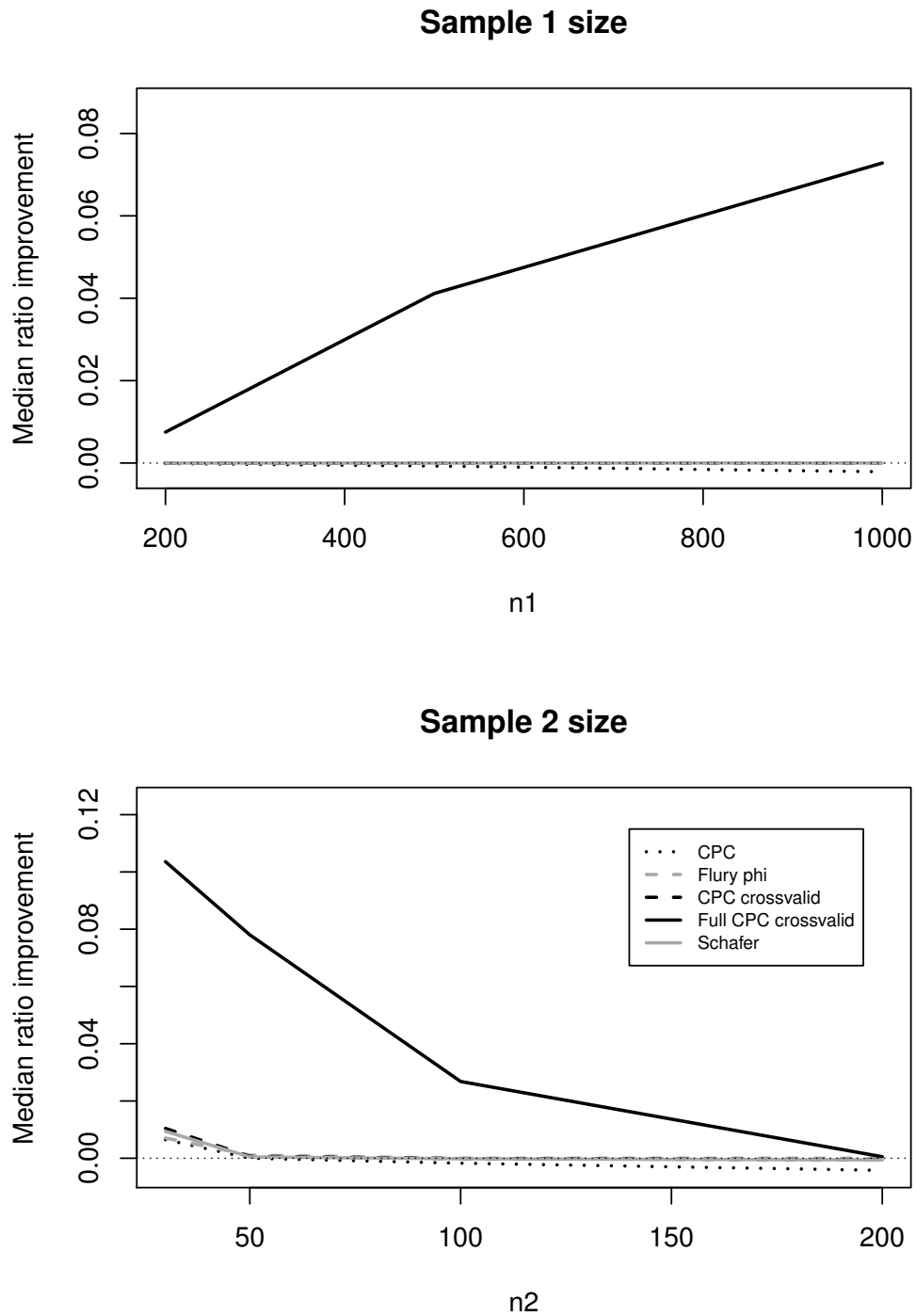
Unrelated covariance matrices: Effect of sample sizes

Figure 5.17: Effect of sample size on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

Unrelated covariance matrices: Effect of data type

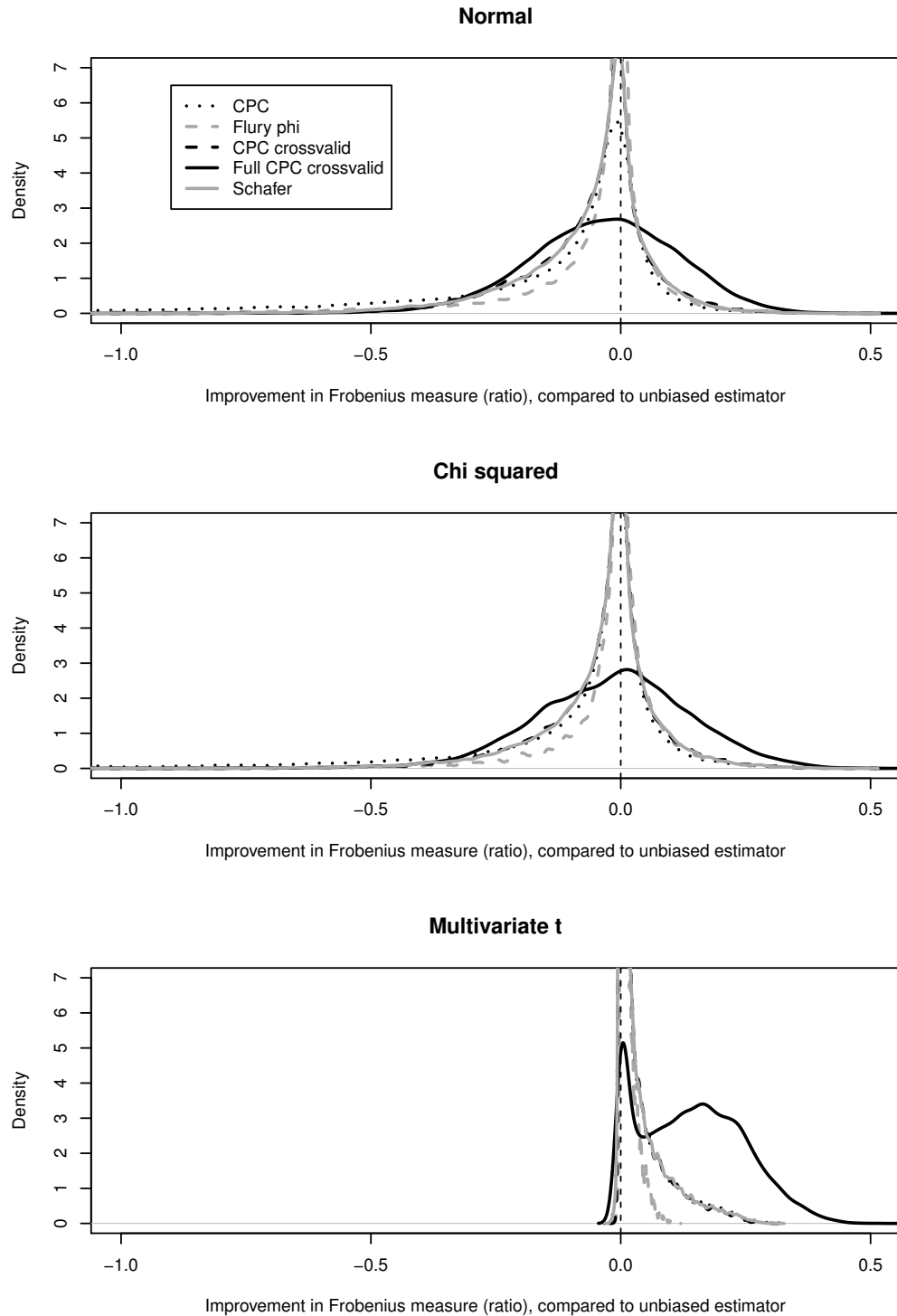


Figure 5.18: Effect of multivariate distribution type on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

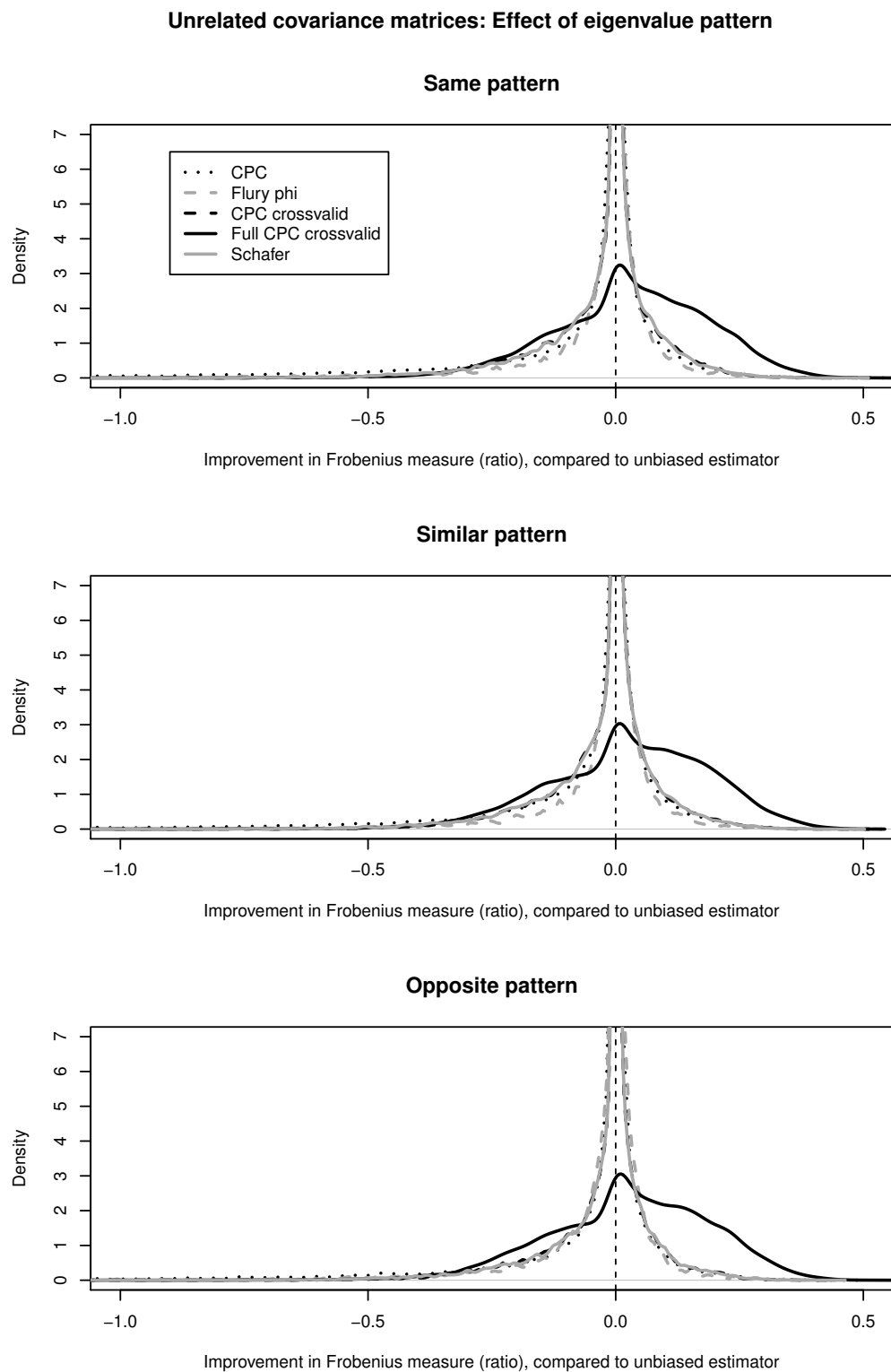


Figure 5.19: Effect of eigenvalue pattern on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

Unrelated covariance matrices: Effect of eigenvalue separation

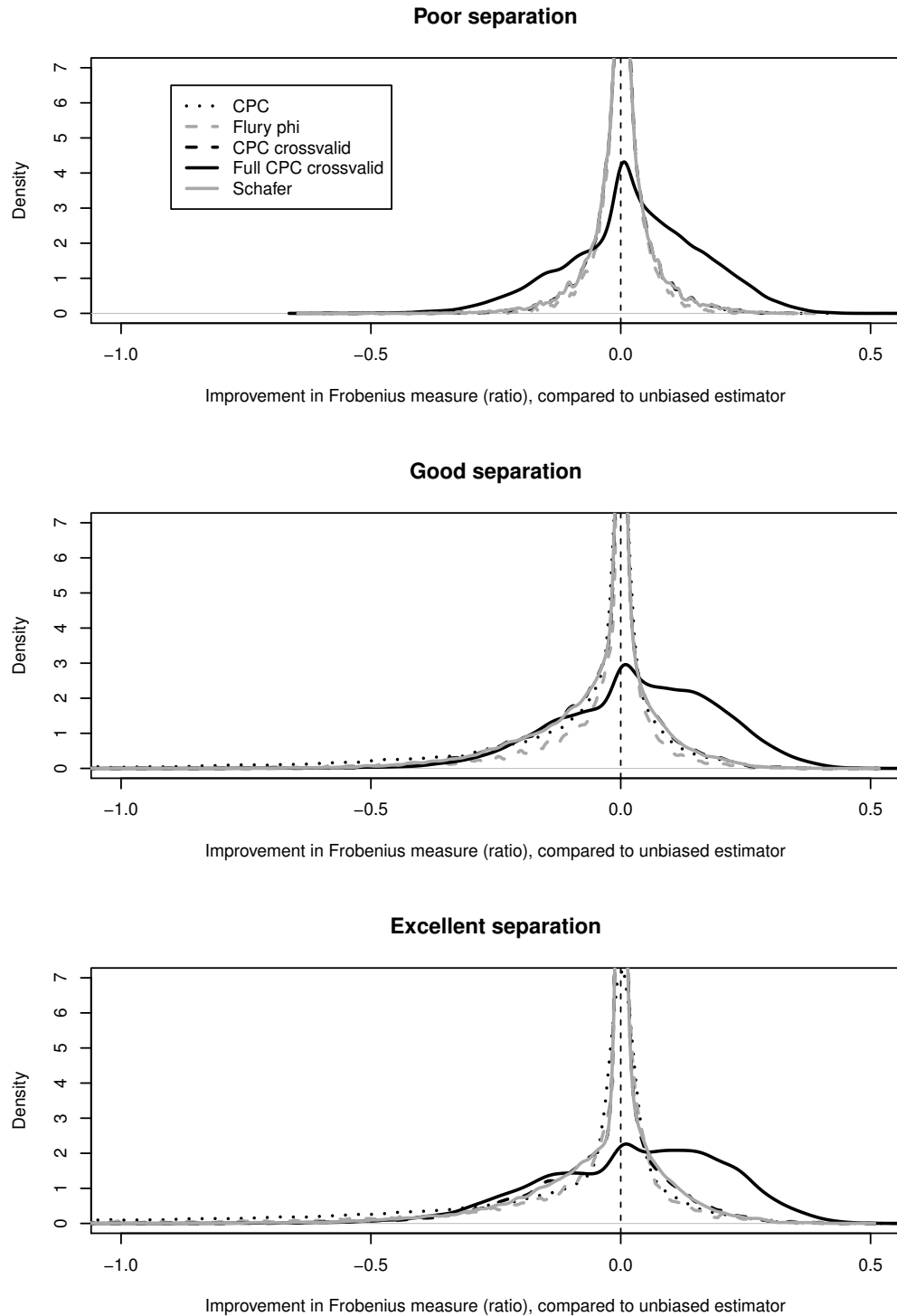


Figure 5.20: Effect of eigenvalue separation on the ratio improvement in the modified Frobenius measure, compared to the unbiased sample covariance matrix estimator, for the unrelated covariance matrices case.

5.6.5 Effect of correlations among the variables

It was surmised that the correlations among the variables per group may also affect the accuracy of the covariance matrix estimators, so a small follow-up Monte Carlo simulation study was performed for $k = 2$ populations with $p = 5, 10$ and 20 variables and full CPC covariance structures (i.e. $q = p$). The effect of correlations under the partial CPC scenario was not considered as no way could be found to simulate data from populations with partial common eigenvectors for specified correlation structures.

Sample sizes for the two groups were held constant at $n_1 = 1000$ and $n_2 = 100$. The separation between subsequent eigenvalues were 40% for the first group and 60% for the second group. The eigenvalues per group were thus well separated.

Data were simulated from multivariate normal, multivariate chi-squared with two degrees of freedom, and multivariate t with one degree of freedom distributions. The population covariance matrices were specified in such a way that there were either strong correlations between the variables, or no correlation (i.e. orthogonal variables). The results from this second set of simulations (1800 runs in total) to investigate the effect of correlations among the variables on the estimation of Σ_2 (the population covariance matrix of the second group, from which a smaller sample is available) are given in Table 5.13.

Strong correlations among the variables leads to an improvement, compared to *Unbiased*, in the estimation of the covariance matrices of normally distributed populations and populations with marginal chi-squared distributions. However, in multivariate t distributed populations, the improvement of the estimators, compared to *Unbiased*, is larger when there are no correlations between the variables than when the correlations are strong.

Thus correlations between the variables do appear to affect the improvement in covariance matrix estimation, and this topic may be worthy of further investigation.

5.7 Application to the VON data

The *CPC* and *Full CPC crossvalid* covariance matrix estimators proposed in this chapter were applied to the VON 2009 cohort by using information about common eigenvectors in the population covariance matrices. In the *Caesarean* ($n_1 = 2387$) and *Vaginal* ($n_2 = 439$) groups, there seem to be three common eigenvectors. In the *South Africa* ($n_1 = 2713$) and *Namibia* ($n_2 = 113$) groups, there seem to be six common eigenvectors. For both

Table 5.13: The ratio improvement in the modified Frobenius measure of the covariance matrix estimators, compared to the unbiased sample covariance matrix estimator, for multivariate populations with no correlations, or strong correlations, between the variables. Confidence intervals are given in brackets.

Distribution	No correlation		Strong correlation	
	Estimate	95% C.I.	Estimate	95% C.I.
Normal				
CPC	0.209	(0.194; 0.228)	0.301	(0.284; 0.321)
Flury phi	0.149	(0.133; 0.166)	0.213	(0.197; 0.232)
CPC crossvalid	0.203	(0.187; 0.219)	0.282	(0.267; 0.297)
Full CPC crossvalid	0.202	(0.187; 0.219)	0.282	(0.267; 0.298)
Schäfer	0.206	(0.190; 0.224)	0.292	(0.277; 0.310)
Pooled	-2.298	(-2.448; -2.158)	-1.935	(-2.081; -1.789)
Chi-squared (2 df)				
CPC	0.120	(0.107; 0.133)	0.264	(0.247; 0.281)
Flury phi	0.077	(0.067; 0.090)	0.166	(0.149; 0.184)
CPC crossvalid	0.115	(0.104; 0.125)	0.244	(0.230; 0.258)
Full CPC crossvalid	0.114	(0.103; 0.125)	0.241	(0.228; 0.255)
Schäfer	0.115	(0.104; 0.128)	0.248	(0.232; 0.263)
Pooled	-1.364	(-1.542; -1.192)	-1.203	(-1.345; -1.081)
Multivariate t (1 df)				
CPC	0.201	(0.189; 0.212)	0.179	(0.166; 0.193)
Flury phi	0.002	(0.000; 0.003)	0.000	(0.000; 0.003)
CPC crossvalid	0.174	(0.163; 0.184)	0.167	(0.154; 0.178)
Full CPC crossvalid	0.174	(0.162; 0.182)	0.167	(0.154; 0.179)
Schäfer	0.182	(0.170; 0.193)	0.172	(0.160; 0.185)
Pooled	-8.065	(-11.231; -6.218)	-9.362	(-15.622; -6.453)

groupings, the eigenvalue patterns in the groups are the same. The VON data are multivariate non-normal and contains six numerical variables with strong correlations among them, and the *Full CPC crossvalid* estimator should thus provide more accurate estimates of the population covariance matrices of the delivery mode groups and regional groups, respectively, compared to the *Unbiased* covariance matrix estimator.

The *Full CPC crossvalid* estimator is particularly useful in the estimation of the covariance matrix of the *Namibia* region, as the sample from this region is relatively small ($n = 113$) and a large gain in accuracy can be obtained by using the knowledge of the six eigenvectors the population covariance matrix of this region share with the population covariance matrix of the *South Africa* region.

5.7.1 Delivery mode

In Chapter 4 it was found that population covariance matrices of the *Caesarean* ($n_1 = 2549$) and *Vaginal* ($n_2 = 492$) delivery mode groups in the VON 2009 cohort share three common eigenvectors, accounting for more than 95% of the observed variation in each of the groups. Knowledge of the three common eigenvectors can be used to find estimates of the population covariance matrices of the two groups under the CPC model.

Using the *CPC* estimator in (5.5), the off-diagonal elements of the \mathbf{L}_i matrices are shrunk to zero, which, combined with the common eigenvector matrix, \mathbf{B} , yield the following estimates for the covariance matrices of the two delivery mode groups under the CPC model:

- *Caesarean*:

$$\begin{aligned} \mathbf{S}_{1(\text{CPC})} &= \mathbf{B}\mathbf{L}_1^0\mathbf{B}' \\ &= \begin{bmatrix} 0.637 & 0.318 & 0.255 & 2.243 & 2.137 & 0.170 \\ 0.318 & 3.371 & 2.112 & 1.430 & 1.309 & 0.247 \\ 0.255 & 2.112 & 2.262 & 1.261 & 1.031 & 0.198 \\ 2.243 & 1.430 & 1.261 & 11.663 & 8.752 & 0.670 \\ 2.137 & 1.309 & 1.031 & 8.752 & 10.110 & 0.637 \\ 0.170 & 0.247 & 0.198 & 0.670 & 0.637 & 0.506 \end{bmatrix} \end{aligned}$$

- *Vaginal*:

$$\begin{aligned} \mathbf{S}_{2(\text{CPC})} &= \mathbf{B}\mathbf{L}_2^0\mathbf{B}' \\ &= \begin{bmatrix} 0.876 & 0.449 & 0.366 & 3.335 & 3.141 & 0.260 \\ 0.449 & 5.401 & 3.704 & 2.057 & 1.815 & 0.391 \\ 0.366 & 3.704 & 3.448 & 1.796 & 1.461 & 0.307 \\ 3.335 & 2.057 & 1.796 & 16.768 & 13.391 & 0.990 \\ 3.141 & 1.815 & 1.461 & 13.391 & 14.415 & 0.920 \\ 0.260 & 0.391 & 0.307 & 0.990 & 0.920 & 0.840 \end{bmatrix}. \end{aligned}$$

Using the crossvalidation method described in Section 5.5.2, the shrinkage intensity parameters for the delivery mode groups were estimated as

- *Caesarean*

$$\hat{\alpha}_1 = 0.4031,$$

and

- *Vaginal*

$$\hat{\alpha}_2 = 0.7061.$$

Finally, plugging the shrinkage intensity parameters into (5.10) yields the following estimates of the population covariance matrices (with the *Full CPC crossvalid* estimator):

- *Caesarean*:

$$\begin{aligned} \mathbf{S}_{1(\text{CPC})}^* &= \hat{\alpha}_1\mathbf{S}_1 + (1 - \hat{\alpha}_1)\mathbf{S}_{1(\text{CPC})} \\ &= \begin{bmatrix} 0.648 & 0.325 & 0.257 & 2.254 & 2.173 & 0.171 \\ 0.325 & 3.385 & 2.116 & 1.438 & 1.341 & 0.226 \\ 0.257 & 2.116 & 2.260 & 1.256 & 1.047 & 0.185 \\ 2.254 & 1.438 & 1.256 & 11.547 & 8.753 & 0.645 \\ 2.173 & 1.341 & 1.047 & 8.753 & 10.208 & 0.622 \\ 0.171 & 0.226 & 0.185 & 0.645 & 0.622 & 0.500 \end{bmatrix} \end{aligned}$$

- *Vaginal*:

$$\begin{aligned} \mathbf{S}_{2(\text{CPC})}^* &= \hat{\alpha}_2\mathbf{S}_2 + (1 - \hat{\alpha}_2)\mathbf{S}_{2(\text{CPC})} \\ &= \begin{bmatrix} 0.775 & 0.377 & 0.339 & 3.242 & 2.821 & 0.280 \\ 0.377 & 5.251 & 3.637 & 1.884 & 1.447 & 0.651 \\ 0.339 & 3.637 & 3.433 & 1.758 & 1.248 & 0.589 \\ 3.242 & 1.884 & 1.758 & 17.826 & 13.397 & 1.339 \\ 2.821 & 1.447 & 1.248 & 13.397 & 13.545 & 1.170 \\ 0.280 & 0.651 & 0.589 & 1.339 & 1.170 & 0.918 \end{bmatrix}. \end{aligned}$$

5.7.2 Regions

In Chapter 4 it was concluded that the population covariance matrices of the *South Africa* ($n_1 = 2921$) and *Namibia* ($n_2 = 120$) regional groups have a common eigenvector structure (full CPC). Knowledge about the common eigenvectors can be used to obtain estimates of the population covariance matrices of the two groups under the CPC model, in particular for the *Namibia* region from which relatively little data are available.

Using the *CPC* estimator, the off-diagonal elements of the \mathbf{L}_i matrices are shrunk to zero, which, combined with the \mathbf{B} matrix yield the following estimates for the covariance matrices of the two regional groups under the CPC assumption:

- *South Africa*:

$$\begin{aligned} \mathbf{S}_{1(\text{CPC})} &= \mathbf{B}\mathbf{L}_1^0\mathbf{B}' \\ &= \begin{bmatrix} 0.671 & 0.323 & 0.261 & 2.398 & 2.261 & 0.189 \\ 0.323 & 3.704 & 2.392 & 1.441 & 1.347 & 0.285 \\ 0.261 & 2.392 & 2.465 & 1.274 & 1.082 & 0.259 \\ 2.398 & 1.441 & 1.274 & 12.389 & 9.307 & 0.738 \\ 2.261 & 1.347 & 1.082 & 9.307 & 10.628 & 0.689 \\ 0.189 & 0.285 & 0.259 & 0.738 & 0.689 & 0.556 \end{bmatrix} \end{aligned}$$

- *Namibia*:

$$\begin{aligned} \mathbf{S}_{2(\text{CPC})} &= \mathbf{B}\mathbf{L}_2^0\mathbf{B}' \\ &= \begin{bmatrix} 0.877 & 0.502 & 0.405 & 3.384 & 3.159 & 0.268 \\ 0.502 & 3.610 & 2.172 & 2.270 & 2.099 & 0.306 \\ 0.405 & 2.172 & 2.503 & 1.951 & 1.708 & 0.301 \\ 3.384 & 2.270 & 1.951 & 16.973 & 13.468 & 1.057 \\ 3.159 & 2.099 & 1.708 & 13.468 & 14.430 & 0.979 \\ 0.268 & 0.306 & 0.301 & 1.057 & 0.979 & 0.700 \end{bmatrix}. \end{aligned}$$

The crossvalidation method described in Section 5.5.2 gave the following values for the shrinkage intensity parameters for the two regions:

- *South Africa*

$$\hat{\alpha}_1 = 0.0351,$$

and

- *Namibia*

$$\hat{\alpha}_2 = 0.6201.$$

Finally, substituting the estimated shrinkage intensity values into (5.10) yields the following estimates of the population covariance matrices (with the *Full CPC crossvalid* estimator):

- *South Africa:*

$$\begin{aligned} \mathbf{S}_{1(\text{CPC})}^* &= \hat{\alpha}_1 \mathbf{S}_1 + (1 - \hat{\alpha}_1) \mathbf{S}_{1(\text{CPC})} \\ &= \begin{bmatrix} 0.671 & 0.323 & 0.261 & 2.399 & 2.261 & 0.190 \\ 0.323 & 3.703 & 2.391 & 1.439 & 1.345 & 0.285 \\ 0.261 & 2.391 & 2.465 & 1.273 & 1.081 & 0.260 \\ 2.399 & 1.439 & 1.273 & 12.392 & 9.307 & 0.740 \\ 2.261 & 1.345 & 1.081 & 9.307 & 10.627 & 0.690 \\ 0.190 & 0.285 & 0.260 & 0.740 & 0.690 & 0.556 \end{bmatrix} \end{aligned}$$

- *Namibia:*

$$\begin{aligned} \mathbf{S}_{2(\text{CPC})}^* &= \hat{\alpha}_2 \mathbf{S}_2 + (1 - \hat{\alpha}_2) \mathbf{S}_{2(\text{CPC})} \\ &= \begin{bmatrix} 0.870 & 0.573 & 0.435 & 3.157 & 3.304 & 0.125 \\ 0.573 & 4.149 & 2.250 & 2.916 & 2.581 & 0.069 \\ 0.435 & 2.250 & 2.305 & 2.447 & 2.021 & 0.089 \\ 3.157 & 2.916 & 2.447 & 15.976 & 13.456 & 0.209 \\ 3.304 & 2.581 & 2.021 & 13.456 & 15.218 & 0.402 \\ 0.125 & 0.069 & 0.089 & 0.209 & 0.402 & 0.575 \end{bmatrix}. \end{aligned}$$

It is interesting to compare the *Full CPC crossvalid* estimate of the covariance matrix for *Namibia* to the *Unbiased* estimate for this group:

$$\mathbf{S}_2 = \begin{bmatrix} 0.866 & 0.617 & 0.453 & 3.017 & 3.393 & 0.037 \\ 0.617 & 4.479 & 2.299 & 3.312 & 2.877 & -0.076 \\ 0.453 & 2.299 & 2.184 & 2.751 & 2.212 & -0.041 \\ 3.017 & 3.312 & 2.751 & 15.365 & 13.449 & -0.311 \\ 3.393 & 2.877 & 2.212 & 13.449 & 15.702 & 0.048 \\ 0.037 & -0.076 & -0.041 & -0.311 & 0.048 & 0.498 \end{bmatrix}$$

Using the *Unbiased* estimate of the covariance matrix, the conclusion would have been that the sixth variable (*ATEMP* = worst temperature measured within one hour of birth) has negative relationships with the two Apgar score variables (*AP1* and *AP5*) and gestational age (*GESTAGE*). However, from $\mathbf{S}_{2(\text{CPC})}^*$ it is seen that this conclusion is probably incorrect—in fact, the temperature variable has positive relationships with all of the other perinatal variables. The correlations between birth weight (*BWGT*) and temperature, and birth head circumference (*BHEADCIR*) and temperature are also probably not as weak as would have been concluded by an interpretation of the *Unbiased* covariance matrix values.

Chapter 6

Using the CPC model in discriminant analysis

6.1 Introduction

After obtaining improved estimates of two or more population covariance matrices using the CPC model (if appropriate) as described in Chapter 5, it is of interest to investigate whether these estimates can be used to improve the misclassification error rate in discriminant analysis. If there is more accurate information available about the structures of the population covariance matrices, it should be easier to determine to which group a new observation belongs. Plugging the CPC covariance matrix estimators into the quadratic discriminant function leads to what is known as the *CPC discriminant function*, which is introduced in Section 6.2.

The topic of CPC discriminant analysis was first studied by Schmid (1987), with the main results discussed again by Flury (1988). Flury and Schmid (1992) derived asymptotic results for the discrimination coefficients under the homogeneous covariance matrices, proportionality, CPC and unrelated covariance matrices models, respectively. They showed that, although CPC discrimination can improve on the misclassification rate of ordinary quadratic discrimination in some situations, usually such improvement is not substantial. They suggested that both the proportional and CPC models can perform well in classification applications where the number of variables and/or number of groups are large relative to the sample sizes, and that these models offer a compromise between the assumption of equal covariance matrices on the one extreme and the assumption of unrelated covariance matrices on the other. For sparse data, theoretically incorrect but more parsimonious models can sometimes outperform theoretically correct models in

discriminant analysis applications, because the discriminant function coefficients for a more parsimonious model will generally be less variable (Flury and Schmid, 1992).

Flury et al. (1994) followed up on these investigations with a simulation study to determine the misclassification error rates for the different covariance matrix estimators when plugged into the quadratic discriminant function. They showed that CPC discrimination often does not have a great advantage over ordinary quadratic discrimination (using the unbiased sample estimators of the population covariance matrices), and that in many situations the performance of quadratic discrimination under the proportional model compares well with the results achieved by using the linear discriminant rule. They concluded that ordinary quadratic discrimination should be avoided if possible, as it performs poorly in all scenarios except when there are large samples available from populations with unrelated covariance matrices.

Friedman (1989) proposed a technique known as *regularised discriminant analysis* for high-dimensional, sparse data settings. This method is briefly discussed in Section 6.3, pointing out a relation to CPC discrimination based on the new CPC shrinkage covariance matrix estimator developed in Chapter 5.

Section 6.4 contains a short discussion of the shapes of the ellipses formed by different covariance matrix estimators in the bivariate normal setting, and offers an explanation of why certain estimators will perform poorly for population covariance matrices at specific levels in Flury's hierarchy.

Bianco et al. (2008) used partial influence functions to derive the asymptotic variances of the discriminant function coefficients under the same models considered by Flury and Schmid (1992), and their results confirmed the order relations of the coefficient variances from the earlier study. They also performed a simulation study using multivariate normal as well as contaminated data (i.e. a mixture of data from two different normal distributions), and their results concerning the performance of the different discriminant functions confirmed what was found previously by Flury et al. (1994). The results of the simulation study presented in Section 6.5 of this chapter will be compared to these earlier studies on CPC discrimination.

The chapter is concluded with applications of the CPC discrimination procedure to the delivery mode, regional, and mortality groupings of the VON 2009 cohort in Section 6.6.

6.2 Discriminant analysis under the CPC model

Suppose that samples from $k = 2$ multivariate normally distributed populations with unequal mean vectors and common covariance matrix, Σ , are available. Indicate the estimators of the population means vectors of the first and second populations with $\bar{\mathbf{x}}_1$ and $\bar{\mathbf{x}}_2$, respectively, and let

$$c = \frac{1}{2}(\bar{\mathbf{x}}_1' \mathbf{S}_p^{-1} \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2' \mathbf{S}_p^{-1} \bar{\mathbf{x}}_2), \quad (6.1)$$

where \mathbf{S}_p indicates the pooled covariance matrix estimator as in (4.2). Assuming equal costs of misclassification and equal prior probabilities of occurrence, a new observation with unknown group membership status, \mathbf{x}_{new} , is allocated to the first group (i.e. belonging to the first population) if

$$(\bar{\mathbf{x}}_1' - \bar{\mathbf{x}}_2') \mathbf{S}_p^{-1} \mathbf{x}_{\text{new}} \geq c, \quad (6.2)$$

otherwise it is allocated to the second group. Equation (6.2) is known as the *linear classification rule* (Johnson and Wichern, 2002). The purpose of the linear discriminant function is to find the linear combination of the p variables giving the greatest separation between the group centroids in the p -dimensional space.

Fisher (1938) derived the same linear classification rule in (6.2) without the multivariate normality assumption. The multivariate normality assumption is thus not necessary for *linear discriminant analysis (LDA)*, and the method can also be applied to multivariate non-normal data.

If the assumption of homogeneous covariance matrices for the two populations is untenable, *quadratic discriminant analysis (QDA)* should be used instead. Unlike the linear discriminant function, the quadratic discriminant function depends on the assumption that the populations have multivariate normal distributions. QDA should therefore not be used if the normality assumption seems doubtful.

For samples from $k = 2$ multivariate normally distributed populations, let

$$c = \frac{1}{2} \ln \left(\frac{|\mathbf{S}_1|}{|\mathbf{S}_2|} \right) + \frac{1}{2}(\bar{\mathbf{x}}_1' \mathbf{S}_1^{-1} \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2' \mathbf{S}_2^{-1} \bar{\mathbf{x}}_2), \quad (6.3)$$

where \mathbf{S}_1 and \mathbf{S}_2 indicate the unbiased sample estimators of the covariance matrices of the first and second populations, respectively.

Assume equal costs of misclassification for the two groups and equal prior probabilities of occurrence. According to the *quadratic classification rule*, a new observation, \mathbf{x}_{new} , is allocated to the first group if

$$-\frac{1}{2}\mathbf{x}'_{\text{new}}(\mathbf{S}_1^{-1} - \mathbf{S}_2^{-1})\mathbf{x}_{\text{new}} + (\bar{\mathbf{x}}'_1\mathbf{S}_1^{-1} - \bar{\mathbf{x}}'_2\mathbf{S}_2^{-1})\mathbf{x}_{\text{new}} \geq c, \quad (6.4)$$

otherwise it is allocated to the second group (Johnson and Wichern, 2002).

Under the multivariate normality assumption, more accurate estimators of the population covariance matrices in (6.3) and (6.4) can lead to improved classification rules and lower misclassification error rates. This hypothesis have been investigated by Schmid (1987), Flury (1988), Flury and Schmid (1992), Flury et al. (1994) and Bianco et al. (2008), and the results from this chapter will be compared to these studies. For CPC discrimination, the unbiased covariance matrix estimators in (6.3) and (6.4) are simply replaced by the CPC estimators. Let

$$c = \frac{1}{2}\ln\left(\frac{|\mathbf{S}_{1(\text{CPC})}|}{|\mathbf{S}_{2(\text{CPC})}|}\right) + \frac{1}{2}(\bar{\mathbf{x}}'_1\mathbf{S}_{1(\text{CPC})}^{-1}\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}'_2\mathbf{S}_{2(\text{CPC})}^{-1}\bar{\mathbf{x}}_2), \quad (6.5)$$

where $\mathbf{S}_{1(\text{CPC})}$ and $\mathbf{S}_{2(\text{CPC})}$ are the *CPC* covariance matrix estimators in (5.5) for the first and second groups, respectively. Under the CPC assumption, the quadratic discriminant rule becomes: Allocate a new observation, \mathbf{x}_{new} , to the first group if

$$-\frac{1}{2}\mathbf{x}'_{\text{new}}(\mathbf{S}_{1(\text{CPC})}^{-1} - \mathbf{S}_{2(\text{CPC})}^{-1})\mathbf{x}_{\text{new}} + (\bar{\mathbf{x}}'_1\mathbf{S}_{1(\text{CPC})}^{-1} - \bar{\mathbf{x}}'_2\mathbf{S}_{2(\text{CPC})}^{-1})\mathbf{x}_{\text{new}} \geq c, \quad (6.6)$$

otherwise allocate it to the second group.

Flury and Schmid (1992) have shown that the asymptotic variances of the discriminant function coefficients are the same for ordinary QDA and CPC discrimination for $k = 2$ groups when the CPC model holds. In particular, if $\lambda_{1j} - \lambda_{1h} = \lambda_{2h} - \lambda_{2j}$ for all (j, h) pairs of the eigenvalues from two population covariance matrices, CPC discrimination and ordinary QDA should perform about equally well. However, if $\lambda_{1j}^{-1} - \lambda_{1h}^{-1} = \lambda_{2j}^{-1} - \lambda_{2h}^{-1}$ for all (j, h) , the variances of some of the CPC discriminant function coefficients can be smaller than those obtained from the ordinary quadratic discriminant rule and CPC discrimination may perform better. Thus the improvement in misclassification error rate from the use of CPC covariance matrix estimators (instead of the unbiased covariance matrix estimators) in the quadratic discriminant rule should generally not be large.

In the partial common eigenvector situation, estimators of the covariance matrices under the $\text{CPC}(q)$ model can be plugged into (6.3) and (6.4) to provide a partial CPC quadratic discriminant rule. However, in light of the small improvement in misclassification error expected from CPC discrimination compared to ordinary QDA, use of partial CPC discrimination will

probably not be of any practical significance and therefore was not be explored for the purpose of this dissertation.

Although it may seem that LDA will not be widely applicable due to the very restrictive common covariance matrix assumption, O'Neill (1984) found that the linear classification rule is quite robust against deviations from this assumption. Relatively large sample sizes are needed for ordinary QDA to outperform LDA, even when the population covariance matrices are not nearly equal. Flury et al. (1994) made the more general observation that using a more parsimonious but theoretically incorrect model in the Flury hierarchy of covariance matrices often leads to better classification. This is particularly true in situations where the number of groups and/or number of variables are large relative to the sample sizes, as the stricter constraints imposed on the covariance matrices lead to more stable estimators and often to improved discrimination between the groups.

6.3 Regularized discriminant analysis

From the theory for spectral decomposition (Johnson and Wichern, 2002) it is known that the inverse of the i^{th} sample covariance matrix can be written in terms of its eigenvectors and eigenvalues as

$$\mathbf{S}_i^{-1} = \sum_{j=1}^p \frac{\mathbf{e}_{ij}\mathbf{e}_{ij}'}{d_{ij}}, \quad i = 1, \dots, k. \quad (6.7)$$

Consequently the quadratic discriminant rule in (6.4) can be written as

$$\begin{aligned} & -\frac{1}{2}\mathbf{x}'_{\text{new}} \left[\sum_{j=1}^p \left(\frac{\mathbf{e}_{1j}\mathbf{e}_{1j}'}{d_{1j}} - \frac{\mathbf{e}_{2j}\mathbf{e}_{2j}'}{d_{2j}} \right) \right] \mathbf{x}_{\text{new}} \\ & + \left(\bar{\mathbf{x}}'_1 \sum_{j=1}^p \frac{\mathbf{e}_{1j}\mathbf{e}_{1j}'}{d_{1j}} - \bar{\mathbf{x}}'_2 \sum_{j=1}^p \frac{\mathbf{e}_{2j}\mathbf{e}_{2j}'}{d_{2j}} \right) \mathbf{x}_{\text{new}} \geq c. \end{aligned} \quad (6.8)$$

From (6.8) it can be seen that the smallest eigenvalues of \mathbf{S}_i can have a large influence on the discriminant function, and most of the variability in the discriminant function can usually be traced back to the subspace spanned by the eigenvectors associated with the smallest eigenvalues (Friedman, 1989). For small samples or high-dimensional data, the sample covariance matrix elements, and consequently the eigenvectors and eigenvalues, will not be estimated very precisely, causing instability in the discriminant function. Thus any method which enables more precise estimation of the eigenvectors and

eigenvalues without introducing an unacceptable amount of bias should decrease the variability of the discriminant function. According to Friedman (1989), this explains why LDA, using the biased but more precise pooled covariance matrix estimator, outperforms QDA in small-sample situations even for instances where it employs a theoretically incorrect covariance matrix model.

The degree of ellipsoidal symmetry of the population distributions had been found to be a more important aspect in discriminative accuracy than the detailed shape of the distributions (Friedman, 1989). This may explain why, when the sample size is relatively large, there seems to be little difference in misclassification error rate between the quadratic discriminant functions based on the different covariance matrix estimators.

Regularised discriminant analysis (RDA), proposed by Friedman (1989), finds a weighted estimate,

$$\mathbf{S}_i(\lambda) = (1 - \lambda)\mathbf{S}_i + \lambda\mathbf{S}_p, \quad (6.9)$$

between the unbiased sample covariance matrix for the i^{th} group, \mathbf{S}_i , and the pooled covariance matrix, \mathbf{S}_p . The shrinkage intensity parameter, λ , is determined by crossvalidation. To counteract the inordinate effect of the smallest eigenvalues of $\mathbf{S}_i(\lambda)$ in (6.8), the weighted covariance matrix in (6.9) is shrunk towards a multiple of the identity matrix,

$$\mathbf{S}_i(\lambda, \gamma) = (1 - \gamma)\mathbf{S}_i(\lambda) + \frac{\gamma}{p} \text{tr}[\mathbf{S}_i(\lambda)] \mathbf{I}. \quad (6.10)$$

Note that the target matrix in (6.10) is a multiple of the identity matrix with the average eigenvalue of $\mathbf{S}_i(\lambda)$ on the diagonal. The optimal value for the second shrinkage intensity parameter, γ , is also determined by crossvalidation. The regularised covariance matrices, $\mathbf{S}_i(\lambda, \gamma)$, are plugged into the quadratic discriminant function to perform RDA.

Friedman (1989) concludes that RDA is useful in a small sample size, high-dimensional context. Although the method is not greatly disadvantaged with well-conditioned data as considered in this dissertation, it is not recommended.

The reason for the discussion of RDA here is that it is similar to CPC discrimination based on the new CPC shrinkage estimators proposed in Chapter 5, in the sense that both methods plug regularised estimators of the covariance matrices into the quadratic discriminant function to perform classification. Plugging the CPC shrinkage estimators (from equation (5.10)) into (6.4) gives the classification rule,

$$-\frac{1}{2}\mathbf{x}'_{\text{new}}(\mathbf{S}_{1(\text{CPC})}^{\star-1} - \mathbf{S}_{2(\text{CPC})}^{\star-1})\mathbf{x}_{\text{new}} + (\bar{\mathbf{x}}'_1\mathbf{S}_{1(\text{CPC})}^{\star-1} - \bar{\mathbf{x}}'_2\mathbf{S}_{2(\text{CPC})}^{\star-1})\mathbf{x}_{\text{new}} \geq c, \quad (6.11)$$

where

$$c = \frac{1}{2}\ln\left(\frac{|\mathbf{S}_{1(\text{CPC})}^{\star}|}{|\mathbf{S}_{2(\text{CPC})}^{\star}|}\right) + \frac{1}{2}(\bar{\mathbf{x}}'_1\mathbf{S}_{1(\text{CPC})}^{\star-1}\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}'_2\mathbf{S}_{2(\text{CPC})}^{\star-1}\bar{\mathbf{x}}_2). \quad (6.12)$$

The estimator in (5.10) shrinks the unbiased sample covariance matrix towards the CPC estimator, whereas RDA shrinks the unbiased sample covariance matrix consecutively towards the pooled estimator and a multiple of the identity matrix. CPC discrimination can thus be viewed as a form of RDA, with regularisation performed in a different manner.

6.4 Shapes of the covariance matrix estimates

Different estimators of the population covariance matrices of several groups have different shapes in p -dimensional space, with the shapes depending on the available sample data and the constraints (if any) imposed on the covariance matrices. In the context of discriminant analysis, the classification accuracy depends on the accuracy of the sample covariance matrix estimates as representations of the population covariance structures. Pronounced differences between the population covariance matrices and the sample estimates will in most cases affect the classification accuracy negatively.

To demonstrate some of the typical shapes formed by different covariance matrix estimators for data from multivariate normally distributed populations, consider the $p = 2$, $k = 2$ situation when the two population covariance matrices are exactly equal. In this case the *Pooled* covariance matrix estimator should be the most precise, as it combines the information regarding the common eigenvectors as well as the common eigenvalues to estimate the equal population covariance matrices. The CPC and *Full CPC cross-valid* estimators only utilise information about the common eigenvectors in the two covariance matrices, allowing the two sets of eigenvalues to differ. The *Unbiased* estimator should perform worst of the four estimators in this situation, especially for small samples, as it has the fewest degrees of freedom to estimate the two sets of eigenvectors and two sets of eigenvalues. It is unnecessary in this case, as the sets of eigenvectors and eigenvalues are common to the two population covariance matrices. From the graph in the top left of Figure 6.1 it can be seen that the shapes formed by the 95%

confidence ellipses for all four covariance matrix estimators are close to the ellipses formed by the actual population covariance matrices.

When the population covariance matrices are not equal, but have common eigenvectors and the rank order of the common eigenvectors are the same for both covariance matrices (Figure 6.1, top right), the accuracy of the *Pooled* estimator should deteriorate as the absolute differences between the sets of eigenvalues increases. The CPC estimators and *Unbiased* should perform about equally well for larger sample sizes, but for smaller sample sizes *CPC* and *Full CPC crossvalid* should be slightly more accurate in the estimation of the population covariance matrices, because the common eigenvectors are estimated by pooling the information from both groups.

For the situation where there are common eigenvectors but the rank orders of the eigenvectors are exactly the opposite in the two population covariance matrices (Figure 6.1, bottom left), the *Pooled* estimator should perform poorly. For example, if the magnitudes of the eigenvalues in the two groups are equal, but they are associated with different eigenvectors, the *Pooled* estimator will give an estimate of the covariance structure which is more or less spherical. For higher dimensional populations, the same principle will hold: If the rank orders of two or more common eigenvectors are very different (or the exact opposite) in the two covariance matrices, the *Pooled* estimator of the covariance matrices will be more spherical in the subspace spanned by these common eigenvectors, estimating the population covariance structure poorly.

The CPC estimators should still perform relatively well in this situation, even with the greater number of parameters associated with it. Because information about an eigenvector associated with a smaller eigenvalue (with a larger standard error) in the one covariance matrix is combined with information about an eigenvector associated with a larger eigenvalue (with a smaller standard error) in the other covariance matrix, the CPC estimators should perform slightly worse than in the same common eigenvector rank order situation, but should still estimate the population covariance structures accurately. The accuracy of the *Unbiased* estimator relative to the CPC estimators should be similar to that observed in the same eigenvector rank order scenario.

Should the population covariance matrices be unrelated (Figure 6.1, bottom right), the *Pooled*, *CPC* and *Full CPC crossvalid* estimators will all incorrectly assume common eigenvectors, which will hamper their accuracy when compared to the *Unbiased* estimator. The *Unbiased* estimator should perform the best in this case, even with the fewer degrees of freedom to estimate the two sets of eigenvectors and two sets of eigenvalues. From data simulated from two bivariate normally distributed populations with unrelated

6.4. SHAPES OF THE COVARIANCE MATRIX ESTIMATES

191

covariance matrices, it was observed that the *CPC* estimator will generally model the one covariance matrix well and the other one poorly.

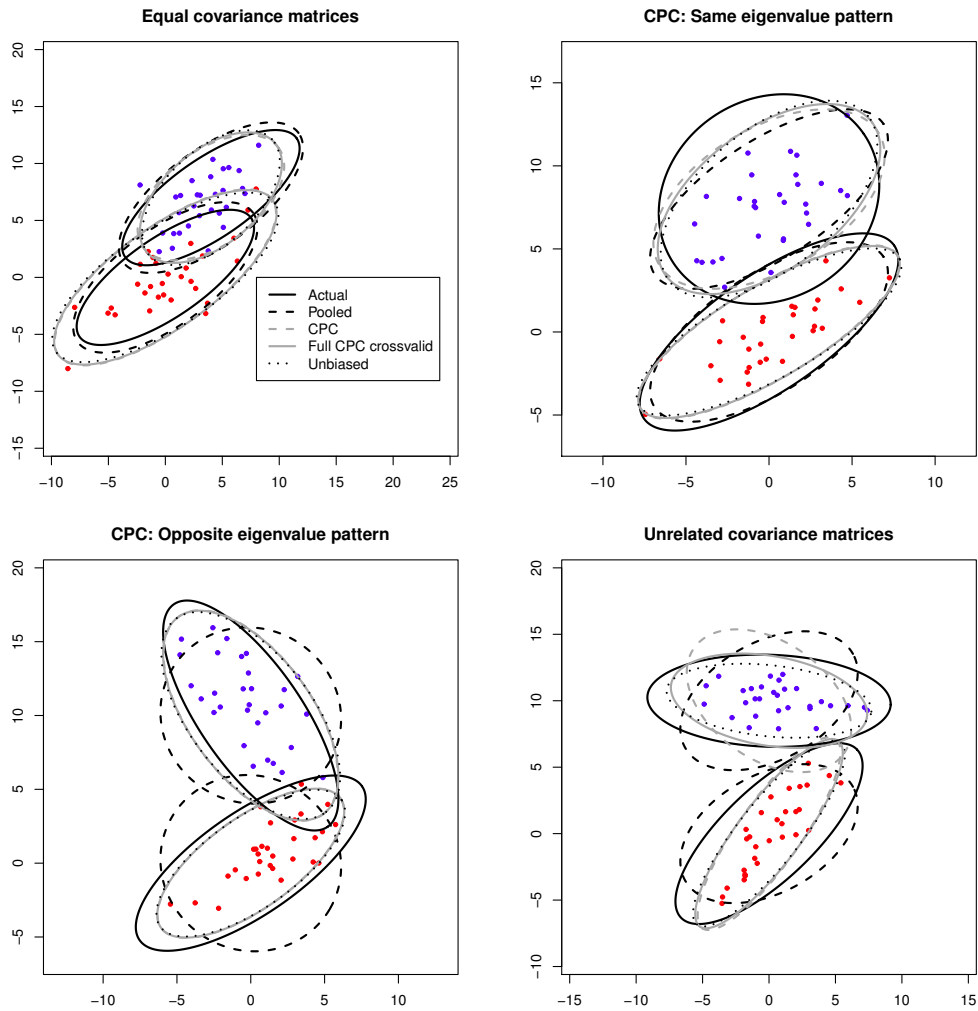


Figure 6.1: 95% confidence ellipses for different covariance matrix estimators, for samples of sizes $n_1 = n_2 = 30$ from two bivariate normal populations. The solid black ellipses indicates the actual population covariance matrices in each case.

