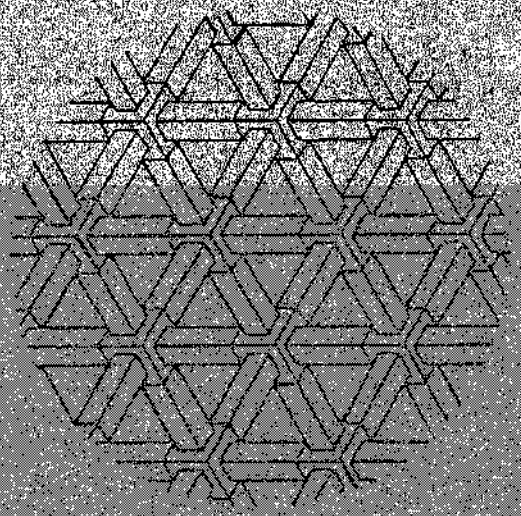
ECONOMETRIC ANALYSIS OF MULTIDIMENSIONAL MODELS



J. VERHEES

STELLINGEN

bij het proefschrift

ECONOMETRIC ANALYSIS OF MULTIDIMENSIONAL MODELS

van

Jaap Verhees

1. Het begrip 'dimension' is voor meer uitleg vatbaar in het onderzoeksveld van multidimensionale data–analyse. Het begrip kan tegelijkertijd duiden op het aantal richtingen, op de omschrijving van een richting, en op het kardinale getal van de richtingen van een en dezelfde data–struktuur. Om verwarring te voorkomen, kan men gebruik maken van de begrippen 'way', 'dimension' en 'mode' in de verschillende contexten.

2. Een k–factoriële Kronecker–product–struktuur als vorm van restrikties op de elementen van een matrix levert een reduktie in aantal parameters van $\ell^2 \equiv \prod_{i=1}^{k} \ell_i^2$ naar $q \equiv \sum_{i=1}^{k} \ell_i^2$. Is symmetrie aan de orde, dan is de reduktie van $\frac{1}{2}\ell(\ell+1)$ naar $\frac{1}{2}(q + \sum_{i=1}^{k} \ell_i)$.

3. De $pq \times mn$ matrix $\tilde{A}$ speelt een grote rol in het k–dimensionale covariantie–struktuur–analyse–model. De matrix $\tilde{A}$ die wordt aangeduid met de term tilde–transformatie van de $mp \times nq$ matrix A, wordt verkregen door een bewerkelijke maar systematische verplaatsing van alle elementen van A. Matrix $\tilde{A}$ is namelijk gedefinieerd als

$$\tilde{A} \equiv (a_{11},a_{21},\ldots,a_{m1},a_{12},a_{22},\ldots,a_{m2},\ldots,a_{1n},a_{2n},\ldots,a_{mn}),$$

waarbij a_{ij} de 'vec' van blok–matrix A_{ij} in matrix A is. Een specifieke (0,1)–matrix B_i verduidelijkt de systematiek in de transformatie van A naar $\tilde{A}$ en vice versa (p. 25 en verder).

4. De kanonieke oplossing en alternerende gewone kleinste–kwadraten–oplossing in k–dimensionale principale–componenten–analyse zijn altijd gelijk voor $k = 2$. Voor $k > 2$ geldt deze gelijkheid in het algemeen niet. Voor $k > 2$ geldt deze gelijkheid als bij de kleinste–kwadraten–oplossing $A_j^{(r)}$ voor A_j in iteratie–stap r geldt dat $A_i^{(r)} = I_{\ell_i}$, $i \neq j$ (p. 44).

5. De aanwezigheid van (0,1)–matrices in de multidimensionale data–analyse–modellen kan voor een programmeur die rekening houdt met geheugenbeslag en rekentijd, een reden zijn om geen 'rechttoe–rechtaan' matrix–multiplicaties uit te voeren. Deze programmeur kiest een van drie manieren ter beschrijving van de operaties van (0,1)–matrices, te weten: door functies, door vectoren van indices, of door toevoeging en permutatie van 'do–loops' (hoofdstuk 9).

6. De noniteratieve ongewogen kleinste-kwadraten-schatting $\hat{\sigma}_i$ van de vec-representatie van een covariantie matrix Σ_i in het k-modale covariantie-struktuur-analyse-model is de eigenvector die samenhangt met de grootste eigenwaarde van een matrix met struktuur $\tilde{S}_i\tilde{S}_i'$ en is dientengevolge de vec-representatie van een symmetrische matrix (p. 127 en 128). Matrix S_i is gedefinieerd als $S_i \equiv C_iSC_i'$, waarbij S een covariantie-matrix is en C_i een commutatie-matrix is (p. 22 t/m 26).

7. Het programmeren van iteratieve functie-optimalisatie-technieken en nulpunt-zoektechnieken maakt de student meer bewust van de numerieke problemen en diverse eigenschappen van deze technieken, en dient daarom een onderdeel van informatica-vakken binnen de studierichting Econometrie te blijven. (Faculteitsgids '89-'90, FEW, RUG: p. 180, 182, 186, en 196).