6.5 Simulation study

A number of Monte Carlo simulation experiments were executed to compare the performance of LDA to the quadratic discriminant functions using the *Unbiased*, *CPC* and *Full CPC crossvalid* covariance matrix estimators for two groups, as discussed in Chapter 5. Samples of sizes $n_i = 30, 50, 100, 200$ were simulated from $k = 2$ multivariate normally distributed populations with $p = 2, 5$ and 10 variables, respectively. Each simulated sample of size n_i (per group) was randomly divided into a 70% training sample and a 30% test sample. Per simulation run, the four different discriminant functions were estimated from the training samples of the two groups, and the misclassification error rates calculated for the test samples.

To improve readability of the results reported in this section, the following designations are used for each of the four different discriminant functions:

- **QDA**. The quadratic classification rule, using the *Unbiased* sample covariance matrices of the two groups.
- **CPC**. The quadratic classification rule, using the *CPC* estimators of the covariance matrices of the two groups.
- **CPC***. The quadratic classification rule, using the *Full CPC crossvalid* estimators of the covariance matrices of the two groups.
- **LDA**. The linear classification rule, using the *Pooled* covariance matrix estimator.

6.5.1 $p = 2$ variables case

For the simulation experiment in the $p = 2$ variables case, data were simulated from two bivariate normally distributed populations with the following covariance matrices:

- 1) **Equal covariance matrices** (Σ_{EQUAL})
- 2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME}). The common eigenvector associated with the largest eigenvalue of the first covariance matrix is also associated with the largest eigenvalue of the second covariance matrix.
- 3) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE}). The common eigenvector associated with the largest eigenvalue of the first covariance matrix is associated with the smallest eigenvalue of the second covariance matrix, and vice versa.

4) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

The population covariance matrices are given in Appendix D. For all four of the scenarios described above, the mean vectors for the two populations were

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{\mu}_2 = \begin{bmatrix} 8 \\ 8 \end{bmatrix}.$$

A total of 1000 simulation runs were performed for each of the ($n_i \times$ four covariance structures considered) scenarios. Misclassification error rates for the test samples per simulation run in the experiment are reported in Table 6.1.

Table 6.1: Simulation results for $k = 2$ samples of equal sizes drawn from bivariate normally distributed populations. Each of the values in the table were calculated from 1000 simulation runs. Standard errors are given in brackets.

Structure	n_i	Misclassification error (%)							
		QDA		CPC		CPC*		LDA	
Σ_{EQUAL}	30	5.66	(0.73)	5.47	(0.72)	5.56	(0.72)	5.33	(0.71)
	50	5.21	(0.70)	5.13	(0.70)	5.13	(0.70)	5.08	(0.69)
	100	5.04	(0.69)	4.95	(0.69)	5.01	(0.69)	4.94	(0.69)
	200	4.90	(0.68)	4.87	(0.68)	4.89	(0.68)	4.86	(0.68)
Σ_{SAME}	30	10.29	(0.96)	9.90	(0.94)	10.13	(0.95)	10.26	(0.96)
	50	9.44	(0.92)	9.28	(0.92)	9.39	(0.92)	9.63	(0.93)
	100	9.31	(0.92)	9.31	(0.92)	9.32	(0.92)	9.66	(0.93)
	200	9.36	(0.92)	9.35	(0.92)	9.34	(0.92)	9.77	(0.94)
Σ_{OPPOSITE}	30	10.68	(0.98)	10.32	(0.96)	10.39	(0.97)	10.51	(0.97)
	50	10.12	(0.95)	10.02	(0.95)	10.00	(0.95)	10.11	(0.95)
	100	9.99	(0.95)	9.97	(0.95)	10.00	(0.95)	10.16	(0.96)
	200	9.65	(0.93)	9.61	(0.93)	9.63	(0.93)	9.80	(0.94)
$\Sigma_{\text{UNRELATED}}$	30	10.18	(0.96)	10.16	(0.96)	10.12	(0.95)	10.15	(0.95)
	50	10.22	(0.96)	10.28	(0.96)	10.15	(0.96)	10.21	(0.96)
	100	9.75	(0.94)	10.01	(0.95)	9.77	(0.94)	9.95	(0.95)
	200	9.52	(0.93)	9.81	(0.94)	9.54	(0.93)	9.74	(0.94)

For equal population covariance matrices (Σ_{EQUAL}), *LDA* showed the smallest misclassification error rate, followed closely by *CPC* and *CPC**. *QDA* performed the worst of the four discriminant functions in this scenario.

In the *CPC* scenario where the common eigenvectors have the same rank order in the two population covariance matrices (Σ_{SAME}), *CPC* performed the best and *LDA* the worst. Only in for the largest sample size ($n_i = 200$) did *CPC** slightly outperform *CPC*. It also seems as if *QDA* and *CPC* perform about equally well for larger sample sizes ($n_1 = 100, 200$) in this scenario.

When the common eigenvectors have exactly the opposite rank orders in the two population covariance matrices (Σ_{OPPOSITE}), the CPC discriminant functions performed the best. The advantage of *CPC* and *CPC** over *QDA* and *LDA* became less clear as the sample size increased though.

In the unrelated covariance matrices scenarios ($\Sigma_{\text{UNRELATED}}$), *CPC** fared the best for smaller sample sizes ($n_i = 30, 50$), but was outperformed by the theoretically correct *QDA* discriminant function in the larger sample sizes ($n_i = 100, 200$). *LDA* had lower misclassification error rates than *CPC* in this scenario.

However, when considering the standard errors of the misclassification error rates (given in brackets in Table 6.1), it is clear that there were no statistically significant differences between the different discriminant functions in any of the scenarios in the $p = 2$ case.

6.5.2 $p = 5$ variables case

A second simulation experiment was performed for $k = 2$ multivariate normally distributed populations with $p = 5$ variables. The following five sets of population covariance matrix scenarios were considered:

- 1) **Equal covariance matrices** (Σ_{EQUAL})
- 2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME}): Common eigenvectors with different sets of eigenvalues (not proportional), but with the exact same rank order in the two population covariance matrices.
- 3) **CPC: Similar rank orders of the common eigenvectors** (Σ_{SIMILAR}): Common eigenvectors with similar rank orders in the two population covariance matrices. Thus the common eigenvectors associated with the largest eigenvalues in the first covariance matrix are also associated with the largest eigenvalues in the second covariance matrix, and common eigenvectors associated with the smallest eigenvalues in the first covariance matrix are also associated with the smallest eigenvalues in the second covariance matrix.
- 4) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE}): Common eigenvectors with exactly the opposite rank orders in the two population covariance matrices. Thus the common eigenvectors associated with the largest eigenvalues in the first covariance matrix are associated with the smallest eigenvalues in the second covariance matrix, and vice versa.

5) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

The population covariance matrices are given in Appendix D. For each of the covariance matrix scenarios outlined above, the population mean vectors were specified as

$$\boldsymbol{\mu}'_1 = [0, 0, 0, 0, 0], \quad \text{and}$$

$$\boldsymbol{\mu}'_2 = [2, 2, 2, 2, 2].$$

A total of 1000 simulation runs were performed for each of the ($n_i \times$ five covariance structures considered) scenarios. The misclassification error rates and mean ranks calculated from the 30% test samples are reported in Table 6.2.

When the population covariance matrices were equal, *LDA* gave the smallest misclassification error rates for all of the sample sizes, as expected. However, the differences in misclassification error rates were not statistically significant, as can be seen from the standard errors reported in Table 6.2. Although the differences in error rates between *LDA* and the other three discriminant functions decreased with an increase in sample size, the equal covariance matrices model is the most parsimonious (compared to the *CPC* and unrelated covariance matrix models) and therefore performs the best. *QDA* is also theoretically correct but employs the least parsimonious of the covariance matrix models, and performed the worst for all sample sizes. These results concur with the results reported by Flury et al. (1994) and Bianco et al. (2008).

For the *CPC* situation when the rank orders of the common eigenvectors in the two population covariance matrices were exactly the same, *CPC* performed the best, followed by *CPC** and *QDA*. *LDA* performed the worst for this covariance structure. For population covariance matrices with similar rank orders of the p common eigenvectors in the population covariance matrices, *CPC* and *CPC** again performed the best, and *LDA* the worst. Again, for both Σ_{SAME} and Σ_{SIMILAR} , the misclassification error rates for the different discriminant functions were not significantly different.

CPC discrimination seems to offer a real improvement over *QDA* and *LDA*, particularly for smaller sample sizes, in the *CPC* case where the common eigenvectors have exactly opposite rank orders in the two population covariance matrices. In these scenarios *CPC* and *CPC** performed the best, followed by the (also theoretically correct) *QDA* discriminant function. *LDA* gave very large misclassification error rates compared to the other three discriminant functions.

Table 6.2: Simulation results for $k = 2$ samples of equal sizes drawn from multivariate normally distributed populations with $p = 5$ variables. Each of the values in the table were calculated from 1000 simulation runs. Standard errors are given in brackets.

Structure	n_i	Misclassification error (%)							
		QDA		CPC		CPC*		LDA	
Σ_{EQUAL}	30	28.62	(1.43)	24.89	(1.37)	25.42	(1.38)	24.40	(1.36)
	50	25.96	(1.39)	23.71	(1.34)	23.94	(1.35)	23.38	(1.34)
	100	23.66	(1.34)	22.57	(1.32)	22.61	(1.32)	22.50	(1.32)
	200	22.70	(1.32)	22.02	(1.31)	22.12	(1.31)	21.94	(1.31)
Σ_{SAME}	30	22.76	(1.33)	19.07	(1.24)	19.58	(1.25)	23.04	(1.33)
	50	20.54	(1.28)	18.63	(1.23)	18.88	(1.24)	22.61	(1.32)
	100	18.30	(1.22)	17.39	(1.20)	17.52	(1.20)	21.35	(1.30)
	200	17.57	(1.20)	17.08	(1.19)	17.15	(1.19)	20.68	(1.28)
Σ_{SIMILAR}	30	10.42	(0.97)	8.23	(0.87)	8.36	(0.88)	10.78	(0.98)
	50	8.66	(0.89)	7.47	(0.83)	7.69	(0.84)	9.71	(0.94)
	100	7.47	(0.83)	6.93	(0.80)	7.01	(0.81)	8.96	(0.90)
	200	6.94	(0.80)	6.74	(0.79)	6.76	(0.79)	8.57	(0.89)
Σ_{OPPOSITE}	30	9.46	(0.93)	8.06	(0.86)	8.29	(0.87)	20.37	(1.27)
	50	7.98	(0.86)	7.27	(0.82)	7.35	(0.83)	18.87	(1.24)
	100	7.22	(0.82)	6.96	(0.80)	6.92	(0.80)	17.48	(1.20)
	200	7.02	(0.81)	6.84	(0.80)	6.86	(0.80)	16.79	(1.18)
$\Sigma_{\text{UNRELATED}}$	30	11.86	(1.02)	13.02	(1.06)	11.56	(1.01)	30.63	(1.46)
	50	10.33	(0.96)	12.29	(1.04)	10.58	(0.97)	29.24	(1.44)
	100	9.37	(0.92)	11.64	(1.01)	9.62	(0.93)	28.35	(1.43)
	200	9.06	(0.91)	11.42	(1.01)	9.29	(0.92)	27.86	(1.42)

In the unrelated covariance matrices scenario, *QDA* fared the best, except for the smallest sample size ($n_1 = 30$) where it was marginally outperformed by *CPC**. *LDA* clearly performed the worst of the four discriminant functions in this scenario, giving very large misclassification error rates.

The benefit of using the more parsimonious *CPC* model becomes more apparent as the number of dimensions increases (Flury and Schmid, 1992). As p and/or k increase, the difference in number of parameters between the *CPC* covariance matrix estimator and the *Unbiased* estimator increases. The value of the *CPC* model in the discriminant analysis context seems to be in situations where there are common eigenvectors in the population covariance matrices. In this case the *CPC* estimators will generally approximate the shape of the population covariance matrices better than the pooled covariance matrix estimator, and will give more precise estimates than when using the unbiased sample covariance matrices, particularly for smaller samples.

6.5.3 $p = 10$ variables case

Lastly, a simulation experiment was performed for $k = 2$ multivariate normally distributed populations with $p = 10$ variables for each of the following population covariance matrix scenarios:

- 1) **Equal covariance matrices** (Σ_{EQUAL})
- 2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME})
- 3) **CPC: Similar rank orders of the common eigenvectors** (Σ_{SIMILAR})
- 4) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE})
- 5) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

The population covariance matrices are given in Appendix D. For each of the populations in the $p = 10$ variables case, the mean vectors were specified as

$$\mu'_1 = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0],$$

and

$$\mu'_2 = [2, 2, 2, 2, 2, 2, 2, 2, 2, 2].$$

Misclassification error rates calculated from the 30% test samples per simulation run, together with the mean ranks of the discriminant functions

in each of the scenarios, are reported in Table 6.3. A total of 1000 simulation runs were used for each of the ($n_i \times$ five covariance structures considered) scenarios.

As in the $p = 2, 5$ cases, *LDA* had the smallest misclassification error rate when the population covariance matrices were equal. *QDA* performed significantly worse than the other discriminant functions in this scenario.

For CPC with the same rank order of the common eigenvectors in the population covariance matrices, *CPC* gave the smallest misclassification error rate, followed closely by *CPC**. When the rank orders of the common eigenvectors in the population covariance matrices were similar, *CPC* and *CPC** again performed the best, as was also found in the $p = 2$ and 5 cases.

With opposite rank orders of the common eigenvectors in the two population covariance matrices, the *CPC* and *CPC** discriminant functions clearly gave the smallest misclassification error rates. This is the situation where CPC discrimination offers the greatest advantage over *QDA* and *LDA*.

Table 6.3: Simulation results for $k = 2$ samples of equal sizes drawn from multivariate normally distributed populations with $p = 10$ variables. Each of the values in the table were calculated from 1000 simulation runs. Standard errors are given in brackets.

Structure	n_i	Misclassification error (%)							
		QDA		CPC		CPC*		LDA	
Σ_{EQUAL}	30	42.06	(1.56)	33.88	(1.50)	34.48	(1.50)	32.72	(1.48)
	50	37.79	(1.53)	31.65	(1.47)	31.67	(1.47)	30.68	(1.46)
	100	34.01	(1.50)	29.25	(1.44)	29.53	(1.44)	28.44	(1.43)
	200	31.27	(1.47)	28.25	(1.42)	28.35	(1.43)	27.70	(1.42)
Σ_{SAME}	30	30.27	(1.45)	20.03	(1.27)	20.54	(1.28)	34.01	(1.50)
	50	23.94	(1.35)	17.49	(1.20)	17.71	(1.21)	31.39	(1.47)
	100	19.85	(1.26)	16.32	(1.17)	16.63	(1.18)	29.73	(1.45)
	200	17.29	(1.20)	15.69	(1.15)	15.69	(1.15)	28.44	(1.43)
Σ_{SIMILAR}	30	28.58	(1.43)	18.12	(1.22)	18.77	(1.23)	33.52	(1.49)
	50	22.96	(1.33)	16.50	(1.17)	16.81	(1.18)	30.43	(1.45)
	100	18.12	(1.22)	14.93	(1.13)	15.08	(1.13)	28.80	(1.43)
	200	15.89	(1.16)	14.13	(1.10)	14.26	(1.11)	27.49	(1.41)
Σ_{OPPOSITE}	30	5.20	(0.70)	2.28	(0.47)	2.46	(0.49)	24.73	(1.36)
	50	3.31	(0.57)	2.15	(0.46)	2.22	(0.47)	21.55	(1.30)
	100	2.41	(0.48)	1.95	(0.44)	1.97	(0.44)	18.31	(1.22)
	200	1.99	(0.44)	1.84	(0.42)	1.85	(0.43)	16.56	(1.18)
$\Sigma_{\text{UNRELATED}}$	30	13.78	(1.09)	8.94	(0.90)	8.47	(0.88)	34.93	(1.51)
	50	8.66	(0.89)	8.14	(0.86)	6.94	(0.80)	32.94	(1.49)
	100	5.85	(0.74)	7.15	(0.81)	5.57	(0.73)	30.80	(1.46)
	200	4.89	(0.68)	6.95	(0.80)	4.92	(0.68)	29.76	(1.45)

Flury et al. (1994) and Bianco et al. (2008) found CPC discrimination and QDA to perform equally well for unrelated population covariance matrices. However, in this simulation experiment for populations with $p = 10$ variables, *CPC** fared the best in the unrelated covariance matrices scenario, except for the largest sample size considered ($n_i = 200$) where it was outperformed slightly by QDA. There may be two reasons for this surprising result: Firstly, *CPC** employs a more parsimonious covariance matrix model than QDA. Thus, even though the CPC model is theoretically incorrect in this case, the reduction in number of parameters to estimate makes the estimation process more stable, particularly for smaller sample sizes. Secondly, by using appropriately large values for the shrinkage intensity parameter in (5.10), the *Full CPC crossvalid* estimator (used in *CPC**) can model the unrelated covariance matrices as accurately as the *Unbiased* estimator (used in QDA). However, as the sample sizes increase, the *Unbiased* covariance matrix estimators become more accurate in estimation of the population covariance matrices.

LDA gives the largest misclassification error rate when the covariance matrices are unrelated. This result concurs with what was found in the simulation studies by Flury et al. (1994) and Bianco et al. (2008).

6.6 Application to the VON data

The quadratic discriminant function depends on the assumption that the populations have multivariate normal distributions. As the normality assumption is not valid for the VON data, applying CPC discrimination to the delivery mode, regional, and mortality groupings in the VON 2009 cohort will not be theoretically correct. In addition, because the sample sizes of the VON groups are relatively large compared to the number of variables, the advantage (if any) of CPC discrimination over ordinary quadratic discrimination should be small.

However, it is interesting to compare the results from the QDA, CPC, *CPC** and LDA classification rules on the VON data, which is the reason for including it here. To obtain realistic estimates of the misclassification error rates for each classification rule, each of the VON groups were divided into a 70% training set and a 30% test set. The discriminant functions were estimated on the training data, and the misclassification error rates were calculated on the independent test data.

For the analysis of the delivery mode groups and the regional groups, infants who died before final hospital discharge and those who were transferred to alternative NICU facilities were included. Infants who were transferred

were excluded from the analysis of the mortality groups, as babies with poorer health were probably more likely to be transferred (and perhaps die). This means that there is an association between transferal status and mortality status, but the final outcomes (death/survival) of the transferred infants are not indicated in this data set.

6.6.1 Delivery mode

In Chapter 4, the covariance matrices of the delivery mode groups (*Caesarean* and *Vaginal*) were seen to have three common eigenvectors. However, because it was found in Chapter 5 that the *Full CPC crossvalid* estimator fares the best in estimating the population covariance matrices even in the $CPC(q)$ situation, the common eigenvector matrix for the delivery groups were estimated with the FG algorithm under the assumption of full CPC:

$$B = \begin{bmatrix} 0.15 & -0.02 & 0.03 & -0.01 & 0.06 & 0.99 \\ 0.13 & 0.78 & 0.05 & -0.61 & -0.03 & -0.01 \\ 0.10 & 0.60 & -0.03 & 0.79 & -0.04 & 0.01 \\ 0.72 & -0.10 & -0.68 & -0.04 & -0.03 & -0.09 \\ 0.66 & -0.13 & 0.73 & 0.03 & -0.05 & -0.13 \\ 0.05 & 0.04 & 0.01 & 0.01 & 1.00 & -0.07 \end{bmatrix}.$$

The results from the four discriminant functions on the 30% test data set were as follows: *QDA* gave the smallest misclassification error rate of 25.8% (standard error: 1.45%), and *LDA* had the largest error rate of 35.4% (standard error: 1.58%). The *CPC* and *CPC** discriminant functions were in between, with error rates of 28.4% (standard error: 1.49%) and 26.8% (standard error: 1.47%), respectively. Thus, while *QDA*, *CPC* and *CPC** all led to significantly smaller misclassification error rates than *LDA*, these three methods did not differ significantly from each other.

6.6.2 Regions

It was concluded that the covariance matrices of the *South Africa* and *Namibia* groups in the VON 2009 cohort share six common eigenvectors. This implies the full CPC situation, and for the CPC discrimination procedure demonstrated here, the common eigenvector matrix estimated with the FG algorithm (given in Section 3.11.2) was used.

CPC and *CPC** performed the best for the regional groups, with misclassification error rates of 21.2% (standard error: 1.35%) and 22.9% (standard error: 1.39%), respectively, on the test data set. *QDA* gave an error rate of 25.2% (standard error: 1.44%), and *LDA* performed the worst of the four

with an error rate of 25.4% (standard error: 1.44%). The misclassification error rates of the four methods therefore did not differ significantly from each other with regard to the regional groups in the VON 2009 cohort.

6.6.3 Mortality

Neonatal mortality is one of the important clinical outcomes in the VON research context. The infants in the VON 2009 cohort were divided in the group that *Survived* ($n_1 = 2826$, excluding infants who were transferred) and the group that *Died* ($n_2 = 104$). The covariance matrices for these two groups are shown below:

- *Survived* ($n_1 = 2826$)

$$\mathbf{S}_1 = \begin{bmatrix} 0.64 & 0.23 & 0.18 & 2.15 & 2.04 & 0.16 \\ 0.23 & 3.41 & 2.10 & 0.93 & 0.95 & 0.22 \\ 0.18 & 2.10 & 2.19 & 0.82 & 0.72 & 0.19 \\ 2.15 & 0.93 & 0.82 & 10.85 & 7.90 & 0.53 \\ 2.04 & 0.95 & 0.72 & 7.90 & 9.23 & 0.51 \\ 0.16 & 0.22 & 0.19 & 0.53 & 0.51 & 0.50 \end{bmatrix},$$

- *Died* ($n_2 = 104$)

$$\mathbf{S}_2 = \begin{bmatrix} 0.78 & 0.42 & 0.34 & 4.12 & 4.12 & 0.50 \\ 0.42 & 6.35 & 4.97 & 3.43 & 1.95 & 0.57 \\ 0.34 & 4.97 & 5.27 & 3.15 & 1.62 & 0.74 \\ 4.12 & 3.43 & 3.15 & 26.78 & 24.08 & 3.08 \\ 4.12 & 1.95 & 1.62 & 24.08 & 27.90 & 2.97 \\ 0.50 & 0.57 & 0.74 & 3.08 & 2.97 & 1.60 \end{bmatrix}.$$

Using the multivariate Shapiro-Wilk test for normality, the normality assumption seems untenable for both the *Survived* ($p < 0.0001$) and *Died* ($p < 0.0001$) groups.

There is strong evidence against the equal covariance matrices hypothesis ($p < 0.0001$ for Box's M test), and the AIC method indicates the unrelated covariance matrices model as the most appropriate. However, the *Ensemble* method indicates six common eigenvectors (thus full CPC). The common eigenvector matrix, estimated with the FG algorithm, is

$$\mathbf{B} = \begin{bmatrix} 0.16 & -0.02 & 0.03 & 0.08 & -0.01 & 0.98 \\ 0.10 & 0.79 & 0.02 & 0.01 & -0.60 & -0.01 \\ 0.08 & 0.60 & -0.04 & -0.10 & 0.79 & 0.01 \\ 0.72 & -0.10 & -0.68 & -0.03 & -0.03 & -0.10 \\ 0.66 & -0.08 & 0.73 & -0.05 & 0.03 & -0.13 \\ 0.04 & 0.04 & 0.01 & 0.99 & 0.08 & -0.09 \end{bmatrix}.$$

*CPC** had the smallest misclassification error rate of 15.5% (standard error: 1.22%) on the test data set. *QDA* and *LDA* both gave error rates of 15.9% (standard errors: 1.23% and 1.23%, respectively), and *CPC* performed the worst with a misclassification error rate of 16.6% (standard error: 1.26%). However, as can be seen from the standard errors, the misclassification error rates of the four methods were not significantly different from each other.

Chapter 7

CPC biplots

7.1 Introduction

A *biplot* is the simultaneous display of the rows and columns of a matrix, \mathbf{X} . The “bi-” prefix refers to the fact that both the observations (rows of \mathbf{X}) and the variables (columns of \mathbf{X}) are represented in the display, usually as points and arrows/axes, respectively. It is a descriptive statistical technique, and is often used as either a precursor or complement to more formal statistical analyses of the data. A good introduction to biplots is given in the book by Gower et al. (2011).

The display can be of one-, two- or three-dimensional form, and therefore generally involves approximation of the elements of \mathbf{X} (in a p -variate space) in a lower-dimensional subspace. Multidimensional scaling techniques are employed to reduce the dimensionality of the data set while preserving the maximum possible proportion of the variation in \mathbf{X} . A reduction in dimensionality inevitably involves a loss of information, and a good biplot solution aims to minimise this loss.

To obtain a biplot solution for \mathbf{X} , the general approach is to compute the singular value decomposition (SVD) of \mathbf{X} ,

$$\mathbf{X} = \mathbf{U}\mathbf{L}^{\frac{1}{2}}\mathbf{V}', \quad (7.1)$$

where \mathbf{U} and \mathbf{V} contain the left- and right-singular vectors of \mathbf{X} , respectively, and $\mathbf{L}^{\frac{1}{2}}$ is a diagonal matrix with the singular values of \mathbf{X} on the diagonal. The $(m, j)^{th}$ element of \mathbf{X} is thus expressed as the product,

$$x_{mj} = \mathbf{u}_{(m)}\mathbf{L}^{\frac{1}{2}}\mathbf{v}'_{(j)}, \quad m = 1, \dots, n, \quad j = 1, \dots, p, \quad (7.2)$$

where $\mathbf{u}_{(m)}$ and $\mathbf{v}_{(j)}$ indicate the m^{th} row of \mathbf{U} and the j^{th} row of \mathbf{V} , respectively.

Note that the squared eigenvalues of \mathbf{X} are the eigenvalues of $\mathbf{X}'\mathbf{X}$. The SVD of $\mathbf{X}'\mathbf{X}$ can thus be written as

$$\mathbf{X}'\mathbf{X} = \mathbf{U}\mathbf{L}\mathbf{V}', \quad (7.3)$$

where $\mathbf{U} = \mathbf{V}$ because $\mathbf{X}'\mathbf{X}$ is a symmetric matrix.

The coordinates for the rows (observations) of \mathbf{X} in an r -dimensional biplot are given by the first r columns of

$$\mathbf{X}\mathbf{V} = \mathbf{U}\mathbf{L}^{\frac{1}{2}}, \quad (7.4)$$

but because (7.1) can also be written as

$$\mathbf{X} = (\mathbf{U}\mathbf{L}^{\frac{1}{2}-\alpha})(\mathbf{V}\mathbf{L}^{\alpha}), \quad (7.5)$$

where $\alpha \in [0, 0.5]$, (7.4) is but one of an infinite number of solutions, which can be written more generally as, (Gower et al., 2011),

$$\mathbf{X}\mathbf{V}\mathbf{L}^{\alpha} = \mathbf{U}\mathbf{L}^{\frac{1}{2}-\alpha}. \quad (7.6)$$

The coordinates for the columns (variables) of \mathbf{X} in an r -dimensional biplot are given by the first r columns of

$$\mathbf{X}'\mathbf{U}\mathbf{L}^{\frac{1}{2}-\alpha} = \mathbf{V}\mathbf{L}^{\alpha}. \quad (7.7)$$

Different values for α in (7.6) and (7.7) lead to different biplots, in which the display quality of the rows and columns of \mathbf{X} vary. For $\alpha = 0$, the rows of \mathbf{X} are displayed optimally, while $\alpha = 0.5$ leads to a biplot in which the columns of \mathbf{X} are displayed optimally.

As a dimension reduction technique, PCA plays an integral part in the *principal component analysis (PCA) biplot* developed by Gabriel (1971). The PCA biplot is briefly introduced in Section 7.2. The PCA biplot is constructed from the eigenvectors of the covariance matrix of \mathbf{X} , and can thus be seen as a biplot of $\mathbf{X}'\mathbf{X}$ as, for column centred matrix \mathbf{X} ,

$$\mathbf{S} = \frac{1}{n-1} \mathbf{X}'\mathbf{X}. \quad (7.8)$$

As $\mathbf{X}'\mathbf{X}$ is a symmetric matrix of size $p \times p$, the PCA biplot derived from $\mathbf{X}'\mathbf{X}$ is in effect a display of only the columns (variables) of \mathbf{X} , which is technically known as a *monoplot* (Gower et al., 2011). In this approach the focus is on displaying the covariance structure of \mathbf{X} instead of the elements of \mathbf{X} itself. It is a subtle distinction, as the monoplot constructed from the elements of $\mathbf{X}'\mathbf{X}$ (with representations of the observations superimposed in the display) and the true biplot constructed from the elements of \mathbf{X} may

look similar. The CPC model is concerned with the covariance matrices of several populations, and the PCA “monoplot” approach is therefore taken for the purpose of this dissertation.

Gardner (2001) briefly mentioned the idea of constructing biplots under the CPC model, but did not explore the topic any further. Williams (2013) used the dependent CPC model, proposed by Neuenschwander and Flury (2000), to construct biplots for longitudinal data. Measurements were made on the same experimental group at different time points, and the data at each time point were treated as a separate though dependent group.

To our knowledge, no attempts have been made yet to use the CPC model to construct biplots for data with distinct groups of independent observations. In Section 7.3, different types of principal component (PC) biplots for data with distinct groups are discussed, and an approach to construct CPC biplots using common eigenvectors estimated by any of the simultaneous diagonalisation algorithms discussed in Chapter 3 is proposed. Note that a distinction is made between the terms “PCA biplot” and “PC biplot”: In this chapter, “PCA biplot” refers to the well-known PCA biplot proposed by Gabriel (1971), while the term “PC biplot” is used for a biplot constructed from a set of eigenvectors. “PC biplot” thus includes both PCA and CPC biplots, among others.

Measures of fit for PCA biplots were given by Gardner-Lubbe et al. (2008) and Gower et al. (2011). Various quality measures for PCA biplots and canonical variate analysis (CVA) biplots were also discussed by Brand (2013). For data with distinct groups, the purpose of CVA is to find linear functions which optimally separate the different groups. CVA therefore differs from CPC analysis in the sense that the former is concerned with optimal separation *between* groups, whereas the latter is concerned merely with modelling the covariance structure *within* groups. Various biplot quality measures are investigated in Section 7.4 to determine which are appropriate to evaluate the quality of a CPC biplot display, and to compare the quality of a CPC biplot to other types of PC biplots for any specific data set.

If it is of interest to distinguish between groups (i.e. to investigate the distances between different groups) when the group centroids differ greatly compared to the within-group variation, a biplot using the covariance matrix of the pooled data should be most appropriate. The first principal component of the pooled data will in this case usually account for the between-group distances, with the second principal component accounting for the direction of the largest within-in group differences.

However, if it is of interest to compare specific characteristics within groups (to determine how these differ between the groups), a CPC biplot may be useful, as the within-group and between-group variation are con-

founded in the pooled data biplot. In the study of allometry, for example, comparing size and shape between the sets of morphometric measurements made on different groups may be informative (see Klingenberg, 1996, for an example). In such a case, a CPC biplot may enable comparison of within-group traits better than a biplot based on the covariance matrix of the pooled data or the pooled covariance matrix.

Details of an R function to compare different types of PC biplots with regard to the various quality measures, and to perform a data-based selection of the most appropriate PC biplot type for any specific data set are discussed in Section 7.5. Use of the function is demonstrated on simulated data and two well-known data sets from the literature.

Lastly, an application of the work outlined in this chapter to the delivery mode, regional, and mortality groupings in the VON 2009 cohort are presented in Section 7.6.

7.2 PCA biplots

The PCA biplot was proposed by Gabriel (1971) and the theory extensively documented and extended in Gower and Hand (1996) and Gower et al. (2011). The basic idea is to use the first $r < p$ (usually $r = 2$ or 3) principal components of a data set to optimally represent the variation in the original p -variate data in an r -dimensional subspace.

For a single group of observations in data matrix, \mathbf{X} , with sample covariance matrix, \mathbf{S} , let \mathbf{E}_r be the $p \times r$ matrix containing the first r eigenvectors of \mathbf{S} . The first r principal component scores for the rows of \mathbf{X} ,

$$\mathbf{Y}_r = \mathbf{X}\mathbf{E}_r, \quad (7.9)$$

are used as coordinates to plot the observations as points in the biplot display. The biplot axes should be scaled equally to avoid further distortion of the approximated distances between the observations in \mathbf{X} (Gower and Hand, 1996). Because the eigenvectors in \mathbf{E}_r are orthogonal to each other, a biplot using the eigenvectors as the orthogonal axes introduces no additional distortion in the representation of the data.

The variables (columns of \mathbf{X}) can be represented in the biplot display as vector arrows from the origin. The lengths and directions of the variable vectors are given by the rows of \mathbf{E}_r . This approach, proposed by Gabriel (1971), was refined by Gower and Hand (1996) by extending the variable vectors in both directions from the origin across the entire biplot, and adding calibration markers to these axes. However, in this dissertation the approach of Gabriel (1971) is followed as the purpose is merely to investigate whether

a biplot under the CPC model offers any advantage over the ordinary PCA biplot.

To aid in the interpretation of the biplot, it is helpful to centre the columns of \mathbf{X} to have zero mean. The centroid of the observations will in this case coincide with the origin of the display from which the variable vectors are drawn.

7.3 PC biplots for data with distinct groups

PC biplots for data with distinct groups can be constructed in a number of different ways. The primary difference between the methods considered here is the way in which the projection matrix (i.e. the eigenvectors used to construct the biplot) is estimated.

For a biplot of data from several distinct groups, the choice of projection matrix is not as simple as in the single group case. In addition to the within-group variation, the between-group variation should also be represented adequately in the biplot. Depending on the specific application, different priorities can be given to the representation of these two sources of variation. If the purpose of the biplot is to obtain optimal separation between the groups, little consideration might be given to the quality of representation of the within-group variation. If it is of interest to compare the variation within each group, a biplot which represents this aspect optimally should be constructed.

The following sections briefly explain three possible ways of estimating the biplot projection matrix for data with distinct groups.

7.3.1 Pooled covariance matrix

To obtain a good representation of the within-group variation, the biplot can be constructed using the eigenvectors of the pooled covariance matrix, \mathbf{S}_p in (4.2), under the assumption that $\mathbf{\Sigma}_1 = \dots = \mathbf{\Sigma}_k$. The pooled covariance matrix accounts only for the within-group variation, with the consequence that the between-group distances may be displayed very poorly in such a biplot, depending on the directions of the differences between the group centroids.

Spectral decomposition of \mathbf{S}_p ,

$$\mathbf{D}_p = \mathbf{E}_p' \mathbf{S}_p \mathbf{E}_p, \quad (7.10)$$

gives the eigenvector matrix, \mathbf{E}_p . \mathbf{D}_p is a diagonal matrix containing the eigenvalues of \mathbf{S}_p on the diagonal. For a biplot using the pooled covariance

matrix approach, a matrix containing the first r columns of \mathbf{E}_p is used as the projection matrix in (7.9).

7.3.2 Pooled data

A biplot can be constructed from the pooled data by pooling the (uncentred) data from several groups. Let $\mathbf{X}_i, i = 1, \dots, k$ indicate the data matrix of the i^{th} group, and let $n = \sum_{i=1}^k n_i$. From the $n \times p$ pooled data matrix,

$$\mathbf{X}_{\text{pool}} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_k \end{bmatrix}, \quad (7.11)$$

the covariance matrix,

$$\mathbf{S}_{\text{pool}} = \frac{1}{n-1} (\mathbf{X}'_{\text{pool}} \mathbf{X}_{\text{pool}} - n \bar{\mathbf{x}}_{\text{pool}} \bar{\mathbf{x}}'_{\text{pool}}), \quad (7.12)$$

is calculated, where $\bar{\mathbf{x}}_{\text{pool}}$ indicates a vector containing the column means of \mathbf{X}_{pool} . The columns of \mathbf{X}_{pool} can be centred for the construction of the biplot, as the distances between the group centroids will be preserved if the centering is performed over \mathbf{X}_{pool} , rather than for each group separately.

Through spectral decomposition of \mathbf{S}_{pool} ,

$$\mathbf{D}_{\text{pool}} = \mathbf{E}'_{\text{pool}} \mathbf{S}_{\text{pool}} \mathbf{E}_{\text{pool}}, \quad (7.13)$$

where \mathbf{D}_{pool} indicates the diagonal matrix with the eigenvalues of \mathbf{S}_{pool} on the diagonal, the eigenvector matrix of the covariance matrix of the pooled data, \mathbf{E}_{pool} , is obtained. To construct the biplot using this approach, the first r columns of \mathbf{E}_{pool} is used as the projection matrix.

The \mathbf{S}_{pool} covariance matrix describes both within-group and between-group variation. If \mathbf{S}_{pool} is dominated by the between-group variation, the eigenvector solution and resulting biplot will predominantly reflect this source, giving more weight to an optimal separation between the groups. However, if the group centroids are relatively close together (compared to the within-group variation), \mathbf{S}_{pool} will be dominated by the within-group variation and the biplot will be very similar to the pooled covariance matrix biplot described in the previous section.

7.3.3 Common eigenvectors

If the assumption of equal population covariance matrices seems untenable, and the aim is to compare the within-group variation and not only the between-group distances, the CPC biplot is another alternative to the usual PCA biplot. To construct a CPC biplot for data with distinct groups, the modal matrix, \mathbf{B} , is calculated under the assumption of common eigenvectors, using any one of the simultaneous diagonalisation algorithms described in Section 3.5. Let \mathbf{B}_r be a $p \times r$ matrix containing the r common eigenvectors used in the construction of the CPC biplot. Plugging \mathbf{B}_r as the projection matrix into (7.9), the scores for the first r common principal components are given by

$$\mathbf{Z}_r = \mathbf{X}_{\text{pool}} \mathbf{B}_r, \quad (7.14)$$

with \mathbf{X}_{pool} defined as in (7.11). For optimal representation of the within-group variation in \mathbf{X} , care should be taken to include in \mathbf{B}_r the common eigenvectors which simultaneously account for the most variation in all k groups. If the common eigenvectors used as the biplot axes do not correspond to the largest eigenvalues of a specific group, the (within-group) quality of the biplot display for that group will be poor (Gardner, 2001).

7.4 Measures of fit for different types of PC biplots

Projecting higher-dimensional data onto a lower-dimensional subspace always involves a loss of information, and the quality (or goodness of fit) of the lower-dimensional biplot representation can be measured in a number of ways. The quality of a biplot display for data with distinct groups can be judged with regard to the following aspects:

- a) how well the variation (or distances) between the *observations* are represented,
- b) how well the original *variables* are represented, and
- c) how well the variation between *groups* is represented.

Measures to judge these aspects for different types of PC biplots are given in the following sections. Gardner-Lubbe et al. (2008), Gower et al. (2011) and Brand (2013) also addressed the first two aspects with regard to PCA and CVA biplots.

7.4.1 Overall quality

For a PC biplot of a data matrix, \mathbf{X} , with sample covariance matrix, \mathbf{S} , Gower and Hand (1996), and Cox and Cox (2010) measured the overall quality of the biplot as the proportion of the total variation in the data accounted for by the first r principal components (where r is the number of dimensions used in the lower-dimensional projection). This is given by

$$\text{Overall quality} = \frac{\sum_{j=1}^r d_j}{\sum_{j=1}^p d_j}, \quad (7.15)$$

where the d_j are the eigenvalues of \mathbf{S} , in decreasing order.

For data with distinct groups, \mathbf{X} can be the combined (uncentred) data matrix for all k groups. In this case, the overall quality measure in (7.15) will include both between-group and within-group variation, and its value will therefore be dominated by the greater of these two sources.

Gower and Hand (1996) showed that, under orthogonality conditions, (7.15) is the proportion of the total variation in \mathbf{X} represented in an r -dimensional biplot. If the columns of \mathbf{X} are centred to have zero mean, and letting \mathbf{E}_r indicate the $p \times r$ orthogonal projection matrix, the coordinates of the observations as projected in the biplot subspace is given by (7.9).

The total variation in the data can be calculated as

$$SS_{\text{TOTAL}} = \text{tr}(\mathbf{X}'\mathbf{X}), \quad (7.16)$$

and the fitted variation as represented in the biplot approximation is given by

$$SS_{\text{FITTED}} = \text{tr}(\mathbf{Y}'_r \mathbf{Y}_r). \quad (7.17)$$

The overall quality of the biplot display can thus be calculated as

$$\text{Overall quality} = \frac{SS_{\text{FITTED}}}{SS_{\text{TOTAL}}}, \quad (7.18)$$

which is equivalent to the measure given in (7.15).

7.4.2 Within-group quality

Gardner (2001) extended the measure in (7.15) to the CPC biplot case and proposed calculating the quality of the display for each group individually. With this approach the assumption is made that there is an orthogonal matrix, \mathbf{B} , which diagonalises the k covariance matrices simultaneously. For a projection of the p -dimensional data onto an r -dimensional subspace, the first

r common eigenvectors in \mathbf{B} , indicated by \mathbf{B}_r , can be used. The goodness of fit with respect to the i^{th} group in the CPC biplot is given by

$$\text{Quality for } i^{th} \text{ group} = \frac{\sum_{j=1}^r l_{ij}}{\sum_{j=1}^p d_{ij}}, \quad i = 1, \dots, k, \quad (7.19)$$

where the l_{ij} are the estimators of the eigenvalues of \mathbf{S}_i under the CPC model, the d_{ij} are the eigenvalues of \mathbf{S}_i , with the r eigenvalues in the numerator corresponding to the common eigenvectors used for constructing the biplot display. Equation (7.19) measures the proportion of within-group variation of the i^{th} group accounted for in the biplot subspace. If the common eigenvectors used for the construction of the CPC biplot are not associated with the largest eigenvalues of a specific group, that group will have a relatively low value for (7.19) and will not be represented well in the biplot (Gardner, 2001).

Using the approach of Gower and Hand (1996), letting \mathbf{X}_i and $\mathbf{Y}_i = \mathbf{X}_i \mathbf{B}_r$ indicate the original centred data and the fitted values for the i^{th} group, respectively, the total within-group variation of the i^{th} group is given by

$$SS_{i(\text{TOTAL})} = \text{tr}(\mathbf{X}_i' \mathbf{X}_i), \quad (7.20)$$

and the fitted variation as represented in the biplot approximation is given by

$$SS_{i(\text{FITTED})} = \text{tr}(\mathbf{Y}_i' \mathbf{Y}_i). \quad (7.21)$$

The within-group quality of the biplot display for the i^{th} group can be calculated as

$$\text{Within-group quality (Group } i) = \frac{SS_{i(\text{FITTED})}}{SS_{i(\text{TOTAL})}}, \quad i = 1, \dots, k. \quad (7.22)$$

The within-group quality of the CPC biplot display depends on the projection quality of each of the k groups in the r -dimensional subspace. It is now proposed that a combined measure of the within-group variation represented in the biplot display be calculated as

$$\begin{aligned} \text{Within-group quality} &= \frac{\sum_{i=1}^k SS_{i(\text{FITTED})}}{\sum_{i=1}^k SS_{i(\text{TOTAL})}} \\ &= \frac{\sum_{i=1}^k \sum_{j=1}^r l_{ij}}{\sum_{i=1}^k \sum_{j=1}^p d_{ij}}. \end{aligned} \quad (7.23)$$

7.4.3 Between-group quality

If the purpose of the biplot is to examine the differences between several groups of data, it is of interest to measure how well the true distances between the group centroids are approximated in the display. Let \mathbf{X} contain the uncentred data for all k groups combined, and \mathbf{X}_i the uncentred data for the i^{th} group, respectively. Let

$$\bar{\mathbf{x}} = \mathbf{X}'\mathbf{j} \frac{1}{\sum_{i=1}^k n_i} \quad (7.24)$$

and

$$\bar{\mathbf{x}}_i = \mathbf{X}_i'\mathbf{j} \frac{1}{n_i} \quad (7.25)$$

be the vectors containing the column means of \mathbf{X} and \mathbf{X}_i , respectively, where \mathbf{j} indicates a column vector of n ones. The total between-group sums of squares can be calculated as

$$\sum_{i=1}^k (\bar{\mathbf{x}}_i - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}). \quad (7.26)$$

For the $p \times r$ orthogonal projection matrix \mathbf{B}_r , let $\bar{\mathbf{x}}_r = \mathbf{B}_r'\bar{\mathbf{x}}$ and $\bar{\mathbf{x}}_{ir} = \mathbf{B}_r'\bar{\mathbf{x}}_i$ be the overall mean vector and the mean vector for the i^{th} group as represented in the r -dimensional subspace, respectively. The fitted between-group sums of squares as approximated in the biplot display is given by

$$\sum_{i=1}^k (\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)'(\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r). \quad (7.27)$$

It is now proposed that the quality of the between-group variation of a PC biplot of data with distinct groups be calculated as

$$\text{Between-group quality} = \frac{\sum_{i=1}^k (\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)'(\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)}{\sum_{i=1}^k (\bar{\mathbf{x}}_i - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})}, \quad (7.28)$$

which is the between-group variation in the r -dimensional display expressed as a proportion of the between-group variation in the original p -dimensional space. Note that the measure in (7.28) will be dominated by groups located relatively far (compared to the other groups) from the overall centroid. To clarify, consider the situation where p -variate samples from three populations are available: If the sample size from the first population is small and has a centroid which is relatively far from the overall centroid (of

all the data combined), and the sample sizes from the other two populations are large, $(\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)'(\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)$ and $(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})$ will be much larger for the first group than for the other two groups, and the first group will therefore dominate the between-group quality measure in (7.28). Thus if the sample sizes of the groups differ considerably, a large value for (7.28) does not necessarily indicate that the distances from the group centroids to the overall centroid are displayed well for all of the groups.

One solution to this problem is to use the sample sizes to weigh the contribution of each group to the between-group quality in (7.28). A weighted measure for between-group quality can be calculated as

$$\text{Weighted between-group quality} = \frac{\sum_{i=1}^k n_i (\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)'(\bar{\mathbf{x}}_{ir} - \bar{\mathbf{x}}_r)}{\sum_{i=1}^k n_i (\bar{\mathbf{x}}_i - \bar{\mathbf{x}})'(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})}. \quad (7.29)$$

7.4.4 Adequacy of the variables

In addition to the quality of display of the observations and groups, it should be considered how well the variables are represented in the r -dimensional biplot, referred to as the *adequacy* of the variables by Gower and Hand (1996).

To measure the adequacy of a p -dimensional variable vector in an r -dimensional subspace, the length of the vector as projected in the r -dimensional subspace is compared to the length of the vector in the original p -dimensional space. For a $p \times p$ orthogonal projection matrix, \mathbf{B} , calculated with any of the diagonalisation algorithms as discussed in Section 3.5, each of the column vectors of \mathbf{B} has unit length. The rows of \mathbf{B} contain the loadings of the p original variables on each of the p orthogonal axes and are also of unit length (Rencher, 1998; Gower et al., 2011).

The length of the j^{th} variable vector in the r -dimensional subspace is equal to the sum of the squared loadings of the j^{th} variable in the associated r eigenvectors. Let \mathbf{B}_r indicate the $p \times r$ projection matrix containing r columns of \mathbf{B} . The adequacies of the variables in the selected r -dimensional subspace are given by the diagonal elements of

$$\mathbf{B}_r \mathbf{B}_r'. \quad (7.30)$$

These vector lengths are in the range $[0; 1]$ due to the row vectors having unit length, with greater values indicating better adequacies for the corresponding variables in the biplot display.

The mean adequacy over all p variables will always be equal to $\frac{r}{p}$ for an r -dimensional representation, given that the columns of \mathbf{B} are orthonormal.

It is thus uninformative to compute the mean adequacy of the variables. As a composite measure to compare the adequacies of variables for different types of PC biplots, it is now proposed to calculate the median of the adequacy values,

$$\text{Overall adequacy} = \text{median}[\text{diag}(\mathbf{B}_r \mathbf{B}_r')], \quad (7.31)$$

which is not a constant function of r and p , as the mean adequacy is. A greater overall adequacy is interpreted as a better overall representation of the variables in the biplot display.

7.4.5 Axis predictivities

The terms “prediction” and “interpolation” can be used in reference to biplots (see Gower et al., 2011), but they have different meanings than in the context of regression modelling, for example. With biplots, *prediction* refers to the orthogonal projection of observations onto calibrated axes (representing the variables) in the biplot display, in order to read off the values for the variables from these axes. *Interpolation* refers to the positioning of a new observation (point) on the biplot display in the place at which the values for this observation, projected orthogonally from the calibrated axes, intersect.

For predictive purposes, Gardner-Lubbe et al. (2008) and Gower et al. (2011) discourage the use of adequacy to judge the goodness of fit of variables represented in a PCA biplot and recommend the use of a measure called *axis predictivity* instead. To calculate the axis predictivities for a biplot of the data in \mathbf{X} using the eigenvector (projection) matrix \mathbf{E} , let

$$\mathbf{J} = \begin{bmatrix} \mathbf{I}_r & 0 \\ 0 & 0 \end{bmatrix}, \quad (7.32)$$

where \mathbf{I}_r is a $r \times r$ identity matrix, and calculate the fitted values in the biplot as

$$\hat{\mathbf{X}} = \mathbf{X} \mathbf{E} \mathbf{J} \mathbf{E}'. \quad (7.33)$$

The axis predictivities for the biplot are given by the p diagonal elements of (Gardner-Lubbe et al., 2008)

$$\mathbf{\Pi} = \text{diag}(\hat{\mathbf{X}}' \hat{\mathbf{X}}) [\text{diag}(\mathbf{X}' \mathbf{X})]^{-1}. \quad (7.34)$$

Gower et al. (2011) showed that (7.34) can be written in terms of the singular value decomposition of $\mathbf{X}' \mathbf{X}$ as

$$\mathbf{\Pi} = \text{diag}(\mathbf{V} \mathbf{\Lambda} \mathbf{J} \mathbf{V}') [\text{diag}(\mathbf{V} \mathbf{\Lambda} \mathbf{V}')]^{-1}, \quad (7.35)$$

where \mathbf{V} is the matrix of right eigenvectors of $\mathbf{X}'\mathbf{X}$ and $\mathbf{\Lambda}$ is the matrix containing the eigenvalues of $\mathbf{X}'\mathbf{X}$ on the diagonal. Gower et al. (2011) have shown that, because axis predictivity is the variance accounted for per variable, the overall quality as defined in Section 7.4.1 can also be written as the sum of the weighted predictivities, i.e.

$$\text{Overall quality} = \sum_{j=1}^p w_j \pi_j, \quad (7.36)$$

where π_j is the j^{th} diagonal element of $\mathbf{\Pi}$, and the weight of the j^{th} variable given by

$$w_j = \frac{(\mathbf{V}\mathbf{\Lambda}\mathbf{V}')_{jj}}{\text{tr}(\mathbf{\Lambda})}, \quad (7.37)$$

where $(\mathbf{V}\mathbf{\Lambda}\mathbf{V}')_{jj}$ indicates the j^{th} diagonal element of $\mathbf{V}\mathbf{\Lambda}\mathbf{V}'$.

Thus the overall quality measure in (7.18) also gives an indication of the goodness of fit of the variables, although it is a much cruder measure than axis predictivity.

Replacing the eigenvector matrix, \mathbf{E} , in (7.33) with another orthogonal projection matrix, for example \mathbf{B} , can lead to axis predictivities of greater than one when $r < p$. In this case, a quick empirical check will show that the *Type B orthogonality* requirement for the calculation of axis predictivities (Gower et al., 2011),

$$\mathbf{X}'\mathbf{X} = \hat{\mathbf{X}}'\hat{\mathbf{X}} + (\mathbf{X} - \hat{\mathbf{X}})'(\mathbf{X} - \hat{\mathbf{X}}), \quad (7.38)$$

is violated as the sum of the terms on the right-hand side of the equation is different from the quantity on the left-hand side. In these cases the calculation of axis predictivities does not make sense. Therefore it is usually not applicable to PC biplots other than the ordinary PCA biplot.

7.4.6 Mean standard predictive errors

To evaluate the quality of representation of the variables in a biplot display, Rui Alves (2012) proposed using the *mean standard predictive error (MSPE)*. MSPE is a measure to judge the accuracy of values inferred when making readings directly from a biplot by orthogonal projection of the observations onto a specific variable axis. Rui Alves showed that MSPE and axis predictivity are related, but serves different purposes and concludes that MSPE is also useful for biplots of orthogonal rotations other than the usual eigenvector rotation employed in a ordinary PCA biplot.

To calculate the MSPE of the j^{th} variable in a biplot using r of the orthonormal vectors in \mathbf{B} , let \mathbf{B}_r indicate the $p \times r$ projection matrix containing these orthonormal vectors. Let $\mathbf{b}_{(j)}$ indicate the row of \mathbf{B}_r corresponding to the j^{th} variable under consideration. The orthogonal projection of the m^{th} observation, $\mathbf{x}_{(m)}$, on the axis of the j^{th} variable as represented in the biplot is given by (Rui Alves, 2012)

$$\hat{x}_{mj} = \mathbf{x}_{(m)} \mathbf{B}_r \mathbf{b}'_{(j)}. \quad (7.39)$$

The difference between this direct reading of the value of observation m on variable j from the biplot and the actual value, x_{mj} , is used to calculate the *standard predictive error (SPE)* of variable j for observation m as (Rui Alves, 2012)

$$\epsilon_{mj} = \frac{|x_{mj} - \hat{x}_{mj}|}{s_j}, \quad (7.40)$$

where s_j is the sample standard deviation of variable j .

The MSPE of variable j is the mean of the SPEs over all n observations in \mathbf{X} . From (7.39), the SPE in (7.40) can be written as

$$\epsilon_{mj} = \frac{|x_{mj} - \mathbf{x}_{(m)} \mathbf{B}_r \mathbf{b}'_{(j)}|}{s_j}, \quad (7.41)$$

so that the MSPE of variable j is given by

$$\begin{aligned} \bar{\epsilon}_j &= \frac{1}{n} \sum_{m=1}^n \epsilon_{mj} \\ &= \frac{\mathbf{j}' |\mathbf{x}_j - \mathbf{X} \mathbf{B}_r \mathbf{b}'_{(j)}|}{n s_j} \end{aligned} \quad (7.42)$$

with \mathbf{x}_j indicating the j^{th} column of \mathbf{X} . A smaller value for (7.42) indicates higher predictive value for the j^{th} variable in the biplot. Rui Alves (2012) suggested plotting all variables for which the MSPE is below a pre-determined threshold.

Because the definition of MSPE does not depend on Type B orthogonality as axis predictivity does, it is also useful for biplots constructed from orthogonal projections other than that obtained from the eigenvectors of the data, such as CPC biplots.

7.4.7 Sample predictivities

Gower et al. (2011) also advocates the use of *sample predictivity* as a finer measure to assess how well individual observations in the original data set are represented in the biplot subspace. With \mathbf{X} and $\hat{\mathbf{X}}$ defined as in Section 7.4.5, the predictivities of the observations in an r -dimensional biplot are given by the diagonal elements of

$$\Psi = \text{diag}(\hat{\mathbf{X}}\hat{\mathbf{X}}')[\text{diag}(\mathbf{X}\mathbf{X}')]^{-1}. \quad (7.43)$$

The sample predictivities measure how well the actual Euclidean distances of the observations to the centroid are preserved in the biplot subspace.

The definition of sample predictivities requires *Type A orthogonality*, defined by Gower et al. (2011) as

$$\mathbf{X}\mathbf{X}' = \hat{\mathbf{X}}\hat{\mathbf{X}}' + (\mathbf{X} - \hat{\mathbf{X}})(\mathbf{X} - \hat{\mathbf{X}})'. \quad (7.44)$$

The Type A orthogonality requirement seems to hold also for biplots constructed from orthogonal projections other than that obtained through the use of the eigenvector matrix, \mathbf{E} . Consequently the calculation of sample predictivities should be a useful measure to assess the fit of the observations also in CPC biplots, and any other biplots using orthogonal projection matrices in Euclidean space.

Observations with small sample predictivities are poorly represented in the biplot subspace. By comparison to a predetermined sample predictivity threshold, these observations can be identified as outliers and their plotting in the biplot suppressed. This is similar to the idea suggested by Rui Alves (2012) where the SPEs (of the observations) in (7.40) are used for the same purpose.

7.5 Comparison of PC biplots

In order to obtain the highest quality biplot for any specific data set, it is of interest to compare the biplot quality measures for different PC biplot types. The importance given to the different measures can vary, depending on the purpose for which the biplot display is constructed. If the purpose is to distinguish between groups, biplots with the highest between-group quality will be favoured. If predictive ability is a high priority, the biplot with the smallest MSPE value can be chosen. The different biplots for data with distinct groups discussed in Section 7.3 each have different strengths and weaknesses, and it will be useful to have a data-based method to select the most appropriate type of biplot for any specific data set to be analysed.

CPC biplots should ideally be constructed using the common eigenvectors that account for most of the variation in each of the k groups simultaneously. The stepwise CPC algorithm (Trendafilov, 2010) finds the common eigenvectors sequentially in a way that the rank order of the eigenvectors are the same in all k of the covariance matrices. A consequence of this is that the first $r < p$ CPCs account for the largest proportion of variation in all of the populations simultaneously. In contrast to stepwise CPC, the FG (Flury and Gautschi, 1986) and JADE (Cardoso and Souloumiac, 1996) algorithms do not guarantee that the first r CPCs (associated with the first r columns of the \mathbf{B} matrix) is the subset of r CPCs that accounts for the maximum possible amount of variation in all of the populations simultaneously. With the latter two algorithms, the possibility thus remains that one or more of the groups are represented poorly (or not in the best possible way) in the subspace spanned by the r common eigenvectors used to construct the biplot display.

As it seems that a CPC biplot constructed with stepwise CPCs will generally account for a greater proportion of variation in the data (compared to CPC biplots using the FG and JADE algorithms), it is of interest to investigate whether this is indeed the case. To this end, a number of data sets were simulated and the quality measures were calculated for each type of PC biplot discussed in this chapter.

It is conjectured that CPC biplots will be most useful when the directions of maximum variation within the groups are the same as the directions of maximum variation between the groups. CPC analysis is concerned with the maximisation of within-group variation in all k populations simultaneously, without any regard for between-group differences. Only in the case where the directions of the r common eigenvectors (used to construct the biplot) coincide exactly with the directions of optimal separation between the groups will CPC biplots display the between-group variation optimally. However, the directions of the common eigenvectors will generally not be the same as the directions of the most effective discriminant functions (Rencher, 1998).

To compare the different types of PC biplots for any specific data set, the R function `biplot.choice()` was written. This function calculates the

- *Overall quality* in (7.18),
- *Within-group quality* in (7.23),
- *Between-group quality* in (7.28),
- the median *Adequacy* in (7.31),
- the mean of the j *MSPE* values in (7.42), and

- the mean of the *Sample predictivities* in (7.43)

for the following types of biplots:

- 1) **Pooled S**: Biplot constructed from the eigenvectors of the pooled covariance matrix, as discussed in Section 7.3.1,
- 2) **Pooled data**: Biplot constructed from the eigenvectors of the covariance matrix of the pooled data, as discussed in Section 7.3.2,
- 3) **Flury**: CPC biplot constructed from the common eigenvectors as estimated with the FG algorithm,
- 4) **Stepwise CPC**: CPC biplot constructed from the common eigenvectors as estimated with the stepwise CPC algorithm,
- 5) **JADE**: CPC biplot constructed from the common eigenvectors as estimated with the JADE algorithm.

Users of the `biplot.choice()` function can also specify additional projection matrices to compare with the five types of biplot mentioned above, if required. Example code to compare the different types of two-dimensional biplot for the iris data (Anderson, 1935) is shown below:

```
> data(iris)
> setosa <- iris[1:50, 1:4]
> versicolor <- iris[51:100, 1:4]
> virginica <- iris[101:150, 1:4]
> biplot.choice(datalist = list(setosa, versicolor, virginica),
               rdim = 2)
```

The following sections report the results of applying the `biplot.choice()` function to a number of simulated and well-known real data sets, respectively.

7.5.1 Simulated data

A small Monte Carlo simulation study was performed to study the properties of the five types of PC biplots. Samples of sizes $n_i = 50, i = 1, 2, 3$ were simulated from multivariate normally distributed populations with $p = 5$ variables. All of the eigenvectors of the population covariance matrices were common (i.e. the full CPC situation), and the eigenvalues (with the orders corresponding to the order of the common eigenvectors) were specified as follows:

$$\mathbf{l}'_1 = [5 \ 4 \ 3 \ 2 \ 1],$$

$$\mathbf{l}'_2 = [1 \ 2 \ 3 \ 4 \ 5],$$

and

$$\mathbf{l}'_3 = [2 \ 4 \ 1 \ 5 \ 3].$$

To test various covariance matrix configurations, a new population common eigenvector matrix was randomly selected from all $p \times p$ orthogonal matrices for each simulation run. A total of 1000 replications of the simulation experiment were performed for each of three different cases:

- 1) *Poor separation between the group centroids*, compared to the within-group variation. In this case the population distributions nearly coincided, and the between-group variation was almost negligible compared to the within-group variation. For each simulation run, population mean vectors were selected anew as follows:

- $\boldsymbol{\mu}_1$: $p = 5$ values selected randomly from a uniform distribution on the $[-2; 2]$ interval.
- $\boldsymbol{\mu}_2$: $p = 5$ values selected randomly from a uniform distribution on the $[-2; 2]$ interval.
- $\boldsymbol{\mu}_3$: $p = 5$ values selected randomly from a uniform distribution on the $[-2; 2]$ interval.

- 2) *Medium separation between the group centroids*. The group centroids were allowed to vary more than in the first case, so that the between-group variation formed a non-negligible part of the overall variation in the data. For each simulation run, population mean vectors were selected anew as follows:

- $\boldsymbol{\mu}_1$: $p = 5$ values selected randomly from a uniform distribution on the $[-5; 0]$ interval.
- $\boldsymbol{\mu}_2$: $p = 5$ values selected randomly from a uniform distribution on the $[-2; 3]$ interval.
- $\boldsymbol{\mu}_3$: $p = 5$ values selected randomly from a uniform distribution on the $[0; 5]$ interval.

3) *Good separation between the group centroids.* In this case, the between-group variation was large compared to the within-group variation, dominating the overall variation observed in the data. For each simulation run, population mean vectors were selected anew as follows:

- μ_1 : $p = 5$ values selected randomly from a uniform distribution on the $[-2; 3]$ interval.
- μ_2 : $p = 5$ values selected randomly from a uniform distribution on the $[4; 9]$ interval.
- μ_3 : $p = 5$ values selected randomly from a uniform distribution on the $[10; 15]$ interval.

Poor separation between group centroids

Biplots constructed from the eigenvectors of the covariance matrix of the pooled data (*Pooled data*) for nine of the simulation runs (randomly selected from the total of $r = 1000$) are shown in Figure 7.1. The three groups can be distinguished by the colours of the points (black, gray, or brown) representing the observations. Plotting of the variables was suppressed, as the purpose of these biplots is merely to show examples of the configuration of points for three groups.

The quality measures for each of the five PC biplot types are reported in Table 7.1. The *Pooled data* biplot has the best values for all of the quality measures, with the exception of within-group quality where it is outperformed by the other PC biplot types. If the purpose of the biplot is to compare the variation within the groups, the *Pooled S* biplot should be used. For any other purpose, the *Pooled data* biplot should provide the highest quality display.

Of the CPC biplots, *Stepwise CPC* generally provides the best quality display, as it has the best overall and within-group quality, as well as the largest mean sample predictivity. The *Flury* and *JADE* biplots have performed about equally well on all of the quality measures.

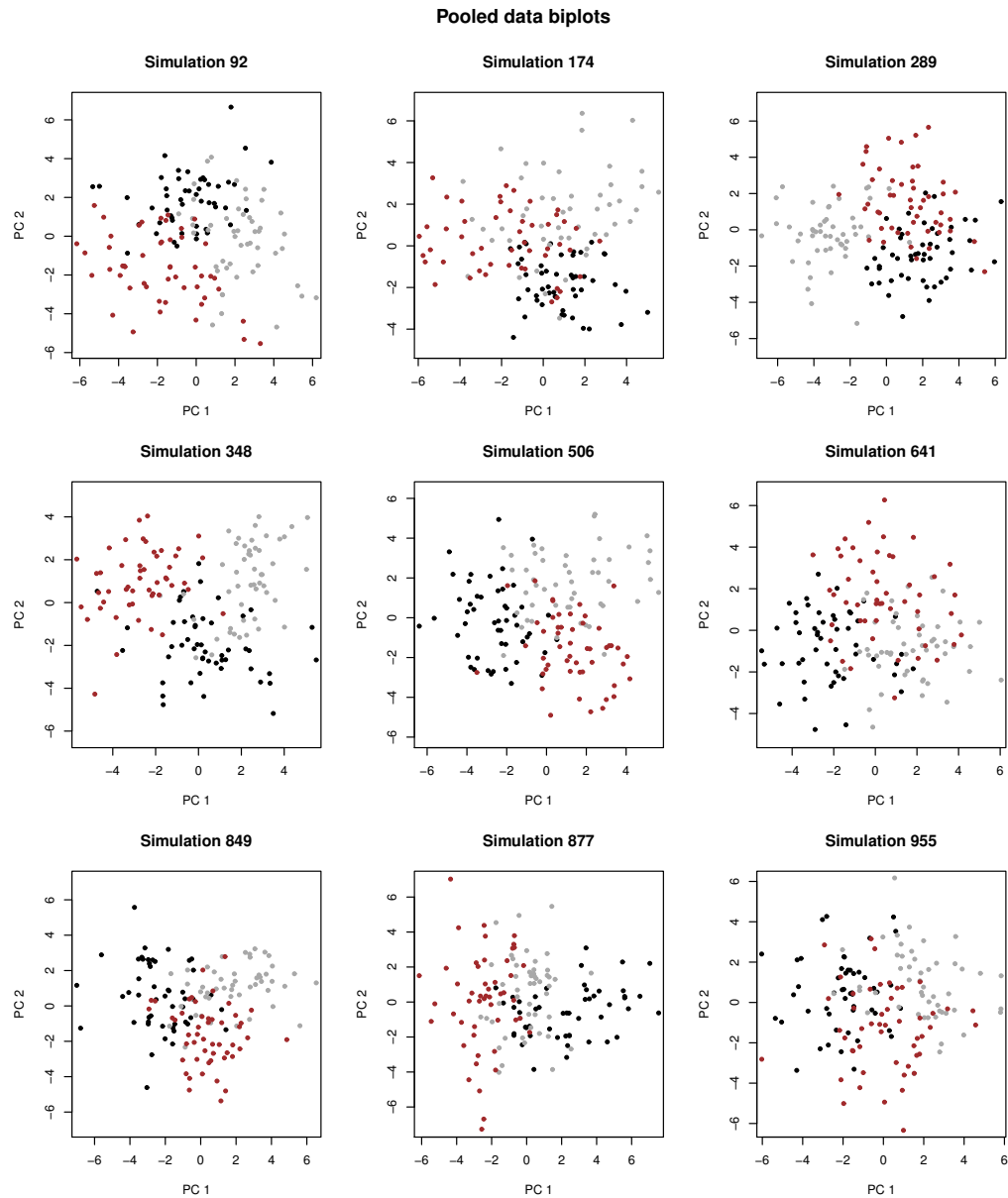


Figure 7.1: A random sample of biplots constructed from the eigenvectors of the covariance matrix of the pooled data in the case when the group centroids are poorly separated. The different coloured points (black, gray, and brown) indicate the three groups. Plotting of the variables was suppressed.

Table 7.1: Quality measures for the five PC biplot types in the case when the group centroids are poorly separated, calculated from a total of $r = 1000$ simulation runs. The best values for each quality measure are indicated in **bold** type.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.48	0.51	0.39	0.36	0.57	0.45
Pooled data	0.57	0.45	0.91	0.36	0.52	0.53
Flury	0.46	0.48	0.40	0.36	0.57	0.44
Stepwise CPC	0.48	0.50	0.39	0.36	0.57	0.45
JADE	0.46	0.48	0.40	0.37	0.57	0.44

Medium separation between group centroids

Pooled data biplots for nine randomly selected simulation runs (from a total of $r = 1000$) in the case when there was increased separation between the group centroids are shown in Figure 7.2. From the quality measures reported in Table 7.2, it can be seen that the *Pooled data* biplot is clearly the best option in this case too. The only weakness of the *Pooled data* biplot is in the quality of representation of the within-group variation, in which the *Pooled S* and *Stepwise CPC* biplots perform the best.

Regarding only the three CPC biplot types, the *Stepwise CPC* biplot again seems to be the best option, as it outperforms the other two CPC biplot types on nearly all of the quality measures.

Table 7.2: Quality measures for the five PC biplot types in the case when the group centroids are better separated (*Medium*), calculated from a total of $r = 1000$ simulation runs. The best values for each quality measure are indicated in **bold** type.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.36	0.50	0.28	0.35	0.65	0.37
Pooled data	0.79	0.41	0.99	0.37	0.38	0.72
Flury	0.33	0.48	0.25	0.36	0.67	0.35
Stepwise CPC	0.37	0.50	0.30	0.35	0.64	0.38
JADE	0.35	0.48	0.28	0.36	0.66	0.36

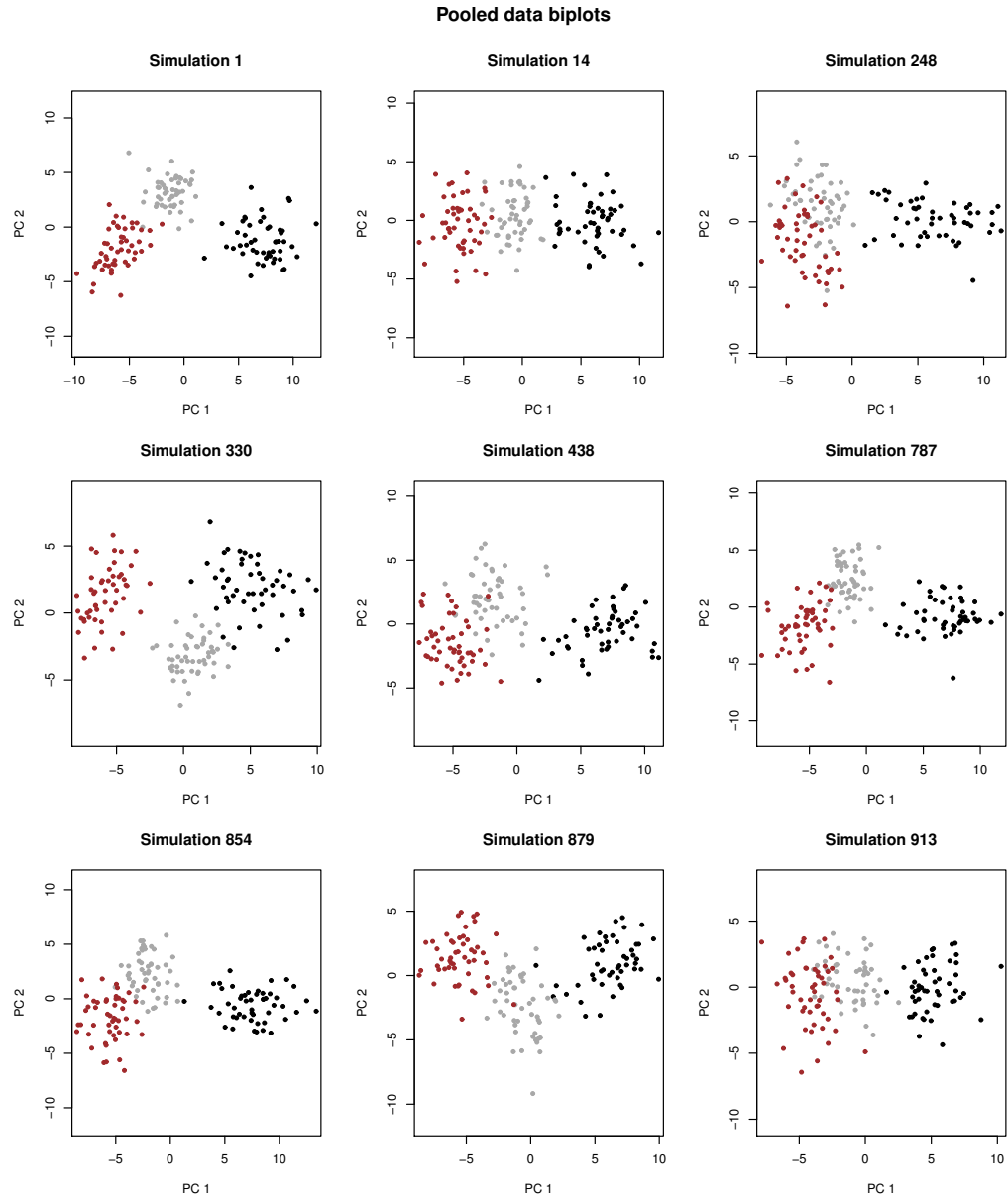


Figure 7.2: A random sample of biplots constructed from the eigenvectors of the covariance matrix of the pooled data in the case when there are better (*Medium*) separation between the group centroids. The different coloured points (black, gray, and brown) indicate the three groups. Plotting of the variables was suppressed.

Good separation between group centroids

Pooled data biplots constructed for the data from a randomly selected sample of nine of the $r = 1000$ simulation runs, in the case when the group centroids were well separated (compared to the within-group variation), are shown in Figure 7.3. Because the between-group variation is the dominant source in the overall observed variation, the first principal component of the pooled data usually accounts mainly for the between-group variation, and provides a good visual separation between the groups. The second principal component accounts for the direction of maximum variation within the groups simultaneously, subject to the usual eigenvector orthogonality constraint.

The biplot quality measures reported in Table 7.3 show the same trends as was seen in the previous two cases. Clearly, even if there are common eigenvectors in the population covariance matrices, the *Pooled data* biplot outperforms the CPC biplots, except in the display quality of the within-group variation. If quality of the representation of the within-group variation takes precedence in the selection of a PC biplot type, the *Stepwise CPC* biplot will be the best choice.

The *Stepwise CPC* biplot again performed the best among the three CPC biplot types.

Table 7.3: Quality measures for the five PC biplot types in the case when the group centroids are well separated, calculated over a total of $r = 1000$ simulation runs. The best values for each quality measure are indicated in **bold** type.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.28	0.50	0.26	0.36	0.68	0.33
Pooled data	0.94	0.41	1.00	0.34	0.20	0.81
Flury	0.25	0.48	0.22	0.36	0.70	0.30
Stepwise CPC	0.30	0.50	0.28	0.36	0.66	0.34
JADE	0.28	0.48	0.26	0.36	0.68	0.32

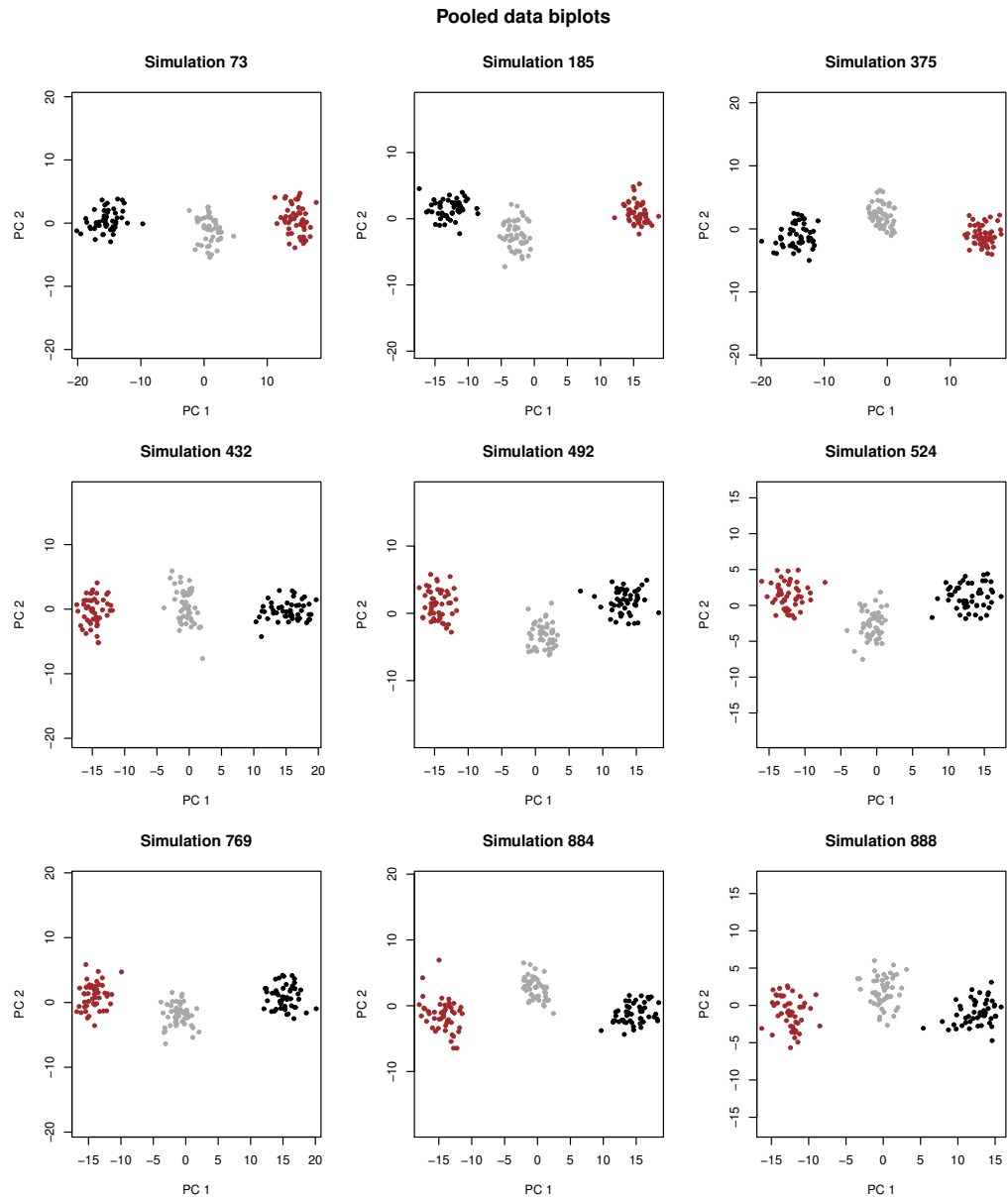


Figure 7.3: A random sample of biplots constructed from the eigenvectors of the covariance matrix of the pooled data in the case when the group centroids are well separated. The different coloured points (black, gray, and brown) indicate the three groups. Plotting of the variables was suppressed.

7.5.2 Iris data

The quality measures as discussed in this chapter were applied to the Iris data (Anderson, 1935) to determine which type of PC biplot will be most suitable to visually represent this data set. For the different types of two-dimensional biplot, the quality measures are reported in Table 7.4.

The biplot constructed from the eigenvectors of the covariance matrix of the pooled data (*Pooled data*) will clearly provide the highest overall quality, best distinction between the three groups, best representation of the variables (due to the lowest mean MSPE value) and best sample predictivities. Representation of the within-group variation is not much poorer than for the other biplot types.

Table 7.4: Quality measures for a two-dimensional biplots of the Iris data (*Setosa*, *Versicolor* and *Virginica* species).

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.7944	0.8721	0.7827	0.5431	0.4333	0.7693
Pooled data	0.9777	0.8445	0.9978	0.5510	0.1795	0.9412
Flury	0.8982	0.8665	0.9030	0.6039	0.3787	0.8707
Stepwise CPC	0.8366	0.8704	0.8315	0.5766	0.4366	0.8135
JADE	0.9006	0.8666	0.9057	0.6224	0.3695	0.8741

The *JADE* biplot seems to strike a good balance across the quality measures, and will be shown here for illustrative purposes. The common eigenvector matrix estimated with the JADE algorithm is:

$$\mathbf{B}_{\text{JADE}} = \begin{bmatrix} 0.73 & 0.20 & -0.61 & 0.23 \\ 0.24 & 0.82 & 0.45 & -0.26 \\ 0.62 & -0.53 & 0.42 & -0.38 \\ 0.15 & -0.05 & 0.49 & 0.86 \end{bmatrix}.$$

The two-dimensional biplot of the Iris data, constructed with the first two common eigenvectors in \mathbf{B}_{JADE} is shown in Figure 7.4. The largest difference between the three species seems to be with regards to petal length, as the axis for this variable lies in the direction of the greatest separation between the groups. *Versicolor* and *Virginica* are similar with regards to sepal width, and the largest variation within each of these groups can be ascribed to differences in sepal length and petal length of the individual flowers. The within-group variation in *Setosa* can mainly be accounted for by differences in sepal width and sepal length of the individual flowers. These conclusions

are supported by an inspection of the means and standard deviations of the four variables for each of the groups (see Table 7.5).

Table 7.5: Means and standard deviations for the four variables in the Iris data set.

	Sepal length	Sepal width	Petal length	Petal width
Means				
<i>Setosa</i>	5.006	3.428	1.462	0.246
<i>Versicolor</i>	5.936	2.770	4.260	1.326
<i>Virginica</i>	6.588	2.974	5.552	2.026
Standard deviations				
<i>Setosa</i>	0.352	0.379	0.174	0.105
<i>Versicolor</i>	0.516	0.314	0.470	0.198
<i>Virginica</i>	0.636	0.322	0.552	0.275

The quality measures for a three-dimensional biplot of the Iris data are reported in Table 7.6. Although the overall, within-group, between-group and sample predictivity measures are very similar for the different biplots, it seems that the *Pooled data* biplot will represent the variables best, as it has the smallest mean MSPE value.

Table 7.6: Quality measures for a three-dimensional biplots of the Iris data (*Setosa*, *Versicolor* and *Virginica* species).

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.9823	0.9632	0.9852	0.8876	0.1351	0.9780
Pooled data	0.9948	0.9609	0.9999	0.8338	0.0688	0.9860
Flury	0.9847	0.9594	0.9885	0.9314	0.0996	0.9798
Stepwise CPC	0.9818	0.9627	0.9847	0.9076	0.1276	0.9780
JADE	0.9879	0.9627	0.9918	0.8934	0.1036	0.9827

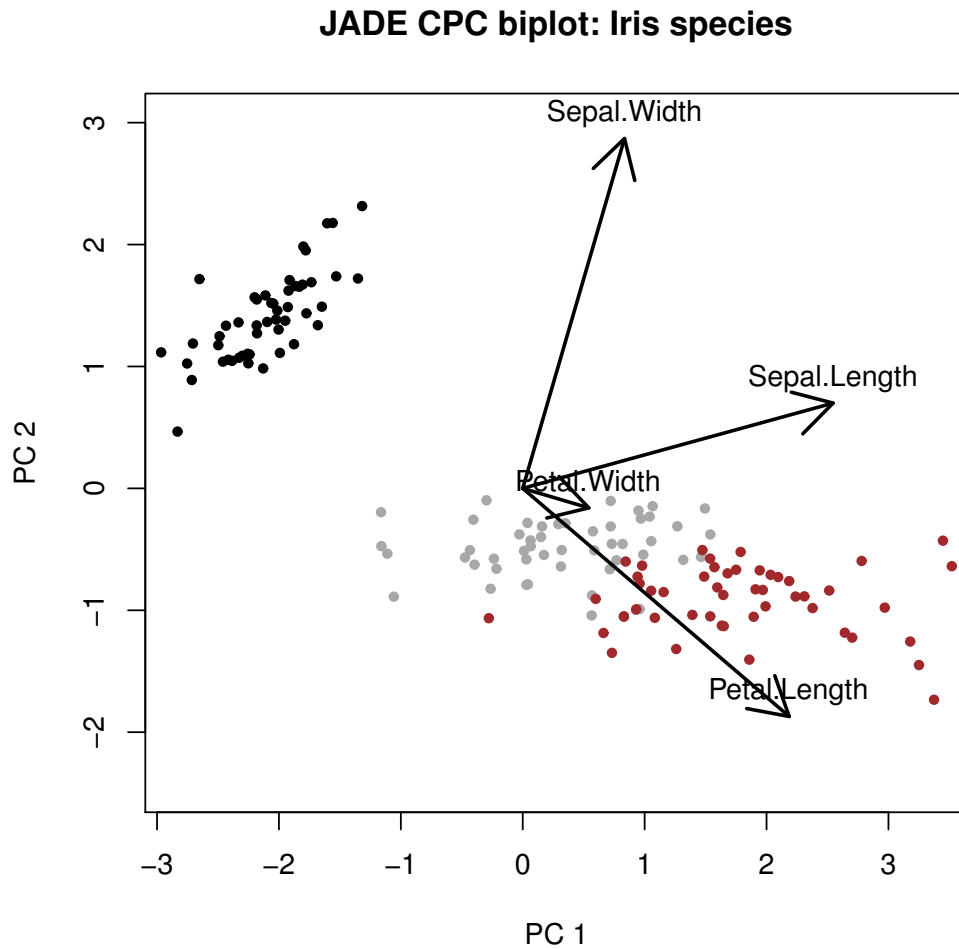


Figure 7.4: Two-dimensional biplot of the Iris data, using the first two common eigenvectors as estimated with the JADE algorithm. The different coloured points indicate the three iris species: *Setosa* = black, *Versicolor* = gray, *Virginica* = brown.

7.5.3 Bank notes data

To select the most appropriate type of two-dimensional PC biplot for the Swiss bank notes data Flury (1988), the biplot quality measures were calculated and are reported in Table 7.7. The *Pooled data* biplot clearly seems the best option in terms of the overall and between-group quality of the display, representation of the variables, as well as the sample predictivities. The within-group quality and adequacy of the *Pooled data* biplot are very similar to those of the other biplot types. The biplot constructed from the common eigenvectors estimated with the FG algorithm (*Flury*) also seems to be a good option.

Table 7.7: Quality measures for a two-dimensional biplots of the Swiss bank notes (*Genuine* and *Forged*).

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.4159	0.7164	0.2058	0.2442	0.8027	0.3548
Pooled data	0.8757	0.6982	0.9998	0.2496	0.4355	0.8470
Flury	0.6491	0.7006	0.6130	0.2115	0.7492	0.6159
Stepwise CPC	0.3515	0.7126	0.0989	0.2522	0.7529	0.3065
JADE	0.4439	0.7137	0.2552	0.2163	0.7881	0.3765

The common eigenvector matrix estimated with the FG algorithm is:

$$\mathbf{B}_{\text{FG}} = \begin{bmatrix} 0.02 & 0.33 & -0.45 & -0.34 & -0.74 & 0.12 \\ 0.04 & 0.48 & -0.19 & -0.21 & 0.30 & -0.77 \\ 0.04 & 0.44 & -0.31 & -0.16 & 0.56 & 0.61 \\ 0.81 & 0.32 & 0.15 & 0.46 & -0.12 & 0.03 \\ -0.58 & 0.46 & 0.03 & 0.66 & -0.13 & 0.01 \\ 0.09 & -0.40 & -0.80 & 0.41 & 0.11 & -0.12 \end{bmatrix}.$$

The two-dimensional biplot of the bank notes data, using the first two common eigenvectors in \mathbf{B}_{FG} , is shown in Figure 7.5. This configuration provides a good separation between the two groups. The greatest separation between the groups are in the directions of *LEFT*, *RIGHT*, *LENGTH* and *DIAG*, which are all measurements of the size and shape of the notes. Within each group, the largest variation is seen along the *TOP* and *BOTTOM* variables, which pertain to the placement of the printed image on the currency paper. The *Genuine* notes display less variation in this aspect (image placement) than the *Forged* notes.

For a three-dimensional biplot of the bank notes data, the biplot quality measures are reported in Table 7.8. With the exception of within-group

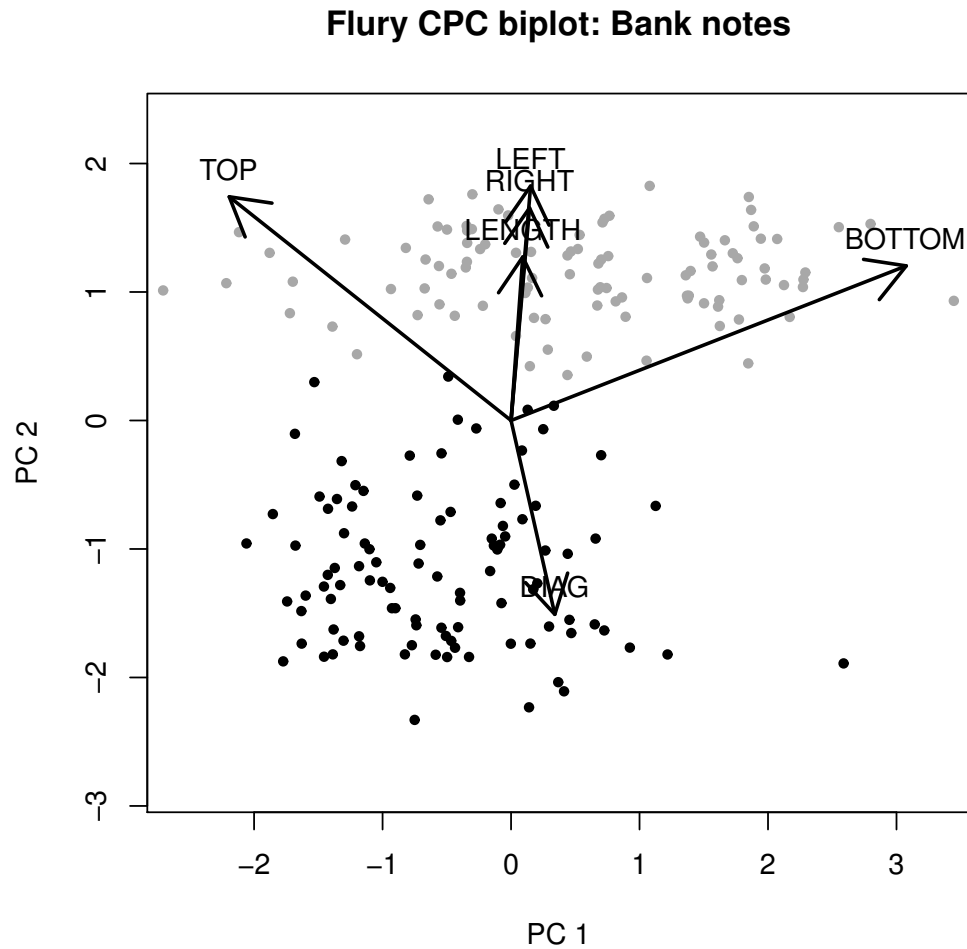


Figure 7.5: Two-dimensional biplot of the Swiss bank notes data, using the first two common eigenvectors as estimated with the FG algorithm. The different coloured points indicate the two groups: *Genuine* = black, *Forged* = gray.

quality, the *Pooled data* outperforms all of the other PC biplot types and also provides the best representation of the variables as it has the smallest mean MSPE value. The between-group quality of display for the *Stepwise CPC* biplot is notably poor, and it also fares worst in terms of representation of the variables as measured by mean MSPE.

Table 7.8: Quality measures for a three-dimensional biplots of the Swiss bank notes data (*Genuine* and *Forged*).

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.7462	0.8421	0.6792	0.4008	0.6403	0.6978
Pooled data	0.9298	0.8298	0.9998	0.4639	0.3416	0.9106
Flury	0.8916	0.8206	0.9413	0.4307	0.4008	0.8689
Stepwise CPC	0.6103	0.8380	0.4510	0.4267	0.7078	0.5588
JADE	0.7822	0.8392	0.7423	0.4374	0.6336	0.7366

To investigate why the *Pooled data* biplot fares better than the other biplot types with regards to between-group quality (i.e. good separation between the groups in the biplot display), the directions of the vectors representing the differences between the group centroids were compared to the directions of the eigenvectors used in the construction of the biplot. The normalised vector of mean differences (between *Genuine* and *Forged*) is

$$\bar{\mathbf{d}}' = [0.04 \quad -0.11 \quad -0.15 \quad -0.69 \quad -0.30 \quad 0.64].$$

Multiplying the transpose of the eigenvector matrix of the covariance matrix of the pooled data, \mathbf{B}_{pool} , with $\bar{\mathbf{d}}$ gives

$$\mathbf{B}'_{\text{pool}}\bar{\mathbf{d}} = \begin{bmatrix} -0.04 & 0.11 & 0.14 & 0.77 & 0.20 & -0.58 \\ 0.01 & 0.07 & 0.07 & -0.56 & 0.66 & -0.49 \\ -0.33 & -0.26 & -0.34 & -0.22 & -0.56 & -0.59 \\ 0.56 & 0.46 & 0.42 & -0.19 & -0.45 & -0.26 \\ 0.75 & -0.35 & -0.53 & 0.10 & 0.10 & -0.08 \\ 0.10 & -0.77 & 0.63 & -0.02 & -0.03 & -0.05 \end{bmatrix} \begin{bmatrix} 0.04 \\ -0.11 \\ -0.15 \\ -0.69 \\ -0.30 \\ 0.64 \end{bmatrix} = \begin{bmatrix} -0.99 \\ -0.14 \\ 0.00 \\ 0.01 \\ -0.00 \\ -0.01 \end{bmatrix},$$

which shows that the first eigenvector and $\bar{\mathbf{d}}$ are nearly collinear as the inner product of these two vectors is -0.99 . The direction of the first eigenvector, used as one of the biplot axes, is also the direction of optimal separation between the two groups. Similar inspections of the eigenvectors of the pooled covariance matrix,

$$\mathbf{B}'_p \bar{\mathbf{d}} = \begin{bmatrix} 0.03 & -0.00 & -0.01 & -0.86 & 0.49 & -0.15 \\ 0.23 & 0.09 & 0.21 & 0.11 & 0.44 & 0.83 \\ -0.38 & -0.51 & -0.45 & -0.27 & -0.32 & 0.48 \\ 0.50 & 0.21 & 0.25 & -0.40 & -0.67 & 0.18 \\ 0.74 & -0.33 & -0.54 & 0.13 & 0.14 & -0.12 \\ -0.09 & 0.76 & -0.63 & -0.03 & -0.02 & 0.11 \end{bmatrix} \begin{bmatrix} 0.04 \\ -0.11 \\ -0.15 \\ -0.69 \\ -0.30 \\ 0.64 \end{bmatrix} = \begin{bmatrix} \mathbf{0.35} \\ \mathbf{0.29} \\ 0.69 \\ 0.55 \\ -0.05 \\ 0.10 \end{bmatrix},$$

and common eigenvectors estimated with the FG algorithm,

$$\mathbf{B}'_{\text{FG}} \bar{\mathbf{d}} = \begin{bmatrix} 0.02 & 0.04 & 0.04 & 0.81 & -0.58 & 0.09 \\ 0.33 & 0.48 & 0.44 & 0.32 & 0.46 & -0.40 \\ -0.45 & -0.19 & -0.31 & 0.15 & 0.03 & -0.80 \\ -0.34 & -0.21 & -0.16 & 0.46 & 0.66 & 0.41 \\ -0.74 & 0.30 & 0.56 & -0.12 & -0.13 & 0.11 \\ 0.12 & -0.77 & 0.61 & 0.03 & 0.01 & -0.12 \end{bmatrix} \begin{bmatrix} 0.04 \\ -0.11 \\ -0.15 \\ -0.69 \\ -0.30 \\ 0.64 \end{bmatrix} = \begin{bmatrix} -\mathbf{0.34} \\ -\mathbf{0.71} \\ -0.57 \\ -0.22 \\ 0.04 \\ -0.10 \end{bmatrix},$$

stepwise CPC algorithm,

$$\mathbf{B}'_{\text{stepwise}} \bar{\mathbf{d}} = \begin{bmatrix} -0.00 & -0.01 & -0.02 & -0.82 & 0.56 & -0.13 \\ 0.37 & 0.29 & 0.37 & 0.23 & 0.49 & 0.59 \\ -0.37 & -0.46 & -0.38 & -0.09 & 0.02 & 0.71 \\ 0.43 & 0.09 & 0.16 & -0.50 & -0.65 & 0.32 \\ 0.73 & -0.34 & -0.55 & 0.13 & 0.14 & -0.13 \\ -0.09 & 0.76 & -0.63 & -0.03 & -0.03 & 0.11 \end{bmatrix} \begin{bmatrix} 0.04 \\ -0.11 \\ -0.15 \\ -0.69 \\ -0.30 \\ 0.64 \end{bmatrix} = \begin{bmatrix} \mathbf{0.31} \\ \mathbf{0.00} \\ 0.59 \\ 0.73 \\ -0.06 \\ 0.10 \end{bmatrix},$$

and JADE algorithm,

$$\mathbf{B}'_{\text{JADE}} \bar{\mathbf{d}} = \begin{bmatrix} 0.34 & 0.49 & 0.43 & 0.32 & 0.47 & -0.38 \\ -0.75 & 0.32 & 0.55 & -0.12 & -0.13 & 0.10 \\ 0.11 & -0.76 & 0.63 & 0.03 & 0.02 & -0.12 \\ -0.05 & -0.01 & 0.00 & 0.89 & -0.43 & 0.16 \\ -0.55 & -0.28 & -0.33 & 0.30 & 0.59 & -0.26 \\ 0.09 & -0.01 & 0.11 & 0.08 & 0.48 & 0.86 \end{bmatrix} \begin{bmatrix} 0.04 \\ -0.11 \\ -0.15 \\ -0.69 \\ -0.30 \\ 0.64 \end{bmatrix} = \begin{bmatrix} -\mathbf{0.70} \\ \mathbf{0.04} \\ -0.11 \\ -0.38 \\ -0.49 \\ 0.34 \end{bmatrix},$$

show that none of the first two (common) eigenvectors used for construction of the biplot in each case correspond closely to the direction of optimal separation between the groups.

7.6 Application to the VON data

The quality measures discussed in this chapter were applied to the groupings in the VON 2009 cohort, to investigate which type of PC biplot is most appropriate for the delivery mode, regional and mortality groupings, respectively. Results are reported in the following three sections.

For the delivery mode and regional groupings, infants transferred to alternative NICU facilities and those who died before final hospital discharge were included. For the mortality groupings (*Survived* and *Died*), transferred infants were excluded as transferal status is expected to be correlated with variables influencing mortality status.

7.6.1 Delivery mode

Quality measures for a two-dimensional biplot of the *Caesarean* and *Vaginal* groups are reported in Table 7.9. As the different biplot types perform almost equally well on each of the quality measures, a decision was made to use the biplot involving the least amount of computation, which is the *Pooled data* biplot (shown in Figure 7.6).

The variation in the data is dominated by the size (*GESTAGE* and *BHEADCIR*) and the feasibility of life (*AP1* and *AP5*) components. There seem to be very little difference between the centroids of the delivery mode groups, compared to the within-group variation.

Table 7.9: Quality measures for two-dimensional biplots of the delivery mode groups (*Caesarean* and *Vaginal*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.8883	0.8898	0.5409	0.4082	0.3214	0.7879
Pooled data	0.8884	0.8898	0.5542	0.4062	0.3213	0.7880
Flury	0.8883	0.8898	0.5405	0.4085	0.3214	0.7879
Stepwise CPC	0.8883	0.8898	0.5395	0.4095	0.3215	0.7878
JADE	0.8874	0.8888	0.5525	0.3968	0.3238	0.7874

Table 7.10 reports the quality measures for different three-dimensional biplot types for the delivery mode groups. The situation is the same as in the two-dimensional case, with the different biplots being practically equal on each of the quality measures.

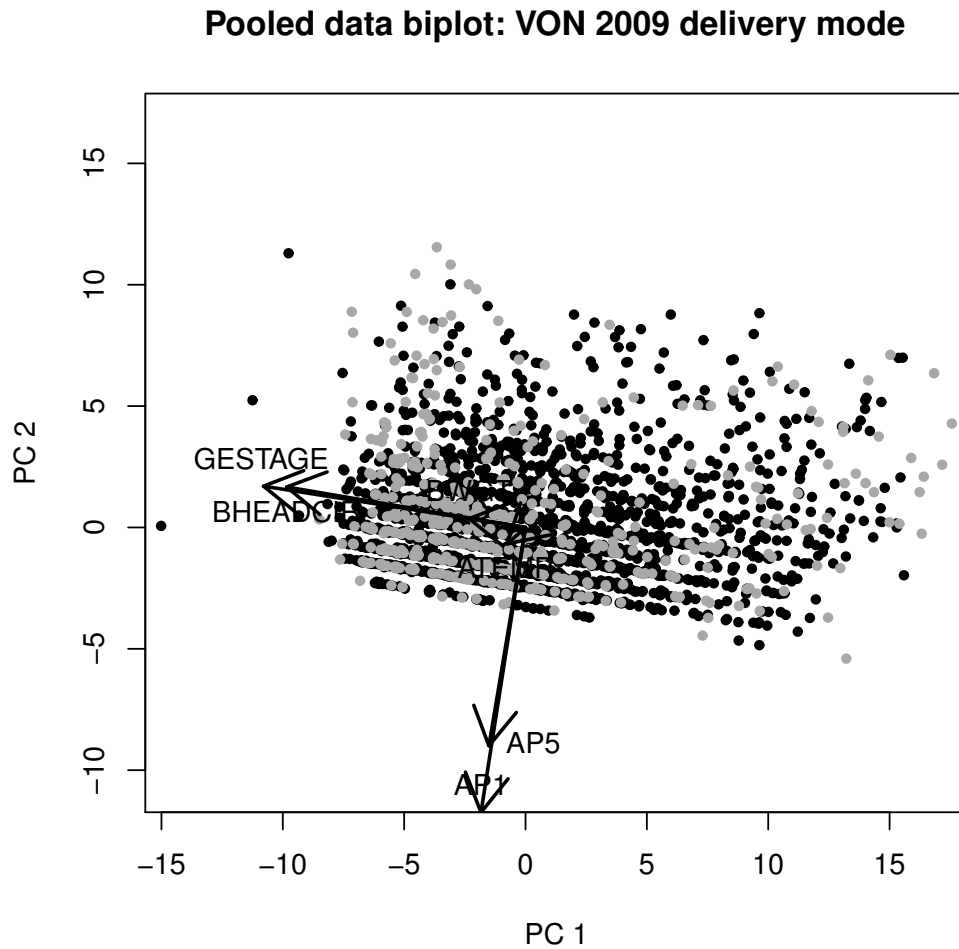


Figure 7.6: Two-dimensional biplot of the delivery mode groups in the VON 2009 cohort, using the first two eigenvectors of the covariance matrix of the pooled data. The different coloured points indicate the two groups: *Cesarean* = black, *Vaginal* = gray.

Table 7.10: Quality measures for three-dimensional biplots of the delivery mode groups (*Caesarean* and *Vaginal*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.9588	0.9587	0.9792	0.4998	0.2487	0.9175
Pooled data	0.9588	0.9587	0.9809	0.4995	0.2485	0.9175
Flury	0.9588	0.9587	0.9793	0.5002	0.2487	0.9175
Stepwise CPC	0.9588	0.9587	0.9793	0.5001	0.2487	0.9175
JADE	0.9584	0.9583	0.9786	0.4969	0.2508	0.9167

7.6.2 Regions

The quality measures calculated for two-dimensional biplots of the regional groups (*South Africa* and *Namibia*) are reported in Table 7.11. The situation is the same as for the delivery mode groups, with all of the biplot types faring nearly equally well on each of the quality measures. The *Pooled data* biplot was again selected because it requires the least amount of computation, and is shown in Figure 7.7.

Table 7.11: Quality measures for two-dimensional biplots of the regional groups (*South Africa* and *Namibia*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.8884	0.8886	0.7565	0.4058	0.3213	0.7881
Pooled data	0.8884	0.8886	0.7576	0.4062	0.3213	0.7880
Flury	0.8884	0.8886	0.7569	0.4064	0.3212	0.7880
Stepwise CPC	0.8884	0.8886	0.7566	0.4052	0.3213	0.7881
JADE	0.8854	0.8857	0.7548	0.4032	0.3291	0.7849

For a three-dimensional biplot of the regional groups, quality measures are reported in Table 7.12. The *Pooled data* biplot can be used as it there is almost no practically significant quality difference between it and the other four types of PC biplots.

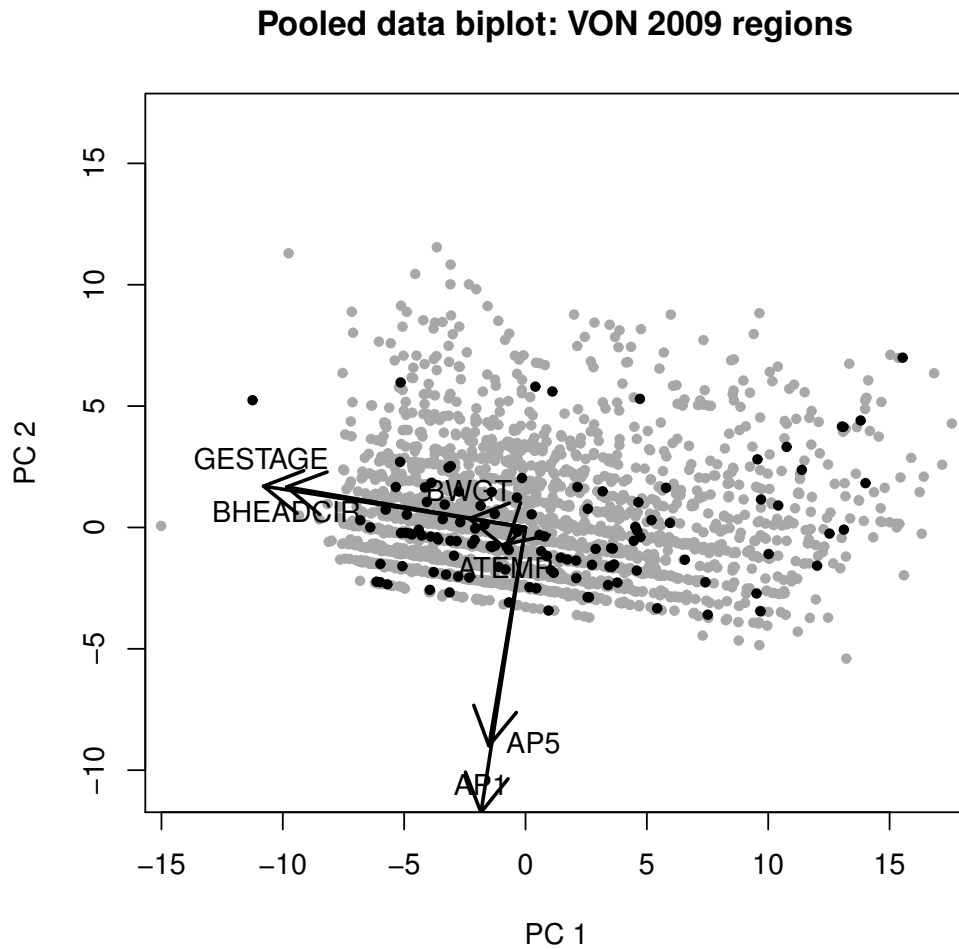


Figure 7.7: Two-dimensional biplot of the regional groups in the VON 2009 cohort, using the first two eigenvectors of the covariance matrix of the pooled data. The different coloured points indicate the two groups: *Namibia* = black, *South Africa* = gray.

Table 7.12: Quality measures for three-dimensional biplots of the regional groups (*South Africa* and *Namibia*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.9588	0.9590	0.8315	0.4994	0.2484	0.9175
Pooled data	0.9588	0.9590	0.8331	0.4995	0.2485	0.9175
Flury	0.9588	0.9590	0.8315	0.4994	0.2483	0.9175
Stepwise CPC	0.9588	0.9590	0.8319	0.4994	0.2484	0.9175
JADE	0.9567	0.9569	0.8228	0.5021	0.2577	0.9159

7.6.3 Mortality

Quality measures for two-dimensional biplot displays of the mortality groups are reported in Table 7.13. The quality of all of the biplots are almost the same on all of the measures. The *Pooled S* biplot of the mortality groups are shown in Figure 7.8.

There seems to be a cluster of observations from the *Died* group towards the right-hand side of the biplot, which indicates that the mortality groups differed most on the *GESTAGE* and *BHEADCIR* variables. This means that infants with low gestational age and/or low birth head circumference were more likely to die, compared to their peers with greater scores on these variables.

A small difference can also be seen between the *Survived* and *Died* groups with regards to the Apgar score variables (*AP1* and *AP5*). There is a greater clustering of observations from the *Survived* group towards the bottom of the graph (i.e. greater Apgar scores), compared to the *Died* group. The infants with greater Apgar scores thus had better chances of survival.

Table 7.13: Quality measures for two-dimensional biplots of the mortality groups (*Survived* and *Died*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.8873	0.8799	0.9986	0.4045	0.3240	0.7858
Pooled data	0.8873	0.8799	0.9989	0.4065	0.3238	0.7857
Flury	0.8873	0.8799	0.9984	0.4022	0.3241	0.7859
Stepwise CPC	0.8872	0.8799	0.9984	0.4002	0.3242	0.7860
JADE	0.8802	0.8724	0.9985	0.4514	0.3447	0.7762

Finally, for selecting a PC biplot type to construct a three-dimensional biplot, quality measures are reported in Table 7.14. As in the two-dimensional case, no clear differences can be seen between the different biplot types.

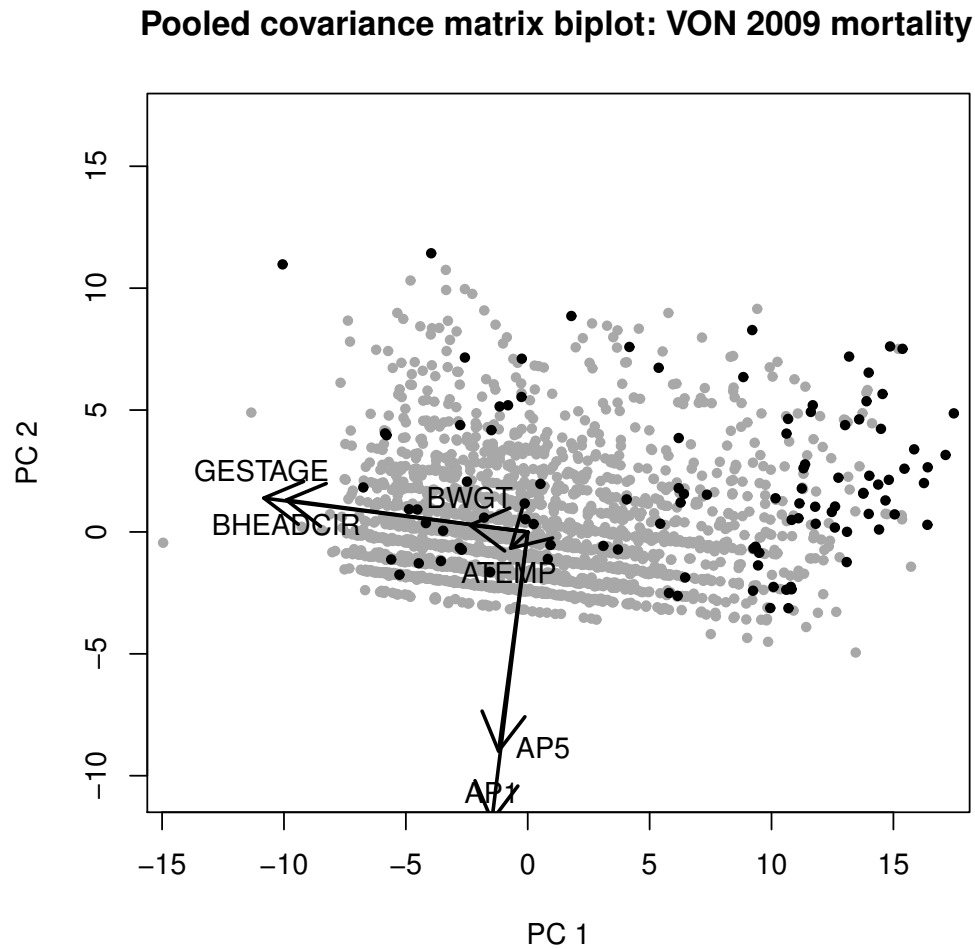


Figure 7.8: Two-dimensional biplot of the mortality groups in the VON 2009 cohort, using the first two eigenvectors of the pooled covariance matrix. The different coloured points indicate the two groups: *Died* = black, *Survived* = gray.

Table 7.14: Quality measures for three-dimensional biplots of the mortality groups (*Survived* and *Died*) in the VON 2009 cohort.

	Overall	Within	Between	Adequacy	MSPE	Sample predictivities
Pooled S	0.9584	0.9557	0.9987	0.4998	0.2506	0.9167
Pooled data	0.9584	0.9557	0.9989	0.4997	0.2504	0.9167
Flury	0.9583	0.9557	0.9985	0.4997	0.2507	0.9168
Stepwise CPC	0.9583	0.9557	0.9984	0.4997	0.2508	0.9168
JADE	0.9559	0.9531	0.9990	0.5001	0.2620	0.9131

Chapter 8

Regression modelling of the VON data

8.1 Introduction

The primary focus in this chapter is on regression modelling of the VON 2009 data set in order to understand and predict neonatal mortality and length of hospital stay for NICU admissions at Southern African private hospitals. It is therefore of a more applied nature than the previous chapters, with the role of the CPC model rendered somewhat less conspicuous. An earlier version of this work was published in Pepler et al. (2012).

The idea of CPC regression, suggested by Flury (1988), is investigated, and comparisons are made to ordinary least squares (OLS) regression, principal component (PC) regression and partial least squares (PLS) regression. These four approaches to regression modelling of grouped multivariate data are briefly discussed in Section 8.2.

Three measures to compare different regression models are mentioned in Section 8.3. These measures are used to compare the different models to predict neonatal mortality and length of stay for admissions to NICUs in the VON 2009 data set in Section 8.4.

The chapter is concluded with some remarks about the practicality of PC and CPC regression in Section 8.5.

8.2 Regression models for data with distinct groups

Although the regression models discussed in the following sections (with the exception of CPC regression) can also be used to analyse data from a single group, the focus here is on illuminating how these different model types can be used to analyse data from distinct groups. For a good general introduction to regression modelling, see Draper and Smith (1998).

8.2.1 Multiple linear regression

With an ordinary *multiple linear regression* model, a hyperplane is fitted to p covariates with a model of the form

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \epsilon, \quad (8.1)$$

where Y is the response variable, β_j is the coefficient for the j^{th} regressor, X_j , β_0 is the intercept, and $\epsilon \sim N(0, \sigma^2)$ is an error term (Johnson and Wichern, 2002).

The *ordinary least squares (OLS)* estimators of the β_j coefficients in (8.1) are

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}, \quad (8.2)$$

where \mathbf{y} is the vector of observed responses on Y and

$$\mathbf{X} = [\mathbf{j} \quad \mathbf{x}_1 \quad \dots \quad \mathbf{x}_p] \quad (8.3)$$

is the design matrix, with \mathbf{j} indicating a column vector of ones and where $\mathbf{x}_i, i = 1, \dots, p$ are the vectors of observations on the covariates, X_1, \dots, X_p .

The fitted values from the multiple linear regression model are given by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}. \quad (8.4)$$

Under the assumption of normality of ϵ , $\hat{\beta}$ as in (8.2) is distributed $N_{p+1}(\beta, \sigma^2(\mathbf{X}'\mathbf{X})^{-1})$, where σ^2 is the error variance of the model (Johnson and Wichern, 2002). If the regression errors are normally and independently distributed with zero mean and equal variance, i.e. $\epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$, the sum of squares for error in the linear regression model is

$$\text{SSE} = \mathbf{y}'\mathbf{y} - \hat{\beta}'\mathbf{X}'\mathbf{y}, \quad (8.5)$$

and σ^2 is estimated with the error variance of the fitted model as

$$s_e^2 = \frac{\text{SSE}}{n - p - 1}. \quad (8.6)$$

The covariance matrix of the estimated regression coefficients can thus be estimated as

$$\widehat{\text{Var}(\hat{\beta})} = s_e^2(\mathbf{X}'\mathbf{X})^{-1}. \quad (8.7)$$

If the regression model is constructed for data with distinct groups, $k - 1$ dummy indicator variables can be added to the design matrix to indicate group membership, for example

$$\mathbf{X} = \begin{bmatrix} \mathbf{j} & \mathbf{X}_1 & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{j} & \mathbf{X}_2 & \mathbf{j} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{j} & \mathbf{X}_3 & \mathbf{0} & \mathbf{j} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \mathbf{j} & \mathbf{X}_k & \mathbf{0} & \mathbf{0} & \dots & \mathbf{j} \end{bmatrix}, \quad (8.8)$$

where \mathbf{X}_i contains the covariates for the i^{th} group, $\mathbf{0}$ indicates a column vector of zeros, and \mathbf{j} indicates a column vector of ones. A model with different intercepts, but equal (partial) slopes for the groups is of the form

$$Y = \beta_0 + \beta_1 D_1 + \dots + \beta_{k-1} D_{k-1} + \beta_k X_1 + \dots + \beta_{k+p-1} X_p + \epsilon, \quad (8.9)$$

where D_1, \dots, D_{k-1} indicate the $k - 1$ indicator variables in (8.8). The effect of including such indicator variables in the regression model is that k parallel hyperplanes are fitted, one for each of the groups. The assumption is therefore that the model effects are identical for the k groups, but that each group has a different intercept.

If the assumption of equal partial slopes (i.e. equal β_j values) for the different groups seems untenable, different partial slopes can be fitted by combining the group indicator variable with each of the regressors as interaction terms in the linear model. Such a model with different intercepts and different (partial) slopes is of the form

$$\begin{aligned} Y = & \beta_0 + \beta_1 D_1 + \dots + \beta_{k-1} D_{k-1} \\ & + \beta_k X_1 + \dots + \beta_{k+p-1} X_p \\ & + \beta_{k+p} D_1 X_1 + \dots + \beta_{k+2p-1} D_1 X_p + \dots \\ & + \beta_{k+(k-1)p} D_{k-1} X_1 + \dots + \beta_{k+kp-1} D_{k-1} X_p + \epsilon. \end{aligned} \quad (8.10)$$

If the number of groups and/or the number of variables included in the regression model become large, model (8.10) will have considerably more parameters to estimate than model (8.9). For small samples, the more parsimonious model (8.9) may provide a more stable solution in such a case, even if theoretically incorrect.

8.2.2 Principal component regression

When fitting linear regression models on regressors which are highly correlated among themselves, the multicollinearity usually inflates the estimated error variance of the model in (8.6). This in turn inflates the variances of the estimated coefficients in (8.7), making the estimates of these coefficients unstable. To alleviate this problem, variable selection can be performed using one of a number of procedures such as stepwise regression (Rawlings et al., 1998), comparison of all regression subsets using Mallows C_p (Rawlings et al., 1998), or even inspection of the eigenvectors of the covariance matrix of \mathbf{X} in order to manually select one variable from each highly collinear subset (see Section 2.9).

Another option is to calculate the principal components of covariance matrix of \mathbf{X} and perform the regression on the principal component scores instead of the original variables (Jolliffe, 2002). Because the eigenvectors are orthogonal and the principal components therefore uncorrelated, there is no multicollinearity to make the estimation process unstable. A disadvantage of this approach is that the estimated regression coefficients are difficult to interpret, as each regressor (i.e. principal component) is a linear combination of the original variables.

Let \mathbf{E} be the eigenvector matrix of \mathbf{S} , the sample covariance matrix of data matrix, $\mathbf{X} : n \times p$. The principal component scores of \mathbf{X} ,

$$\mathbf{Z} = \mathbf{X}\mathbf{E}, \quad (8.11)$$

are used as regressors to construct the *principal component (PC) regression* model,

$$Y = \beta_0 + \beta_1 Z_1 + \dots + \beta_q Z_q + \epsilon, \quad 1 \leq q \leq p, \quad (8.12)$$

where Z_j is the j^{th} principal component of the covariance matrix of \mathbf{X} (Jolliffe, 2002). The number of principal components included in the model can be smaller than the full dimensionality, particularly when the original variables are highly collinear and the first q components account for almost all of the observed variation in the data. In such cases the last $p - q$ components (associated with the smallest eigenvalues) can be assumed to account

predominantly for noise, and can be discarded before fitting the regression model.

In the presence of multicollinearity among the original variables, PC regression on a reduced number of principal components can decrease the variances of the estimated regression coefficients substantially, at the risk of introducing comparatively little bias (Jolliffe, 2002). Suppose that the regression coefficients in (8.12) are estimated as

$$\hat{\boldsymbol{\beta}} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{y}, \quad (8.13)$$

where the columns of \mathbf{Z} are the principal component scores and \mathbf{y} the response vector. If the columns of \mathbf{Z} are centred,

$$(\mathbf{Z}'\mathbf{Z})^{-1} = (n-1) \sum_{j=1}^p d_j^{-1} \mathbf{e}_j \mathbf{e}_j', \quad (8.14)$$

which means that (8.13) can be written as

$$\hat{\boldsymbol{\beta}} = (n-1) \left(\sum_{j=1}^p d_j^{-1} \mathbf{e}_j \mathbf{e}_j' \right) \mathbf{Z}'\mathbf{y}. \quad (8.15)$$

The covariance matrix of the estimated regression coefficients, analogous to (8.7), are estimated by

$$\widehat{\text{Var}}(\hat{\boldsymbol{\beta}}) = s_e^2(n-1) \sum_{j=1}^p d_j^{-1} \mathbf{e}_j \mathbf{e}_j'. \quad (8.16)$$

Multicollinearity will produce at least one very small eigenvalue in the covariance matrix. If a PC regression model is fitted only on $q < p$ components associated with the larger eigenvalues, the smallest eigenvalues (d_j) will be removed from (8.16), leading to a reduction in the variances of the regression coefficients (Jolliffe, 2002).

8.2.3 Common principal component regression

Flury (1988) suggested that the common principal component scores of the \mathbf{X}_i , $i = 1, \dots, k$, data matrices,

$$\mathbf{Z} = \mathbf{X}\mathbf{B}, \quad (8.17)$$

where

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_k \end{bmatrix}, \quad (8.18)$$

and

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_k \end{bmatrix}, \quad (8.19)$$

and \mathbf{B} is the common eigenvector matrix, can be used in a regression model in a similar way as when performing PC regression. The *CPC regression* model is

$$Y = \beta_0 + \beta_1 Z_1 + \dots + \beta_q Z_q + \epsilon, \quad 1 \leq q \leq p, \quad (8.20)$$

where Z_j is the j^{th} common principal component of covariance matrices of the \mathbf{X}_i matrices. As for the OLS and PC regression models, $k - 1$ dummy indicator variables can be added to the design matrix to indicate group membership, for example

$$\begin{bmatrix} j & \mathbf{Z}_1 & 0 & 0 & \dots & 0 \\ j & \mathbf{Z}_2 & j & 0 & \dots & 0 \\ j & \mathbf{Z}_3 & 0 & j & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ j & \mathbf{Z}_k & 0 & 0 & \dots & j \end{bmatrix}, \quad (8.21)$$

where \mathbf{Z}_i contains the common principal component scores for the i^{th} group. This allows the possibility of fitting CPC regression models with different intercepts and/or partial slopes for the different groups.

8.2.4 Partial least squares regression

Partial least squares (PLS) regression works in a similar way as PC and CPC regression, in that a regression model is fitted to a number of orthogonal components which are linear combinations of the original numerical input variables (regressors). The main difference lies therein that both the input and response variables are used in the calculation of the PLS components, instead of only the input variables (as done in PC and CPC regression).

The PLS components are calculated in a way that simultaneously maximises the variance of each component and also its correlation with the response, subject to the constraint that each additional component is orthogonal to the preceding ones. As mentioned by Hastie et al. (2009), the PLS algorithm places more weight on the maximisation of the variance than correlation with the response, which makes it very similar to PC and CPC regression.

The work in this section is mostly derived from Hastie et al. (2009), where a description of the PLS algorithm was given. The columns of the input data matrix $\mathbf{X} : n \times p$ is standardised to have zero mean and unit variance, because the PLS component directions are not scale invariant.

The first PLS component is given by

$$\mathbf{z}_1 = \sum_{j=1}^p \psi_{1j} \mathbf{x}_j, \quad (8.22)$$

where ψ_{1j} is the inner product of the j^{th} column of \mathbf{X} and the response vector, \mathbf{y} , i.e.

$$\psi_{1j} = \mathbf{x}_j' \mathbf{y}. \quad (8.23)$$

To obtain the remaining $p-1$ PLS components, the following procedure is repeated for each component: After calculating the h^{th} , $h = 1, \dots, p$ component, the columns of \mathbf{X} are orthogonalised with respect the h^{th} component, i.e.

$$\mathbf{x}_j^{(h)} = \mathbf{x}_j^{(h-1)} - \left[\frac{\mathbf{z}_h' \mathbf{x}_j^{(h-1)}}{\mathbf{z}_h' \mathbf{z}_h} \right] \mathbf{z}_h, \quad j = 1, \dots, p, \quad (8.24)$$

where $\mathbf{x}_j^{(h)}$ indicates the j^{th} column of \mathbf{X} at the calculation of the h^{th} component. For calculation of the first component, $\mathbf{x}_j^{(h-1)} = \mathbf{x}_j^{(0)}$ is set equal to the original j^{th} input variable.

Each of the remaining $p-1$ components are calculated in turn as

$$\mathbf{z}_h = \sum_{j=1}^p \psi_{hj} \mathbf{x}_j^{(h-1)}, \quad (8.25)$$

where

$$\psi_{hj} = \mathbf{x}_j^{(h-1)'} \mathbf{y}, \quad (8.26)$$

and the $\mathbf{x}_j^{(h)}$ is updated for each additional component.

The PLS regression model is fitted to the first $1 \leq q \leq p$ orthogonal PLS components. If all p components are included in the model, the OLS fit is obtained.

The PLS components in $\mathbf{Z} = [\mathbf{z}_1 \ \dots \ \mathbf{z}_p]$ are obtained by a rotation of the original input matrix, i.e.

$$\mathbf{Z} = \mathbf{X}\mathbf{B}_{\text{PLS}}, \quad (8.27)$$

and the linear combinations (i.e. the columns of \mathbf{B}_{PLS}) to construct the PLS components can therefore be recovered with

$$\begin{aligned} (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Z} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}\mathbf{B}_{\text{PLS}} \\ &= \mathbf{B}_{\text{PLS}}. \end{aligned} \quad (8.28)$$

8.3 Comparison of model fit

To compare the fit of regression models with different numbers of parameters, the Akaike Information Criterion (AIC), (Akaike, 1974),

$$\text{AIC} = -2L + 2p \quad (8.29)$$

can be used, where L is the maximised log-likelihood of the model and p is the number of parameters in the model. The model with the lowest AIC value provides the best fit. For a more complex model to have a better fit, the increase in the log-likelihood should be large enough to compensate for the increase in the number of parameters.

The coefficient of determination (R^2) measures the proportion of variation in a numerical response accounted for by a fitted model. For a simple linear regression model, it is calculated as

$$R^2 = \frac{\sum_{m=1}^n (\hat{y}_m - \bar{y})^2}{\sum_{m=1}^n (y_m - \bar{y})^2}, \quad (8.30)$$

where y_m and \hat{y}_m indicate the actual and fitted response values for the m^{th} observation, respectively, and \bar{y} is the observed mean of the response variable (Draper and Smith, 1998). R^2 is equal to the square of the correlation between the observed response variable and the fitted values. The R^2 statistic can thus also be calculated as the square of the Pearson correlation coefficient between the fitted values and the response (Ryan, 2008), i.e.

$$R^2 = \frac{[\sum_{m=1}^n (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})]^2}{[\sum_{m=1}^n (\hat{y}_i - \bar{\hat{y}})^2] [\sum_{m=1}^n (y_i - \bar{y})^2]}. \quad (8.31)$$

If there are multiple predictor variables in the regression model, R^2 is referred to as the *coefficient of multiple determination* (Ryan, 2008).

However, R^2 is not a good statistic to compare the fit of models with different numbers of regressors, as the R^2 statistic for a specific sample and model will always increase if more regressors are added to the model, up to the maximum value of one. A modified version of the coefficient of multiple determination, the *adjusted R^2*

$$\text{Adjusted } R^2 = 1 - (1 - R^2) \left(\frac{n - 1}{n - p} \right), \quad (8.32)$$

is a measure which penalises an increase in the number of model parameters if the additional variables do not provide a sufficient increase in predictive value (Draper and Smith, 1998). Among competing models with different levels of complexity, the one with the largest adjusted R^2 value can be chosen as the best.

To judge model fit for logistic regression models where the response variable is binary, the area under the *receiver operating characteristic (ROC)* curve is commonly used (Agresti, 2003). With a response variable taking the values 0 or 1, a 2×2 classification table of the observation counts can be constructed for the observed values (0 or 1) against the predicted values (0 or 1, determined by using some cut-off point probability, π) from the fitted model. Defining the measures,

$$\text{Specificity} = P(\hat{y} = 1 | y = 1), \quad (8.33)$$

i.e. the probability of a true positive prediction (with 1 being labelled “positive” for the purpose of this explanation), and

$$\text{Sensitivity} = P(\hat{y} = 0 | y = 0), \quad (8.34)$$

i.e. the probability of a true negative prediction, the ROC curve is a plot of sensitivity as a function of (1 - specificity) for all possible values of π (Agresti, 2003). As sensitivity and specificity are both probabilities and therefore constrained to fall in the interval $[0; 1]$, the area under the ROC curve is also constrained to this same interval. The greater the area under the curve (AUC), the greater is the predictive ability of the model (Agresti, 2003).

8.4 Application to the VON data

Parents of newborn babies admitted to NICUs have a strong interest in the prognosis regarding mortality and the anticipated discharge of their babies

from hospital. These factors are also considered by medical staff and hospital administrators when they have to decide on treatment regimes and allocate resources to the care of these infants.

The purpose of this part of the dissertation is to construct regression models to predict neonatal mortality and length of stay (LOS) for NICU admissions from easily measurable perinatal variables. Such models can be useful to healthcare professionals working in Southern African private hospitals by supplying prognostic estimates to aid in decision making and the counselling of concerned parents.

As mortality and LOS are two separate issues (Hintz et al., 2010), they were modelled separately in the following two sections.

In order to construct the regression models, only perinatal input variables which under normal circumstances are readily available upon admission to a NICU were used. The six continuous measures are Apgar scores at one and five minutes after birth (*AP1* and *AP5*), temperature measured within one hour after birth (*ATEMP*), birth weight (*BWGT*), gestational age (*GESTAGE*) and birth head circumference (*BHEADCIR*). The four nominal variables are delivery mode (*VAGDEL*), gender (*SEX*), maternal ethnicity (*RACE*), and hospital region (*REGION*).

The response variables are survival to hospital discharge and total hospital LOS in days. The mortality response variable is coded as a one if the infant died during hospital stay (including delivery room deaths), and as a zero if the infant survived until discharge.

A total of 2930 observations from the VON 2009 cohort (infants born in 2009, excluding transfers and cases with incomplete data) were used to construct the models. Another 2307 observations were available from the VON 2008 cohort, which were used to test the regression models. Descriptive statistics for these two cohorts were given in Section 1.2. Infants who were transferred to alternative NICU facilities before final discharge were excluded from the regression models, as the mortality status and LOS for these cases are unknown.

Bivariate analyses were performed to gauge the effects of the input variables on each of the two response variables. For mortality, univariate generalised linear models, using a logit link function (Agresti, 2003), were constructed for each of the input variables. The AUC was calculated for each of these models, indicating the usefulness of each of the variables in predicting neonatal mortality.

In a similar way, univariate generalised linear models, with a log link function (Agresti, 2003), were used to test the effect of each of the input variables on LOS. The log link function was necessary to make the distribution of LOS, which typically is positively skewed, closer to normal. The

coefficient of determination was used to determine the usefulness of each of the variables in predicting LOS.

Because there is multicollinearity among the numerical variables, a decision on the inclusion or exclusion of these variables from the regression model had to be made. Applying traditional variable selection techniques such as stepwise regression and C_p -selection failed to correct the problem. Variable selection was therefore performed by selecting one numerical variable from each of three variable subsets obtained from a principal component analysis, as discussed in Section 2.9. The three variable subsets are:

- Subset 1: *GESTAGE*, *BHEADCIR*;
- Subset 2: *AP1*, *AP5*;
- Subset 3: *ATEMP*.

The sixth eigenvector of the covariance matrix of the VON 2009 cohort is dominated by *BWGT* (see Table 2.1), but accounts for only 0.5% of the observed variation, and *BWGT* was therefore excluded. *BWGT* is also strongly correlated with both *GESTAGE* ($r = 0.83$) and *BHEADCIR* ($r = 0.85$).

8.4.1 Mortality

To assess the effects of each of the input variables on mortality, simple logistic regression models (Agresti, 2003) of mortality on each of the input variables were fitted. The results are shown in Table 8.1.

When analysed separately, all of the numerical variables have significant effects on neonatal mortality. Apgar score at one minute ($p < 0.0001$), gestational age ($p < 0.0001$), and temperature ($p < 0.0001$), are the variables from the three subsets identified in Section 8.4 which are most significantly associated with the two mortality groups (*Survived* and *Died*). These three variables are used in the regression models on the original variables, discussed in the rest of this section.

Delivery mode is the only nominal variable showing a significant effect on mortality. The other nominal variables (region, maternal ethnicity and gender) did not have significant predictive power when considered separately.

The following three types of logistic regression models were fitted to the VON 2009 data to predict neonatal mortality:

- **Model M1:** Multiple logistic regression of mortality on *AP1*, *GESTAGE*, *ATEMP* (which were chosen by inspection of the eigenvectors of the covariance matrix of the VON 2009 cohort) and *VAGDEL*;

Table 8.1: Univariate effects of the perinatal input variables on mortality. The area under the receiver operating characteristic curve (AUC) for each of the simple logistic regression models are given in the last column.

	Effect on mortality		AUC
	(odds ratio)	<i>p</i> -value	
Birth weight	0.17	< 0.0001	0.79
Apgar at 1 min	0.67	< 0.0001	0.76
Apgar at 5 mins	0.62	< 0.0001	0.77
Gestational age	0.71	< 0.0001	0.77
Birth head circumference	0.72	< 0.0001	0.75
Temperature	0.50	< 0.0001	0.61
Region			0.50
Namibia	(base)		
South Africa	0.82	0.6810	
Maternal ethnicity			0.59
Asian	(base)		
Black	1.73	0.4500	
Other	0.83	0.8210	
White	0.81	0.7710	
Delivery mode			0.58
Caesarean	(base)		
Vaginal	2.53	< 0.0001	
Gender			0.52
Female	(base)		
Male	0.87	0.4910	

- **Model M2:** Multiple logistic PC regression of mortality on the principal components of the VON 2009 cohort, and *VAGDEL*;
- **Model M3:** Multiple logistic CPC regression of mortality on the common principal components of the VON 2009 cohort, and *VAGDEL*.

For all three types of models, tests for equal intercepts and equal partial slopes for the delivery mode groups (*Caesarean* and *Vaginal*) were performed.

Model M1

The best model of type *M1* (AIC = 688.58) is shown in the table below:

	Estimate	Std. Error	<i>p</i> -value	Odds ratio
Intercept	7.239	0.840	< 0.0001	
Apgar at 1 min	-0.284	0.045	< 0.0001	0.75
Gestational age	-0.270	0.027	< 0.0001	0.76
Delivery mode	0.857	0.247	0.0005	2.36

Temperature (*ATEMP*) was dropped from the model because it was not significant ($p = 0.3546$). There also was not sufficient evidence for different intercepts or different partial slopes for the delivery mode groups at a 5% significance level.

Model M2

The principal components were calculated by using the eigenvectors of the covariance matrix of the six numerical input variables in the VON 2009 data set. For the model of type *M2*, the last four principal components were dropped from the model because they were not significant at a 5% level. There also was insufficient statistical evidence for different intercepts and partial slopes for the two delivery mode groups. The best PC regression model (AIC = 679.94) for neonatal mortality is shown in the table below:

	Estimate	Std. Error	<i>p</i> -value	Odds ratio
Intercept	8.187	0.856	< 0.0001	
PC1	0.228	0.019	< 0.0001	1.26
PC2	0.206	0.036	< 0.0001	1.23
Delivery mode	0.778	0.249	0.0018	2.18

Model M3

The common principal components were calculated by using the common eigenvector matrix of the covariance matrices of the *Survived* and *Died* groups, estimated with the FG algorithm. For the model of type *M3*, the last four common principal components are not significant at a 5% level and were therefore dropped from the model. Again there was insufficient evidence for different intercepts or different partial slopes for the delivery mode groups. The best CPC regression model ($AIC = 679.97$) for mortality is reported in the table below:

	Estimate	Std. Error	<i>p</i> -value	Odds ratio
Intercept	8.151	0.853	< 0.0001	
CPC1	0.222	0.019	< 0.0001	1.25
CPC2	0.213	0.036	< 0.0001	1.24
Delivery mode	0.779	0.249	0.0018	2.18

ROC curves for the three mortality models are shown in Figure 8.1. The AUC values for models *M1*, *M2* and *M3* are 0.8254 (standard error: 0.0253), 0.8338 (standard error: 0.0235) and 0.8338 (standard error: 0.0235), respectively, showing that all three models fit the data about equally well. The standard errors of the AUC values were calculated using the `Hmisc` package in R (Harrell, 2012), with the instructions from Harrell (2009) and Harrell (2011).

Optimal (predicted probability) cut-off points to classify infants into the *Survived* or *Died* groups were determined on the VON 2009 data. The cut-off points for models *M1*, *M2* and *M3* are predicted probabilities of 0.0971, 0.0618 and 0.0617, respectively. The results of applying these cut-off points to classify the infants in the VON 2008 cohort is shown in Table 8.2. Model *M1* has the smallest misclassification error rate, largest specificity, and largest positive predictive value (PPV), (Altman, 1990). However, the sensitivity of model *M1* is smaller than for the other two models, which means that it gives a greater proportion of Type II (false negative) errors. In this setting, sensitivity (in accurately identifying high-risk infants) will be most important, and therefore the Type II errors (false negatives) are of a more serious nature than Type I errors (false positives).

For predictive purposes, the PC or CPC regression models will be preferred. For a predicted death with either of these two models, the odds of dying are about 21 times greater than for a predicted survival. Either model can therefore be used as a simple screening test to identify high mortality risk infants upon their admission to the NICU.

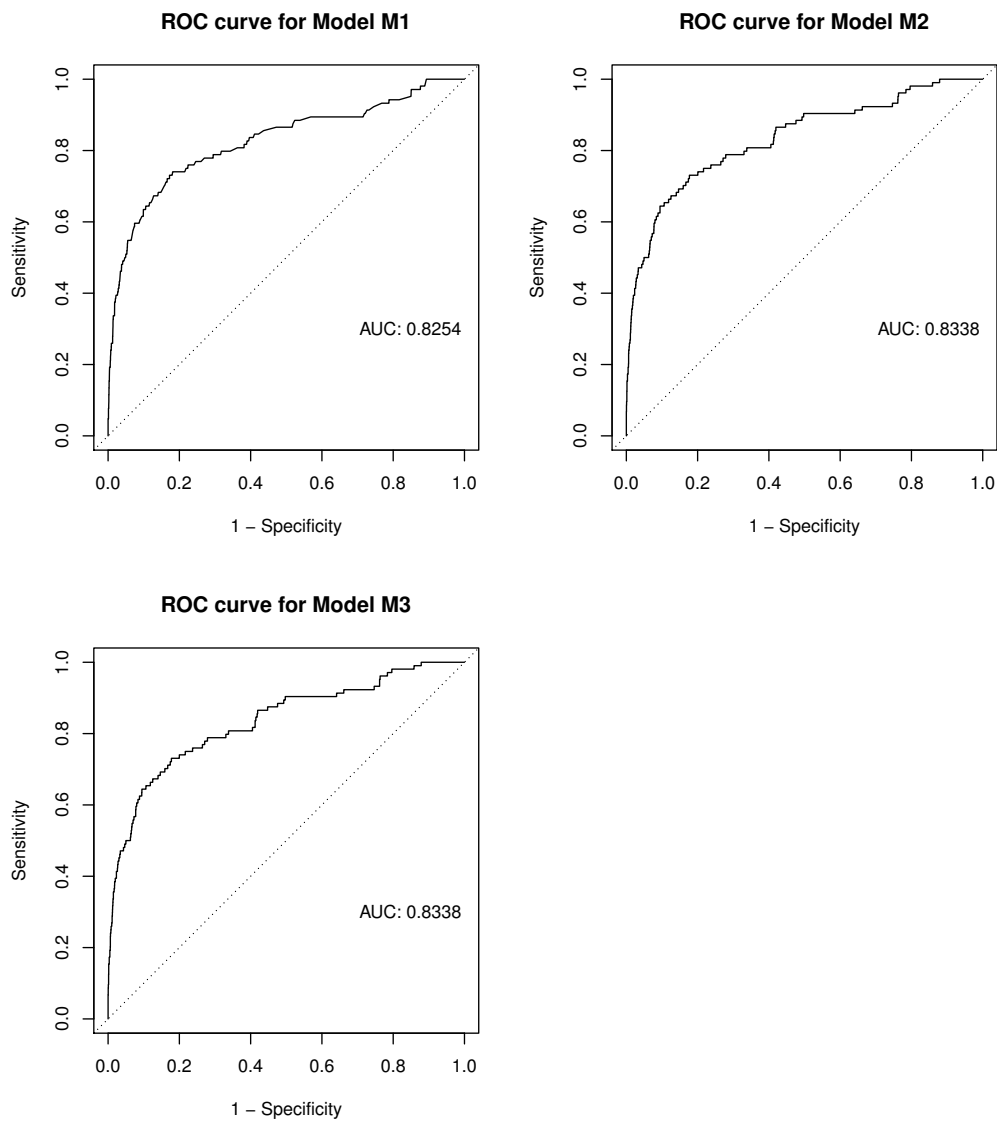


Figure 8.1: ROC curves for the three regression models to predict mortality for NICU admissions.

Table 8.2: Model statistics for classification of the infants in the VON 2008 cohort regarding mortality.

	Misclassification error rate (%)	Sensitivity	Specificity	Positive predictive value	Negative predictive value
Model M1	6.3%	56.0%	95.3%	32.7%	98.1%
Model M2	9.8%	67.0%	91.2%	23.7%	98.5%
Model M3	9.8%	67.0%	91.2%	23.7%	98.5%

However, the parameter estimates of the PC and CPC regression models are more difficult to interpret than those of the regression model constructed on the original variables (*M1*). To gain insight into the factors influencing neonatal mortality, the estimated effects for model *M1* will be interpreted here.

Increases in Apgar score at one minute and gestational age had negative effects on the odds of neonatal death. For each one unit increase in the Apgar score at one minute, the odds of neonatal death decreases on average by roughly 25%. With each week increase in gestational age, the odds of neonatal death decreases by 24% on average.

Vaginal delivery appeared to increase the odds of neonatal death with 136% compared to delivery by Caesarean section, but this effect should be interpreted with caution: It is most probably due to differences in the NICU admission protocol for vaginal and Caesarean deliveries and not due to any real effect. The infants delivered by Caesarean section were on average weaker, sicker, and more premature than their vaginal delivery counterparts and were more readily admitted to NICU for observation only, as was also observed in a Canadian study by Fallah et al. (2011). On the other hand, infants born by vaginal delivery were usually less premature and were admitted to NICU only for more serious birth defects or illnesses. Therefore, given that the infants have been admitted to an NICU, the pool of infants delivered by Caesarean section were on average in better health compared to those born by vaginal delivery, thereby skewing the results.

The predictive value of the fitted models may have been diminished to some extent by the difference in size of the two mortality groups (*Survived* and *Died*). However, the PC and CPC regression models present simple methods to indicate high mortality risk neonates: The odds of death for positive predictions are about 21 times higher than for negative predictions.

8.4.2 Length of stay

LOS is a discrete numerical variable measured on the ratio scale. It has a Poisson type distribution, as can be seen from the top part of Figure 8.2. In order to fit linear regression models to predict LOS, Poisson loglinear models (corrected for overdispersion) were used (Agresti, 2003). This entails fitting a linear regression model to the natural logarithm of LOS, of which the distribution is shown in the bottom part of Figure 8.2.

A scatterplot matrix of $\ln(\text{LOS})$ and the six numerical input variables is shown in Figure 8.3. There seem to be negative linear relationships between $\ln(\text{LOS})$ and each of the numerical variables, with the exception of *ATEMP* which does not show a clear relationship with LOS. The positive correlations between *BWGT*, *GESTAGE* and *BHEADCIR* can clearly be seen, as well as the positive correlation between *AP1* and *AP5*.

To investigate the relationship of each of the input variables with the response, simple Poisson loglinear models of LOS on each of the predictor variables were fitted. The results, together with coefficient of determination (R^2) values are reported in Table 8.3. All of the numerical variables showed statistically significant effects on LOS, although the Apgar scores and temperature had limited usefulness in explaining the variation in LOS (as indicated by their low R^2 values). Region, delivery mode and gender were also significant when considered separately, but their low R^2 values indicate that these variables add little explanatory power. The maternal ethnicity groups did not seem to differ significantly from each other with regard to LOS.

To determine which of the numerical variables from each of the three correlated subsets identified with PCA in Section 8.4 to include in the regression models on the original numerical input variables, Spearman rank-order correlation coefficients (Spearman, 1904) between each of the variables and $\ln(\text{LOS})$ were calculated. These correlation coefficients are reported below:

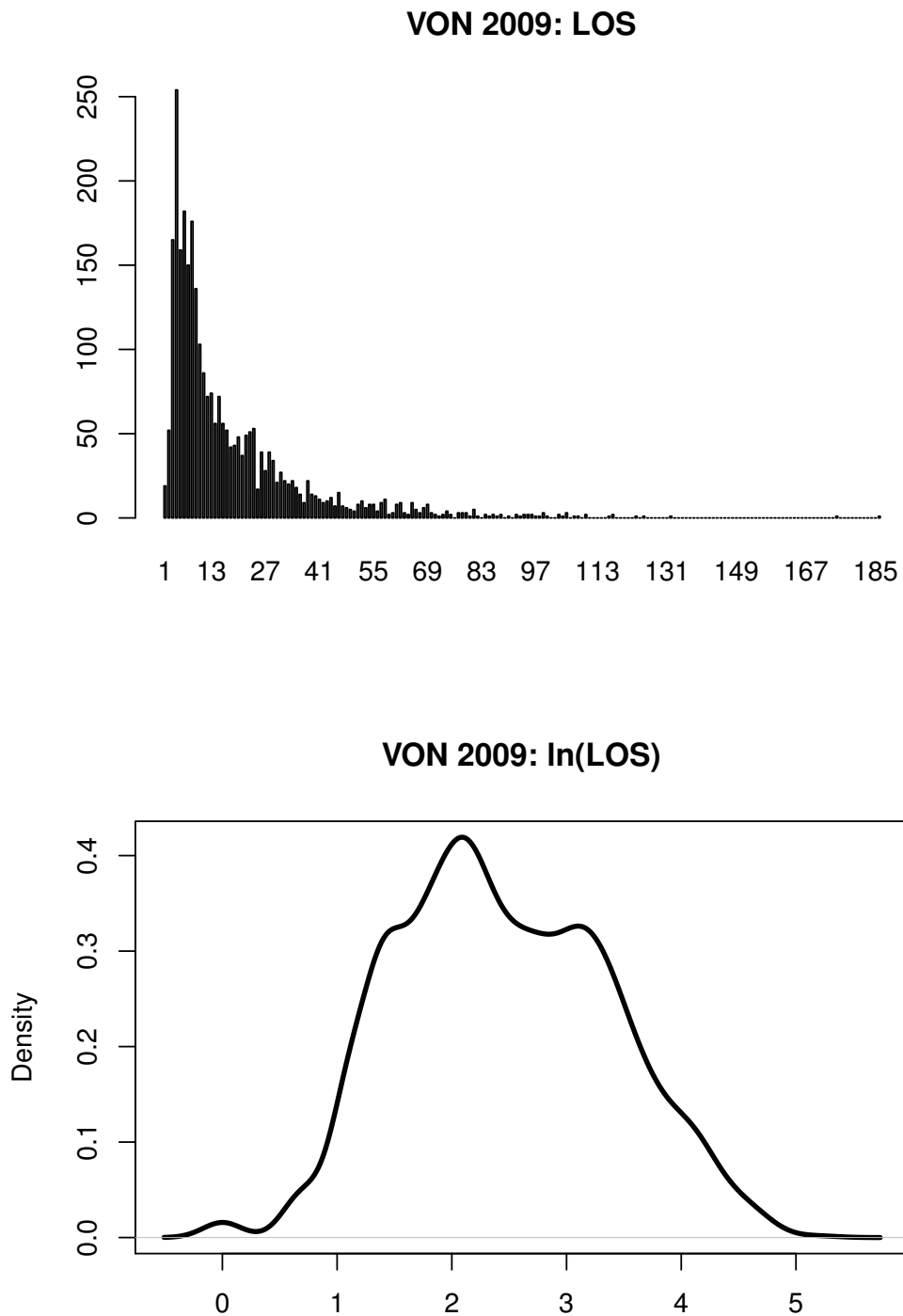


Figure 8.2: Distribution of length of stay (LOS) and the natural logarithm of LOS for the VON 2009 data.

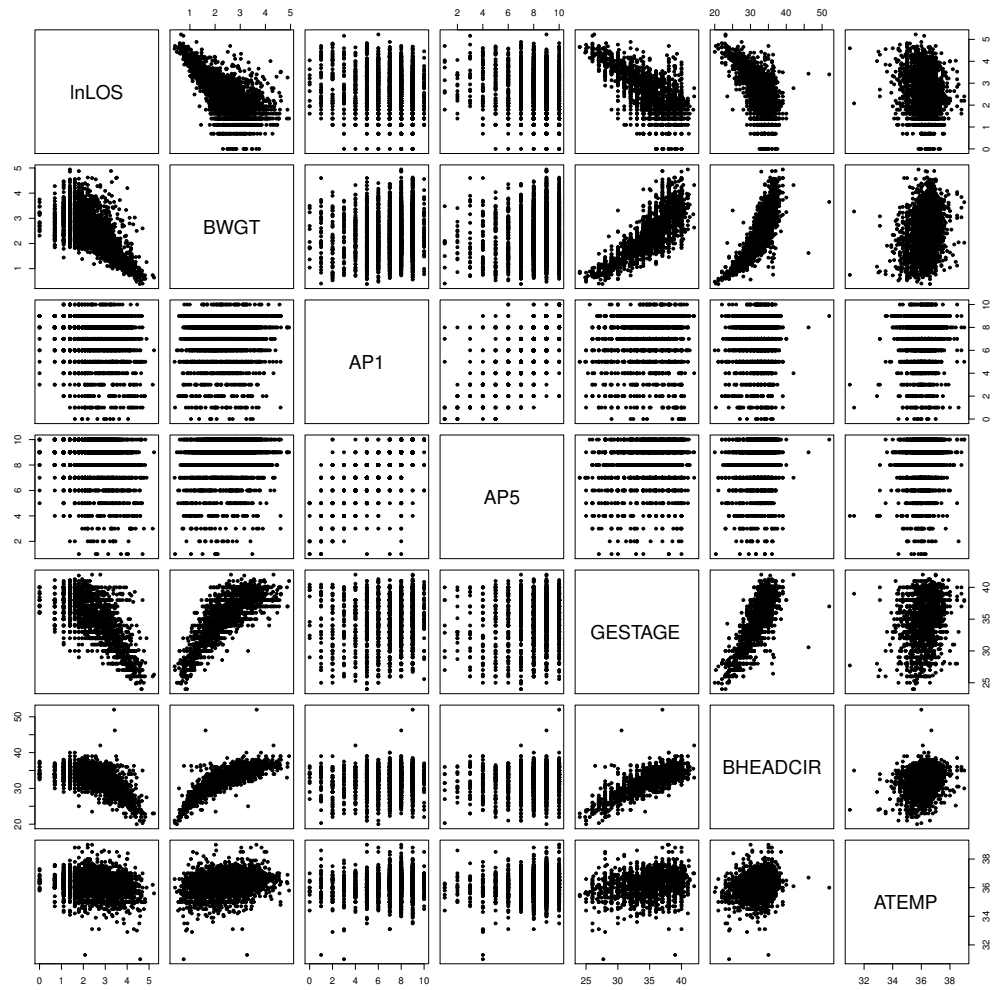


Figure 8.3: Scatter plot matrix of $\ln(\text{LOS})$ and the six numerical variables in the VON 2009 data set.

Table 8.3: Univariate effects of the perinatal input variables on LOS of NICU admissions.

	Effect on LOS	<i>p</i> -value	R ²
Birth weight	−61.5%	< 0.0001	0.67
Apgar at 1 min	−10.3%	< 0.0001	0.05
Apgar at 5 mins	−12.4%	< 0.0001	0.05
Gestational age	−17.7%	< 0.0001	0.63
Birth head circumference	−17.8%	< 0.0001	0.61
Temperature	−23.5%	< 0.0001	0.04
Region			0.003
Namibia	(base)		
South Africa	−22.5%	0.0047	
Maternal ethnicity			0.002
Asian	(base)		
Black	−8.7%	0.4770	
Other	+4.5%	0.7460	
White	−2.0%	0.8760	
Delivery mode			0.004
Caesarean	(base)		
Vaginal	−17.6%	0.0010	
Gender			0.006
Female	(base)		
Male	−14.9%	< 0.0001	

	ln(LOS)	BWGT	AP1	AP5	GES- TAGE	BHEAD- CIR	ATEMP
ln(LOS)	1.00	-0.69	-0.18	-0.18	-0.67	-0.61	-0.16
BWGT	-0.69	1.00	0.17	0.16	0.83	0.86	0.28
AP1	-0.18	0.17	1.00	0.76	0.15	0.15	0.17
AP5	-0.18	0.16	0.76	1.00	0.14	0.13	0.17
GESTAGE	-0.67	0.83	0.15	0.14	1.00	0.75	0.24
BHEADCIR	-0.61	0.86	0.15	0.13	0.75	1.00	0.23
ATEMP	-0.16	0.28	0.17	0.17	0.24	0.23	1.00

The Spearman rank-order correlations confirmed the grouping of variables obtained from PCA: The two Apgar score variables are strongly correlated ($r = 0.76$), and so are the variables pertaining to size of the neonates (all $r \geq 0.75$).

Together with *ATEMP*, *AP5* and *GESTAGE* were the respective variables from the first and second PCA subsets that show the strongest correlation with $\ln(\text{LOS})$. These three numerical variables were therefore included in the regression models constructed on the original variables.

The following four types of Poisson loglinear regression models were fitted to the VON 2009 data to predict LOS:

- **Model L1:** Poisson multiple loglinear regression of LOS on *AP5*, *GESTAGE*, *ATEMP* (which were chosen by inspection of the eigenvectors of the covariance matrix of the VON 2009 cohort), *REGION*, *VAGDEL*, and *SEX*;
- **Model L2:** Poisson multiple loglinear PC regression of LOS on the principal components of the VON 2009 cohort, and *REGION*, *VAGDEL*, and *SEX*;
- **Model L3:** Poisson multiple loglinear CPC regression of LOS on the common principal components of the VON 2009 cohort, and *REGION*, *VAGDEL*, and *SEX*;
- **Model L4:** Poisson multiple loglinear PLS regression of LOS on the PLS components of the VON 2009 cohort, and *REGION*, *VAGDEL*, and *SEX*.

The models were corrected for overdispersion, and variables not significant at a 5% level were sequentially dropped. The assumptions of equal intercepts and equal (partial) slopes were also investigated for each of the nominal variables.

Model L1

The best model of type $L1$ ($R^2 = 0.6385$, adjusted $R^2 = 0.6380$) is shown in the table below:

	Estimate	Std. Error	p -value
Intercept	9.768	0.119	< 0.0001
Apgar at 5 mins	-0.037	0.007	< 0.0001
Gestational age	-0.189	0.003	< 0.0001
Region			
South Africa	-0.147	0.055	0.0078
Delivery mode			
Vaginal	-0.083	0.036	0.0215

Increases in Apgar score at five minutes ($p < 0.0001$) and gestational age ($p < 0.0001$) both had negative effects on $\ln(\text{LOS})$. Greater values for these two measures are thus associated with shorter LOS. The regions differ with regards to LOS ($p = 0.0078$), and it seems from the model estimates that infants admitted to South African NICUs had significantly shorter LOS than infants admitted to NICUs in the participating Namibian hospitals. Vaginal delivery ($p = 0.0215$) is associated with a shorter LOS than Caesarean delivery.

Other studies by Powell et al. (1992) and Hintz et al. (2010) found gestational age to be the strongest predictor for LOS. In this retrospective observational study, gestational age, birth weight and birth head circumference are highly correlated, with gestational age also emerging as the strongest predictor for LOS.

Model L2

The parameter estimates for the best Poisson multiple loglinear PC regression model ($R^2 = 0.7159$, adjusted $R^2 = 0.7153$) are shown below:

	Estimate	Std. Error	p -value
Intercept	7.247	0.299	< 0.0001
PC1	0.155	0.002	< 0.0001
PC3	0.108	0.033	0.0010
PC6	-0.360	0.037	< 0.0001
Region			
South Africa	-0.071	0.054	0.1875
Delivery mode			
Vaginal	-0.070	0.034	0.0419
PC3 \times South Africa	-0.129	0.034	0.0001

Only the first, third and sixth principal components are significantly associated with the response variable. The model intercepts for the two regions do not differ significantly ($p = 0.1875$), but the partial slopes for $PC3$ differ between the two regions ($p = 0.0001$). The third principal component is dominated by a contrast between *GESTAGE* and *BHEADCIR*, and is thus an indication of whether the infant has a small/large head relative to his/her gestational age. From the parameter estimates given above, it seems that this component has a larger influence on LOS in the *Namibia* group than in the *South Africa* group (as the coefficient for $PC3$ is positive, whereas the combination of the coefficients for $PC3$ and $PC3 \times \text{South Africa}$ is comparatively close to zero). The delivery mode groups are again seen to differ with regard to LOS ($p = 0.0419$).

Model L3

The common principal components of the regional groups (*South Africa* and *Namibia*) were calculated for the VON 2009 cohort, and used as regressors in the regression model of type *L3*. The best Poisson multiple loglinear CPC regression model ($R^2 = 0.7157$, adjusted $R^2 = 0.7151$) is shown below:

	Estimate	Std. Error	<i>p</i> -value
Intercept	7.166	0.306	< 0.0001
CPC1	0.155	0.002	< 0.0001
CPC3	0.109	0.033	0.0009
CPC6	-0.359	0.037	< 0.0001
Region			
South Africa	-0.056	0.055	0.3097
Delivery mode			
Vaginal	-0.070	0.034	0.0409
CPC3 \times South Africa	-0.130	0.034	0.0001

The first, third and sixth common principal components of the regional groups are significantly associated with the response variable. As in the case of model *L2*, there is not sufficient evidence for a difference in model intercepts for the regional groups ($p = 0.3097$), but the delivery mode groups differ significantly ($p = 0.0409$). There is a difference between the two regions with regard to the partial slope for $CPC3$ ($p = 0.0001$), with a greater effect for this component on LOS in the *Namibia* group than in the *South Africa* group. Inspection of the common eigenvectors shows that this component is mainly a contrast of the *GESTAGE* and *BHEADCIR* variables (with the interpretation being the same as for $PC3$ in model *L2*).

Model L4

The PLS vectors estimated from the six numerical input variables and $\ln(\text{LOS})$ in the VON 2009 data set are given in Table 8.4.

Table 8.4: PLS vectors for the six numerical input variables and $\ln(\text{LOS})$ in the VON 2009 data.

	PLS1	PLS2	PLS3	PLS4	PLS5	PLS6
BWGT	-0.56	-0.13	0.10	0.11	-0.17	-0.80
AP1	-0.15	0.45	-0.19	0.22	-0.65	0.12
AP5	-0.17	0.34	-0.43	-0.10	0.65	-0.13
GESTAGE	-0.58	-0.42	-0.52	-0.70	-0.15	0.33
BHEADCIR	-0.54	-0.01	0.42	0.63	0.32	0.47
ATEMP	-0.14	0.70	0.57	-0.21	0.00	0.05

Parameter estimates for the best Poisson multiple loglinear PLS regression model ($R^2 = 0.7212$, adjusted $R^2 = 0.7202$) are reported below:

	Estimate	Std. Error	<i>p</i> -value
Intercept	2.703	0.058	< 0.0001
PLS1	0.398	0.007	< 0.0001
PLS2	0.109	0.011	< 0.0001
PLS4	-0.274	0.106	0.0101
PLS5	-0.189	0.092	0.0409
PLS6	0.095	0.032	0.0036
Region			
South Africa	-0.072	0.058	0.2134
Delivery mode			
Vaginal	-0.081	0.035	0.0187
PLS4 \times South Africa	0.327	0.109	0.0027
PLS5 \times South Africa	0.229	0.095	0.0162
PLS2 \times Vaginal	-0.064	0.023	0.0046

The first, second, fourth, fifth and sixth PLS components are statistically significant. The regions do not differ from each other ($p = 0.2134$), but it is retained in the model as it is involved in significant two-way interactions. The delivery mode groups differ significantly from each other ($p = 0.0187$).

The regions differ with regard to the partial slopes for the fourth ($p = 0.0027$) and fifth ($p = 0.0162$) PLS components. The fourth component is dominated by a contrast between *GESTAGE* and *BHEADCIR*, and is thus an indication of whether the head of the infant is small/large for the gestational

8.5. DIFFERENCES BETWEEN PC AND CPC REGRESSION 265

age. An inspection of the parameter estimates shows that this component has a greater effect on LOS in *Namibia* than in *South Africa*, which is the same conclusion as was made from the *L2* and *L3* models.

The fifth component is dominated by a contrast between *AP1* and *AP5*, and thus represents the change in the feasibility of life for the infant, from the first to the fifth minute after delivery. Inspection of the parameter estimates shows that this component also has a greater effect on LOS in *Namibia* than in *South Africa*.

Analysis of the residuals from the respective models showed that the model fits are adequate. There were no extreme outliers or distinctive patterns in any of the residual plots.

The predictive value of the four models were calculated by squaring the correlations between the predicted LOS values and actual LOS values for the VON 2008 data set. These predicted R^2 values are shown in Table 8.5. Model *L4* has the largest predicted R^2 , followed closely by models *L2* and *L3*. *L1* performed the worst of the four models on the VON 2008 test data set. Scatterplots of the predicted LOS for each of the models against the actual LOS values are shown in Figure 8.4.

Table 8.5: Predicted coefficients of determination for models to predict LOS, calculated from the data of the VON 2008 cohort.

	Predicted R^2
Model L1	0.6441
Model L2	0.7182
Model L3	0.7178
Model L4	0.7201

8.5 Differences between PC and CPC regression

In the previous section it was seen that the PC regression and CPC regression models give similar results and have similar model fits when attempting to predict either mortality or LOS for the NICU admissions in the VON data. This will not always be the case.

When the full number of principal components, or alternatively the common principal components, are used in fitting the regression model, the results from PC and CPC regression should be identical (except for any differences due to sign differences between the eigenvectors and common eigenvectors). The reason for this is that both sets of components are obtained

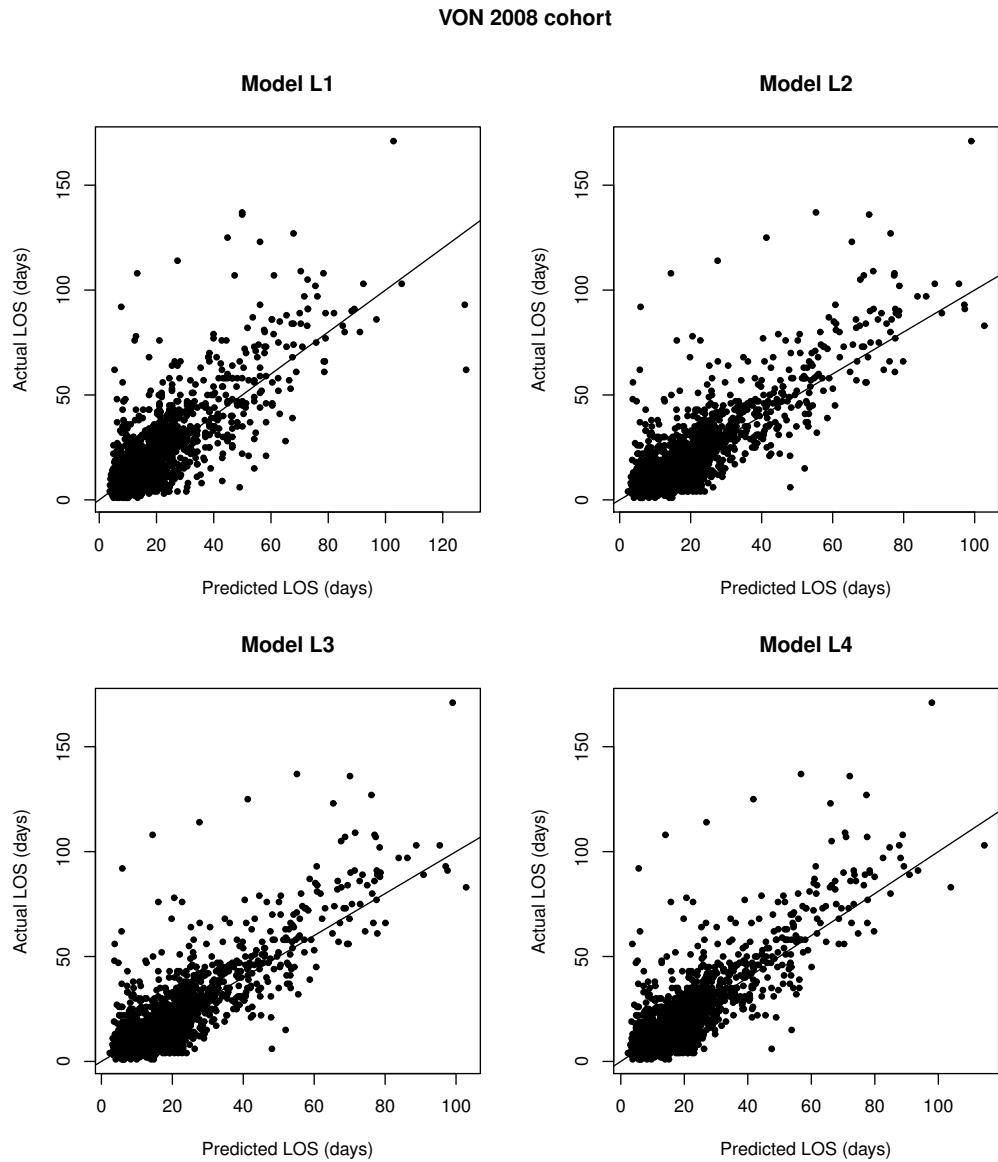


Figure 8.4: Actual vs. predicted values for the four Poisson loglinear regression models to predict LOS.

8.5. DIFFERENCES BETWEEN PC AND CPC REGRESSION 267

from orthogonal projections of the original input variables, and the principal components and common principal components are orthogonal among themselves and together account for all of the variation observed in the original numeric variables. It can be shown that the full set of components obtained from any other orthogonal projection matrix of the required dimensions will also lead to the same model fit. Though the estimated regression coefficients can differ greatly, the model fit and predictive value will remain unchanged.

However, when the PC (or CPC) analysis is performed as a dimension reduction step and a number of the components are discarded before fitting the regression model, the model fits can differ. In this case the best fitting model will be the one for which the retained (common) principal components have the strongest correlation with the response variable. Ideally the set of orthonormal projection vectors should be calculated in a way that the first component have the maximum possible correlation with the response variable, which is one of the goals of PLS regression.

PC and CPC analysis are undirected methods in the sense that their purpose is to describe the covariance structure of numerical variables in a useful way, but the directions of the estimated eigenvectors (or common eigenvectors) are arbitrary with respect to the response variable in a regression context. What makes PC and CPC regression useful is that it offers a way to deal with multicollinearity, at the expense of obtaining a model of which the regression coefficients may be difficult to interpret.

Chapter 9

Summary

9.1 Results and new developments

A summary of the main results from this dissertation on the identification and application of common principal components is presented in this section. In addition to research on the identification of the most appropriate covariance matrix model from Flury's hierarchy for multivariate non-normally distributed populations, the CPC model was applied in new ways to a number of existing statistical techniques. The properties of these applications of the CPC model were studied in Monte Carlo simulation experiments, and the use thereof demonstrated on the VON data set.

Customised software developed for this dissertation was compiled in the `cpc` R package, which is given in Appendix B.

In **Chapter 2** an overview of PCA was given, introducing important concepts such as the eigenvectors and eigenvalues of covariance matrices. Asymptotic results for inference on the eigenvectors and eigenvalues under the assumption of multivariate normality in the population were given. Methods to decide on the number of principal components to retain, for the application of PCA as a dimension reduction technique, were discussed.

PCA was performed on the numerical variables from the VON 2009 cohort, but because the multivariate normality assumption is untenable for the VON data, bootstrap methods were used to construct confidence intervals for the eigenvector elements and the eigenvalues of the covariance matrix. It was seen that the first three principal components (from a total of six) in the VON data together account for more than 95% of the observed variation. The first three principal components were interpreted (by inspection of the eigenvector elements) as pertaining to the size of the infant, feasibility of life, and having a small/large head for the stage of development, respectively.

Variable selection (for regression modelling) by inspection of the eigenvector elements were demonstrated on the VON data. Lastly a number of outliers with regard to the overall covariance structure of the VON data were identified by inspection of the last few principal components.

The CPC and partial CPC models, as extensions of PCA to several groups, were discussed in **Chapter 3**. Asymptotic results for inference on the common eigenvector elements and the eigenvalues under the multivariate normality assumption were given. A geometrical interpretation of the CPC model was given, followed by short discussion of three algorithms (FG, stepwise CPC, and JADE) for the simultaneous diagonalisation of several covariance matrices. Flury's hierarchy for the covariance matrices of several groups was also discussed.

CPC analysis was performed on the delivery mode (*Caesarean* and *Vaginal*) and regional (*South Africa* and *Namibia*) groupings in the VON 2009 cohort, respectively. It was seen that the eigenvectors of the covariance matrices of the delivery mode groups appear very similar, but that they differ in importance in the two groups. Common eigenvectors for the delivery mode groups were estimated with the FG, stepwise CPC, and JADE algorithms, respectively, and it was seen that the solutions are very similar. Due to the lack of multivariate normality, bootstrap methods were used to construct confidence intervals for the common eigenvector elements and the eigenvalues under the CPC model.

The CPC analysis on the regional groups revealed that the eigenvector of the two groups appear very similar. Common eigenvectors were estimated with the FG algorithm, and bootstrap methods were used to construct confidence intervals for the common eigenvector elements and the eigenvalues of the two covariance matrices.

Methods to select a common eigenvector model from Flury's hierarchy was investigated in **Chapter 4**. At the time of writing, a shortened version of the work in this chapter has been accepted for publication in Pepler et al. (2014). Likelihood ratio tests for the first two levels in Flury's hierarchy (equality and proportionality of the covariance matrices) were given, followed by a discussion on the identification of common eigenvectors in two population covariance matrices. In addition to the chi-squared statistics and use of the AIC statistics as proposed by Flury (1988), a number of new methods based on bootstrap methods were proposed, namely the *BootTest*, *RVC*, *BVD*, *BCR*, and *Ensemble* methods. The *BVD* and *Ensemble* methods were seen to outperform the parametric methods, as well as the *BootTest* and *RVC* methods (which are modifications of proposals by Klingenberg (1996) and Klingenberg and McIntyre (1998)), in the majority of situations considered in the Monte Carlo simulation study. The new non-parametric methods also

do not depend on any assumptions about the population distributions, which is an advantage in practical applications.

The methods discussed in this chapter were applied to a number of well-known data sets from the literature (the Bank notes data, the Swiss heads data, and the Iris data) and to the delivery mode and regional groups in the VON 2009 cohort. While the AIC method indicates the unrelated covariance matrices model for the delivery mode groups, it was found that the covariance matrices of *Caesarean* and *Vaginal* have three common eigenvectors and that the CPC(3) model is the most appropriate. For the regional groups, the AIC method also indicates no common eigenvectors, but the new non-parametric methods show that the covariance matrices of *South Africa* and *Namibia* have six common eigenvectors and the full CPC model is thus appropriate.

Estimation of population covariance matrices under the CPC model was explored in **Chapter 5**. The CPC estimator proposed by (Flury, 1988) was used in defining a new CPC shrinkage covariance matrix estimator. Three methods for estimation of the shrinkage intensity parameter were put forward. A modification to the Frobenius matrix norm was proposed for the comparison of symmetric matrices, and this modified version was used to compare the various covariance matrix estimators under a number of simulated scenarios. Of the estimators considered (including the *Unbiased* and *Pooled* estimators), the CPC shrinkage estimator under the assumption of full CPC in the population covariance matrices, and with the shrinkage intensity parameter estimated by crossvalidation (*Full CPC crossvalid*), performed the best overall. Even when the population covariance matrices are unrelated, the proposed CPC shrinkage estimator still approximates the population covariance matrices better than the other estimators.

The *Full CPC crossvalid* estimator was used to obtain estimates of the population covariance matrices of the delivery mode and regional groups in the VON 2009 cohort. For the *Namibia* region, the *Full CPC crossvalid* estimate of the covariance matrix was seen to differ from the *Unbiased* estimate to such an extent that the interpretation of the covariances involving the temperature variable (*ATEMP*) changed substantially.

In **Chapter 6** the CPC and CPC shrinkage estimators (*Full CPC crossvalid*) were used to construct CPC discriminant functions. Following a discussion of the ellipses formed by a number of covariance matrix estimators, CPC discrimination was compared to ordinary quadratic discrimination (QDA) and linear discrimination (LDA) in a Monte Carlo simulation study. It was shown that CPC discrimination outperforms both QDA and LDA when two population covariance matrices are not equal or proportional, but have common eigenvectors. As expected, LDA performs the best when the population covariance matrices are equal, and QDA generally performs the

best when the covariance matrices are unrelated.

QDA, CPC, and LDA discrimination were applied to the delivery mode, regional, and mortality status groups in the VON 2009 cohort with mixed results. For the delivery mode groups, QDA gave the smallest misclassification error rate, followed by CPC and LDA. CPC discrimination performed the best for the regional groups. For the mortality status groups (*Survived* and *Died*), use of the CPC shrinkage estimators gave the smallest misclassification error rate, but the plain CPC covariance matrix estimators gave the largest misclassification error rate.

Construction of biplots under the CPC model is the topic that was explored in **Chapter 7**. Some background about biplots was given, noting that the PC biplot is constructed from the covariance matrix and not the data matrix itself. This approach is also followed for CPC biplots, which are constructed using the common eigenvectors of several covariance matrices. For data with distinct groups, two alternatives (a biplot of the pooled covariance matrix, and a biplot of the covariance matrix of the pooled data) to the CPC biplot were briefly discussed. A number of quality measures applicable to biplots for data with distinct groups were given. These include measures of the quality of representation of the observations, groups and variables, respectively, in the biplot display.

In a discussion on the comparison of the different biplot types, details about an R function (written for the purpose of this dissertation) to compare the biplots with regard to the quality measures are given. A small simulation study showed that a biplot constructed from the eigenvectors of the covariance matrix of the pooled data generally gives the best display for data with distinct groups, with the exception of the within-group quality of display. For display of the within-group variation, the biplot constructed from the eigenvectors of the pooled covariance matrix, or one of the CPC biplots usually fare better. Among the three CPC biplot types, it was found that the stepwise CPC biplot generally gives the highest quality display.

The different biplot types were compared on the Iris data and the Bank notes data, as well as the delivery mode, regional, and mortality groupings in the VON 2009 cohort. For the Iris data, construction of a CPC biplot from the common eigenvectors estimated with the JADE algorithm was demonstrated. A biplot using the common eigenvectors estimated with the FG algorithm was constructed for the Bank notes data. For the VON groupings, *Pooled data* biplots seem to provide the best quality displays.

In **Chapter 8**, a number of regression models were constructed to predict neonatal mortality and length of hospital stay for NICU admissions in the VON data. An earlier version of the work in this chapter was published in Pepler et al. (2012). After a brief discussion of four types of regression

models (multiple linear regression, PC regression, CPC regression, and PLS regression) for data with distinct groups, three measures of model fit (AIC, area under the ROC curve, and R^2) were given.

The different regression model types were applied to the VON data and the results were compared. It was found that readily available day-of-admission data provide a good source of information to predict neonatal death and length of stay (LOS) in the Southern African private hospital setting. For the regression models on the original numerical variables, variable selection was performed by inspecting the loadings of the eigenvectors of the covariance matrix of the VON 2009 data.

Of the available perinatal variables, Apgar score at one minute, gestational age and delivery mode have significant effects on the odds of neonatal mortality. A low Apgar score at one minute and low gestational age are associated with an increase in the probability of death. The PC and CPC regression models for mortality gave very similar results, with only the first two (common) principal components, together with the delivery mode variable, significantly associated with the response. Either of the PC or CPC regression models can be used as a simple screening test to identify high mortality risk infants: For a predicted death, the odds of dying are about 21 times greater than for a predicted survival.

Poisson multiple loglinear regression models were fitted to predict LOS for the infants in the VON data set. Infants in the *South Africa* group had significantly shorter LOS than infants in the *Namibia* group. There are also differences in the delivery mode groups regarding LOS. Increases in Apgar score at five minutes and gestational age are associated with a decrease in LOS. From the PC regression model for LOS it was seen that the principal component giving an indication of whether the infant had a small/large head for the stage of development had a significantly larger influence in *Namibia* than in *South Africa*. The same effect was also seen in the CPC and PLS regression models for LOS. The PLS component describing the change in the feasibility of life from the first to the fifth minute after birth also had a greater effect on LOS in *Namibia* than in *South Africa*.

While the PC, CPC and PLS regression models for LOS all fit the data reasonably well, the PLS regression model gives the best fit. The PLS components are calculated using both the input and response variables (i.e. it is a “directed” method), while PCA and CPC analysis are merely concerned with modelling the covariance structures of the input variables and therefore give suboptimal regression model fits.

9.2 Future research

A number of questions arising from the research presented in this dissertation may be worthy of further investigation. These are briefly listed in the sections below.

Identification of common eigenvectors

The first remarks concern the methods for identification of the most appropriate model from Flury's hierarchy for several covariance matrices. Though the *AIC* method is based on likelihood estimation (with the assumption of multivariate normality in the populations), it may be of interest to investigate whether an adjustment to the penalisation term in (4.5) can improve its performance against the non-parametric methods.

Regarding the *BVD* method, the vector correlation cut-off point of 0.71 was chosen based on the fact that it is the midpoint between collinearity and orthogonality for two normalised vectors. There is some indication that this cut-off point may be too great for very small samples (and perhaps too small for very large samples). Optimisation of the performance of the *BVD* method by adjusting the cut-off point based on the sizes of the sample from the different populations may be worth exploring further.

The CPC model identification methods were compared on two independent populations (with various covariance structure configurations) only. It is of interest to extend the proposed methodology to more than two groups, and to compare the different methods in this setting. Extension of the bootstrap-based methods (*BootTest*, *RVC*, *BVD* and *BCR*) to analyse data with dependent groups, such as measurements observed on the same individuals at different time points (i.e. longitudinal data), is another topic to consider.

Estimation of covariance matrices

The proposed estimators for population covariance matrices under the CPC model were studied only in the context of well-conditioned covariance matrices where the sample sizes are greater than the number of variables. It will be interesting to determine whether (and to what extent) the performance of the *CPC* and *Full CPC crossvalid* estimators deteriorate in cases where $p \geq n$ and/or $p \geq n_i$.

It was briefly mentioned in Section 5.6.5 that the nature of the correlations between the original variables (per group) seems to affect the covariance matrix estimators in the case when there are full CPC in the population covariance matrices. Further research is needed to illuminate the effects of

correlations among the variables on the proposed estimators, and it will be interesting to see whether these effects are also significant for population covariance matrices at each of the other levels of Flury's hierarchy.

CPC discriminant analysis

The rank orders of the common eigenvectors in the covariance matrices, and the locations of the different populations (i.e. the population centroids) influence the orientations and positions of the estimated covariance matrix shapes in p -dimensional space. Flury et al. (1994) and Bianco et al. (2008) hinted at the influence of these factors, but more work is needed to clarify the exact nature of their influence on the four discriminant functions studied in Chapter 6.

Schmid (1987) and Flury et al. (1994) have shown that discrimination under the assumption of proportional covariance matrices perform well even in situations where it is theoretically incorrect (as in the case when the covariance matrices are not proportional but have common eigenvectors). It will be interesting to compare the performance of regularised CPC discriminant analysis to discriminant analysis using the proportional covariance matrix estimators, particularly in the CPC and partial CPC contexts.

CVA biplots under the CPC model

The CVA biplot and the CPC biplot are both applicable to data with distinct groups. However, the purpose of CVA is to optimally separate the groups while CPC analysis is concerned with modelling of the covariance structures *within* the groups. It may be of interest to explore whether the CPC model can be employed profitably in the construction of CVA biplots, and to determine in which situations such a CPC-CVA biplot may be useful.

VON regression models

The regression models to predict mortality and LOS for infants in the VON data set are not intended to provide admission criteria for NICUs, as only infants that have been admitted to an NICU (and recorded in the VON database) were included in the analysis. The objective was to develop prediction models on data readily available on the day of admission. The predicted values can be used as benchmarks from which to evaluate and monitor the performance of individual NICUs in terms of neonatal mortality and LOS. Ideally this should form part of a greater NICU quality control programme, of which the results may be used by hospital managers for improved direction

of resources, hopefully resulting in an improvement in the quality of patient care.

The prediction models can probably be improved by using knowledge of later-occurring morbidities (Hintz et al., 2010). The effect of unobserved variables such as congenital malformations and possible early infections should also be discounted. When interpreting the model effects, it should be kept in mind that there may be confounding or proxy effects attributed to the day-of-admission variables (Powell et al., 1992). Infants with low gestational age may, for example, be more prone to hospital acquired infections (increasing their LOS) and because of the model specification this effect is attributed to the gestational age variable. Lastly, another factor to consider for future research, especially in the Southern African context, is the role of maternal HIV status on mortality and LOS predictions.

These challenges may be addressed in future research endeavours.

Appendix A

Simulating CPC and CPC(q) data

A.1 Simulating data for a specified eigenvector structure

In order to construct covariance matrices,

$$\Sigma_i = \mathbf{B}_i \mathbf{\Lambda}_i \mathbf{B}_i', \quad i = 1, \dots, k, \quad (\text{A.1})$$

for populations with q common eigenvectors, $0 \leq q \leq p$, appropriate \mathbf{B}_i and $\mathbf{\Lambda}_i$ matrices should be chosen. The `betasim()` function in the `cpc` package gives \mathbf{B}_i matrices for specified values of k , p and q . If $q = p$, all of the eigenvectors are common and the \mathbf{B}_i matrices will be equal.

$\mathbf{\Lambda}_i$ is a diagonal matrix with the eigenvalues of the i^{th} covariance matrix on the diagonal. The diagonal values of the $\mathbf{\Lambda}_i$ matrices can therefore be chosen to reflect specified rank orders of the common eigenvectors of the covariance matrices (*Same*, *Similar* or *Opposite* rank orders, for example). To simulate data from populations with equal covariance matrices, equal $\mathbf{\Lambda}_i$ and \mathbf{B}_i matrices should be used. For data from populations with proportional covariance matrices, proportional $\mathbf{\Lambda}_i$ matrices are used with a common \mathbf{B} matrix.

If required, different population mean vectors, $\boldsymbol{\mu}_i, i = 1, \dots, k$, can be specified. Data can be simulated from multivariate normal or multivariate non-normal distributions for the $\boldsymbol{\mu}_i$ vectors and Σ_i matrices.

A.2 A method for simulating multivariate non-normal data

For a given orthogonal matrix \mathbf{B} (or matrices \mathbf{B}_i , in the case of partial CPC), and specified $\mathbf{\Lambda}_i$ matrices, the population covariance matrices are as given in (A.1). Let X_{ij} , $j = 1, \dots, p$ be p independent random variables with prespecified non-normal distributions, and let

$$Z_{ij} = \frac{X_{ij}}{\sqrt{\text{Var}(X_{ij})}} \quad (\text{A.2})$$

be the p standardised non-normal variables. The p -variate stochastic vector $\mathbf{z}'_i = [Z_1 \ Z_2 \ \dots \ Z_p]$ is distributed

$$\mathbf{z}_i \sim (\boldsymbol{\mu}_i, \mathbf{I}_p). \quad (\text{A.3})$$

The covariance matrix of $\mathbf{W}_i = \boldsymbol{\Sigma}_i^{\frac{1}{2}} \mathbf{z}_i$ is

$$\begin{aligned} \text{Cov}(\mathbf{W}_i) &= \boldsymbol{\Sigma}_i^{\frac{1}{2}} \text{Cov}(\mathbf{z}_i) \boldsymbol{\Sigma}_i^{\frac{1}{2}} \\ &= \boldsymbol{\Sigma}_i^{\frac{1}{2}} \mathbf{I}_p \boldsymbol{\Sigma}_i^{\frac{1}{2}} \\ &= \boldsymbol{\Sigma}_i. \end{aligned} \quad (\text{A.4})$$

To simulate multivariate non-normal data for the i^{th} population with covariance matrix $\boldsymbol{\Sigma}_i$, n_i pseudo-random samples should be generated from p predetermined univariate non-normal distributions, i.e.

$$X_{ij} \sim \text{non-normal distribution}, \quad j = 1, \dots, p. \quad (\text{A.5})$$

The simulated \mathbf{x}_{ij} vectors will be independent of each other, and can be standardised to have unit variance, i.e.

$$\mathbf{z}_{ij} = \frac{\mathbf{x}_{ij}}{\sqrt{\text{Var}(X_{ij})}}, \quad j = 1, \dots, p. \quad (\text{A.6})$$

Putting matrix $\mathbf{Z}_i = [\mathbf{z}_{i1} \ \mathbf{z}_{i2} \ \dots \ \mathbf{z}_{ip}]$, and letting

$$\mathbf{W}_i = \mathbf{Z}_i \boldsymbol{\Sigma}_i^{\frac{1}{2}}, \quad (\text{A.7})$$

where

$$\boldsymbol{\Sigma}_i^{\frac{1}{2}} = \mathbf{E}_i \boldsymbol{\Delta}_i^{\frac{1}{2}} \mathbf{E}_i', \quad (\text{A.8})$$

and

A.2. A METHOD FOR SIMULATING MULTIVARIATE NON-NORMAL DATA 279

$$\Delta_i^{\frac{1}{2}} = \begin{bmatrix} \sqrt{\delta_{i1}} & 0 & \dots & 0 \\ 0 & \sqrt{\delta_{i2}} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \sqrt{\delta_{ip}} \end{bmatrix} \quad (\text{A.9})$$

is a diagonal matrix with the square roots of the eigenvalues of the covariance matrix of the i^{th} group on the diagonal, the covariance matrix of \mathbf{W}_i should approximate Σ_i adequately for sufficiently large n_i .

Appendix B

R code

B.1 The cpc package (version 0.1-4)

B.1.1 Auxiliary functions

```
box.mtest <- function(covmats, nvec){
  # Chi-square approximation to Box's M test for equality of k
  # covariance matrices, as described in Rencher (2002, p255).
  # Returns the p value for the test.

  # covmats: array of k covariance matrices to be tested for
  # equality
  # nvec: vector of sample sizes of the k groups

  k <- dim(covmats)[3]
  p <- dim(covmats)[2]
  ntot <- sum(nvec)

  Sp <- matrix(0, nrow = p, ncol = p)
  for(i in 1:k){
    Sp <- Sp + (nvec[i] - 1) * covmats[, , i]
  }
  Sp <- Sp / (ntot - k)

  temp <- 0
  for(i in 1:k){
    temp <- temp + 1 / (nvec[i] - 1)
  }

  c1 <- (temp - (1 / (ntot - k))) * (2 * (p^2) + 3 * p - 1) /
```

```

      (6 * (p + 1) * (k - 1))

temp2 <- 0
for(i in 1:k){
  temp2 <- temp2 + (nvec[i] - 1) * log(det(covmats[, , i]))
}

lnM <- 0.5 * (temp2) - 0.5 * (ntot - k) * log(det(Sp))

chi2 <- -2 * (1 - c1) * lnM

v <- 0.5 * (k - 1) * p * (p + 1)

pval <- pchisq(q = chi2, df = v, lower.tail = FALSE)

return(pval)
}

```

```

flury.phi <- function(datamat){
  # Calculates phi measure of goodness of diagonalisation as
  # proposed by Flury (1988)
  # Interpretation: phi smaller value --> better diagonalisation

  # datamat: square symmetric matrix

  return(det(diag(diag(datamat))) / det(datamat))
}

```

```

standcol <- function(x, centre = FALSE, stand = TRUE)
{
  # Standardize columns to have zero mean

  if(centre){
    for(i in 1:ncol(x)){
      x[, i] <- x[, i] - mean(x[, i])
    }
  }

  if (stand){
    for(i in 1:ncol(x)){

```

B.1. THE CPC PACKAGE

283

```
      x[, i] <- x[, i] / sd(x[, i])
    }
  }

  return(x)
}
```

```
tr <- function(X){
  # Calculates the trace of p x p matrix X

  return(sum(diag(X)))
}
```

```
vec <- function(datamat){
  # Stacks the columns of a data matrix in a vector

  n <- nrow(datamat)
  outputvec <- NULL
  for(i in 1:n){
    outputvec <- append(outputvec, datamat[, i])
  }
  return(outputvec)
}
```

```
vecs <- function(datamat){
  # Stacks the columns of the lower diagonal of a symmetric matrix
  # in a vector

  p <- ncol(datamat)
  outputvec <- NULL
  for(j in 1:p){
    outputvec <- append(outputvec, datamat[j:p, j])
  }
  return(outputvec)
}
```

B.1.2 Simulation functions

```

betasim <- function(k, p, q){
  # Finds BETA matrices for the required number of groups (k),
  # variables (p) and common principal components (q) for data
  # simulation

  # k: number of groups
  # p: number of variables
  # q: number of common principal components, as in CPC(q)

  dotprod.cutoff <- cos(((90 - 2 * k) * pi) / (180 * k))
  # maximum value for dot product of eigenvectors that are NOT
  # common (0.766 corresponds to 40 degree angle)
  dotprod <- 1
  BETA.matrices <- array(NA, dim = c(p, p, k))
  library(gtools)

  while (dotprod > dotprod.cutoff){
    mat <- matrix(c(runif(n = p * q, min = 1, max = 100)),
      nrow = p, ncol = q)
    for(i in 1:k){
      tempmat <- cbind(mat, matrix(c(runif(n = p * (p - q),
        min = 1, max = 100)), nrow = p, ncol = (p - q)))
      BETA.matrices[, , i] <- qr.Q(qr(tempmat))
    }
    if (q >= (p - 1)){
      break
    }
    BETA.test <- BETA.matrices[, (q + 1):p, ]
    permsmat <- permutations(n = p - q, r = 2,
      repeats.allowed = TRUE)
    numperms <- nrow(permsmat)
    groupcombsmat <- combinations(n = k, r = 2,
      repeats.allowed = FALSE)
    numgroupcombs <- nrow(groupcombsmat)

    dotvec <- rep(NA, times = numperms * numgroupcombs)
    for(g in 1:numgroupcombs){
      testmats <- BETA.test[, , groupcombsmat[g, ]]
      for(i in 1:numperms){
        dotvec[(g - 1) * numperms + i]
          <- abs(testmats[, permsmat[i, 1], 1] %*%)
      }
    }
  }
}

```



```

        testmats[, permsmat[i, 2], 2])
    }
}
dotprod <- max(dotvec)
}
return(BETA.matrices)
}

```

```

nonnormaldata.sim <- function(Sigma, n = 100,
  df = rep(2, times = ncol(Sigma))){
  # Generates multivariate nonnormal data with the specified
  # covariance structures (in Sigma)

  # Sigma: covariance structure for which to simulate the data
  # n: sample size
  # df: vector of chi-squared degrees of freedom - to control
  # skewness of the variables (skew = sqrt(8/df))

  p <- ncol(Sigma)[1]

  xdata <- matrix(NA, nrow = n, ncol = p)
  zdata <- matrix(NA, nrow = n, ncol = p)
  for(j in 1:p){
    xdata[, j] <- rchisq(n = n, df = df[j])
    zdata[, j] <- xdata[, j] / sqrt(2 * df[j])
  }

  wdata <- matrix(NA, nrow = n, ncol = p)
  Sigma.eigen <- eigen(Sigma)
  sqrt.Sigma <- Sigma.eigen$vectors %*%
    diag(Sigma.eigen$values^0.5) %*% solve(Sigma.eigen$vectors)
  wdata <- zdata %*% sqrt.Sigma

  return(wdata)
}

```

B.1.3 Simultaneous diagonalisation algorithms

```

FG <- function(covmats, nvec){
  # Implementation of the FG algorithm as described in

```

```

# Flury (1988) p178

# covmats: array of covariance matrices to be simultaneously
# diagonalized, created by a command such as
# covmats <- array(NA, dim = c(p, p, k))
# nvec: vector of sample sizes for the covariance matrices in
# covmats

p <- dim(covmats)[2]
B <- diag(p)
DIFF <- 100
k <- dim(covmats)[3]
while (DIFF > 1e-09){
  B.old <- B
  T.mat <- array(NA, dim = c(2, 2, k))
  for(m in 1:(p-1)){
    for(j in (m + 1):p){
      vek <- c(m, j)
      for(i in 1:k){
        T.mat[, , i] <- t(B[, vek]) %*% covmats[, , i] %*%
          B[,vek]
      }
      J <- G.algorithm(T.mat, nvec)
      B[, vek] <- B[, vek] %*% J
    }
  }
  for(i in 1:p){
    for(j in 1:p){
      DIFF <- abs(B[i, j] - B.old[i, j])
    }
  }
}

# Order the columns of B

diagvals <- 0
for(i in 1:k){
  diagvals <- diagvals + diag(t(B) %*% covmats[, , i] %*% B)
}

B <- B[, order(diagvals, decreasing = TRUE)]

diagvalsmat <- matrix(NA, nrow = p, ncol = k)

```

B.1. THE CPC PACKAGE

287

```

for(i in 1:k){
  diagvalsmat[, i] <- diag(t(B) %*% covmats[, , i] %*% B)
}

return(list(B = B, diagvals = diagvalsmat))
}

```

```

G.algorithm <- function(T.mat, nvec){
  # Implementation of the G algorithm as described in
  # Flury (1988) p181

  k <- dim(T.mat)[3] # number of groups
  Q <- matrix(rep(200, 4), nrow = 2, ncol = 2)
  Q.nuut <- diag(2)
  delta <- matrix(NA, nrow = k, ncol = 2)

  while (abs(Q[2, 1] - Q.nuut[2, 1]) > 1e-07){
    Q <- Q.nuut
    for(i in 1:k){
      for(j in 1:2){
        delta[i, j] <- t(Q[, j]) %*% T.mat[, , i] %*% Q[, j]
      }
    }
    U <- matrix(rep(0, 4), nrow = 2, ncol = 2)
    for(i in 1:k){
      U <- U + nvec[i] * ((delta[i, 1] - delta[i, 2]) /
        (delta[i, 1] * delta[i, 2])) * T.mat[, , i]
    }
    Q.nuut <- eigvec(U)
  }
  return(Q.nuut)
}

```

```

eigvec <- function(U){
  if (U[1, 2] != 0){
    ratio <- (U[2, 2] - U[1, 1]) / U[1, 2]
  } else {
    ratio = 0
  }
  discr <- sqrt(ratio^2 + 4)

```

```

T1 <- (ratio + discr) / 2
T2 <- (ratio - discr) / 2
if (abs(T1) > abs(T2)){
  T.waarde <- T2
} else {
  T.waarde <- T1
}
C.waarde <- 1 / sqrt(1 + T.waarde^2)
S <- T.waarde * C.waarde
J <- matrix(NA, nrow = 2, ncol = 2)
J[1, 1] <- C.waarde
J[2, 1] <- S
J[1, 2] <- -S
J[2, 2] <- C.waarde
return(J)
}

```

```

stepwisecpc <- function(covmats, nvec) {
  # Stepwise CPC as described in the paper by N. Trendafilov (2010)

  # covmats: array containing the covariance matrices for the
  # groups, created with a command such as
  # covmats <- array(NA, dim = c(p, p, k)), where p refers to the
  # number of rows/columns of each covariance
  # matrix, and k is the number of groups (or covariance matrices)
  # nvec: vector containing the sample sizes of the k groups

  p <- dim(covmats)[1]
  k <- length(nvec)
  ntot <- sum(nvec)

  # Calculate pooled covariance matrix

  Spooled <- matrix(0, nrow = p, ncol = p)
  for(j in 1:k){
    Spooled <- Spooled + (nvec[j] - 1) * covmats[, , j]
  }
  Spooled <- Spooled / (ntot - k)

  # Initial values for stepwise CPC algorithm

  Qmat <- matrix(0, nrow = p, ncol = p)

```

```

Qtilde <- eigen(Spooled, symmetric = TRUE)$vectors
pimat <- diag(p)
muvec <- rep(0, k)
elmax <- 10 # maximum number of iterations

# Calculate stepwise CPCs

for(j in 1:p){
  xvec <- as.vector(Qtilde[, j])
  xvec <- pimat %*% xvec

  for(i in 1:k){
    muvec[i] <- t(xvec) %*% covmats[, , i] %*% xvec
  }
  for(el in 1:elmax){
    Smat <- matrix(0, nrow = p, ncol = p)
    for(i in 1:k){
      Smat <- Smat + (nvec[i] - 1) * covmats[, , i] / muvec[i]
    }
    yvec <- pimat %*% Smat %*% xvec
    xvec <- yvec / as.numeric(sqrt(t(yvec) %*% yvec))
    for(i in 1:k){
      muvec[i] <- t(xvec) %*% covmats[, , i] %*% xvec
    }
  }
  Qmat[, j] <- xvec
  pimat <- pimat - xvec %*% t(xvec)
}

eigenvals <- matrix(0, p, k)
for(i in 1:k){
  eigenvals[, i] <- t((diag(t(Qmat) %*% covmats[, , i] %*%
    Qmat)))
}

results <- list(B = Qmat, eigenvals = eigenvals)
return(results)
}

```

```

B.partial <- function(covmats, nvec,
  B = FG(covmats = covmats, nvec = nvec)$B, commonvec.order, q){
  # Estimates matrices of common (and non-common) eigenvectors for

```

```

# k groups

# covmats: array of sample covariance matrices for the k groups
# nvec: vector of sample sizes of the k groups
# B: matrix of common eigenvectors (estimated under the
# assumption of full CPC)
# commonvec.order: order of the common eigenvectors in B (with
# the q truly common eigenvectors in the first q positions)
# q: number of eigenvectors common to all k groups

k <- dim(covmats)[3]
p <- dim(covmats)[1]
B <- B[, commonvec.order]
Bmats <- array(NA, dim = c(p, p, k))
for(i in 1:k){
  B1 <- B[, 1:q, drop = FALSE]
  B2 <- B[, (q+1):p]
  Q1 <- eigen(t(B2) %*% covmats[, , i] %*% B2)$vectors
  B21 <- B2 %*% Q1
  Bmats[, , i] <- cbind(B1, B21)
}
return(Bmats)
}

```

B.1.4 Identify common eigenvector functions

```

flury.test <- function(covmats, nvec, B = FG(covmats, nvec)$B,
  p = dim(covmats)[1], qmax = p - 2,
  commonvec.order = findcpc(covmats = covmats, B = B,
    plotting = FALSE)$commonvec.order){
  # Calculates the partial chi-squared statistics and AIC values
  # for all the models in Flury's (1988) hierarchy for covariance
  # matrices

  # covmats: array of covariance matrices to be tested, created by
  # a command such as covmats <- array(NA, dim = c(p, p, k))
  # nvec: vector of sample sizes of the k groups
  # B: modal matrix (orthogonal p x p matrix diagonalising the k
  # covariance matrices simultaneously)
  # commonvec.order: vector indicating the order of the most
  # likely candidates of common eigenvectors - from 1 (most

```

```

# likely) to p (least likely)
# p: number of variables
# qmax: maximum for q when estimating the CPC(q) models

if ((qmax + 2) > p){
  qmax <- p - 2
  model.names <- c("Equality", "Proportionality", "CPC",
    paste("CPC(", seq(from = qmax, to = 1), ")", sep = ""),
    "Heterogeneity")
  No.of.CPCs <- c(p, p, p, (p - 2):1, 0)
} else {
  if (qmax < 1){
    qmax <- 0
    model.names <- c("Equality", "Proportionality", "CPC",
      "Heterogeneity")
    No.of.CPCs <- c(p, p, p, 0)
  } else {
    model.names <- c("Equality", "Proportionality", "CPC",
      paste("CPC(", seq(from = qmax, to = 1), ")", sep = ""),
      "Heterogeneity")
    No.of.CPCs <- c(p, p, p, (p - 2):1, 0)
  }
}

nmodels <- length(model.names)
chi.square <- rep(NA, times = nmodels)
df <- rep(NA, times = nmodels)
model.AIC <- rep(NA, times = nmodels)

# Equality

equal.test.output <- equal.test(covmats, nvec)
chi.square[1] <- equal.test.output$chi.square
df[1] <- equal.test.output$df
model.AIC[1] <- flury.AIC(equal.test.output$covmats.equal,
  covmats, nvec, df = equal.test.output$df)

# Proportionality

prop.test.output <- prop.test(covmats, nvec)
chi.square[2] <- prop.test.output$chi.square
df[2] <- prop.test.output$df
model.AIC[2] <- flury.AIC(prop.test.output$covmats.prop,
  covmats, nvec, df = prop.test.output$df)

```

```

# CPC

cpc.test.output <- cpc.test(covmats = covmats, nvec = nvec,
  B = B)
chi.square[3] <- cpc.test.output$chi.square
df[3] <- cpc.test.output$df
model.AIC[3] <- flury.AIC(cpc.test.output$covmats.cpc, covmats,
  nvec, df = cpc.test.output$df)

# CPC(q)

if (qmax > 0){
  B <- B[, commonvec.order]
  q <- qmax
  for(i in 1:qmax){
    cpcq.test.output<-cpcq.test(covmats,nvec,B,q=q)
    chi.square[3+i]<-cpcq.test.output$chi.square
    df[3 + i] <- cpcq.test.output$df
    model.AIC[3 + i] <- flury.AIC(cpcq.test.output$covmats.cpcq,
      covmats, nvec, df = cpcq.test.output$df)
    q <- q - 1
  }
}

# Heterogeneity

model.AIC[3 + qmax + 1] <- flury.AIC(covmats, covmats, nvec,
  df = 0)

chi.square[1:(nmodels - 2)] <- chi.square[1:(nmodels - 2)] -
  chi.square[2:(nmodels - 1)]
df[1:(nmodels - 2)] <- df[1:(nmodels - 2)] - df[2:(nmodels - 1)]
chi.div.df <- chi.square / df
chi.square <- round(chi.square, 2)
chi.div.df <- round(chi.div.df, 2)
model.AIC <- round(model.AIC, 2)

resultmat <- data.frame(Model = model.names,
  Chi.square = chi.square, DF = df, Chi2.div.df = chi.div.df,
  AIC = model.AIC, No.of.CPCs = No.of.CPCs)
return(resultmat)
}

```

```

equal.test <- function(covmats, nvec){
  # covmats: array of covariance matrices to be tested for
  # homogeneity vs. heterogeneity
  # nvec: vector of sample sizes of the k groups

  k <- dim(covmats)[3]
  p <- dim(covmats)[1]

  covmats.pooltotal <- 0

  for(i in 1:k){
    covmats.pooltotal <- covmats.pooltotal + covmats[, , i] *
      (nvec[i] - 1)
  }

  covmats.pool <- covmats.pooltotal / (sum(nvec) - k)

  chi2total <- 0
  covmats.equal <- array(NA, dim = c(p, p, k))

  for(i in 1:k){
    covmats.equal[, , i] <- covmats.pool
    chi2total <- chi2total + (nvec[i] - 1) *
      log(det(covmats.equal[, , i]) / det(covmats[, , i]))
  }

  df <- k * (0.5 * p * (p - 1) + p) - (0.5 * p * (p - 1) + p)

  return(list(chi.square = chi2total, df = df,
    covmats.equal = covmats.equal))
}

```

```

prop.test <- function(covmats, nvec){
  # covmats: array of covariance matrices to be tested for
  # proportionality vs. heterogeneity
  # nvec: vector of sample sizes of the k groups

  k <- dim(covmats)[3]
  p <- dim(covmats)[1]

```

```

rvec <- (nvec - 1) / (sum(nvec) - k)

# Step PCM0

rho <- rep(1, times = k)
maxrho <- 1
prevmaxrho <- 100

while (abs(maxrho - prevmaxrho) > 1e-09){

  # Step PCM1

  covmats.total <- 0
  for(i in 1:k){
    covmats.total <- covmats.total + rvec[i] * covmats[, , i] /
      rho[i]
  }
  B <- eigen(covmats.total)$vectors
  avec <- matrix(NA, ncol = p, nrow = k)
  for(i in 1:k){
    avec[i, ] <- diag(t(B) %*% covmats[, , i] %*% B)
  }

  # Step PCM2

  lambda <- rep(NA, times = p)
  for(j in 1:p){
    lambda[j] <- sum(rvec * avec[, j] / rho)
  }

  # Step PCM3

  for(i in 2:k){
    rho[i] <- 1 / p * sum(avec[i, ] / lambda)
  }

  # Step PCM4

  prevmaxrho <- maxrho
  maxrho <- max(rho[2:k])
}

covmats.prop <- array(NA, dim = c(p, p, k))

```

```

for(i in 1:k){
  covmats.prop[, , i] <- B %*% (diag(lambda) * rho[i]) %*% t(B)
}

chi2total <- 0

for(i in 1:k){
  chi2total <- chi2total + (nvec[i] - 1) *
    log(det(covmats.prop[, , i]) / det(covmats[, , i]))
}

df <- k * (0.5 * p * (p - 1) + p) - (0.5 * p * (p - 1) + p +
  k - 1)

return(list(chi.square = chi2total, df = df,
  covmats.prop = covmats.prop))
}

```

```

cpc.test <- function(covmats, nvec, B){
  # covmats: array of covariance matrices to be tested for CPC vs.
  # heterogeneity
  # nvec: vector of sample sizes of the k groups
  # B: modal matrix (orthogonal p x p matrix diagonalising the k
  # covariance matrices simultaneously)

  k <- dim(covmats)[3]
  p <- dim(covmats)[1]

  covmats.cpc <- array(NA, dim = c(p, p, k))
  chi2total <- 0
  for(i in 1:k){
    lambda <- diag(t(B) %*% covmats[, , i] %*% B)
    covmats.cpc[, , i] <- B %*% diag(lambda) %*% t(B)
    chi2total <- chi2total + (nvec[i] - 1) *
      log(det(covmats.cpc[, , i]) / det(covmats[, , i]))
  }

  df <- k * (0.5 * p * (p - 1) + p) - (0.5 * p * (p - 1) + k * p)

  return(list(chi.square = chi2total, df = df,
    covmats.cpc = covmats.cpc))
}

```

```

cpcq.test <- function(covmats, nvec, B, q){
  # covmats: array of covariance matrices to be tested for CPC vs.
  # heterogeneity
  # nvec: vector of sample sizes of the k groups
  # B: modal matrix with the q common eigenvectors in the first q
  # columns
  # q: number of common eigenvectors (in B)

  k <- dim(covmats)[3]
  p <- dim(covmats)[1]

  B1 <- B[, c(1:q)]
  B2 <- B[, c((q + 1):p)]

  covmats.cpcq <- array(NA, dim = c(p, p, k))
  for(i in 1:k){
    Q <- eigen(t(B2) %*% covmats[, , i] %*% B2)$vectors
    Bi <- cbind(B1, B2 %*% Q)
    Li <- diag(diag(t(Bi) %*% covmats[, , i] %*% Bi))
    covmats.cpcq[, , i] <- Bi %*% Li %*% t(Bi)
  }

  chi2total <- 0

  for(i in 1:k){
    chi2total <- chi2total + (nvec[i] - 1) *
      log(det(covmats.cpcq[, , i]) / det(covmats[, , i]))
  }

  df <- k * (0.5 * p * (p - 1) + p) - (0.5 * p * (p - 1) + k *
    p + 0.5 * (k - 1) * (p - q) * (p - q - 1))

  return(list(chi.square = chi2total, df = df,
    covmats.cpcq = covmats.cpcq))
}

```

```

flury.AIC <- function(covmats.high, covmats.low, nvec, df){
  # covmats.high: estimates of the covariance matrices under the
  # "higher" model
  # covmats.low: estimates of the covariance matrices under the

```

```

# "lower" model (usually unrelated/individual covariance
# matrices)
# nvec: vector of sample sizes
# df: degrees of freedom of the higher model, versus unrelated
# covariance matrices

k <- length(nvec)
p <- dim(covmats.high)[1]
aic.total <- 0
for(i in 1:k){
  aic.total <- aic.total + (nvec[i] - 1) *
    (sum(diag(solve(covmats.high[, , i]) %*%
      covmats.low[, , i])) + log(det(covmats.high[, , i])) -
    sum(diag(solve(covmats.low[, , i]) %*% covmats.low[, , i]))
    - log(det(covmats.low[, , i])))
}

npar <- k * (0.5 * p * (p - 1) + p) - (0.5 * p * (p - 1) + p) -
  df

aic.criterion <- aic.total + 2 * npar
return(aic.criterion)
}

```

```

BVD <- function(origdata, reps = 1000)
{
  # Identifies the number of common eigenvectors using the
  # bootstrap vector correlation distributions (BVD method)

  # origdata: list of the original data sets
  # reps: number of bootstrap replications to use

  k <- 2 # can handle only two groups at this stage
  p <- ncol(origdata[[1]])
  nvec <- rep(NA, times = k)
  covmats <- array(NA, dim = c(p, p, k))
  for(i in 1:k){
    nvec[i] <- nrow(origdata[[i]])
    covmats[, , i] <- cov(origdata[[i]])
  }

  B <- FG(covmats, nvec)$B

```

```

findcpc.out <- findcpc(covmats, B = B, plotting = FALSE)
commonvecnums <- findcpc.out$all.correlations[1:p, ]
for(i in 2:(k + 1)){
  j <- 2
  while (j <= p){
    if (length(unique(commonvecnums[1:j, i])) ==
        length(unique(commonvecnums[1:(j - 1), i]))) {
      commonvecnums <- commonvecnums[-j, ]
      p <- p - 1
    }
    j <- j + 1
  }
}

commonvec.order <- commonvecnums[, "B"]

bootreps <- bootveccor(origdata = origdata,
  veccormat = commonvecnums[, 1:k], nvec = nvec, reps = reps)

commonvecs <- rep(0, times = p)

for(j in 1:p){
  # BVD: median > 0.71 AND median +- distance between median and
  # 2.5th percentile >= 1
  llim <- quantile(bootreps[, j], probs = 0.025)
  temp.med <- median(bootreps[, j])
  temp.dist <- temp.med - llim
  temp.upper <- temp.med + temp.dist
  if ((temp.med > 0.71) & (temp.upper >= 1)){
    commonvecs[j] <- 1
  }
}

commonvecs <- rbind("Common eigenvector" = commonvec.order,
  commonvecs)
return(commonvecs)
}

```

```

findcpc <- function(covmats, B = NULL, cutoff = 0.95,
  plotting = TRUE, main = "Vector correlations for the
  permutations"){
  # Identifies possibly common eigenvectors in k data groups by

```

```

# investigating the vectors correlations of all pairwise
# combinations of eigenvectors

# covmats: array of covariance matrices from the groups to
# compare (made by command such as
# covmats <- array(NA, dim = c(row, col, nummatrices)))
# B: modal matrix (orthogonal p x p matrix diagonalising the k
# covariance matrices simultaneously)
# plotting: should a plot of the dot products be given?
# cutoff: cutoff point for indicating significance of the
# geometric mean of the dot products for all pairwise
# comparisons of a combination of eigenvectors

library(gtools)

if (is.null(B)){
  k <- dim(covmats)[3]
} else {
  k <- dim(covmats)[3] + 1
}
p <- dim(covmats)[1]

PCA.array <- array(NA, dim = c(p, p, k))

if (is.null(B)){
  for(i in 1:k){
    PCA.array[, , i] <- eigen(covmats[, , i])$vectors
  }
} else {
  PCA.array[, , 1] <- B
  for(i in 2:k){
    PCA.array[, , i] <- eigen(covmats[, , (i - 1)])$vectors
  }
}

# Calculating the sum of the vector correlations of all pairwise
# vector comparisons per vector combination

permsmat <- permutations(p, k, repeats.allowed = TRUE)
# Sets up matrix with all possible combinations of the columns
# of the k sets of eigenvectors
numperms <- nrow(permsmat)
# Total number of combinations to test for commonness of the

```

```

# vectors
numcomparisons <- ncol(permsmat) - 1
# Total number of pairwise comparisons to be made per vector
# combination
dotvec <- rep(0, times = numperms)

for(i in 1:numperms){
  temp<-rep(NA,times=numcomparisons)
  for(j in 1:numcomparisons){
    testvecs <- cbind(PCA.array[, (permsmat[i, j]), j],
                      PCA.array[, (permsmat[i, j + 1]), j + 1])
    temp[j] <- abs(t(testvecs[, 1]) %*% testvecs[, 2])
  }
  dotvec[i] <- exp(mean(log(temp)))
  # Calculate the geometric mean of the vector correlations of
  # all pairwise comparisons for the ith combination of
  # eigenvectors
}

resultmat <- cbind(permsmat, dot.products = dotvec)
resultmat <- resultmat[order(resultmat[, "dot.products"],
                             decreasing = TRUE), ]

dot.sig <- resultmat[resultmat[, "dot.products"] >= cutoff, ,
                     drop = FALSE]

plotnum <- max(c((p + 2), 20))
plotnum <- min(c(plotnum, numperms))
resultmat <- resultmat[c(1:plotnum), ]
if (plotting){
  resultmat <- resultmat[order(resultmat[, "dot.products"],
                              decreasing = FALSE), ]

  par(pch = 20, xaxt = "n")
  plot(x = c(1:plotnum), y = resultmat[, "dot.products"],
       type = "b", main = main,
       ylab = "Absolute vector correlation", xlab = "",
       ylim = c(0, 1))
  abline(h = 1, lty = 3)
  par(adj = 0, srt = 90)
  for(i in 1:plotnum){
    if (resultmat[i, "dot.products"] >= cutoff){
      combtext <- toString(resultmat[i, 1:k])
    }
  }
}

```



```

        points(x = i, y = resultmat[i, "dot.products"],
              col = "red")
        text(x = i, y = resultmat[i, "dot.products"],
             labels = combtext, pos = 1, cex = 0.7, offset = 1)
      }
    }
    resultmat <- resultmat[order(resultmat[, "dot.products"],
                                decreasing = TRUE), ]
  }

  if (is.null(B)){
    colnames(resultmat) <- c(paste("Group", 1:k, sep = ""),
                           "correlations")
  } else {
    colnames(resultmat) <- c("B", paste("Group", 1:(k - 1),
                                         sep = ""), "correlations")
  }

  if (!is.null(B)){
    commonvec.order <- resultmat[1:p, 1]
    commonvec.order <- append(unique(commonvec.order),
                             sort(c(1:p)[-unique(commonvec.order)]))
  } else {
    commonvec.order <- NULL
  }

  par(adj = 0.5)

  return(list(all.correlations = resultmat,
             commonvec.order = commonvec.order))
}

bootveccor <- function(origdata, veccormat, nvec, reps = 1000)
{
  # Function to calculate the bootstrap vector correlations

  # origdata: list of the grouped sample data
  # veccormat: matrix of the p eigenvector combinations with the
  # largest dot products
  # nvec: vector of group sizes
  # reps: number of bootstrap replications

```

```

numcomb <- nrow(veccormat)
p <- ncol(origdata[[1]]) # number of variables
k <- length(nvec) # number of groups
bootreps <- matrix(NA, ncol = numcomb, nrow = reps)
for(r in 1:reps){
  group.PCA <- array(NA, dim = c(p, p, k))
  for(i in 1:k){
    bootdata <- origdata[[i]][sample(c(1:nvec[i]),
      size = nvec[i], replace = TRUE),]
    group.PCA[, , i] <- eigen(cov(bootdata))$vectors
  }
  for(c in 1:numcomb){
    bootreps[r, c] <- abs(t(group.PCA[, veccormat[c, 1], 1]) %*%
      group.PCA[, veccormat[c, 2], 2])
  }
}
return(bootreps)
}

```

```

RVC <- function(covmats, reps = 100000)
{
  # Randomisation test for common eigenvectors as proposed by
  # Klingenberg (1996) -- Random vector correlations (RVC) method
  # H_0: eigenvector pair are not common
  # H_1: eigenvector pair are common

  # covmats: array of covariance matrices of the k groups
  # reps: number of randomisations to use

  k <- 2 # works only for k = 2 groups at this stage
  p <- dim(covmats)[1]

  E <- array(NA, dim = c(p, p, k))
  for(i in 1:k){
    E[, , i] <- eigen(covmats[, , i])$vectors
  }

  rand.dotproducts <- rep(NA, times = reps)
  for(r in 1:reps){
    randvec1 <- runif(n = p, min = -1, max = 1)
    randvec1 <- randvec1 / sqrt(randvec1 %*% randvec1)
    randvec2 <- runif(n = p, min = -1, max = 1)
  }
}

```

```

    randvec2 <- randvec2 / sqrt(randvec2 %*% randvec2)
    rand.dotproducts[r] <- abs(randvec1 %*% randvec2)
  }

  commonvecnums <- findcpc(covmats,
    plotting = FALSE)$all.correlations[1:p, ]
  for(i in 1:k){
    j <- 2
    while (j <= p){
      if (length(unique(commonvecnums[1:j, i])) ==
        length(unique(commonvecnums[1:(j - 1), i]))){
        commonvecnums <- commonvecnums[-j, ]
        p <- p - 1
      }
      j <- j + 1
    }
  }

  commonvec.order <- commonvecnums[, 1:2]

  orig.dotproducts <- abs(diag(t(E[, , 1][, commonvec.order[,1]])
    %*% (E[, , 2][, commonvec.order[, 2]])))

  pvals <- rep(NA, times = p)
  for(j in 1:p){
    pvals[j] <- length(rand.dotproducts[rand.dotproducts >=
      orig.dotproducts[j]]) / reps
  }

  return(data.frame(commonvec.order,
    vec.correlations = orig.dotproducts, p.values = pvals))
}

```

```

BootTest <- function(origdata, q = ncol(origdata[[1]]),
  reps = 1000){
  # Bootstrap test for common eigenvectors as proposed by
  # Klingenberg (1996) -- Bootstrap hypothesis test (BootTest)
  # method
  # H_0: eigenvector pair are common
  # H_1: eigenvector pair are not common

  # origdata: list of the original data for the k groups

```

```

# q: number of common components to test for
# reps: number of bootstrap replications to use

k <- 2 # works only for k = 2 groups at this stage
p <- ncol(origdata[[1]])

nvec <- rep(NA, times = k)
covmats <- array(NA, dim = c(p, p, k))
for(i in 1:k){
  nvec[i] <- nrow(origdata[[i]])
  covmats[, , i] <- cov(origdata[[i]])
}

B <- FG(covmats, nvec)$B
findcpc.out <- findcpc(covmats, B = B, plotting = FALSE)
commonvecnums <- findcpc.out$all.correlations[1:p, ]

temp.p <- p
for(i in 1:(k + 1)){
  j <- 2
  while (j <= temp.p){
    if (length(unique(commonvecnums[1:j, i])) ==
        length(unique(commonvecnums[1:(j - 1), i]))){
      commonvecnums <- commonvecnums[-j, ]
      temp.p <- temp.p - 1
      j <- j - 1
    }
    j <- j + 1
  }
}

commonvec.order <- commonvecnums[, "B"]
eigenvec.order <- commonvecnums[, c("Group1", "Group2")]
q <- min(q, nrow(eigenvec.order))
eigenvec.order.full <- matrix(NA, nrow = p, ncol = k)
if (q < p){
  for(i in 1:k){
    tempvec <- c(1:p)
    tempvec <- tempvec[-eigenvec.order[, i]]
    eigenvec.order.full[, i] <- c(eigenvec.order[, i], tempvec)
  }
} else {
  eigenvec.order.full <- eigenvec.order
}

```

```

    }

    E <- array(NA, dim = c(p, p, k))
    data.rotated <- origdata
    for(i in 1:k){
      E[, , i] <- eigen(covmats[, , i])$vectors[,
        eigenvec.order.full[, i]]
      data.rotated[[i]] <- as.matrix(origdata[[i]]) %*% E[, , i] %*%
        t(B)
    }
    orig.dotproducts <- abs(diag(t(E[, , 1]) %*% E[, , 2])[1:q])

    bootreps <- matrix(NA, ncol = q, nrow = reps)
    for(r in 1:reps){
      rep.eigenvecs <- array(NA, dim = c(p, q, k))
      for(i in 1:k){
        bootdata <- data.rotated[[i]][sample(c(1:nvec[i]),
          size = nvec[i], replace = TRUE),]
        rep.eigenvecs[, , i] <- eigen(cov(bootdata))$vectors[,
          eigenvec.order[1:q, i]]
      }
      for(j in 1:q){
        bootreps[r, j] <- abs(rep.eigenvecs[, j, 1] %*%
          rep.eigenvecs[, j, 2])
      }
    }

    pvals <- rep(NA, times = q)
    for(j in 1:q){
      pvals[j] <- nrow(bootreps[bootreps[, j] <=
        orig.dotproducts[j], , drop = FALSE]) / reps
    }

    return(data.frame(eigenvec.order[1:q, c("Group1", "Group2")],
      drop = FALSE, vec.correlations = orig.dotproducts[1:q],
      p.values = pvals[1:q]))
  }

```

```

BCR <- function(origdata, reps = 1000)
{
  # Bootstrap confidence regions (BCR) method
  # Calculates 95% bootstrap confidence regions for eigenvector

```

```

# pairs; if regions overlap, the eigenvectors are common

# Testing the hypothesis:
# H0: the pair of eigenvectors are common
# H1: the pair of eigenvectors are NOT common

# origdata: list of the grouped sample data
# reps: number of bootstrap replications to use

k <- 2 # for two groups only
p <- ncol(origdata[[1]]) # for two groups only

nvec <- rep(NA, times = k)
covmats <- array(NA, dim = c(p, p, k))
E <- array(NA, dim = c(p, p, k))
for(i in 1:k){
  nvec[i] <- nrow(origdata[[i]])
  covmats[, , i] <- cov(origdata[[i]])
  E[, , i] <- eigen(covmats[, , i])$vectors
}

B <- FG(covmats, nvec)$B
findcpc.out <- findcpc(covmats, B = B, plotting = FALSE)
commonvecnums <- findcpc.out$all.correlations[1:p, ]
p.new <- p
for(i in 2:(k + 1)){
  j <- 2
  while (j <= p.new){
    if (length(unique(commonvecnums[1:j, i])) ==
        length(unique(commonvecnums[1:(j - 1), i]))){
      commonvecnums <- commonvecnums[-j, ]
      p.new <- p.new - 1
    }
    j <- j + 1
  }
}

commonvec.order <- commonvecnums[, "B"]
eigenvec.order <- commonvecnums[, c("Group1", "Group2")]

common.ind <- rep(1, times = p.new)

for(j in 1:p.new){

```

```

bootreps1 <- matrix(NA, ncol = p, nrow = reps)
bootreps2 <- matrix(NA, ncol = p, nrow = reps)
dotprod1 <- rep(NA, times = reps)
dotprod2 <- rep(NA, times = reps)
for(r in 1:reps){
  bootreps1[r, ] <- eigen(cov(origdata[[1]][sample(
    c(1:nvec[1]), size = nvec[1], replace = TRUE), ]))$vectors
    [, eigenvec.order[j, 1]]
  bootreps2[r, ] <- eigen(cov(origdata[[2]][sample(
    c(1:nvec[2]), size = nvec[2], replace = TRUE), ]))$vectors
    [, eigenvec.order[j, 2]]
  dotprod1[r] <- abs((bootreps1[r, , drop = FALSE]) %*%
    (E[, , 1][, eigenvec.order[j, 1], drop = FALSE]))
  dotprod2[r] <- abs((bootreps2[r, , drop = FALSE]) %*%
    (E[, , 2][, eigenvec.order[j, 2], drop = FALSE]))
}
bootreps1.cutoff <- quantile(dotprod1, probs = 0.05)
bootreps1.trim <- bootreps1[dotprod1 > bootreps1.cutoff, ]
temp <- abs(bootreps1.trim %*% E[, , 2]
  [, eigenvec.order[j, 2]])
between.dotprod <- max(temp)
within.dotprod <- quantile(dotprod2, probs = 0.05)
if (within.dotprod > between.dotprod){
  common.ind[j] <- 0
}
}
return(data.frame("Common eigenvector" = commonvec.order,
  common.ind = common.ind))
}

```

```

ensemble.test <- function(origdata, standardize = FALSE)
{
  # Function to identify common eigenvectors using the ensemble
  # method

  # Ensemble method to identify common eigenvectors in k groups:
  # majority vote on number of common eigenvectors from the AIC,
  # Bootstrap Vector Correlation Distribution (BVD), Bootstrap
  # Confidence Regions (BCR), Random Vector Correlations (RVC) and
  # Bootstrap hypothesis test (BootTest) methods

  # origdata: list of the sample groups data

```

```

# standardize: should the data be standardized (mean=0, stdev=1)?

myFunc <- function(datavec){
  return((datavec - mean(datavec)) / sd(datavec))
}

standcol <- function(datamat){
  return(apply(datamat, 2, myFunc))
}

nvec <- c(nrow(origdata[[1]]), nrow(origdata[[2]]))
p <- ncol(origdata[[1]])
covmats <- array(NA, dim = c(p, p, 2))
if (standardize){
  origdata[[1]] <- standcol(origdata[[1]])
  origdata[[2]] <- standcol(origdata[[2]])
}
covmats[, , 1] <- cov(origdata[[1]])
covmats[, , 2] <- cov(origdata[[2]])

B <- FG(covmats, nvec)$B
commonvec.order <- findcpc(covmats, B = B,
  plotting = FALSE)$commonvec.order

flury.out <- flury.test(covmats, nvec, B = B,
  p = dim(covmats)[1], qmax = p - 2,
  commonvec.order = commonvec.order)
fluryAIC.vote <- flury.out[which.min(flury.out
  [, "AIC"]), "No.of.CPCs"]
fluryAIC.ind <- c(rep(1, times = fluryAIC.vote),
  rep(0, times = (p - fluryAIC.vote)))

BVD.ind <- BVD(origdata, reps = 1000)
if (length(BVD.ind) < p){
  BVD.ind <- c(BVD.ind, rep(0, times = (p - length(BVD.ind))))
}

BCR.ind <- BCR(origdata, reps = 1000)[, "common.ind"]
if (length(BCR.ind) < p){
  BCR.ind <- c(BCR.ind, rep(0, times = (p - length(BCR.ind))))
}

bonferroni.sig <- 1 - 0.95^(1 / p)

```



```

RVC.out <- RVC(covmats, reps = 100000)[, "p.values"]
RVC.ind <- rep(0, times = length(RVC.out))
RVC.ind[RVC.out <= bonferroni.sig] <- 1
if (length(RVC.ind) < p){
  RVC.ind <- c(RVC.ind, rep(0, times = (p - length(RVC.ind))))
}

BootTest.out <- BootTest(origdata)[, "p.values"]
BootTest.ind <- rep(0, times = length(BootTest.out))
BootTest.ind[BootTest.out > bonferroni.sig] <- 1
if (length(BootTest.ind) < p){
  BootTest.ind <- c(BootTest.ind, rep(0,
    times = (p - length(BootTest.ind))))
}

# Return majority vote on number of common eigenvectors (ties
# broken by choosing maximum mode)

resultmat <- rbind("Common eigenvector" = commonvec.order,
  AIC = fluryAIC.ind, BVD = BVD.ind, BCR = BCR.ind,
  RVC = RVC.ind, BootTest = BootTest.ind)
common.votes <- apply(resultmat[2:6, ], 2, sum)
commonvecs <- rep(0, times = p)
commonvecs[common.votes > 2] <- 1
resultmat <- rbind(resultmat, "Common vectors" = commonvecs)
commonvecnums <- commonvec.order[commonvecs > 0]
return(list(Results = resultmat, commonvecs = commonvecnums,
  commonvecmat = B[, commonvecnums, drop = FALSE]))
# Row 1: order of common eigenvectors in B;
# Row 2-5: results from AIC, BVD, BCR and RVC tests
# (1 = eigenvector common);
# Row 6: ensemble test common eigenvector indicator
# (1 = eigenvector common)
}

```

B.1.5 Covariance matrix estimation functions

```

alpha.crossvalid <- function(datamat, B, reps = 100){
  # Estimates alpha weighting parameter by cross-validation, for
  # improved estimation of population covariance matrix

```

```

# datamat: matrix containing sample data for the ith group
# B: matrix of estimated common (and possibly non-common)
# eigenvectors
# reps: number of replications to use in cross-validation

p <- ncol(datamat)
numobs <- nrow(datamat)
train.n <- round(numobs * 0.7, 0)
alpha.vals <- seq(from = 0, to = 1, by = 0.01)
alpha.n <- length(alpha.vals)
min.error.alpha <- rep(NA, times = reps)

for(i in 1:reps){
  sampledata <- datamat[sample(1:numobs, size = numobs,
    replace = FALSE), ]
  traindata <- sampledata[1:train.n, ]
  testdata <- sampledata[(train.n + 1):(numobs), ]
  traindata.covmat <- cov(traindata)
  testdata.covmat <- cov(testdata)
  L.diag <- diag(diag(t(B) %*% traindata.covmat %*% B))
  S.cpc <- B %*% L.diag %*% t(B)
  frobvals <- rep(NA, times = alpha.n)
  for(j in 1:alpha.n){
    S.new <- alpha.vals[j] * traindata.covmat + (1 -
      alpha.vals[j]) * S.cpc
    frobvals[j] <- frobenius(S.new, testdata.covmat)
  }
  min.error.alpha[i] <- alpha.vals[which.min(frobvals)]
}
return(mean(min.error.alpha))
}

```

```

alpha.schafer <- function(datamat, B, reps = 1000){
  # Estimates alpha weighting parameter by the method proposed in
  # Schafer & Strimmer (2005), for "Target D", for improved
  # estimation of population covariance matrix

  # datamat: matrix containing sample data for the ith group
  # B: matrix of estimated common (and possibly non-common)
  # eigenvectors
  # reps: number of bootstrap replications to use for estimation

```

B.1. THE CPC PACKAGE

311

```

# of the variances of the off-diagonal elements of the L_i

p <- ncol(datamat)
numobs <- nrow(datamat)
S <- cov(datamat)
L <- t(B) %*% S %*% B
L.offdiagvals <- NULL

for(i in 1:reps){
  bootdata <- datamat[sample(1:numobs, size = numobs,
    replace = TRUE), ]
  bootdata.cov <- cov(bootdata)
  L.boot <- t(B) %*% bootdata.cov %*% B
  L.offdiagvals <- rbind(L.offdiagvals, offdiag.vec(L.boot))
}
numer <- sum(apply(X = L.offdiagvals, MARGIN = 2, FUN = var))
denom <- sum((offdiag.vec(L))^2)
return(1 - min(numer / denom, 1))
}

```

```

offdiag.vec <- function(datamat){
  # Stacks the rows of a square matrix (excluding diagonal
  # elements) in a vector

  p <- ncol(datamat)
  datavec <- NULL
  for(j in 1:p){
    datavec <- c(datavec, datamat[j, (1:p)[-j]])
  }
  return(datavec)
}

```

```

frobenius <- function(datamat, targetmat){
  # Modified version of the Frobenius measure for a square
  # symmetric matrix with p(p + 1) / 2 estimable parameters

  # datamat: symmetric square matrix for which to calculate the
  # Frobenius measure
  # targetmat: target matrix of same size as datamat, to compare
  # datamat against

```

```

p <- ncol(datamat)
frobttotal <- 0
frobmat <- (datamat - targetmat)^2
for(j in 1:p){
  for(h in j:p){
    frobttotal <- frobttotal + frobmat[j, h]
  }
}
return(sqrt(frobttotal))
}

```

B.1.6 Discriminant analysis function

```

discriminant.qda <- function(origdata, group,
  method = c("unbiased", "pooled", "cpc", "fullcpccrossvalid"),
  B = NULL, standardize = FALSE){
  # origdata: matrix containing the sample data for two groups
  # group: vector (with values 1 and 2) indicating group
  # membership for the rows in origdata
  # method: unbiased, cpc, fullcpccrossvalid, pooled
  # standardize: should the standardised covariance matrices (i.e.
  # the correlation matrices) be used?

  n.all1 <- nrow(origdata[group == 1, ])
  n.all2 <- nrow(origdata[group == 2, ])
  n.train1 <- round(n.all1 * 0.7, 0)
  n.train2 <- round(n.all2 * 0.7, 0)
  n.test1 <- n.all1 - n.train1
  n.test2 <- n.all2 - n.train2
  p <- ncol(origdata)
  group.predict <- rep(NA, times = (n.test1 + n.test2))

  myFunc <- function(datavec){
    return(datavec / sd(datavec))
  }

  standcol <- function(datamat){
    return(apply(datamat, 2, myFunc))
  }

```

```

if (standardize){
  group1data <- origdata[group == 1, ]
  group2data <- origdata[group == 2, ]
  group1data <- standcol(group1data)
  group2data <- standcol(group2data)
  temp.origdata <- rbind(group1data, group2data)
} else {
  temp.origdata <- origdata
}

group1data <- temp.origdata[group == 1, ][1:n.train1, ]
group2data <- temp.origdata[group == 2, ][1:n.train2, ]
S1 <- cov(group1data)
S2 <- cov(group2data)
testdata <- rbind(temp.origdata[group == 1, ]
  [(n.train1 + 1):n.all1, ], temp.origdata[group == 2, ]
  [(n.train2 + 1):n.all2, ])

if (method[1] == "pooled"){
  Sp <- (S1 * (nrow(group1data) - 1) + S2 * (nrow(group2data)
    - 1)) / (nrow(group1data) + nrow(group2data) - 2)
  S1 <- Sp
  S2 <- Sp
}

if (method[1] == "cpc"){
  S1 <- B %%% diag(diag(t(B) %%% S1 %%% B)) %%% t(B)
  S2 <- B %%% diag(diag(t(B) %%% S2 %%% B)) %%% t(B)
}

if (method[1] == "fullcpccrossvalid"){
  S1.cpc <- B %%% diag(diag(t(B) %%% S1 %%% B)) %%% t(B)
  S2.cpc <- B %%% diag(diag(t(B) %%% S2 %%% B)) %%% t(B)
  alpha1 <- alpha.crossvalid(group1data, B = B, reps = 100)
  alpha2 <- alpha.crossvalid(group2data, B = B, reps = 100)
  S1 <- alpha1 * S1 + (1 - alpha1) * S1.cpc
  S2 <- alpha2 * S2 + (1 - alpha2) * S2.cpc
}

S1.inv <- solve(S1)
S2.inv <- solve(S2)
xbar1 <- apply(group1data, 2, mean)
xbar2 <- apply(group2data, 2, mean)

```

```

c <- 0.5 * (log(det(S1) / det(S2)) + (xbar1 %*% S1.inv %*%
  xbar1 - xbar2 %*% S2.inv %*% xbar2))

for(i in 1:(n.test1 + n.test2)){
  newobs <- as.matrix(testdata[i, , drop = FALSE])
  classvalue <- -0.5 * newobs %*% (S1.inv - S2.inv) %*%
    t(newobs) + (xbar1 %*% S1.inv - xbar2 %*% S2.inv) %*%
    t(newobs)
  if (classvalue >= c){
    group.predict[i] <- 1
  } else {
    group.predict[i] <- 2
  }
}

group.test <- c(group[group == 1][(n.train1 + 1):n.all1],
  group[group == 2][(n.train2 + 1):n.all2])
discrep <- group.test - group.predict
misclassrate <- length(discrep[discrep != 0])
misclassrate <- misclassrate / (n.test1 + n.test2)
return(list(cbind(testdata, group = group.test,
  group.predict = group.predict), misclassrate))
}

```

B.1.7 Biplot functions

```

biplot <- function(datalist, B, D3 = FALSE, varex = 1,
  plotvar = TRUE, main = "CPC biplot", col = c("blue", "red",
  "green", "orange", "brown", "purple"), radius = 0.1, lwd = 3){
  # Draws a 2- or 3-dimensional biplot of the data in datamat
  # (with different colours indicating the different groups),
  # rotated with the orthogonal matrix B.

  # datalist: list of the data from the k groups
  # B: orthogonal projection matrix
  # D3: should a 3-dimensional biplot be drawn?
  # varex: expansion factor for drawing the variables on the biplot
  # plotvar: should the variables be drawn on the biplot?
  # main: title of the biplot
  # col: colors for the data points of the k groups

```

```

k <- length(datalist)
datamat <- NULL
nvec <- rep(NA, times = k)
for(i in 1:k){
  datamat <- rbind(datamat, datalist[[i]])
  nvec[i] <- nrow(datalist[[i]])
}
p <- ncol(datamat)
varnames <- colnames(datamat)

# Standardize the columns of datamat by subtracting the column
# means
datamat <- t(t(datamat) - apply(datamat, 2, mean))

plotpoints <- as.matrix(datamat) %*% B

# 3-dimensional biplot
if (D3){
  library(rgl)
  rgl.open()
  rgl.bg(color = "white", sphere = TRUE)
  plot3d(x = plotpoints[, 1], y = plotpoints[, 2],
        z = plotpoints[, 3], xlab = "PC 1", ylab = "PC 2",
        zlab = "PC 3", type = "n")
  decorate3d(main = main, xlab = NULL, ylab = NULL, zlab = NULL)
  begin <- 1
  for(i in 1:k){
    end <- begin + nvec[i] - 1
    spheres3d(x = plotpoints[begin:end, 1],
             y = plotpoints[begin:end, 2],
             z = plotpoints[begin:end, 3], col = col[i],
             radius = radius)
    begin <- end + 1
  }

  if (plotvar){
    for(j in 1:p){
      lines3d(x = c(0, B[j, 1] * varex),
             y = c(0, B[j, 2] * varex), z = c(0, B[j, 3] * varex),
             lwd = lwd)
    }
    text3d(x = B[, 1] * varex, y = B[, 2] * varex,

```

```

        z = B[, 3] * varex, texts = varnames)
    }
} else {

# 2-dimensional biplot
library(MASS)
par(pch = 20)
eqscplot(x = plotpoints[, 1], y = plotpoints[, 2], type = "n",
         xlab = "PC 1", ylab = "PC 2", main = main)

begin <- 1
for(i in 1:k){
  end <- begin + nvec[i] - 1
  points(x = plotpoints[begin:end, 1],
        y = plotpoints[begin:end, 2], type = "p", col = col[i])
  begin <- end + 1
}

if (plotvar){
  arrows(x0 = rep(0, times = p), x1 = B[, 1] * varex,
        y0 = rep(0, times = p), y1 = B[, 2] * varex, lwd = lwd)
  text(x = B[, 1] * varex, y = B[, 2] * varex, pos = 3,
       labels = varnames)
}
}
}

```

```

biplot.measures <- function(datalist, projectmat, rdim){
  # Calculates goodness of fit measures for r-dimensional
  # principal component biplot of data with distinct groups

  # datalist: list containing all data
  # projectmat: orthogonal projection matrix for biplot
  # rdim: number of dimensions of biplot

  k <- length(datalist)
  p <- ncol(datalist[[1]])
  if (rdim > p){
    cat("Number of biplot dimensions cannot be larger than number
        of variables in data!\n")
    return()
  }
}

```


B.1. THE CPC PACKAGE

317

```

varnames <- colnames(datalist[[1]])

# Overall quality of biplot display

X <- NULL
for(i in 1:k){
  X <- rbind(X, as.matrix(datalist[[i]]))
}
n <- nrow(X)
Xmean <- apply(X, 2, mean)
X <- t(t(X) - Xmean)
Y <- X %%% projectmat[, 1:rdim]
totalvar <- sum(diag(t(X) %%% X))
fittedvar <- sum(diag(t(Y) %%% Y))
overall.quality <- fittedvar / totalvar
  # Overall quality of display of points in biplot (within and
  # between group variation)

# Mean quality of biplot display of variation within groups

withingroup.total <- rep(NA, times = k)
withingroup.fitted <- rep(NA, times = k)
within.total <- 0
within.fitted <- 0
between.total <- 0
between.fitted <- 0
Xmean.fit <- t(projectmat[, 1:rdim]) %%% Xmean
for(i in 1:k){
  groupdata <- as.matrix(datalist[[i]])
  groupmean <- apply(groupdata, 2, mean)
  groupdata <- t(t(groupdata) - groupmean)
  fitteddata <- groupdata %%% projectmat[, 1:rdim]

  withingroup.total[i] <- sum(diag(t(groupdata) %%% groupdata))
  within.total <- within.total + withingroup.total[i]
  withingroup.fitted[i] <- sum(diag(t(fitteddata) %%%
    fitteddata))
  within.fitted <- within.fitted + withingroup.fitted[i]

  between.total <- between.total + t(groupmean - Xmean) %%%
    (groupmean - Xmean)
  groupfitted.mean <- t(projectmat[, 1:rdim]) %%% groupmean
  between.fitted <- between.fitted + t(groupfitted.mean -

```

```

      Xmean.fit) %*% (groupfitted.mean - Xmean.fit)
}
within.quality <- withingroup.fitted / withingroup.total
  # Within group variation quality of display
within.quality <- matrix(within.quality, ncol = k, byrow = TRUE)
colnames(within.quality) <- paste("Group ", c(1:k), sep = "")
within.quality.mean <- within.fitted / within.total
  # Mean quality of display of points (within group variation)

# Quality of between group variation displayed in biplot

between.quality <- as.numeric(between.fitted / between.total)
  # Overall quality of between group variation as represented in
  # r-dimensional biplot

# Adequacies of the variables

adequacies <- diag(projectmat[, 1:rdim] %*%
  t(projectmat[,1:rdim]))
  # adequacy of the variables as represented in a r-dimensional
  # biplot
adequacies.median <- median(adequacies)
adequacies <- matrix(adequacies, ncol = p, byrow = TRUE)
colnames(adequacies) <- varnames

# Axis predictivities (predictivities of the variables)

J <- diag(c(rep(1, times = rdim), rep(0, times = (p - rdim))))
X.fitted <- X %*% projectmat %*% J %*% t(projectmat)
axis.predictivities <- diag(diag(diag(t(X.fitted) %*% X.fitted))
  %*% solve(diag(diag(t(X) %*% X)))) # Axis predictivities
axis.predictivities.mean <- sum(diag(diag(diag(t(X.fitted)
  %*% X.fitted)) %*% solve(diag(diag(t(X) %*% X))))) / p
  # Mean predictivity of axes (variables)

axis.predictivities <- matrix(axis.predictivities, ncol = p,
  byrow = TRUE)
colnames(axis.predictivities) <- varnames

# Sample predictivities (predictivities of the observations)

sample.predictivities <- diag(diag(diag(X.fitted %*%
  t(X.fitted))) %*% solve(diag(diag(X %*% t(X)))))

```

```

# Sample predictivities
sample.predictivities.mean <- sum(diag(diag(diag(X.fitted %*%
  t(X.fitted))) %*% solve(diag(diag(X %*% t(X))))) / nrow(X)
# Mean predictivity of samples (observations)

# Mean standard predictive errors (Alves 2012)

Xsd <- apply(X, 2, sd)
onevec <- matrix(1, nrow = n, ncol = 1)
mspe <- rep(NA, times = p)
for(j in 1:p){
  mspe[j] <- (t(onevec) %*% abs(X[, j] - X %*%
    projectmat[, 1:rdim] %*% t(projectmat[j, 1:rdim,
      drop = FALSE]))) / (n * Xsd[j])
}
mspe.mean <- mean(mspe)

return(list(overall.quality = overall.quality,
  within.quality = within.quality,
  within.quality.mean = within.quality.mean,
  between.quality = between.quality, adequacies = adequacies,
  adequacies.median = adequacies.median,
  axis.predictivities = axis.predictivities,
  axis.predictivities.mean = axis.predictivities.mean,
  sample.predictivities = sample.predictivities,
  sample.predictivities.mean = sample.predictivities.mean,
  mspe = mspe, mspe.mean = mspe.mean))
}

```

```

biplot.choice <- function(datalist, rdim, add.projectmats = NULL){
  # Gives biplot goodness of fit measures for different types of
  # principal components biplots for data with distinct groups

  # datalist: list containing all data
  # rdim: number of dimensions of biplot
  # add.projectmats: additional orthogonal projections matrices to
  # compute biplot fit measures for

  k <- length(datalist)
  p <- ncol(datalist[[1]])
  nvec <- rep(NA, times = k)

```

```

# Eigenvectors of pooled covariance matrix

dfpooled <- 0
SSpooled <- 0
for(i in 1:k){
  nvec[i] <- nrow(datalist[[i]])
  dfpooled <- dfpooled + nvec[i] - 1
  SSpooled <- SSpooled + cov(as.matrix(datalist[[i]])) *
    (nvec[i] - 1)
}
Sp <- SSpooled / dfpooled
Ep <- eigen(Sp)$vectors
pooledcov.output <- biplot.measures(datalist = datalist,
  projectmat = Ep, rdim = rdim)

# Eigenvectors of covariance matrix of pooled data

datamat <- NULL
for(i in 1:k){
  datamat <- rbind(datamat, as.matrix(datalist[[i]]))
}
E <- eigen(cov(datamat))$vectors
pooleddata.output <- biplot.measures(datalist = datalist,
  projectmat = E, rdim = rdim)

# CPC: FG algorithm

S <- array(NA, dim = c(p, p, k))
for(i in 1:k){
  S[, , i] <- cov(as.matrix(datalist[[i]]))
}
B.flury <- cpc::FG(covmats = S, nvec = nvec)$B
flury.output <- biplot.measures(datalist = datalist,
  projectmat = B.flury, rdim = rdim)

# CPC: Stepwise CPC

B.stepwise <- stepwisecpc(covmats = S, nvec = nvec)$B
stepwise.output <- biplot.measures(datalist = datalist,
  projectmat = B.stepwise, rdim = rdim)

# CPC: JADE algorithm

```

```

library(JADE)
B.jade <- rjd(X = S)$V
lvals <- rep(0, times = p)
for(i in 1:k){
  lvals <- lvals + diag(t(B.jade) %*% S[, , i] %*% B.jade)
}
jade.order <- order(lvals, decreasing = TRUE)
B.jade <- B.jade[, jade.order]
jade.output <- biplot.measures(datalist = datalist,
  projectmat = B.jade, rdim = rdim)

# Produce output table

pooledcov<-c(pooledcov.output$overall.quality,
  pooledcov.output$within.quality.mean,
  pooledcov.output$between.quality,
  pooledcov.output$adequacies.median,
  pooledcov.output$mspe.mean,
  pooledcov.output$sample.predictivities.mean)

pooleddata<-c(pooleddata.output$overall.quality,
  pooleddata.output$within.quality.mean,
  pooleddata.output$between.quality,
  pooleddata.output$adequacies.median,
  pooleddata.output$mspe.mean,
  pooleddata.output$sample.predictivities.mean)

flury<-c(flury.output$overall.quality,
  flury.output$within.quality.mean,
  flury.output$between.quality,
  flury.output$adequacies.median,
  flury.output$mspe.mean,
  flury.output$sample.predictivities.mean)

stepwise<-c(stepwise.output$overall.quality,
  stepwise.output$within.quality.mean,
  stepwise.output$between.quality,
  stepwise.output$adequacies.median,
  stepwise.output$mspe.mean,
  stepwise.output$sample.predictivities.mean)

jade<-c(jade.output$overall.quality,
  jade.output$within.quality.mean,

```

```

        jade.output$between.quality,
        jade.output$adequacies.median,
        jade.output$mspe.mean,
        jade.output$sample.predictivities.mean)

resultsmat <- rbind(pooledcov, pooleddata, flury, stepwise, jade)
rownames(resultsmat) <- c("Pooled S", "Pooled data", "Flury",
  "Stepwise CPC", "JADE")
colnames(resultsmat) <- c("Overall", "Within", "Between",
  "Adequacy", "MSPE", "Sample predictivities")

return(resultsmat)
}

```

B.1.8 PLS regression function

```

pls.est <- function(X, Y){
  # Implementation of the partial least squares (PLS) algorithm
  # as described in Elements of Statistical Learning -- Hastie et
  # al. (2009)

  # X: independent variables matrix
  # Y: dependent/response variable vector

  X.orig <- as.matrix(X)
  X.sdinv <- diag(1 / apply(X.orig, 2, sd))
  X.orig <- t(t(X.orig) - apply(X.orig, 2, mean)) %*% X.sdinv
  # standardise columns of X.orig to have zero mean and unit
  # variance
  Y <- as.vector(Y)

  # PLS, as from Algorithm 3.3 (p81) in Hastie et al. (2009)
  X <- X.orig
  p <- ncol(X)
  y0 <- mean(Y) * rep(1, times = nrow(X))
  Z <- matrix(0, nrow = nrow(X), ncol = p)
  Ymat <- matrix(NA, nrow = nrow(X), ncol = p)
  for(m in 1:p){
    for(j in 1:p){
      phi <- X[, j] %*% Y
      Z[, m] <- Z[, m] + phi * X[, j]
    }
  }
}

```

```

    }
    theta <- (Z[, m] %*% Y) / (Z[, m] %*% Z[, m])
    if (m == 1){
      Ymat[, m] <- y0 + theta * Z[, m]
    } else {
      Ymat[, m] <- Ymat[, m - 1] + theta * Z[, m]
    }
    for(j in 1:p){
      X[, j] <- X[, j] - ((Z[, m] %*% X[, j]) / (Z[, m] %*%
        Z[, m])) %*% Z[, m]
    }
  }

  B <- solve(t(X.orig) %*% X.orig) %*% t(X.orig) %*% Z
  for(j in 1:p){
    B[, j] <- B[, j] / sqrt(B[, j] %*% B[, j])
  }

  return(list(pls.scores = Z, pls.loadings = B))
}

```

B.2 VON data analysis script

B.2.1 Load the necessary R packages

```
library(cpc)
library(mvnormtest)
library(JADE)
require(ROCR, quietly = TRUE)
library(Hmisc)
```

B.2.2 Read in the VON data files

```
von2008 <- read.csv("VON_2008_Data.csv")
von2009 <- read.csv("VON_2009_Data.csv")
von2008[, "BWGT"] <- von2008[, "BWGT"] / 1000
  # convert birthweight from gram to kg
von2009[, "BWGT"] <- von2009[, "BWGT"] / 1000
  # convert birthweight from gram to kg
numvarnames <- c("BWGT", "AP1", "AP5", "GESTAGE", "BHEADCIR",
  "ATEMP")
numvarunits <- c("Kilograms", "Score", "Score", "Weeks",
  "Centimetres", "Degrees Celsius")
catvarnames <- c("REGION", "MRACE", "VAGDEL", "SEX")
extravarnames <- c("HOSPNO", "DIED", "TRANSFERRED")
```

B.2.3 Chapter 1

```
# Covariance matrix of the numerical variables in the VON 2009
# cohort
S2009 <- cov(von2009[, numvarnames])
S2009

# Empirical distributions of the numerical variables
par(mfrow = c(3, 2), oma = c(0, 0, 2, 0), mar = c(4, 4, 4, 2)
  + 0.1)
plot(density(von2009[, numvarnames[1]]), main = numvarnames[1],
  lwd = 2, xlab = numvarunits[1])
barplot(table(von2009[, numvarnames[2]]), main = numvarnames[2],
  xlab = numvarunits[2], ylab = "Frequency", col = "darkgray",
  width = 10, space = 5)
```



```

barplot(table(von2009[, numvarnames[3]]), main = numvarnames[3],
  xlab = numvarunits[3], ylab = "Frequency", col = "darkgray",
  width = 10, space = 5)
plot(density(von2009[, numvarnames[4]]), main = numvarnames[4],
  lwd = 2, xlab = numvarunits[4])
plot(density(von2009[, numvarnames[5]]), main = numvarnames[5],
  lwd = 2, xlab = numvarunits[5])
plot(density(von2009[, numvarnames[6]]), main = numvarnames[6],
  lwd = 2, xlab = numvarunits[6])
title(main = "VON 2009 cohort: Numerical variables", outer = TRUE)

```

B.2.4 Chapter 2

```

# Total variance in the data
tr(S2009)

# Eigenvectors and eigenvalues
S2009.eigen <- eigen(S2009)
S2009.eigen

# Scree plot of the eigenvalues
par(pch = 20)
plot(S2009.eigen$values, type = "b",
  main = "VON 2009 cohort: Scree plot of the eigenvalues",
  xlab = "Principal component", ylab = "Eigenvalue")

# Pearson correlations of the numerical variables with LOS
cor(von2009[, c("LOS1", numvarnames)], method = "pearson")

# Kruskal-Wallis tests of the mortality groups with regard to the
# numerical variables
kruskal.test(x = von2009[, numvarnames[1]], g = von2009[, "DIED"])
kruskal.test(x = von2009[, numvarnames[2]], g = von2009[, "DIED"])
kruskal.test(x = von2009[, numvarnames[3]], g = von2009[, "DIED"])
kruskal.test(x = von2009[, numvarnames[4]], g = von2009[, "DIED"])
kruskal.test(x = von2009[, numvarnames[5]], g = von2009[, "DIED"])
kruskal.test(x = von2009[, numvarnames[6]], g = von2009[, "DIED"])

# Testing for multivariate normality in the population
mshapiro.test(t(von2009[, numvarnames]))

# Parametric methods for inference on the eigenvalues and

```

```

# eigenvectors (incorrectly assuming multivariate normality)
p <- ncol(S2009)
eigvec.SE <- matrix(NA, nrow = p, ncol = p)
for(j in 1:p){
  for(h in 1:p){
    tempsum <- 0
    for(u in 1:p){
      if(u != j){
        tempsum <- tempsum + (S2009.eigen$values[u] /
          (S2009.eigen$values[u] - S2009.eigen$values[j])^2)
          * (S2009.eigen$vectors[h, u])^2
      }
    }
    eigvec.SE[h, j] <- sqrt(1 / (nrow(von2009) - 1)
      * S2009.eigen$values[j] * tempsum)
  }
}
eigvec.SE # standard errors of the eigenvector loadings

eigval.SE <- sqrt(2 / (nrow(von2009) - 1) * S2009.eigen$values)
eigval.SE # standard errors of the eigenvalues

eigval.LCL <- S2009.eigen$values / (1 + qnorm(p = 0.975)
  * sqrt(2 / (nrow(von2009) - 1)))
eigval.UCL <- S2009.eigen$values / (1 - qnorm(p = 0.975)
  * sqrt(2 / (nrow(von2009) - 1)))
eigval.LCL # parametric lower 95% confidence limits
eigval.UCL # parametric upper 95% confidence limits

# Bootstrap sampling to estimate the standard errors and confidence
# intervals for the eigenvector loadings and the eigenvalues
reps <- 1000
n <- nrow(von2009)
p <- ncol(von2009[, numvarnames])
maxabsvals <- apply(abs(S2009.eigen$vectors), 2, max)
maxpos <- rep(NA, times = p)
maxvals <- rep(NA, times = p)
for(j in 1:p){
  maxpos[j] <- which(abs(S2009.eigen$vectors[, j])
    == maxabsvals[j])
  maxvals[j] <- S2009.eigen$vectors[maxpos[j], j]
}
booteigvecs <- array(NA, dim = c(p, p, reps))

```

```

booteigvals <- matrix(NA, nrow = reps, ncol = p)
booteigvalvars <- matrix(NA, nrow = reps, ncol = p)
booteigvalcumvars <- matrix(NA, nrow = reps, ncol = p)
for(r in 1:reps){
  bootsamp <- von2009[sample(1:n, size = n,
    replace = T), numvarnames]
  bootsamp.eigen <- eigen(cov(bootsamp))
  booteigvecs[, , r] <- bootsamp.eigen$vectors
  booteigvals[r, ] <- bootsamp.eigen$values
  booteigvalvars[r, ] <- bootsamp.eigen$values
    / sum(bootsamp.eigen$values) * 100
  temp <- rep(0, times = p)
  for(j in 1:p){
    temp[j] <- sum(bootsamp.eigen$values[1:j])
      / sum(bootsamp.eigen$values) * 100
    bootmaxval <- booteigvecs[, , r][maxpos[j], j]
    if((sign(bootmaxval) * sign(maxvals[j])) != 1){
      booteigvecs[, j, r] <- booteigvecs[, j, r] * (-1)
    }
  }
  booteigvalcumvars[r, ] <- temp
}
(apply(booteigvals, 2, quantile, probs = c(0.84))
 - apply(booteigvals, 2, quantile, probs = c(0.16))) / 2
# bootstrap standard errors of eigenvalues

apply(booteigvals, 2, quantile, probs = c(0.025, 0.975))
# bootstrap C.I. for eigenvalues
apply(booteigvalvars, 2, quantile, probs = c(0.025, 0.975))
# bootstrap C.I. for variance accounted for per principal
# component
apply(booteigvalcumvars, 2, quantile, probs = c(0.025, 0.975))
# bootstrap C.I. for cumulative variance accounted for per
# principal component

lowerlims <- matrix(NA, nrow = p, ncol = p)
upperlims <- matrix(NA, nrow = p, ncol = p)
boot.se <- matrix(NA, nrow = p, ncol = p)
for(j in 1:p){
  for(h in 1:p){
    tempvec <- booteigvecs[j, h, ]
    confint <- quantile(tempvec, probs = c(0.025, 0.975))
  }
}

```

```

    lowerlims[j, h] <- confint[1]
    upperlims[j, h] <- confint[2]
    boot.se[j, h] <- (quantile(tempvec, probs = 0.84)
      - quantile(tempvec, probs = 0.16)) / 2
  }
}
lowerlims # lower 95% confidence limits for the eigenvector
# elements
upperlims # upper 95% confidence limits for the eigenvector
# elements
boot.se # bootstrap standard errors of the eigenvector loadings

# Plot of the bootstrap distributions of the loadings in the first
# eigenvector
par(mfrow = c(3, 2), oma = c(0, 0, 2, 0), mar = c(3, 4, 4, 2)
  + 0.1)
plot(density(booteigvecs[1, 1, ]), xlim = c(-1, 1), main = "BWGT",
  xlab = "")
polygon(density(booteigvecs[1, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
plot(density(booteigvecs[2, 1, ]), xlim = c(-1, 1), main = "AP1",
  xlab = "")
polygon(density(booteigvecs[2, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
plot(density(booteigvecs[3, 1, ]), xlim = c(-1, 1), main = "AP5",
  xlab = "")
polygon(density(booteigvecs[3, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
plot(density(booteigvecs[4, 1, ]), xlim = c(-1, 1),
  main = "GESTAGE", xlab = "")
polygon(density(booteigvecs[4, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
plot(density(booteigvecs[5, 1, ]), xlim = c(-1, 1),
  main = "BHEADCIR", xlab = "")
polygon(density(booteigvecs[5, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
plot(density(booteigvecs[6, 1, ]), xlim = c(-1, 1),
  main = "ATEMP", xlab = "")
polygon(density(booteigvecs[6, 1, ]), col = "darkgray")
abline(v = 0, lty = 3)
title(main = "VON 2009 cohort: Eigenvector 1", outer = TRUE)

# Computing eigenvector residuals to identify outliers with regard

```

```

# to the covariance structure
pcscores <- as.matrix(von2009[, numvarnames]) %*%
  S2009.eigen$vectors
pcresiduals <- apply(pcscores[, 4:6]^2, 1, sum)
par(pch = 1)
boxplot(pcresiduals, horizontal = TRUE,
  main = "VON 2009 cohort: Reduced PCA residuals")

# Plot of last two principal components to identify outliers
pc5.outliers <- pcscores[pcscores[, 5] < 28.5, ]
pc5.outliers.pos <- rep(NA, times = nrow(pc5.outliers))
for(i in 1:nrow(pc5.outliers)){
  pc5.outliers.pos[i] <- which(pcscores[, 5] ==
    pc5.outliers[i, 5])
}
pc5.outliers.pos # row numbers of the outliers (PC5)
pc6.outliers <- pcscores[((pcscores[, 6] < (-9.3)) |
  (pcscores[, 6] > (-5.5))), ]
pc6.outliers.pos <- rep(NA, times = nrow(pc6.outliers))
for(i in 1:nrow(pc6.outliers)){
  pc6.outliers.pos[i] <- which(pcscores[, 6] == pc6.outliers[i, 6])
}
pc6.outliers.pos # row numbers of the outliers (PC6)
par(pch = 20)
plot(x = pcscores[, 5], y = pcscores[, 6], type = "p",
  main = "VON 2009 cohort", xlab = "PC5", ylab = "PC6",
  ylim = c(-9.9, -5.2))
text(x = pc5.outliers[, 5], y = pc5.outliers[, 6], pos = 1,
  labels = pc5.outliers.pos, cex=0.7)
text(x = pc6.outliers[, 5], y = pc6.outliers[, 6], pos = 1,
  labels = pc6.outliers.pos, cex=0.7)

```

B.2.5 Chapter 3

```

# Delivery mode groups
caesarean <- von2009[von2009[, "VAGDEL"] ==
  "Caesarean", numvarnames]
vaginal <- von2009[von2009[, "VAGDEL"] == "Vaginal", numvarnames]
nvek <- c(nrow(caesarean), nrow(vaginal))

# Covariance matrices of the delivery mode groups
S <- array(NA, dim = c(6, 6, 2))

```

```

S[, , 1] <- cov(caesarean)
S[, , 2] <- cov(vaginal)
S

# Separate sets of eigenvectors for the delivery mode groups
caesarean.eigen <- eigen(S[, , 1])
vaginal.eigen <- eigen(S[, , 2])
caesarean.eigen$vectors
vaginal.eigen$vectors

# Angles between the separate sets of eigenvectors, and similarity
# measure
acos(abs(t(caesarean.eigen$vectors) %*% vaginal.eigen$vectors)) /
  (pi / 2) * 90 # angles (in degrees) between the separate sets
# of eigenvectors
tr(abs(t(caesarean.eigen$vectors) %*% vaginal.eigen$vectors))
  # similarity measure for the eigenvectors of the delivery mode
  # groups
tr(abs(t(caesarean.eigen$vectors) %*%
  vaginal.eigen$vectors[, c(1, 2, 3, 5, 4, 6)]))
  # similarity measure for the corresponding eigenvectors

# Common eigenvectors estimated with the FG algorithm
delivery.cpc <- cpc::FG(covmats = S, nvec = nvek)
delivery.cpc$B
  # common eigenvectors estimated with the FG algorithm
delivery.cpc$diagvals[, 1]
  # Caesarean eigenvalues under the CPC model (FG algorithm)
delivery.cpc$diagvals[, 2]
  # Vaginal eigenvalues under the CPC model (FG algorithm)

# Common eigenvectors estimated with the JADE algorithm
delivery.jade <- rjd(X = S)$V[, c(4, 2, 5, 3, 6, 1)]
delivery.jade
  # common eigenvectors estimated with the JADE algorithm
diag(t(delivery.jade) %*% S[, , 1] %*% delivery.jade)
  # Caesarean eigenvalues under the CPC model (JADE algorithm)
diag(t(delivery.jade) %*% S[, , 2] %*% delivery.jade)
  # Vaginal eigenvalues under the CPC model (JADE algorithm)

# Common eigenvectors estimated with the Stepwise CPC algorithm
delivery.stepwise <- stepwisecpc(covmats = S, nvec = nvek)
delivery.stepwise$B

```

```

# common eigenvectors estimated with the Stepwise CPC algorithm
delivery.stepwise$eigenvals[, 1]
# Caesarean eigenvalues under the CPC model (Stepwise CPC
# algorithm)
delivery.stepwise$eigenvals[, 2]
# Vaginal eigenvalues under the CPC model (Stepwise CPC
# algorithm)

# Parametric methods for inference on the eigenvalues and
# eigenvectors (incorrectly assuming multivariate normality)
eigval.SE <- sqrt(2 / (nrow(caesarean) - 1)
  * delivery.cpc$diagvals[, 1])
eigval.SE # standard errors of the Caesarean eigenvalues
eigval.SE <- sqrt(2 / (nrow(vaginal) - 1)
  * delivery.cpc$diagvals[, 2])
eigval.SE # standard errors of the Vaginal eigenvalues

eigval.LCL <- delivery.cpc$diagvals[, 1] / (1 + qnorm(p = 0.975)
  * sqrt(2 / (nrow(caesarean) - 1)))
eigval.UCL <- delivery.cpc$diagvals[, 1] / (1 - qnorm(p = 0.975)
  * sqrt(2 / (nrow(caesarean) - 1)))
eigval.LCL # parametric lower 95% confidence limits for the
# Caesarean eigenvalues
eigval.UCL # parametric upper 95% confidence limits for the
# Caesarean eigenvalues

eigval.LCL <- delivery.cpc$diagvals[, 2] / (1 + qnorm(p = 0.975)
  * sqrt(2 / (nrow(vaginal) - 1)))
eigval.UCL <- delivery.cpc$diagvals[, 2] / (1 - qnorm(p = 0.975)
  * sqrt(2 / (nrow(vaginal) - 1)))
eigval.LCL # parametric lower 95% confidence limits for the
# Vaginal eigenvalues
eigval.UCL # parametric upper 95% confidence limits for the
# Vaginal eigenvalues

p <- ncol(delivery.cpc$B)
k <- 2
theta.hmean <- matrix(NA, nrow = p, ncol = p)
for(h in 1:p){
  for(j in 1:p){
    theta <- rep(NA, times = k)
    for(i in 1:k){
      if(h != j){

```

```

        theta[i] <- (sum(nvek) - 1) / (nvek[i] - 1)
        * (delivery.cpc$diagvals[j, i]
        * delivery.cpc$diagvals[h, i])
        / (delivery.cpc$diagvals[j, i]
        - delivery.cpc$diagvals[h, i])^2
    }
  }
  theta.hmean[j, h] <- 1 / sum(1 / theta)
}
}
commoneigvec.se <- matrix(NA, nrow = p, ncol = p)
for(h in 1:p){
  for(m in 1:p){
    tempsum <- 0
    for(j in 1:p){
      if(h != j){
        tempsum <- tempsum + theta.hmean[j, h]
        * (delivery.cpc$B[m, j])^2
      }
    }
    commoneigvec.se[m, h] <- sqrt(tempsum / (sum(nvek) - 1))
  }
}
commoneigvec.se # parametric standard errors of the common
# eigenvector loadings

# Bootstrap standard errors for the common eigenvector elements
# and the eigenvalues under the CPC model
set.seed(4959)
reps <- 1000
n <- nrow(von2009)
p <- ncol(von2009[, numvarnames])
maxabsvals <- apply(abs(delivery.cpc$B), 2, max)
maxpos <- rep(NA, times = p)
maxvals <- rep(NA, times = p)
for(j in 1:p){
  maxpos[j] <- which(abs(delivery.cpc$B[, j]) == maxabsvals[j])
  maxvals[j] <- delivery.cpc$B[maxpos[j], j]
}
bootcommoneigvecs <- array(NA, dim = c(p, p, reps))
booteigvals1 <- matrix(NA, nrow = reps, ncol = p)
booteigvals2 <- matrix(NA, nrow = reps, ncol = p)
booteigvalvars1 <- matrix(NA, nrow = reps, ncol = p)

```



```

booteigvalvars2 <- matrix(NA, nrow = reps, ncol = p)
booteigvalcumvars1 <- matrix(NA, nrow = reps, ncol = p)
booteigvalcumvars2 <- matrix(NA, nrow = reps, ncol = p)
boot.S <- array(NA, dim = c(6, 6, 2))
for(r in 1:reps){
  bootsamp1 <- caesarean[sample(1:nvek[1], size = nvek[1],
    replace = T), numvarnames]
  bootsamp2 <- vaginal[sample(1:nvek[2], size = nvek[2],
    replace = T), numvarnames]
  boot.S[, , 1] <- cov(bootsamp1)
  boot.S[, , 2] <- cov(bootsamp2)
  bootcommoneigvecs[, , r] <- cpc::FG(covmat = boot.S, nvec = nvek)$B
  ###
  # adjustment for 4th and 5th eigenvectors having different orders
  # in the two covariance matrices
  if((abs(bootcommoneigvecs[, 4, r] %>% delivery.cpc$B[, 4])
    + abs(bootcommoneigvecs[, 5, r] %>% delivery.cpc$B[, 5]))
    < (abs(bootcommoneigvecs[, 4, r] %>% delivery.cpc$B[, 5])
    + abs(bootcommoneigvecs[, 5, r] %>% delivery.cpc$B[, 4]))) {
    bootcommoneigvecs[, , r]
      <- bootcommoneigvecs[, c(1, 2, 3, 5, 4, 6), r]
  }
  ###
  booteigvals1[r, ] <- diag(t(bootcommoneigvecs[, , r]) %>%
    boot.S[, , 1] %>% bootcommoneigvecs[, , r])
  booteigvals2[r, ] <- diag(t(bootcommoneigvecs[, , r]) %>%
    boot.S[, , 2] %>% bootcommoneigvecs[, , r])
  booteigvalvars1[r, ] <- booteigvals1[r, ]
    / sum(booteigvals1[r, ]) * 100
  booteigvalvars2[r, ] <- booteigvals2[r, ]
    / sum(booteigvals2[r, ]) * 100
  temp1 <- rep(0, times = p)
  temp2 <- rep(0, times = p)
  for(j in 1:p){
    temp1[j] <- sum(booteigvals1[r, 1:j])
      / sum(booteigvals1[r, ]) * 100
    temp2[j] <- sum(booteigvals2[r, 1:j])
      / sum(booteigvals2[r, ]) * 100
    bootmaxval <- bootcommoneigvecs[, , r][maxpos[j], j]
    if((sign(bootmaxval) * sign(maxvals[j])) != 1){
      bootcommoneigvecs[, j, r] <- bootcommoneigvecs[, j, r] * (-1)
    }
  }
}

```

```

    booteigvalcumvars1[r, ] <- temp1
    booteigvalcumvars2[r, ] <- temp2
  }

  (apply(booteigvals1, 2, quantile, probs = c(0.84))
   - apply(booteigvals1, 2, quantile, probs = c(0.16))) / 2
  # bootstrap standard errors of the Caesarean eigenvalues
  (apply(booteigvals2, 2, quantile, probs = c(0.84))
   - apply(booteigvals2, 2, quantile, probs = c(0.16))) / 2
  # bootstrap standard errors of the Vaginal eigenvalues

  apply(booteigvals1, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence intervals for the Caesarean eigenvalues
  apply(booteigvalvars1, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence intervals for variance accounted for
  # per common principal component in the Caesarean group
  apply(booteigvalcumvars1, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence interval for the cumulative variance
  # accounted for per common principal component in the Caesarean
  # group
  apply(booteigvals2, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence interval for the Vaginal eigenvalues
  apply(booteigvalvars2, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence interval for the variance accounted
  # for per common principal component in the Vaginal group
  apply(booteigvalcumvars2, 2, quantile, probs = c(0.025, 0.975))
  # 95% bootstrap confidence interval for the cumulative variance
  # accounted for per common principal component in the Vaginal
  # group

  lowerlims <- matrix(NA, nrow = p, ncol = p)
  upperlims <- matrix(NA, nrow = p, ncol = p)
  boot.se <- matrix(NA, nrow = p, ncol = p)
  for(j in 1:p){
    for(h in 1:p){
      tempvec <- bootcommoneigvecs[j, h, ]
      confint <- quantile(tempvec, probs = c(0.025, 0.975))
      lowerlims[j, h] <- confint[1]
      upperlims[j, h] <- confint[2]
      boot.se[j, h] <- (quantile(tempvec, probs = 0.84)
        - quantile(tempvec, probs = 0.16)) / 2
    }
  }
}

```

B.2. VON DATA ANALYSIS SCRIPT

335

```

lowerlims # bootstrap 95% lower confidence limits for the
# common eigenvector loadings
upperlims # bootstrap 95% upper confidence limits for the
# common eigenvector loadings
boot.se # bootstrap standard errors of the common eigenvector
# loadings

# Regional groups
southafrica <- von2009[von2009[, "REGION"] == "RSA",
  numvarnames]
namibia <- von2009[von2009[, "REGION"] == "Namibia",
  numvarnames]
nvek <- c(nrow(southafrica), nrow(namibia))

# Covariance matrices of the regional groups
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(southafrica)
S[, , 2] <- cov(namibia)
S

# Separate sets of eigenvectors for the regional groups
southafrica.eigen <- eigen(S[, , 1])
namibia.eigen <- eigen(S[, , 2])
southafrica.eigen$eigenvectors
namibia.eigen$eigenvectors

# Angles between the separate sets of eigenvectors, and similarity measure
acos(abs(t(southafrica.eigen$eigenvectors) %*%
  namibia.eigen$eigenvectors)) / (pi / 2) * 90
# angles (in degrees) between the separate sets of eigenvectors
tr(abs(t(southafrica.eigen$eigenvectors) %*% namibia.eigen$eigenvectors))
# similarity measure

# Common eigenvectors estimated with the FG algorithm
region.cpc <- cpc::FG(covmats = S, nvec = nvek)
region.cpc$B # common eigenvectors estimated with the FG algorithm
region.cpc$diagvals[, 1]
# South Africa eigenvalues under the CPC model (FG algorithm)
region.cpc$diagvals[, 2]
# Namibia eigenvalues under the CPC model (FG algorithm)

# Common eigenvectors estimated with the JADE algorithm
region.jade <- rjd(X = S)$V

```

```

region.jade
  # common eigenvectors estimated with the JADE algorithm
diag(t(region.jade) %*% S[, , 1] %*% region.jade)
  # South Africa eigenvalues under the CPC model (JADE algorithm)
diag(t(region.jade) %*% S[, , 2] %*% region.jade)
  # Namibia eigenvalues under the CPC model (JADE algorithm)

# Common eigenvectors estimated with the Stepwise CPC algorithm
region.stepwise <- stepwisecpc(covmats = S, nvec = nvec)
region.stepwise$B
  # common eigenvectors estimated with the Stepwise CPC algorithm
region.stepwise$eigenvals[, 1]
  # South Africa eigenvalues under the CPC model (Stepwise CPC
  # algorithm)
region.stepwise$eigenvals[, 2]
  # Namibia eigenvalues under the CPC model (Stepwise CPC
  # algorithm)

# Parametric methods for inference on the eigenvalues and
# eigenvectors (incorrectly assuming multivariate normality)
eigval.SE <- sqrt(2 / (nrow(southafrica) - 1)
  * region.cpc$diagvals[, 1])
eigval.SE # standard errors of the South Africa eigenvalues
eigval.SE <- sqrt(2 / (nrow(namibia) - 1)
  * region.cpc$diagvals[, 2])
eigval.SE # standard errors of the Namibia eigenvalues

eigval.LCL <- region.cpc$diagvals[, 1] / (1 + qnorm(p = 0.975)
  * sqrt(2 / (nrow(southafrica) - 1)))
eigval.UCL <- region.cpc$diagvals[, 1] / (1 - qnorm(p = 0.975)
  * sqrt(2 / (nrow(southafrica) - 1)))
eigval.LCL # parametric lower 95% confidence limits for the South
  # Africa eigenvalues
eigval.UCL # parametric upper 95% confidence limits for the South
  # Africa eigenvalues

eigval.LCL <- region.cpc$diagvals[, 2] / (1 + qnorm(p = 0.975)
  * sqrt(2 / (nrow(namibia) - 1)))
eigval.UCL <- region.cpc$diagvals[, 2] / (1 - qnorm(p = 0.975)
  * sqrt(2 / (nrow(namibia) - 1)))
eigval.LCL # parametric lower 95% confidence limits for the
  # Namibia eigenvalues
eigval.UCL # parametric upper 95% confidence limits for the

```

B.2. VON DATA ANALYSIS SCRIPT

337

```

# Namibia eigenvalues

p <- ncol(region.cpc$B)
k <- 2
theta.hmean <- matrix(NA, nrow = p, ncol = p)
for(h in 1:p){
  for(j in 1:p){
    theta <- rep(NA, times = k)
    for(i in 1:k){
      if(h != j){
        theta[i] <- (sum(nvek) - 1) / (nvek[i] - 1)
          * (region.cpc$diagvals[j, i] * region.cpc$diagvals[h, i])
          / (region.cpc$diagvals[j, i] - region.cpc$diagvals[h, i])^2
      }
    }
    theta.hmean[j, h] <- 1 / sum(1 / theta)
  }
}
commoneigvec.se <- matrix(NA, nrow = p, ncol = p)
for(h in 1:p){
  for(m in 1:p){
    tempsum <- 0
    for(j in 1:p){
      if(h != j){
        tempsum <- tempsum + theta.hmean[j, h]
          * (region.cpc$B[m, j])^2
      }
    }
    commoneigvec.se[m, h] <- sqrt(tempsum / (sum(nvek) - 1))
  }
}
commoneigvec.se # parametric standard errors of the common
# eigenvector loadings

# Bootstrap standard errors of the common eigenvector elements and
# the eigenvalues under the CPC model
set.seed(3660)
reps <- 1000
n <- nrow(von2009)
p <- ncol(von2009[, numvarnames])
maxabsvals <- apply(abs(region.cpc$B), 2, max)
maxpos <- rep(NA, times = p)
maxvals <- rep(NA, times = p)

```

```

for(j in 1:p){
  maxpos[j] <- which(abs(region.cpc$B[, j]) == maxabsvals[j])
  maxvals[j] <- region.cpc$B[maxpos[j], j]
}
bootcommoneigvecs <- array(NA, dim = c(p, p, reps))
booteigvals1 <- matrix(NA, nrow = reps, ncol = p)
booteigvals2 <- matrix(NA, nrow = reps, ncol = p)
booteigvalvars1 <- matrix(NA, nrow = reps, ncol = p)
booteigvalvars2 <- matrix(NA, nrow = reps, ncol = p)
booteigvalcumvars1 <- matrix(NA, nrow = reps, ncol = p)
booteigvalcumvars2 <- matrix(NA, nrow = reps, ncol = p)
boot.S <- array(NA, dim = c(6, 6, 2))
for(r in 1:reps){
  bootsamp1 <- southafrica[sample(1:nvek[1], size = nvek[1],
    # replace = T), numvarnames]
  bootsamp2 <- namibia[sample(1:nvek[2], size = nvek[2],
    # replace = T), numvarnames]
  boot.S[, , 1] <- cov(bootsamp1)
  boot.S[, , 2] <- cov(bootsamp2)
  bootcommoneigvecs[, , r] <- cpc::FG(covmat = boot.S, nvec = nvek)$B
  booteigvals1[r, ] <- diag(t(bootcommoneigvecs[, , r])
    %*% boot.S[, , 1] %*% bootcommoneigvecs[, , r])
  booteigvals2[r, ] <- diag(t(bootcommoneigvecs[, , r])
    %*% boot.S[, , 2] %*% bootcommoneigvecs[, , r])
  booteigvalvars1[r, ] <- booteigvals1[r, ]
    / sum(booteigvals1[r, ]) * 100
  booteigvalvars2[r, ] <- booteigvals2[r, ]
    / sum(booteigvals2[r, ]) * 100
  temp1 <- rep(0, times = p)
  temp2 <- rep(0, times = p)
  for(j in 1:p){
    temp1[j] <- sum(booteigvals1[r, 1:j]) / sum(booteigvals1[r, ])
      * 100
    temp2[j] <- sum(booteigvals2[r, 1:j]) / sum(booteigvals2[r, ])
      * 100
    bootmaxval <- bootcommoneigvecs[, , r][maxpos[j], j]
    if((sign(bootmaxval) * sign(maxvals[j])) != 1){
      bootcommoneigvecs[, j, r] <- bootcommoneigvecs[, j, r] * (-1)
    }
  }
  booteigvalcumvars1[r, ] <- temp1
  booteigvalcumvars2[r, ] <- temp2
}

```

```

(apply(booteigvals1, 2, quantile, probs = c(0.84))
 - apply(booteigvals1, 2, quantile, probs = c(0.16))) / 2
# bootstrap standard errors of the South Africa eigenvalues
(apply(booteigvals2, 2, quantile, probs = c(0.84))
 - apply(booteigvals2, 2, quantile, probs = c(0.16))) / 2
# bootstrap standard errors of the Namibia eigenvalues

apply(booteigvals1, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence intervals for the South Africa
# eigenvalues
apply(booteigvalvars1, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence intervals for the variance accounted
# for per common principal component in the South Africa group
apply(booteigvalcumvars1, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence intervals for the cumulative variance
# accounted for per common principal component in the South
# Africa group
apply(booteigvals2, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence intervals for the Namibia eigenvalues
apply(booteigvalvars2, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence intervals for the variance accounted
# for per common principal component in the Namibia group
apply(booteigvalcumvars2, 2, quantile, probs = c(0.025, 0.975))
# 95% bootstrap confidence interval for the cumulative variance
# accounted for per common principal component in the Namibia
# group

lowerlims <- matrix(NA, nrow = p, ncol = p)
upperlims <- matrix(NA, nrow = p, ncol = p)
boot.se <- matrix(NA, nrow = p, ncol = p)
for(j in 1:p){
  for(h in 1:p){
    tempvec <- bootcommoneigvecs[j, h, ]
    confint <- quantile(tempvec, probs = c(0.025, 0.975))
    lowerlims[j, h] <- confint[1]
    upperlims[j, h] <- confint[2]
    boot.se[j, h] <- (quantile(tempvec, probs = 0.84)
      - quantile(tempvec, probs = 0.16)) / 2
  }
}
lowerlims # 95% bootstrap lower confidence limits for the common
# eigenvector loadings

```

```
upperlims # 95% bootstrap upper confidence limits for the common
# eigenvector loadings
boot.se # bootstrap standard errors of the common eigenvector
# loadings
```

B.2.6 Chapter 4

```
# Delivery mode groups
nvek <- c(nrow(caesarean), nrow(vaginal))
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(caesarean)
S[, , 2] <- cov(vaginal)

# Test for multivariate normality
mshapiro.test(t(caesarean))
mshapiro.test(t(vaginal))

# Test equality and check for proportionality of the covariance
# matrices
box.mtest(covmats = S, nvec = nvek)
# Box's M test for equality of covariance matrices
S[, , 1] / S[, , 2] # check for proportionality

# Find the most likely common eigenvector combinations
findcpc(covmats = S, cutoff = 0.65, plotting = TRUE,
  main = "Delivery mode: Vector correlations for the permutations")

# AIC and chi-square statistic methods
flury.test(covmats = S, nvec = nvek)

# Ensemble test
ensemble.test(origdata = list(caesarean, vaginal))

# Regional groups
nvek <- c(nrow(southafrica), nrow(namibia))
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(southafrica)
S[, , 2] <- cov(namibia)

# Test for multivariate normality
mshapiro.test(t(southafrica))
mshapiro.test(t(namibia))
```



```

# Test equality and check for proportionality of covariance
# matrices
box.mtest(covmats = S, nvec = nvek)
  # Box's M test for equality of covariance matrices
S[, , 1] / S[, , 2]  # check for proportionality

# Find the most likely common eigenvector combinations
findcpc(covmats = S, cutoff = 0.65, plotting = TRUE,
  main = "Regions: Vector correlations for the permutations")

# AIC and chi-square statistic methods
flury.test(covmats = S, nvec = nvek)

# Ensemble test
ensemble.test(origdata = list(southafrica,namibia))

```

B.2.7 Chapter 5

```

# Delivery mode groups
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(caesarean)
S[, , 2] <- cov(vaginal)
nvek <- c(nrow(caesarean), nrow(vaginal))

# Full CPC crossvalidation: Shrinkage intensity parameter estimates
B <- cpc::FG(covmats = S, nvec = nvek)$B
alpha.fullcpccross1 <- alpha.crossvalid(datamat = caesarean,
  B = B, reps = 100)
alpha.fullcpccross2 <- alpha.crossvalid(datamat = vaginal,
  B = B, reps = 100)
alpha.fullcpccross1  # Caesarean shrinkage intensity estimate
alpha.fullcpccross2  # Vaginal shrinkage intensity estimate

# CPC covariance matrix estimates
S.cpc <- array(NA, dim = c(6, 6, 2))
S.cpc[, , 1] <- B %%% diag(diag((t(B) %%% S[, , 1] %%% B))) %%%
  t(B)
S.cpc[, , 2] <- B %%% diag(diag((t(B) %%% S[, , 2] %%% B))) %%%
  t(B)
S.cpc  # CPC covariance matrix estimates

```

```

# Full CPC crossvalidation: covariance matrix estimates
S.fullcpccrossvalid <- array(NA, dim = c(6, 6, 2))
S.fullcpccrossvalid[, , 1] <- alpha.fullcpccross1 * S[, , 1] +
  (1 - alpha.fullcpccross1) * S.cpc[, , 1]
S.fullcpccrossvalid[, , 2] <- alpha.fullcpccross2 * S[, , 2] +
  (1 - alpha.fullcpccross2) * S.cpc[, , 2]
S.fullcpccrossvalid
# Full CPC crossvalid covariance matrix estimates

# Regional groups
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(southafrica)
S[, , 2] <- cov(namibia)
nvek <- c(nrow(southafrica), nrow(namibia))

# Full CPC crossvalidation: Shrinkage intensity parameter estimates
B <- cpc::FG(covmats = S, nvek = nvek)$B
alpha.fullcpccross1 <- alpha.crossvalid(datamat = southafrica,
  B = B, reps = 100)
alpha.fullcpccross2 <- alpha.crossvalid(datamat = namibia,
  B = B, reps = 100)
alpha.fullcpccross1 # South Africa shrinkage intensity estimate
alpha.fullcpccross2 # Namibia shrinkage intensity estimate

# CPC covariance matrix estimates
S.cpc <- array(NA, dim = c(6, 6, 2))
S.cpc[, , 1] <- B %%% diag(diag((t(B) %%% S[, , 1] %%% B))) %%%
  t(B)
S.cpc[, , 2] <- B %%% diag(diag((t(B) %%% S[, , 2] %%% B))) %%%
  t(B)
S.cpc # CPC covariance matrix estimates

# Full CPC crossvalidation: covariance matrix estimates
S.fullcpccrossvalid <- array(NA, dim = c(6, 6, 2))
S.fullcpccrossvalid[, , 1] <- alpha.fullcpccross1 * S[, , 1] +
  (1 - alpha.fullcpccross1) * S.cpc[, , 1]
S.fullcpccrossvalid[, , 2] <- alpha.fullcpccross2 * S[, , 2] +
  (1 - alpha.fullcpccross2) * S.cpc[, , 2]
S.fullcpccrossvalid
# Full CPC crossvalid covariance matrix estimates

```

B.2.8 Chapter 6

```

# Delivery mode groups
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(caesarean)
S[, , 2] <- cov(vaginal)
nvek <- c(nrow(caesarean), nrow(vaginal))

# Discriminant analysis
group.ind <- c(rep(1, times = nvek[1]), rep(2, times = nvek[2]))
discriminant.qda(origdata = rbind(caesarean, vaginal),
  group = group.ind, method = "unbiased") # QDA discrimination
discriminant.qda(origdata = rbind(caesarean, vaginal),
  group = group.ind, method = "cpc", B = delivery.cpc$B)
# CPC discrimination
discriminant.qda(origdata = rbind(caesarean, vaginal),
  group = group.ind, method = "fullcpccrossvalid",
  B = delivery.cpc$B) # CPC* discrimination
discriminant.qda(origdata = rbind(caesarean, vaginal),
  group = group.ind, method = "pooled") # LDA discrimination

# Regional groups
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(southafrica)
S[, , 2] <- cov(namibia)
nvek <- c(nrow(southafrica), nrow(namibia))

# Discriminant analysis
group.ind <- c(rep(1, times = nvek[1]), rep(2, times = nvek[2]))
discriminant.qda(origdata = rbind(southafrica, namibia),
  group = group.ind, method = "unbiased") # QDA discrimination
discriminant.qda(origdata = rbind(southafrica, namibia),
  group = group.ind, method = "cpc", B = region.cpc$B)
# CPC discrimination
discriminant.qda(origdata = rbind(southafrica, namibia),
  group = group.ind, method = "fullcpccrossvalid",
  B = region.cpc$B) # CPC* discrimination
discriminant.qda(origdata = rbind(southafrica, namibia),
  group = group.ind, method = "pooled") # LDA discrimination

# Mortality groups
survived <- von2009[((von2009[, "TRANSFERRED"] == 0) &
  (von2009[, "DIED"] == 0)), numvarnames]
died <- von2009[((von2009[, "TRANSFERRED"] == 0) &

```

```

    (von2009[, "DIED"] == 1)), numvarnames]
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(survived)
S[, , 2] <- cov(died)
S # covariance matrices of the mortality status groups
nvek <- c(nrow(survived), nrow(died))

# Test for multivariate normality
mshapiro.test(t(as.matrix(survived)))
mshapiro.test(t(as.matrix(died)))

# Test equality of the covariance matrices
box.mtest(covmats = S, nvec = nvek)

# AIC and chi-square statistic methods
flury.test(covmats = S, nvec = nvek)

# Ensemble test
ensemble.test(origdata = list(survived, died))

# Common eigenvectors estimated with the FG algorithm
survival.cpc <- cpc::FG(covmats = S, nvec = nvek)
survival.cpc$B
  # common eigenvectors estimated with the FG algorithm

# Discriminant analysis
group.ind <- c(rep(1, times = nvek[1]), rep(2, times = nvek[2]))
discriminant.qda(origdata = rbind(survived, died),
  group = group.ind, method = "unbiased") # QDA discrimination
discriminant.qda(origdata = rbind(survived, died),
  group = group.ind, method = "cpc", B = survival.cpc$B)
  # CPC discrimination
discriminant.qda(origdata = rbind(survived, died),
  group = group.ind, method = "fullcpccrossvalid",
  B = survival.cpc$B) # CPC* discrimination
discriminant.qda(origdata = rbind(survived, died),
  group = group.ind, method = "pooled") # LDA discrimination

```

B.2.9 Chapter 7

```

# Delivery mode groups
delivery.choice <- biplot.choice(list(caesarean, vaginal),

```

B.2. VON DATA ANALYSIS SCRIPT

345

```

    rdim = 2)
delivery.choice

# 2-dimensional pooled data biplot
B.pool <- eigen(cov(rbind(caesarean, vaginal)))$vectors
B.pool # eigenvectors of the covariance matrix of the pooled data
biplot(datalist = list(caesarean, vaginal), B = B.pool,
       D3 = FALSE, varex = 15, lwd = 2,
       main = "Pooled data biplot: VON 2009 delivery mode",
       col = c("black", "darkgray"))

# Measures for 3-dimensional biplot
delivery.choice <- biplot.choice(list(caesarean, vaginal),
                                rdim = 3)
delivery.choice

# Regional groups
region.choice <- biplot.choice(list(southafrica, namibia),
                                rdim = 2)
region.choice

# 2-dimensional pooled data biplot
B.pool <- eigen(cov(rbind(southafrica, namibia)))$vectors
B.pool # eigenvectors of the covariance matrix of the pooled data
biplot(datalist = list(southafrica, namibia), B = B.pool,
       D3 = FALSE, varex = 15, lwd = 2,
       main = "Pooled data biplot: VON 2009 regions",
       col = c("darkgray", "black"))

# Measures for 3-dimensional biplot
region.choice <- biplot.choice(list(southafrica, namibia),
                                rdim = 3)
region.choice

# Mortality groups
mortality.choice <- biplot.choice(list(survived, died), rdim = 2)
mortality.choice

# 2-dimensional pooled covariance matrix biplot
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(survived)
S[, , 2] <- cov(died)
nvek <- c(nrow(survived), nrow(died))

```

```

Sp <- (S[, , 1] * (nvek[1] - 1) + S[, , 2] * (nvek[2] - 1)) /
  (sum(nvek) - 2) # pooled covariance matrix
Bp <- eigen(Sp)$vectors
Bp # eigenvectors of the pooled covariance matrix
biplot(datalist = list(survived, died), B = Bp, D3 = FALSE,
  varex = 15, lwd = 2,
  main = "Pooled covariance matrix biplot: VON 2009 mortality",
  col = c("darkgray", "black"))

# Measures for 3-dimensional biplot
mortality.choice <- biplot.choice(list(survived, died), rdim = 3)
mortality.choice

```

B.2.10 Chapter 8

```

# Mortality
mortality2009 <- von2009[von2009[, "TRANSFERRED"] == 0,
  c(numvarnames, catvarnames, "DIED")]
died2009 <- mortality2009[mortality2009[, "DIED"] == 1, ]
survived2009 <- mortality2009[mortality2009[, "DIED"] == 0, ]
mortality2008 <- von2008[von2008[, "TRANSFERRED"] == 0,
  c(numvarnames, catvarnames, "DIED")]
died2008 <- mortality2008[mortality2008[, "DIED"] == 1, ]
survived2008 <- mortality2008[mortality2008[, "DIED"] == 0, ]

# Pearson correlations between BWGT and GESTAGE, BHEADCIR,
# respectively
cor(mortality2009[, "BWGT"], mortality2009[, "GESTAGE"])
cor(mortality2009[, "BWGT"], mortality2009[, "BHEADCIR"])

# Fit univariate GLMs with logit link to predict mortality

# BWGT
temp.model1 <- glm(DIED ~ BWGT, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for BWGT
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

```

B.2. VON DATA ANALYSIS SCRIPT

347

```
# AP1
temp.model1 <- glm(DIED ~ AP1, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for AP1
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# AP5
temp.model1 <- glm(DIED ~ AP5, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for AP5
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# GESTAGE
temp.model1 <- glm(DIED ~ GESTAGE, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2])
  # Odds ratio estimate for GESTAGE
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# BHEADCIR
temp.model1 <- glm(DIED ~ BHEADCIR, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2])
  # Odds ratio estimate for BHEADCIR
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# ATEMP
```

```
temp.model1 <- glm(DIED ~ ATEMP, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for ATEMP
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# REGION
temp.model1 <- glm(DIED ~ REGION, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2])
  # Odds ratio estimate for South Africa
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# MRACE
temp.model1 <- glm(DIED ~ MRACE, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for Black
exp(coefficients(temp.model1)[3]) # Odds ratio estimate for Other
exp(coefficients(temp.model1)[4]) # Odds ratio estimate for White
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# VAGDEL
temp.model1 <- glm(DIED ~ VAGDEL, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2])
  # Odds ratio estimate for Vaginal
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve
```


B.2. VON DATA ANALYSIS SCRIPT

349

```

# SEX
temp.model1 <- glm(DIED ~ SEX, family = binomial(logit),
  data = mortality2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Odds ratio estimate for Male
temp.model1.pr <- predict(temp.model1, type = "response",
  mortality2009)
performance(prediction(temp.model1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve

# Model M1: 3 numerical variables (chosen by inspection of
# eigenvectors), 1 nominal variable
mortality.M1 <- glm(DIED ~ AP1 + GESTAGE + VAGDEL,
  family = binomial(logit), data = mortality2009)
summary(mortality.M1)
exp(coefficients(mortality.M1)[2]) # Odds ratio estimate for AP1
exp(coefficients(mortality.M1)[3])
  # Odds ratio estimate for GESTAGE
exp(coefficients(mortality.M1)[4])
  # Odds ratio estimate for VAGDEL

# Model M2: PCs, 1 nominal variable
E <- eigen(cov(mortality2009[, numvarnames]))$vectors
temp <- as.matrix(mortality2009[, numvarnames]) %*% E
colnames(temp) <- paste('PC', 1:6, sep='')
mortality2009 <- cbind(mortality2009, temp)
mortality.M2 <- glm(DIED ~ PC1 + PC2 + VAGDEL,
  family = binomial(logit), data = mortality2009)
summary(mortality.M2)
exp(coefficients(mortality.M2)[2]) # Odds ratio estimate for PC1
exp(coefficients(mortality.M2)[3]) # Odds ratio estimate for PC2
exp(coefficients(mortality.M2)[4])
  # Odds ratio estimate for VAGDEL

# Model M3: CPCs, 1 nominal variable
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(survived2009[, numvarnames])
S[, , 2] <- cov(died2009[, numvarnames])
nvek <- c(nrow(survived2009), nrow(died2009))
B <- cpc::FG(covmats = S, nvec = nvek)$B
temp <- as.matrix(mortality2009[, numvarnames]) %*% B
colnames(temp) <- paste('CPC', 1:6, sep='')
mortality2009 <- cbind(mortality2009, temp)

```

```

mortality.M3 <- glm(DIED ~ CPC1 + CPC2 + VAGDEL,
  family = binomial(logit), data = mortality2009)
summary(mortality.M3)
exp(coefficients(mortality.M3)[2]) # Odds ratio estimate for CPC1
exp(coefficients(mortality.M3)[3]) # Odds ratio estimate for CPC2
exp(coefficients(mortality.M3)[4])
  # Odds ratio estimate for VAGDEL

# ROC curves for the mortality models
par(mfrow = c(2, 2))
mortality.M1.pr <- predict(mortality.M1, type = "response",
  mortality2009)
performance(prediction(mortality.M1.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve
rcorr.cens(x = mortality.M1.pr, S = mortality2009$DIED)["S.D."] / 2
  # Standard error of the AUC
mortality.M1.perf <- performance(prediction(mortality.M1.pr,
  mortality2009$DIED), "tpr", "fpr")
plot(mortality.M1.perf, lty = 1, main = "ROC curve for Model M1",
  xlab = "1 - Specificity", ylab = "Sensitivity")
abline(a = 0, b = 1, lty = 3)
text(x = c(0.85), y = c(0.3), labels = c("AUC: 0.8254"), cex = 1)
mortality.M2.pr <- predict(mortality.M2, type = "response",
  mortality2009)
performance(prediction(mortality.M2.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve
rcorr.cens(x = mortality.M2.pr, S = mortality2009$DIED)["S.D."] / 2
  # Standard error of the AUC
mortality.M2.perf <- performance(prediction(mortality.M2.pr,
  mortality2009$DIED), "tpr", "fpr")
plot(mortality.M2.perf, lty = 1, main = "ROC curve for Model M2",
  xlab = "1 - Specificity", ylab = "Sensitivity")
abline(a = 0, b = 1, lty = 3)
text(x = c(0.85), y = c(0.3), labels = c("AUC: 0.8338"), cex = 1)
mortality.M3.pr <- predict(mortality.M3, type = "response",
  mortality2009)
performance(prediction(mortality.M3.pr, mortality2009$DIED),
  "auc") # Area under the ROC curve
rcorr.cens(x = mortality.M3.pr, S = mortality2009$DIED)["S.D."] / 2
  # Standard error of the AUC
mortality.M3.perf <- performance(prediction(mortality.M3.pr,
  mortality2009$DIED), "tpr", "fpr")
plot(mortality.M3.perf, lty = 1, main = "ROC curve for Model M3",

```

B.2. VON DATA ANALYSIS SCRIPT

351

```

    xlab = "1 - Specificity", ylab = "Sensitivity")
abline(a = 0, b = 1, lty = 3)
text(x = c(0.85), y = c(0.3), labels = c("AUC: 0.8338"), cex = 1)

# Prediction for VON 2008 cohort
temp <- as.matrix(mortality2008[, numvarnames]) %*% E
colnames(temp) <- paste('PC', 1:6, sep='')
mortality2008 <- cbind(mortality2008, temp)
temp <- as.matrix(mortality2008[, numvarnames]) %*% B
colnames(temp) <- paste('CPC', 1:6, sep='')
mortality2008 <- cbind(mortality2008, temp)
temp <- predict(mortality.M1, newdata = mortality2008)
mortality2008.M1pred <- exp(temp) / (1 + exp(temp))
temp <- predict(mortality.M2, newdata = mortality2008)
mortality2008.M2pred <- exp(temp) / (1 + exp(temp))
temp <- predict(mortality.M3, newdata = mortality2008)
mortality2008.M3pred <- exp(temp) / (1 + exp(temp))

# Model M1
predgroup <- rep(0, times = nrow(mortality2008))
predgroup[mortality2008.M1pred >= 0.0971] <- 1
tempdata <- data.frame(Pred = as.factor(predgroup),
  Actual = as.factor(mortality2008$DIED))
.Table <- xtabs(~ Pred + Actual, data = tempdata)
(.Table[1, 2] + .Table[2, 1]) / sum(.Table)
  # Misclassification error rate
.Table[2, 2] / sum(.Table[, 2]) # Sensitivity
.Table[1, 1] / sum(.Table[, 1]) # Specificity
.Table[2, 2] / sum(.Table[2, ]) # Positive predictive value (PPV)
.Table[1, 1] / sum(.Table[1, ]) # Negative predictive value (NPV)

# Model M2
predgroup <- rep(0, times = nrow(mortality2008))
predgroup[mortality2008.M2pred >= 0.0618] <- 1
tempdata <- data.frame(Pred = as.factor(predgroup),
  Actual = as.factor(mortality2008$DIED))
.Table <- xtabs(~ Pred + Actual, data = tempdata)
(.Table[1, 2] + .Table[2, 1]) / sum(.Table)
  # Misclassification error rate
.Table[2, 2] / sum(.Table[, 2]) # Sensitivity
.Table[1, 1] / sum(.Table[, 1]) # Specificity
.Table[2, 2] / sum(.Table[2, ]) # Positive predictive value (PPV)
.Table[1, 1] / sum(.Table[1, ]) # Negative predictive value (NPV)

```

```

# Model M3
predgroup <- rep(0, times = nrow(mortality2008))
predgroup[mortality2008.M3pred >= 0.0617] <- 1
tempdata <- data.frame(Pred = as.factor(predgroup),
  Actual = as.factor(mortality2008$DIED))
.Table <- xtabs(~ Pred + Actual, data = tempdata)
(.Table[1, 2] + .Table[2, 1]) / sum(.Table)
# Misclassification error rate
.Table[2, 2] / sum(.Table[, 2]) # Sensitivity
.Table[1, 1] / sum(.Table[, 1]) # Specificity
.Table[2, 2] / sum(.Table[2, ]) # Positive predictive value (PPV)
.Table[1, 1] / sum(.Table[1, ]) # Negative predictive value (NPV)

# Length of stay (LOS)
los2009 <- von2009[((von2009[, "TRANSFERRED"] == 0) &
  (von2009[, "DIED"] == 0)), c(numvarnames, catvarnames, "LOS1")]
los2009 <- cbind(los2009, lnLOS = log(los2009[, "LOS1"]))
los2008 <- von2008[((von2008[, "TRANSFERRED"] == 0) &
  (von2008[, "DIED"] == 0)), c(numvarnames, catvarnames, "LOS1")]
los2008 <- cbind(los2008, lnLOS = log(los2008[, "LOS1"]))
southafrica2009 <- los2009[los2009[, "REGION"] == "RSA", ]
namibia2009 <- los2009[los2009[, "REGION"] == "Namibia", ]

# Distribution of LOS and ln(LOS)
par(mfrow = c(2, 1))
temp <- seq(from = min(los2009[, "LOS1"]),
  to = max(los2009[, "LOS1"]), by = 1)
ntemp <- length(temp)
freqvals <- rep(NA, times = ntemp)
for(i in 1:ntemp){
  freqvals[i] <- nrow(los2009[los2009[, "LOS1"] == temp[i], ])
}
barplot(height = freqvals, names.arg = temp, space = 1,
  main = "VON 2009: LOS")
plot(density(los2009[, "lnLOS"]), lwd = 3, xlab = "",
  main = "VON 2009: ln(LOS)")

# Scatterplot matrix of ln(LOS) and the numerical variables
temp <- los2009[, c("lnLOS", numvarnames)]
par(pch = 20)
pairs(temp)

```

B.2. VON DATA ANALYSIS SCRIPT

353

```
# Fit univariate GLM models on VON 2009 data to predict LOS

# BWGT
temp.model1 <- glm(LOS1 ~ BWGT, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of BWGT
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# AP1
temp.model1 <- glm(LOS1 ~ AP1, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of AP1
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# AP5
temp.model1 <- glm(LOS1 ~ AP5, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of AP5
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# GESTAGE
temp.model1 <- glm(LOS1 ~ GESTAGE, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of GESTAGE
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# BHEADCIR
temp.model1 <- glm(LOS1 ~ BHEADCIR, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of BHEADCIR
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# ATEMP
```

```

temp.model1 <- glm(LOS1 ~ ATEMP, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of ATEMP
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# REGION
temp.model1 <- glm(LOS1 ~ REGION, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of South Africa
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# MRACE
temp.model1 <- glm(LOS1 ~ MRACE, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of Black
exp(coefficients(temp.model1)[3]) # Effect of Other
exp(coefficients(temp.model1)[4]) # Effect of White
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# VAGDEL
temp.model1 <- glm(LOS1 ~ VAGDEL, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of Vaginal
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# SEX
temp.model1 <- glm(LOS1 ~ SEX, family = quasipoisson(log),
  data = los2009)
summary(temp.model1)
exp(coefficients(temp.model1)[2]) # Effect of Male
(cor(fitted.values(temp.model1), los2009[, "LOS1"]))^2
  # Coefficient of determination

# Spearman rank correlations of numerical variables with ln(LOS)
cor(los2009[,c("lnLOS", numvarnames)], method = "spearman")

```

```

# Model L1: 3 numerical variables (chosen by inspection of
# eigenvectors), 3 nominal variables
los.L1 <- glm(LOS1 ~ AP5 + GESTAGE + REGION + VAGDEL,
  family = quasipoisson(log), data = los2009)
summary(los.L1)
R2 <- (cor(fitted.values(los.L1), los2009[, "LOS1"]))^2
R2 # Coefficient of determination
p <- length(coefficients(los.L1))
1 - (1 - R2) * ((nrow(los2009) - 1) / (nrow(los2009) - p))
# Adjusted coefficient of determination
oldpar <- par(oma = c(0, 0, 3, 0), mfrow = c(2, 2))
plot(los.L1) # diagnostic plots
par(oldpar)

# Model L2: 3 PCs, 4 nominal variables
E <- eigen(cov(los2009[, numvarnames]))$vectors
temp <- as.matrix(los2009[, numvarnames]) %*% E
colnames(temp) <- paste('PC', 1:6, sep = '')
los2009 <- cbind(los2009, temp)
los.L2 <- glm(LOS1 ~ PC1 + PC3 + PC6 + REGION*PC3 + VAGDEL,
  family = quasipoisson(log), data = los2009)
summary(los.L2)
R2 <- (cor(fitted.values(los.L2), los2009[, "LOS1"]))^2
R2 # Coefficient of determination
p <- length(coefficients(los.L2))
1 - (1 - R2) * ((nrow(los2009) - 1) / (nrow(los2009) - p))
# Adjusted coefficient of determination
oldpar <- par(oma = c(0, 0, 3, 0), mfrow = c(2, 2))
plot(los.L2) # diagnostic plots
par(oldpar)

# Model L3: CPCs, 3 nominal variables
S <- array(NA, dim = c(6, 6, 2))
S[, , 1] <- cov(southafrica2009[, numvarnames])
S[, , 2] <- cov(namibia2009[, numvarnames])
nvek <- c(nrow(southafrica2009), nrow(namibia2009))
B <- cpc::FG(covmats = S, nvec = nvek)$B
temp <- as.matrix(los2009[, numvarnames]) %*% B
colnames(temp) <- paste('CPC', 1:6, sep = '')
los2009 <- cbind(los2009, temp)
los.L3 <- glm(LOS1 ~ CPC1 + CPC3 + CPC6 + REGION*CPC3 + VAGDEL,
  family = quasipoisson(log), data = los2009)

```

```

summary(los.L3)
R2 <- (cor(fitted.values(los.L3), los2009[, "LOS1"]))^2
R2 # Coefficient of determination
p <- length(coefficients(los.L3))
1 - (1 - R2) * ((nrow(los2009) - 1) / (nrow(los2009) - p))
# Adjusted coefficient of determination
oldpar <- par(oma = c(0, 0, 3, 0), mfrow = c(2, 2))
plot(los.L3) # diagnostic plots
par(oldpar)

# Model L4: PLSs, 3 nominal variables
B.pls <- pls.est(X = los2009[, numvarnames],
  Y = los2009[, "lnLOS"])$pls.loadings
B.pls # PLS loadings matrix
los2009.numvarmeans <- apply(los2009[, numvarnames], 2, mean)
los2009.numvarsd <- apply(los2009[, numvarnames], 2, sd)
temp <- as.matrix(t((t(los2009[, numvarnames]) -
  los2009.numvarmeans) / los2009.numvarsd)) %*% B.pls
colnames(temp) <- paste("PLS", 1:6, sep = "")
los2009 <- cbind(los2009, temp)
los.L4 <- glm(LOS1 ~ PLS1 + PLS2 + PLS4 + PLS5 + PLS6 +
  REGION*PLS4 + REGION*PLS5 + VAGDEL*PLS2,
  family = quasipoisson(log), data = los2009)
summary(los.L4)
R2 <- (cor(fitted.values(los.L4), los2009[, "LOS1"]))^2
R2 # Coefficient of determination
p <- length(coefficients(los.L4))
1 - (1 - R2) * ((nrow(los2009) - 1) / (nrow(los2009) - p))
# Adjusted coefficient of determination
oldpar <- par(oma = c(0, 0, 3, 0), mfrow = c(2, 2))
plot(los.L4) # diagnostic plots
par(oldpar)

# Prediction for VON 2008 cohort
temp <- as.matrix(los2008[, numvarnames]) %*% E
colnames(temp) <- paste('PC', 1:6, sep = '')
los2008 <- cbind(los2008, temp)
temp <- as.matrix(los2008[, numvarnames]) %*% B
colnames(temp) <- paste('CPC', 1:6, sep = '')
los2008 <- cbind(los2008, temp)
temp <- as.matrix(t((t(los2008[, numvarnames]) -
  los2009.numvarmeans) / los2009.numvarsd)) %*% B.pls
colnames(temp) <- paste('PLS', 1:6, sep = '')

```



```
los2008 <- cbind(los2008, temp)
par(mfrow = c(2, 2), oma = c(0, 0, 2, 0), mar = c(4, 4, 4, 2)
    + 0.1, pch = 20)
temp <- exp(predict(los.L1, newdata = los2008))
(cor(temp, los2008[, "LOS1"]))^2 # Predicted R-squared
plot(x = temp, y = los2008[, "LOS1"], type = "p",
     main = "Model L1", xlab = "Predicted LOS (days)",
     ylab = "Actual LOS (days)")
abline(a = 0, b = 1, lty = 1)
temp <- exp(predict(los.L2, newdata = los2008))
(cor(temp, los2008[, "LOS1"]))^2 # Predicted R-squared
plot(x = temp, y = los2008[, "LOS1"], type = "p",
     main = "Model L2", xlab = "Predicted LOS (days)",
     ylab = "Actual LOS (days)")
abline(a = 0, b = 1, lty = 1)
temp <- exp(predict(los.L3, newdata = los2008))
(cor(temp, los2008[, "LOS1"]))^2 # Predicted R-squared
plot(x = temp, y = los2008[, "LOS1"], type = "p",
     main = "Model L3", xlab = "Predicted LOS (days)",
     ylab = "Actual LOS (days)")
abline(a = 0, b = 1, lty = 1)
temp <- exp(predict(los.L4, newdata = los2008))
(cor(temp, los2008[, "LOS1"]))^2 # Predicted R-squared
plot(x = temp, y = los2008[, "LOS1"], type = "p",
     main = "Model L4", xlab = "Predicted LOS (days)",
     ylab = "Actual LOS (days)")
abline(a = 0, b = 1, lty = 1)
title(main = "VON 2008 cohort", outer = TRUE)
```

Appendix C

Chapter 5 simulation study: Eigenvalues

The sets of population eigenvalues used for the simulation study in Chapter 5 are given below, for each eigenvalue pattern and degree of separation.

a) **Same pattern:**

- $p = 5$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{2.0, 1.7, 1.4, 1.2, 1.0\}$$

- $p = 5$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{4.9, 3.3, 2.2, 1.5, 1.0\}$$

- $p = 5$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{12.9, 6.8, 3.6, 1.9, 1.0\}$$

- $p = 10$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{5.0, 4.2, 3.5, 2.9, 2.4, 2.0, 1.7, 1.4, 1.2, 1.0\}.$$

- $p = 10$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{37.3, 24.9, 16.6, 11.1, 7.4, 4.9, 3.3, 2.2, 1.5, 1.0\}.$$

- $p = 10$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{196.7, 109.3, 60.7, 33.7, 18.7, 10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{318.8, 167.8, 88.3, 46.5, 24.5, 12.9, 6.8, 3.6, 1.9, 1.0\}.$$

- $p = 20$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{6.1, 5.5, 5.0, 4.5, 4.1, 3.7, 3.4, 3.1, 2.8, 2.5, 2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{31.0, 25.8, 21.5, 17.9, 14.9, 12.4, 10.3, 8.6, 7.2, 6.0, 5.0, 4.2, 3.5, 2.9, 2.4, 2.0, 1.7, 1.4, 1.2, 1.0\}.$$

- $p = 20$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{609.1, 435.1, 310.8, 222.0, 158.6, 113.3, 80.9, 57.8, 41.3, 29.5, 21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{2148.3, 1432.2, 954.8, 636.5, 424.3, 282.9, 188.6, 125.7, 83.8, 55.9, 37.3, 24.9, 16.6, 11.1, 7.4, 4.9, 3.3, 2.2, 1.5, 1.0\}.$$

- $p = 20$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{70239.2, 39021.8, 21678.8, 12043.8, 6691.0, 3717.2, 2065.1, 1147.3, 637.4, 354.1, 196.7, 109.3, 60.7, 33.7, 18.7, 10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{195443.5, 102865.0, 54139.5, 28494.5, 14997.1, 7893.2, 4154.3, 2186.5, 1150.8, 605.7, 318.8, 167.8, 88.3, 46.5, 24.5, 12.9, 6.8, 3.6, 1.9, 1.0\}.$$

b) **Similar pattern:**

- $p = 5$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.7, 2.0, 1.4, 1.0, 1.2\}$$

- $p = 5$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{3.3, 4.9, 2.2, 1.0, 1.5\}$$

- $p = 5$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{6.8, 12.9, 3.6, 1.0, 1.9\}$$

- $p = 10$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{3.5, 4.2, 5.0, 2.9, 2.4, 2.0, 1.7, 1.0, 1.2, 1.4\}$$

- $p = 10$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{16.6, 24.9, 37.3, 11.1, 7.4, 4.9, 3.3, 1.0, 1.5, 2.2\}$$

- $p = 10$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{196.7, 109.3, 60.7, 33.7, 18.7, 10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{88.3, 167.8, 318.8, 46.5, 24.5, 12.9, 6.8, 1.0, 1.9, 3.6\}$$

- $p = 20$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{6.1, 5.5, 5.0, 4.5, 4.1, 3.7, 3.4, 3.1, 2.8, 2.5, 2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{17.9, 21.5, 25.8, 31.0, 14.9, 12.4, 10.3, 8.6, 7.2, 6.0, 5.0, 4.2, 3.5, 2.9, 2.4, 2.0, 1.0, 1.2, 1.4, 1.7\}.$$

- $p = 20$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{609.1, 435.1, 310.8, 222.0, 158.6, 113.3, 80.9, 57.8, 41.3, 29.5, 21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{636.5, 954.8, 1432.2, 2148.3, 424.3, 282.9, 188.6, 125.7, 83.8, 55.9, 37.3, 24.9, 16.6, 11.1, 7.4, 4.9, 1.0, 1.5, 2.2, 3.3\}.$$

- $p = 20$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{70239.2, 39021.8, 21678.8, 12043.8, 6691.0, 3717.2, 2065.1, 1147.3, 637.4, 354.1, 196.7, 109.3, 60.7, 33.7, 18.7, 10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{28494.5, 54139.5, 102865.0, 195443.5, 14997.1, 7893.2, 4154.3, 2186.5, 1150.8, 605.7, 318.8, 167.8, 88.3, 46.5, 24.5, 12.9, 1.0, 1.9, 3.6, 6.8\}.$$

c) **Opposite pattern:**

- $p = 5$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.2, 1.4, 1.7, 2.0\}$$

- $p = 5$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.5, 2.2, 3.3, 4.9\}$$

- $p = 5$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.9, 3.6, 6.8, 12.9\}$$

- $p = 10$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.2, 1.4, 1.7, 2.0, 2.4, 2.9, 3.5, 4.2, 5.0\}.$$

- $p = 10$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.5, 2.2, 3.3, 4.9, 7.4, 11.1, 16.6, 24.9, 37.3\}.$$

- $p = 10$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{196.7, 109.3, 60.7, 33.7, 18.7, 10.4, 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.9, 3.6, 6.8, 12.9, 24.5, 46.5, 88.3, 167.8, 318.8\}.$$

- $p = 20$ (poor separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{6.1, 5.5, 5.0, 4.5, 4.1, 3.7, 3.4, 3.1, 2.8, 2.5, 2.3, 2.1, 1.9, 1.7, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.2, 1.4, 1.7, 2.0, 2.4, 2.9, 3.5, 4.2, 5.0, 6.0, 7.2, 8.6, 10.3, 12.4, 14.9, 17.9, 21.5, 25.8, 31.0\}.$$

- $p = 20$ (good separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{609.1, 435.1, 310.8, 222.0, 158.6, 113.3, 80.9, 57.8, 41.3, 29.5, 21.1, 15.1, 10.8, 7.7, 5.5, 3.9, 2.8, 2.0, 1.4, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.5, 2.2, 3.3, 4.9, 7.4, 11.1, 16.6, 24.9, 37.3, 55.9, 83.8, 125.7, 188.6, 282.9, 424.3, 636.5, 954.8, 1432.2, 2148.3\}.$$

- $p = 20$ (excellent separation):

$$\text{diag}(\mathbf{\Lambda}_1) = \{70239.2, 39021.8, 21678.8, 12043.8, 6691.0, 3717.2, 2065.1, \\ 1147.3, 637.4, 354.1, 196.7, 109.3, 60.7, 33.7, 18.7, 10.4, \\ 5.8, 3.2, 1.8, 1.0\}$$

$$\text{diag}(\mathbf{\Lambda}_2) = \{1.0, 1.9, 3.6, 6.8, 12.9, 24.5, 46.5, 88.3, 167.8, 318.8, 605.7, \\ 1150.8, 2186.5, 4154.3, 7893.2, 14997.1, 28494.5, 54139.5, \\ 102865.0, 195443.5\}.$$

Appendix D

Chapter 6 simulation study: Covariance matrices

The covariance matrices used for the simulation study presented in Chapter 6 are given below.

For $k = 2$ multivariate normally distributed populations with $p = 2$ variables, the following four sets of population covariance matrices were used:

- 1) **Equal covariance matrices** (Σ_{EQUAL})

$$\Sigma_1 = \Sigma_2 = \begin{bmatrix} 15.90 & 0.87 \\ 0.87 & 8.10 \end{bmatrix}$$

- 2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME})

$$\Sigma_1 = \begin{bmatrix} 14.20 & 3.34 \\ 3.34 & 9.80 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 22.20 & 3.34 \\ 3.34 & 17.80 \end{bmatrix}$$

- 3) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE})

$$\Sigma_1 = \begin{bmatrix} 21.12 & 2.68 \\ 2.68 & 13.88 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 17.08 & -3.27 \\ -3.27 & 25.92 \end{bmatrix}$$

- 4) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

$$\Sigma_1 = \begin{bmatrix} 14.97 & 3.72 \\ 3.72 & 20.03 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 17.17 & -3.40 \\ -3.40 & 25.83 \end{bmatrix}$$

For $k = 2$ multivariate normally distributed populations with $p = 5$ variables, the following five sets of population covariance matrices were used:

1) **Equal covariance matrices** (Σ_{EQUAL})

$$\Sigma_1 = \Sigma_2 = \begin{bmatrix} 7.00 & 2.14 & 1.43 & 2.63 & 3.31 \\ 2.14 & 5.91 & -1.65 & 4.27 & 1.87 \\ 1.43 & -1.65 & 4.57 & -1.15 & 2.06 \\ 2.63 & 4.27 & -1.15 & 7.14 & 4.21 \\ 3.31 & 1.87 & 2.06 & 4.21 & 6.38 \end{bmatrix}.$$

2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME})

$$\Sigma_1 = \begin{bmatrix} 7.00 & 2.14 & 1.43 & 2.63 & 3.31 \\ 2.14 & 5.91 & -1.65 & 4.27 & 1.87 \\ 1.43 & -1.65 & 4.57 & -1.15 & 2.06 \\ 2.63 & 4.27 & -1.15 & 7.14 & 4.21 \\ 3.31 & 1.87 & 2.06 & 4.21 & 6.38 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 9.81 & 2.50 & 3.61 & 3.88 & 6.20 \\ 2.50 & 8.78 & -4.08 & 7.50 & 2.35 \\ 3.61 & -4.08 & 7.92 & -2.78 & 4.29 \\ 3.88 & 7.50 & -2.78 & 10.22 & 6.05 \\ 6.20 & 2.35 & 4.29 & 6.05 & 9.06 \end{bmatrix}.$$

3) **CPC: Similar rank orders of the common eigenvectors** (Σ_{SIMILAR})

$$\Sigma_1 = \begin{bmatrix} 12.13 & 6.51 & 4.37 & 0.81 & 5.92 \\ 6.51 & 11.63 & -1.42 & 0.62 & 2.57 \\ 4.37 & -1.42 & 8.86 & 4.25 & 3.59 \\ 0.81 & 0.62 & 4.25 & 7.79 & -1.28 \\ 5.92 & 2.57 & 3.59 & -1.28 & 4.68 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 8.33 & 5.36 & 3.17 & 0.36 & 4.47 \\ 5.37 & 13.40 & -7.78 & -4.39 & 1.54 \\ 3.17 & -7.78 & 14.39 & 8.98 & 2.79 \\ 0.36 & -4.39 & 8.99 & 9.21 & -0.32 \\ 4.47 & 1.54 & 2.79 & -0.32 & 3.17 \end{bmatrix}.$$

4) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE})

$$\Sigma_1 = \begin{bmatrix} 3.07 & 1.88 & 2.89 & 0.41 & 2.43 \\ 1.88 & 9.71 & 5.45 & -0.32 & 0.98 \\ 2.89 & 5.45 & 8.37 & -0.60 & 1.64 \\ 0.41 & -0.32 & -0.60 & 4.16 & 2.66 \\ 2.43 & 0.98 & 1.64 & 2.66 & 5.69 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 11.58 & -0.08 & -2.89 & 1.51 & -4.80 \\ -0.08 & 2.60 & -1.70 & -0.12 & 0.13 \\ -2.89 & -1.70 & 4.06 & 0.73 & 0.02 \\ 1.51 & -0.12 & 0.73 & 6.19 & -3.73 \\ -4.80 & 0.13 & 0.02 & -3.73 & 6.58 \end{bmatrix}.$$

5) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

$$\Sigma_1 = \begin{bmatrix} 7.21 & 1.18 & 1.78 & 1.01 & -0.65 \\ 1.18 & 4.27 & 0.70 & 1.24 & -0.05 \\ 1.78 & 0.70 & 5.69 & 4.01 & 4.66 \\ 1.01 & 1.24 & 4.01 & 6.68 & 5.05 \\ -0.65 & -0.05 & 4.66 & 5.05 & 7.16 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 5.11 & 2.79 & 6.86 & -0.33 & 2.91 \\ 2.79 & 12.22 & 4.94 & 9.47 & 0.15 \\ 6.86 & 4.94 & 9.99 & 0.29 & 3.30 \\ -0.33 & 9.47 & 0.29 & 12.79 & -1.12 \\ 2.91 & 0.15 & 3.30 & -1.12 & 5.69 \end{bmatrix}.$$

For $k = 2$ multivariate normally distributed populations with $p = 10$ variables, the following five sets of population covariance matrices were used:

1) **Equal covariance matrices** (Σ_{EQUAL})

$$\Sigma_1 = \Sigma_2 = \begin{bmatrix} 9.75 & 5.35 & 0.07 & 0.55 & 2.54 & 1.07 & 2.94 & 1.07 & 1.93 & 3.77 \\ 5.35 & 13.04 & 3.60 & 0.26 & 1.38 & -0.70 & 4.18 & 0.67 & 3.90 & 2.66 \\ 0.07 & 3.60 & 14.95 & -0.47 & 1.92 & -0.87 & 6.22 & 6.40 & 1.71 & 3.90 \\ 0.55 & 0.26 & -0.47 & 9.80 & 2.87 & 4.27 & 2.22 & 0.85 & 7.04 & 1.19 \\ 2.54 & 1.38 & 1.92 & 2.87 & 9.69 & 0.68 & 0.78 & 4.28 & 0.33 & -0.97 \\ 1.07 & -0.70 & -0.87 & 4.27 & 0.68 & 10.74 & 2.72 & 2.15 & 2.34 & -0.92 \\ 2.94 & 4.18 & 6.22 & 2.22 & 0.78 & 2.72 & 11.58 & 5.28 & 2.09 & 2.74 \\ 1.07 & 0.67 & 6.40 & 0.85 & 4.28 & 2.15 & 5.28 & 9.92 & -2.12 & 2.48 \\ 1.93 & 3.90 & 1.71 & 7.04 & 0.33 & 2.34 & 2.09 & -2.12 & 11.90 & 1.11 \\ 3.77 & 2.66 & 3.90 & 1.19 & -0.97 & -0.92 & 2.74 & 2.48 & 1.11 & 8.64 \end{bmatrix}.$$

2) **CPC: Same rank order of the common eigenvectors** (Σ_{SAME})

$$\Sigma_1 = \begin{bmatrix} 9.75 & 5.35 & 0.07 & 0.55 & 2.54 & 1.07 & 2.94 & 1.07 & 1.93 & 3.77 \\ 5.35 & 13.04 & 3.60 & 0.26 & 1.38 & -0.70 & 4.18 & 0.67 & 3.90 & 2.66 \\ 0.07 & 3.60 & 14.95 & -0.47 & 1.92 & -0.87 & 6.22 & 6.40 & 1.71 & 3.90 \\ 0.55 & 0.26 & -0.47 & 9.80 & 2.87 & 4.27 & 2.22 & 0.85 & 7.04 & 1.19 \\ 2.54 & 1.38 & 1.92 & 2.87 & 9.69 & 0.68 & 0.78 & 4.28 & 0.33 & -0.97 \\ 1.07 & -0.70 & -0.87 & 4.27 & 0.68 & 10.74 & 2.72 & 2.15 & 2.34 & -0.92 \\ 2.94 & 4.18 & 6.22 & 2.22 & 0.78 & 2.72 & 11.58 & 5.28 & 2.09 & 2.74 \\ 1.07 & 0.67 & 6.40 & 0.85 & 4.28 & 2.15 & 5.28 & 9.92 & -2.12 & 2.48 \\ 1.93 & 3.90 & 1.71 & 7.04 & 0.33 & 2.34 & 2.09 & -2.12 & 11.90 & 1.11 \\ 3.77 & 2.66 & 3.90 & 1.19 & -0.97 & -0.92 & 2.74 & 2.48 & 1.11 & 8.64 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 12.72 & 11.23 & -0.14 & 1.02 & 3.24 & 0.21 & 4.10 & 0.23 & 4.07 & 5.54 \\ 11.23 & 19.39 & 5.03 & -0.34 & 0.61 & -3.83 & 5.90 & -1.27 & 7.33 & 6.78 \\ -0.14 & 5.03 & 21.54 & -1.95 & 2.58 & -2.00 & 10.94 & 11.85 & -0.07 & 6.70 \\ 1.02 & -0.34 & -1.95 & 15.25 & 4.51 & 10.17 & 3.77 & 0.92 & 12.78 & -0.19 \\ 3.24 & 0.61 & 2.58 & 4.51 & 11.18 & 3.73 & 2.78 & 7.58 & -0.04 & -1.06 \\ 0.21 & -3.83 & -2.00 & 10.17 & 3.73 & 14.98 & 4.10 & 4.24 & 5.45 & -2.37 \\ 4.11 & 5.90 & 10.94 & 3.77 & 2.78 & 4.10 & 13.79 & 9.13 & 3.38 & 4.93 \\ 0.23 & -1.27 & 11.85 & 0.92 & 7.58 & 4.24 & 9.13 & 14.98 & -5.05 & 2.85 \\ 4.07 & 7.33 & -0.07 & 12.78 & -0.04 & 5.45 & 3.38 & -5.05 & 18.38 & 2.08 \\ 5.54 & 6.78 & 6.70 & -0.19 & -1.06 & -2.37 & 4.93 & 2.85 & 2.08 & 8.60 \end{bmatrix}$$

3) **CPC: Similar rank orders of the common eigenvectors** (Σ_{SIMILAR})

$$\Sigma_1 = \begin{bmatrix} 8.95 & -1.55 & 1.36 & 2.30 & 5.69 & 3.48 & 4.19 & 2.30 & 4.26 & 1.76 \\ -1.55 & 13.94 & -1.02 & 2.27 & 1.26 & -0.73 & 1.52 & -0.70 & 0.81 & 4.15 \\ 1.36 & -1.02 & 11.31 & 5.43 & 5.24 & 4.12 & 3.28 & 5.02 & -0.50 & -2.51 \\ 2.30 & 2.27 & 5.43 & 11.06 & 2.31 & 1.66 & 1.94 & 4.01 & 1.45 & 1.35 \\ 5.69 & 1.25 & 5.24 & 2.31 & 11.77 & 3.04 & 2.93 & 3.74 & 2.49 & 2.19 \\ 3.48 & -0.73 & 4.12 & 1.66 & 3.04 & 10.93 & 0.31 & 3.06 & 7.14 & 0.53 \\ 4.19 & 1.52 & 3.28 & 1.94 & 2.93 & 0.31 & 10.47 & 0.05 & 1.29 & -0.71 \\ 2.30 & -0.70 & 5.02 & 4.01 & 3.74 & 3.06 & 0.05 & 8.09 & 0.24 & 0.91 \\ 4.26 & 0.81 & -0.50 & 1.45 & 2.49 & 7.14 & 1.29 & 0.24 & 10.43 & 4.07 \\ 1.77 & 4.15 & -2.51 & 1.35 & 2.19 & 0.53 & -0.71 & 0.91 & 4.07 & 13.05 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 9.77 & -5.31 & -0.52 & -0.99 & 5.81 & 6.14 & 3.97 & 0.98 & 7.92 & 1.82 \\ -5.31 & 28.64 & -0.10 & 9.34 & 2.44 & -8.83 & 4.72 & -0.35 & -4.49 & 10.69 \\ -0.52 & -0.10 & 15.59 & 9.47 & 5.81 & 0.84 & 4.93 & 7.72 & -6.45 & -7.19 \\ -0.99 & 9.34 & 9.47 & 14.26 & 3.67 & -2.57 & 3.44 & 6.19 & -4.42 & 0.99 \\ 5.81 & 2.44 & 5.81 & 3.67 & 10.44 & 2.07 & 4.62 & 3.92 & 1.57 & 2.12 \\ 6.14 & -8.83 & 0.84 & -2.57 & 2.07 & 15.45 & -2.89 & 2.26 & 13.23 & 0.92 \\ 3.97 & 4.72 & 4.93 & 3.44 & 4.62 & -2.89 & 12.35 & -0.11 & -1.98 & -1.97 \\ 0.98 & -0.35 & 7.72 & 6.19 & 3.92 & 2.26 & -0.11 & 7.79 & -2.02 & -0.73 \\ 7.92 & -4.49 & -6.46 & -4.42 & 1.57 & 13.23 & -1.98 & -2.02 & 17.54 & 7.99 \\ 1.82 & 10.69 & -7.19 & 0.99 & 2.12 & 0.92 & -1.97 & -0.73 & 7.99 & 18.97 \end{bmatrix}$$

4) **CPC: Opposite rank orders of the common eigenvectors** (Σ_{OPPOSITE})

$$\Sigma_1 = \begin{bmatrix} 12.82 & 2.71 & 1.07 & 0.93 & 4.69 & 3.71 & 0.76 & 1.94 & 4.00 & 0.25 \\ 2.71 & 12.80 & 1.97 & 4.33 & 6.05 & 3.29 & -1.64 & 0.82 & 1.29 & -2.78 \\ 1.07 & 1.97 & 5.18 & 3.41 & 1.69 & 1.84 & 2.60 & 1.16 & 1.06 & 4.24 \\ 0.93 & 4.33 & 3.41 & 7.42 & 4.03 & 2.34 & 1.88 & 2.71 & 1.56 & 2.51 \\ 4.69 & 6.05 & 1.68 & 4.03 & 11.40 & 0.68 & 1.50 & 0.64 & 2.72 & 2.56 \\ 3.71 & 3.29 & 1.84 & 2.34 & 0.68 & 8.58 & 3.19 & 4.57 & -0.67 & 1.88 \\ 0.76 & -1.64 & 2.60 & 1.88 & 1.50 & 3.19 & 11.42 & 7.83 & -3.20 & 8.64 \\ 1.94 & 0.82 & 1.16 & 2.71 & 0.64 & 4.57 & 7.83 & 10.03 & -2.66 & 2.81 \\ 4.00 & 1.29 & 1.06 & 1.56 & 2.72 & -0.67 & -3.20 & -2.66 & 12.86 & 0.79 \\ 0.25 & -2.78 & 4.24 & 2.51 & 2.56 & 1.88 & 8.64 & 2.81 & 0.79 & 16.49 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 6.32 & 0.96 & -1.44 & 2.37 & -3.34 & -3.17 & 0.07 & -1.07 & -1.95 & 1.09 \\ 0.96 & 8.11 & -2.50 & -1.56 & -4.42 & -3.28 & 1.71 & -0.32 & 0.28 & 2.36 \\ -1.44 & -2.50 & 17.33 & -6.64 & 2.61 & -0.28 & -2.16 & 2.41 & -0.28 & -3.68 \\ 2.37 & -1.56 & -6.64 & 13.40 & -3.50 & -1.20 & 1.53 & -3.82 & -1.41 & -0.10 \\ -3.34 & -4.42 & 2.61 & -3.50 & 9.13 & 3.14 & -2.02 & 1.36 & -0.15 & -1.98 \\ -3.17 & -3.28 & -0.28 & -1.20 & 3.14 & 10.33 & 0.31 & -3.36 & 0.84 & -1.69 \\ 0.07 & 1.71 & -2.16 & 1.53 & -2.02 & 0.31 & 17.27 & -11.01 & 2.59 & -6.53 \\ -1.07 & -0.32 & 2.41 & -3.82 & 1.36 & -3.36 & -11.01 & 14.26 & 0.20 & 3.17 \\ -1.95 & 0.28 & -0.28 & -1.41 & -0.15 & 0.84 & 2.59 & 0.20 & 5.37 & -1.38 \\ 1.09 & 2.36 & -3.68 & -0.10 & -1.98 & -1.69 & -6.53 & 3.17 & -1.38 & 7.47 \end{bmatrix}.$$

5) **Unrelated covariance matrices** ($\Sigma_{\text{UNRELATED}}$)

$$\Sigma_1 = \begin{bmatrix} 6.19 & 2.35 & -0.76 & 2.34 & 3.18 & 2.97 & 1.81 & 1.20 & 2.73 & -0.82 \\ 2.35 & 6.21 & 0.48 & 2.32 & 1.51 & 1.38 & 3.95 & -0.46 & 4.31 & 0.82 \\ -0.76 & 0.48 & 5.98 & -0.10 & -0.34 & 0.23 & 2.20 & 2.56 & -1.39 & 1.00 \\ 2.34 & 2.32 & -0.10 & 3.75 & 0.03 & 0.60 & 0.75 & 0.21 & 1.62 & 1.40 \\ 3.18 & 1.51 & -0.34 & 0.03 & 6.89 & 3.69 & 1.07 & 1.55 & 4.17 & -0.15 \\ 2.97 & 1.38 & 0.23 & 0.60 & 3.69 & 6.14 & 0.49 & 3.49 & 1.76 & -0.02 \\ 1.81 & 3.95 & 2.20 & 0.75 & 1.07 & 0.49 & 10.65 & -3.07 & 1.36 & 0.14 \\ 1.20 & -0.46 & 2.56 & 0.21 & 1.55 & 3.49 & -3.07 & 9.59 & -1.02 & 0.59 \\ 2.73 & 4.31 & -1.39 & 1.62 & 4.17 & 1.76 & 1.36 & -1.02 & 8.79 & 1.07 \\ -0.82 & 0.82 & 1.00 & 1.40 & -0.15 & -0.02 & 0.14 & 0.59 & 1.07 & 5.62 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 10.89 & 5.23 & 7.56 & 11.23 & -2.78 & 2.50 & -1.29 & 2.14 & 0.19 & 0.84 \\ 5.23 & 12.66 & 1.12 & 9.43 & 0.20 & 0.85 & 5.51 & 2.39 & -0.23 & 2.38 \\ 7.56 & 1.12 & 22.15 & 6.96 & -5.91 & 7.38 & 0.79 & 7.60 & 5.92 & 2.52 \\ 11.23 & 9.43 & 6.96 & 22.06 & 5.87 & 7.04 & 2.96 & -0.82 & -1.19 & 7.93 \\ -2.78 & 0.20 & -5.91 & 5.87 & 41.71 & 6.03 & 10.24 & 0.72 & 5.89 & 22.49 \\ 2.50 & 0.85 & 7.38 & 7.04 & 6.03 & 9.52 & 3.53 & -0.29 & 4.18 & 6.27 \\ -1.29 & 5.51 & 0.79 & 2.96 & 10.24 & 3.53 & 14.46 & 4.70 & 5.90 & 11.44 \\ 2.14 & 2.39 & 7.60 & -0.82 & 0.72 & -0.29 & 4.70 & 14.89 & 5.40 & 7.47 \\ 0.19 & -0.23 & 5.92 & -1.19 & 5.89 & 4.18 & 5.90 & 5.40 & 11.78 & 5.62 \\ 0.84 & 2.38 & 2.52 & 7.93 & 22.49 & 6.27 & 11.44 & 7.47 & 5.62 & 21.48 \end{bmatrix}.$$

References

- Ackermann, R. R. and Cheverud, J. M. (2000). Phenotypic covariance structure in tamarins (genus *Saguinus*): A comparison of variation patterns using matrix correlation and common principal component analysis. *American Journal of Physical Anthropology*, 111(4):489–501.
- Agresti, A. (2003). *Categorical Data Analysis*. Wiley.
- Airoidi, J. P. and Flury, B. K. (1988). An application of common principal component analysis to cranial morphometry of *Microtus californicus* and *M. ochrogaster* (Mammalia, Rodentia). *Journal of Zoology*, 216:21–36.
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.
- Altman, D. G. (1990). *Practical Statistics for Medical Research*. Chapman and Hall/CRC.
- Altman, M., Vanpée, M., Cnattingius, S., and Norman, M. (2009). Moderately preterm infants and determinants of length of hospital stay. *Archives of Disease in Childhood – Fetal and Neonatal Edition*, 94(6):F414–F418.
- Anderson, E. (1935). The irises of the Gaspé peninsula. *Bulletin of the American Iris Society*, 59:2–5.
- Anderson, T. W. (1963). Asymptotic theory for principal component analysis. *The Annals of Mathematical Statistics*, 34(1):122–148.
- Anderson, T. W. (2003). *An Introduction to Multivariate Statistical Analysis*. Wiley.
- Arnold, S. J. and Phillips, P. C. (1999). Hierarchical comparison of genetic variance-covariance matrices. II. Coastal-inland divergence in the garter snake, *Thamnophis elegans*. *Evolution*, 53(5):1516–1527.

- Bartlett, M. S. (1951). The effect of standardization on a χ^2 approximation in factor analysis. *Biometrika*, 38(3/4):337–344.
- Bartolotti, S., Flury, B. D., and Nel, D. G. (1999). Allometric extension. *Biometrics*, 55(4):1210–1214.
- Berner, D. (2011). Size correction in biology: How reliable are approaches based on (common) principal component analysis? *Oecologia*, 166(4):961–971.
- Bianco, A., Boente, G., Pires, A. M., and Rodrigues, I. M. (2008). Robust discrimination under a hierarchy on the scatter matrices. *Journal of Multivariate Analysis*, 99(6):1332–1357.
- Bien, J. and Tibshirani, R. J. (2011). Sparse estimation of a covariance matrix. *Biometrika*, 98(4):807–820.
- Boente, G., Molina, J., and Sued, M. (2010a). On the asymptotic behavior of general projection-pursuit estimators under the common principal components model. *Statistics and Probability Letters*, 80(3):228–235.
- Boente, G. and Orellana, L. (2001). A robust approach to common principal components. In *Statistics in Genetics and in the Environmental Sciences*, Trends in Mathematics, pages 117–145. Birkhäuser Basel.
- Boente, G. and Orellana, L. (2004). Robust plug-in estimators in proportional scatter models. *Journal of Statistical Planning and Inference*, 122:95–110.
- Boente, G., Pires, A. M., and Rodrigues, I. M. (2002). Influence functions and outlier detection under the common principal components model: A robust approach. *Biometrika*, 89(4):861–875.
- Boente, G., Pires, A. M., and Rodrigues, I. M. (2006). General projection-pursuit estimators for the common principal components model: Influence functions and Monte Carlo study. *Journal of Multivariate Analysis*, 97(1):124–147.
- Boente, G., Pires, A. M., and Rodrigues, I. M. (2008). Estimators for the common principal components model based on reweighting: Influence functions and Monte Carlo study. *Metrika*, 67(2):189–218.
- Boente, G., Pires, A. M., and Rodrigues, I. M. (2009). Robust tests for the common principal components model. *Journal of Statistical Planning and Inference*, 139(4):1332–1347.

- Boente, G., Pires, A. M., and Rodrigues, I. M. (2010b). Detecting influential observations in principal components and common principal components. *Computational Statistics and Data Analysis*, 54(12):2967–2975.
- Boente, G., Rodriguez, D., and Sued, M. (2010c). Inference under functional proportional and common principal component models. *Journal of Multivariate Analysis*, 101(2):464–475.
- Boik, R. J. (2002). Spectral models for covariance matrices. *Biometrika*, 89(1):159–182.
- Box, G. E. P. (1949). A general distribution theory for a class of likelihood criteria. *Biometrika*, 36(3/4):317–346.
- Box, G. E. P. (1950). Problems in the analysis of growth and wear curves. *Biometrics*, 6(4):362–389.
- Box, G. E. P. and Draper, N. R. (1987). *Empirical Model-building and Response Surfaces*. Wiley.
- Brand, H. (2013). *PCA and CVA Biplots: A Study of Their Underlying Theory and Quality Measures*. Master's thesis, Stellenbosch University.
- Browne, M. W. (1968). A comparison of factor analytic techniques. *Psychometrika*, 33(3):267–334.
- Browne, R. P. and McNicholas, P. D. (2014a). Estimating common principal components in high dimensions. *Advances in Data Analysis and Classification*, 8(2):217–226.
- Browne, R. P. and McNicholas, P. D. (2014b). Orthogonal Stiefel manifold optimization for eigen-decomposed covariance parameter estimation in mixture models. *Statistics and Computing*, 24(2):203–210.
- Cardoso, J.-F. and Souloumiac, A. (1996). Jacobi angles for simultaneous diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 17(1):161–164.
- Cattell, R. B. and Jaspers, J. (1967). *A General Plasmode (No. 30-10-5-2) for Factor Analytic Exercises and Research*. Society of Multivariate Experimental Psychology.
- Cheverud, J. M. and Marroig, G. (2007). Comparing covariance matrices: Random skewers method compared to the common principal components model. *Genetics and Molecular Biology*, 30(2):461–469.

- Coffey, N., Harrison, A. J., Donoghue, O. A., and Hayes, K. (2011). Common functional principal components analysis: A new approach to analyzing human movement data. *Human Movement Science*, 30(6):1144–1166.
- Cox, T. F. and Cox, A. A. (2010). *Multidimensional Scaling*. Chapman and Hall/CRC.
- Daniels, M. J. and Kass, R. E. (2001). Shrinkage estimators for covariance matrices. *Biometrics*, 57(4):1173–1184.
- Devlin, S. J., Gnanadesikan, R., and Kettenring, J. R. (1975). Robust estimation and outlier detection with correlation coefficients. *Biometrika*, 62(3):531–545.
- Diaconis, P. and Efron, B. (1983). Computer-intensive methods in statistics. *Scientific American*, 248(5):116–130.
- Draper, N. R. and Smith, H. (1998). *Applied Regression Analysis*. Wiley.
- Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Chapman and Hall/CRC.
- Fallah, S., Chen, X., Lefebvre, D., Kurji, J., Hader, J., and Leeb, K. (2011). Babies admitted to NICU/ICU: Province of birth and mode of delivery matter. *Healthcare Quarterly*, 14(2):16–20.
- Fisher, R. A. (1938). The statistical utilization of multiple measurements. *Annals of Eugenics*, 8(4):376–386.
- Flury, B. (1988). *Common Principal Components and Related Multivariate Models*. Wiley.
- Flury, B. D. and Neuenschwander, B. E. (1995). Principal component models for patterned covariance matrices with applications to canonical correlation analysis of several sets of variables. In *Recent Advances in Descriptive Multivariate Analyses*, pages 90–112. Oxford University Press.
- Flury, B. K. (1987). Two generalizations of the common principal component model. *Biometrika*, 74(1):59–69.
- Flury, B. N. (1984). Common principal components in k groups. *Journal of the American Statistical Association*, 79(388):892–898.
- Flury, B. N. (1986). Asymptotic theory for common principal component analysis. *The Annals of Statistics*, 14(2):418–430.

- Flury, B. N. and Gautschi, W. (1986). An algorithm for simultaneous orthogonal transformation of several positive definite symmetric matrices to nearly diagonal form. *SIAM Journal on Scientific and Statistical Computing*, 7(1):169–184.
- Flury, B. W. and Schmid, M. J. (1992). Quadratic discriminant functions with constraints on the covariance matrices: Some asymptotic results. *Journal of Multivariate Analysis*, 40(2):244–261.
- Flury, B. W., Schmid, M. J., and Narayanan, A. (1994). Error rates in quadratic discrimination with constraints on the covariance matrices. *Journal of Classification*, 11(1):101–120.
- Friedman, J. H. (1989). Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175.
- Gabriel, K. R. (1971). The biplot graphic display of matrices with application to principal component analysis. *Biometrika*, 58(3):453–467.
- Gardner, S. (2001). *Extensions of Biplot Methodology to Discriminant Analysis with Application of Non-parametric Principal Components*. PhD thesis, Stellenbosch University.
- Gardner-Lubbe, S., Le Roux, N. J., and Gower, J. C. (2008). Measures of fit in principal component and canonical variate analyses. *Journal of Applied Statistics*, 35(9):947–965.
- Gnanadesikan, R. and Kettenring, J. R. (1972). Robust estimates, residuals, and outlier detection with multiresponse data. *Biometrics*, 28(1):81–124.
- Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations*. Johns Hopkins University Press.
- Goodnight, C. J. and Schwartz, J. M. (1997). A bootstrap comparison of genetic covariance matrices. *Biometrics*, 53(3):1026–1039.
- Gower, J. C., Gardner-Lubbe, S., and Le Roux, N. J. (2011). *Understanding Biplots*. Wiley.
- Gower, J. C. and Hand, D. J. (1996). *Biplots*. Chapman and Hall/CRC.
- Gu, H. and Fung, W. K. (2001). Influence diagnostics in common principal components analysis. *Journal of Multivariate Analysis*, 79(2):275–294.

- Hallin, M., Paindaveine, D., and Verdebout, T. (2010). Testing for common principal components under heterokurticity. *Journal of Nonparametric Statistics*, 22(7):879–895.
- Harrell, F. E. (2009). How to calculate the area under the curve. R help forum (internet website). <http://r.789695.n4.nabble.com/How-to-calculate-the-area-under-the-curve-td902633.html> [Accessed on 8 October 2014].
- Harrell, F. E. (2011). What does the “S.D.” returned by {Hmisc} rcorr.cens measure? R help forum (internet website). <http://r.789695.n4.nabble.com/what-does-the-quot-S-D-quot-returned-by-Hmisc-rcorr-cens-measure-td3329609.html> [Accessed on 8 October 2014].
- Harrell, F. E. (2012). *Hmisc: Harrell Miscellaneous*. R package version 3.9-3.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- Hills, M. (1982). Allometry. In *Encyclopedia of Statistical Sciences*, volume 1, pages 48–54. Wiley.
- Hintz, S. R., Bann, C. M., Ambalavanan, N., Cotten, C. M., Das, A., and Higgins, R. D. (2010). Predicting time to hospital discharge for extremely preterm infants. *Pediatrics*, 125(1):e146–e154.
- Hollander, M., Wolfe, D. A., and Chicken, E. (2014). *Nonparametric Statistical Methods*. Wiley.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441.
- Houle, D., Mezey, J., and Galpern, P. (2002). Interpretation of the results of common principal components analyses. *Evolution*, 56(3):433–440.
- Huber, P. J. (1977). Robust covariances. In *Statistical Decision Theory and Related Topics II*, pages 165–192. Academic Press.
- Huber, P. J. (2004). *Robust Statistics*. Wiley.
- Jacobi, C. J. G. (1846). Über ein leichtes Verfahren die in der Theorie der Säcularstörungen vorkommenden Gleichungen numerisch aufzulösen. *Journal für die reine und angewandte Mathematik*, 30:51–95.

REFERENCES

377

- Jarek, S. (2012). *mvnrmtest: Normality test for multivariate variables*. R package version 0.1-9.
- Jijon, C. R. and Jijon-Letort, F. X. (1995). Perinatal predictors of duration and cost of hospitalization for premature infants. *Clinical Pediatrics*, 34(2):79–85.
- Johnson, R. A. and Wichern, D. W. (2002). *Applied Multivariate Statistical Analysis*. Prentice Hall.
- Jolicoeur, P. (1963). The multivariate generalization of the allometry equation. *Biometrics*, 19(3):497–499.
- Jolliffe, I. T. (1972). Discarding variables in a principal component analysis. I: Artificial data. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 21(2):160–173.
- Jolliffe, I. T. (1973). Discarding variables in a principal component analysis. II: Real data. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 22(1):21–31.
- Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer.
- Jolliffe, I. T., Trendafilov, N. T., and Uddin, M. (2003). A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics*, 12(3):531–547.
- Keramidas, E. M., Devlin, S. J., and Gnanadesikan, R. (1987). A graphical procedure for comparing the principal components of several covariance matrices. *Communications in Statistics – Simulation and Computation*, 16(1):161–191.
- Klingenberg, C. P. (1996). Multivariate allometry. In *Advances in Morphometrics*, pages 23–49. Springer.
- Klingenberg, C. P. and Froese, R. (1991). A multivariate comparison of allometric growth patterns. *Systematic Zoology*, 40(4):410–419.
- Klingenberg, C. P. and McIntyre, G. S. (1998). Geometric morphometrics of developmental instability: Analyzing patterns of fluctuating asymmetry with Procrustes methods. *Evolution*, 52(5):1363–1375.
- Krzanowski, W. J. (1979). Between-groups comparison of principal components. *Journal of the American Statistical Association*, 74(367):703–707.

- Krzanowski, W. J. (1984). Principal component analysis in the presence of group structure. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 33(2):164–168.
- Krzanowski, W. J. (1990). Between-group analysis with heterogeneous covariance matrices: The common principal component model. *Journal of Classification*, 7(1):81–98.
- Krzanowski, W. J. (2000). *Principles of Multivariate Analysis*. Oxford University Press.
- Ledoit, O. and Wolf, M. (2003). Improved estimation of the covariance matrix of stock returns with an application to portfolio selection. *Journal of Empirical Finance*, 10(5):603–621.
- Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411.
- Linn, R. L. (1968). A Monte Carlo approach to the number of factors problem. *Psychometrika*, 33(1):37–71.
- Maronna, R. A. (1976). Robust M -estimators of multivariate location and scatter. *The Annals of Statistics*, 4(1):51–67.
- Mezey, J. G. and Houle, D. (2003). Comparing G matrices: Are common principal components informative? *Genetics*, 165(1):411–425.
- Mosteller, F. and Tukey, J. W. (1977). *Data Analysis and Regression*. Addison-Wesley.
- Najarzadeh, D. (2013). *FGalgorithm: Flury and Gautschi algorithms*. R package version 1.0.
- Nel, D. G. and Pienaar, I. (1998). The decomposition of the Behrens-Fisher statistic in q -dimensional common principal component submodels. *Annals of the Institute of Statistical Mathematics*, 50(2):241–252.
- Neuenschwander, B. E. and Flury, B. D. (2000). Common principal components for dependent random vectors. *Journal of Multivariate Analysis*, 75(2):163–183.
- Nordhausen, K., Cardoso, J.-F., Miettinen, J., Oja, H., Ollila, E., and Taskinen, S. (2013). *JADE: JADE and other BSS methods as well as some BSS performance criteria*. R package version 1.9-91.

REFERENCES

379

- O'Neill, T. J. (1984). *A Theoretical Method of Comparing Classification Rules under Non-optimal Conditions with Application to the Estimates of Fisher's Linear and the Quadratic Discriminant Rules under Unequal Covariance Matrices*. Technical report, Stanford University.
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.
- Pepler, P. T., Uys, D. W., and Nel, D. G. (2012). Predicting mortality and length-of-stay for neonatal admissions to private hospital neonatal intensive care units: A Southern African retrospective study. *African Health Sciences*, 12(2):166–173.
- Pepler, P. T., Uys, D. W., and Nel, D. G. (2014). A comparison of some methods for the selection of a common eigenvector model for the covariance matrices of two groups. *Communications in Statistics – Simulation and Computation*. [Accepted for publication].
- Phillips, P. C. and Arnold, S. J. (1999). Hierarchical comparison of genetic variance-covariance matrices. I. Using the Flury hierarchy. *Evolution*, 53(5):1506–1515.
- Phillips, P. C., Whitlock, M. C., and Fowler, K. (2001). Inbreeding changes the shape of the genetic covariance matrix in *Drosophila melanogaster*. *Genetics*, 158(3):1137–1145.
- Powell, P. J., Powell, C. V. E., Hollis, S., and Robinson, M. J. (1992). When will my baby go home? *Archives of Disease in Childhood*, 67(10 Spec No):1214–1216.
- R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.
- Rawlings, J. O., Pantula, S. G., and Dickey, D. A. (1998). *Applied Regression Analysis: A Research Tool*. Springer.
- Rencher, A. C. (1998). *Multivariate Statistical Inference and Applications*. Wiley.
- Rencher, A. C. (2002). *Methods of Multivariate Analysis*. Wiley.
- Reyment, R. A. (1997). Multiple group principal component analysis. *Mathematical Geology*, 29(1):1–16.

- Rublík, F. (2009). A test of the hypothesis of partial common principal components. *Mathematica Slovaca*, 59(5):579–592.
- Rui Alves, M. (2012). Evaluation of the predictive power of biplot axes to automate the construction and layout of biplots based on the accuracy of direct readings from common outputs of multivariate analyses: 1. Application to principal component analysis. *Journal of Chemometrics*, 26(5):180–190.
- Ryan, T. P. (2008). *Modern Regression Methods*. Wiley.
- Schäfer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(1):1175–1189.
- Schmid, M. J. (1987). *Anwendungun der Theorie proportionaler Kovarianzmatrizen und gemeinsamer Hauptkomponenten auf die quadratische Diskriminanzanalyse*. PhD thesis, University of Berne.
- Schott, J. R. (1988). Common principal component subspaces in two groups. *Biometrika*, 75(2):229–236.
- Schott, J. R. (1991a). Some tests for common principal component subspaces in several groups. *Biometrika*, 78(4):771–777.
- Schott, J. R. (1991b). A test for a specific principal component of a correlation matrix. *Journal of the American Statistical Association*, 86(415):747–751.
- Schott, J. R. (1998). Estimating correlation matrices that have common eigenvectors. *Computational Statistics and Data Analysis*, 27(4):445–459.
- Sengupta, S. and Boyle, J. S. (1998). Using common principal components for comparing GCM simulations. *Journal of Climate*, 11(5):816–830.
- Shen, H. and Huang, J. Z. (2008). Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015–1034.
- Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall/CRC.
- Snedecor, G. W. and Cochran, W. G. (1989). *Statistical Methods*. Iowa State University Press.

REFERENCES

381

- Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101.
- Stauffer, D. F., Garton, E. O., and Steinhorst, R. K. (1985). A comparison of principal components from real and random data. *Ecology*, 66(6):1693–1698.
- Steppan, S. J. (1997). Phylogenetic analysis of phenotypic covariance structure. I. Contrasting results from matrix correlation and common principal component analysis. *Evolution*, 51(2):571–586.
- Steppan, S. J., Phillips, P. C., and Houle, D. (2002). Comparative quantitative genetics: Evolution of the G matrix. *Trends in Ecology and Evolution*, 17(7):320–327.
- Tarpey, T. (2000). Parallel principal axes. *Journal of Multivariate Analysis*, 75(2):295–313.
- Trendafilov, N. T. (2010). Stepwise estimation of common principal components. *Computational Statistics and Data Analysis*, 54(12):3446–3457.
- Vermont Oxford Network (2009). *Manual of Operations for Infants Born in 2009*. Vermont Oxford Network Database. Release 13.2.
- Waldmann, P. and Andersson, S. (2000). Comparison of genetic (co)variance matrices within and between *Scabiosa canescens* and *S. columbaria*. *Journal of Evolutionary Biology*, 13(5):826–835.
- Walsh, B. and Lynch, M. (2013). Comparisons of G and its stability. In *Evolution and selection of quantitative traits: I. Foundations*. [Unpublished].
- Williams, D. (2013). *Biplots and Triplots for Exploring Three Mode Data with an Application to the Investigation of the Immune Response to Bacille Calmette Guérin Vaccine in HIV Positive Infants*. Master's thesis, University of Cape Town.
- Yuan, K. H. and Bentler, P. M. (1994). Test of linear trend in eigenvalues of k covariance matrices with applications in common principal components analysis. *Communications in Statistics – Theory and Methods*, 23(11):3141–3156.
- Zernikow, B., Holtmannspötter, K., Michel, E., Hornschuh, F., Groote, K., and Hennecke, K.-H. (1999). Predicting length-of-stay in preterm neonates. *European Journal of Pediatrics*, 158(1):59–62.

- Ziyatdinov, A., Kanaan-Izquierdo, S., Trendafilov, N. T., and Perera-Lluna, A. (2014). *cpc*: *Methods to perform Common Principal Component Analysis (CPCA)*. R package version 0.1.2.
- Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286.