8. Uit een enquête 'Beroep en Opleiding' onder econometristen 'uit de praktijk', die in 1970 door het (toenmalig geheten) Instituut voor Actuariaat en Econometrie van de UvA is gehouden, bleek dat een pas afgestudeerde econometrist een aantal eigenschappen dient te bezitten die verband houden met 'karakter'. Gerangschikt in volgorde van frequentie van de antwoorden zijn de eigenschappen: vermogen tot werken in teamverband, uitdrukkingsvermogen, initiatief, doorzettingsvermogen, nauwgezetheid, aanpassingsvermogen en enthousiasme. Scores op 'karakter' ontbreken op de cijferlijst bij het doctoraal-examen. Zowel de eventuele werkgever als de pas afgestudeerde sollicitant hebben er profijt van als scores op 'karakter' worden vermeld, aangezien een besparing van kosten en tijdsbeslag voor beide partijen voordelig is. Voor de jurering van een karaktertrek kan de faculteit gebruik maken van soortgelijke methoden die psychologische-test-bureaus hanteren. (Directoraat-Generaal voor de Arbeidsvoorziening (1978), Beroepsmonografieën: Econometrist. Den Haag: Staatsuitgeverij).

9. Het voorstel van staatssecretaris Heerma van Volkshuisvesting om de positie van jongeren op de woningmarkt te verbeteren door inkomsten uit verhuur gedeeltelijk belastingvrij te maken, gaat niet ver genoeg om een groter kameraanbod te garanderen. De inkomsten uit verhuur moeten volledig belastingvrij worden gesteld om huishoudens uit de hogere inkomenslagen te prikkelen. (De Volkskrant, dinsdag 22 augustus 1989).

10. Bij stelsels met meer dan twee vergelijkingen is het optreden van hoge paarsgewijze correlaties van de fouten een voldoende maar geen noodzakelijke voorwaarde voor de keuze van 3SLS boven 2SLS. (Belsley, D.A. (1988), Two- or three-stage least squares? *Computer Science in Economics and Management* 1, 21–30).

11. Het zou zakkenrollers en insluipers lelijk opbreken wanneer niemand meer gebruik maakt van die fraaie polstasjes en portefeuilles waarin paspoort, rijbewijs, creditcards en contanten allemaal tegelijk zo handzaam worden opgeborgen.

12. Het deelnemen aan een triathlon en het schrijven van een proefschrift zijn van gelijke vorm; begin en einde van het af te leggen parcours zijn regelmatig in zicht maar ondertussen moeten veel nieuwe wegen worden ingeslagen, waarbij de man met de hamer meerdere malen in geestesvorm zal verschijnen.

RIJKSUNIVERSITEIT GRONINGEN

ECONOMETRIC ANALYSIS OF MULTIDIMENSIONAL MODELS

Proefschrift

ter verkrijging van het doctoraat in de
Economische Wetenschappen
aan de Rijksuniversiteit Groningen
op gezag van de
Rector Magnificus Dr. L.J. Engels
in het openbaar te verdedigen op
donderdag 28 *september* 1989
des namiddags te 2.45 *uur precies*

door

JAAP VERHEES

geboren op 28 *maart* 1961
te Reek, gemeente Schaijk

Promotor:

Prof. Dr. T.J. Wansbeek

Overige leden van de promotiecommissie:

Prof. Dr. H. Neudecker
Prof. Dr. G. Ridder
Prof. Dr. J.M.F. ten Berge

Aan mijn ouders

ACKNOWLEDGEMENTS

The research I have done as reported in this dissertation was carried out at the Econometrics Institute of the Faculty of Economics of Groningen University. My research is to my full knowledge not a result of my sole contribution. I have received support and ideas from several colleagues and I therefore express my feelings of gratitude to them below.

My gratitude in the first and main place goes out to my promoter, Prof. Dr. T.J. Wansbeek. He has given me strong moral support on many occasions, especially during dark rainy days when my scientific aspirations had run low on fuel and I confronted him with these situations. It is also the outcome of the many shared little scientific projects with Professor Wansbeek that have led to this thesis and my positive experience of scientific research in econometrics.

I thank the members of the graduation committee, Prof Dr. J.M.F. ten Berge, Prof Dr. H. Neudecker, and Prof. Dr. G. Ridder for their willingness to read and comment on my thesis, especially after such short notice of invitation.

My colleague Drs. A. Merckens has supported me many times on odd hours of the day. Most of his enthusiastic computer programming assistance took place at night at the Institute. He found the time to translate my brainstorm ideas and unordered thoughts into simple but effective programs. I appreciate his support very much.

I thank Tamara Schroor for her suggestions and corrections to layout and typing errors in the text which I had overlooked.

Finally, I thank my parents for their moral support and efforts in keeping me in healthy shape by serving nice dinners during many weekends when a young guy would rather finish his study than fix himself some decent healthy dinner at normal hours.

CONTENTS

PART II. MULTIDIMENSIONAL DATA ANALYSIS MODELS

LIST OF SYMBOLS, ABBREVIATIONS, AND MATTERS OF NOTATION

Matrices are denoted by (*italic*) capital letters, several of which are reserved and possibly followed by one subscript in which case we speak of indexed matrices. Block–matrices of partitioned matrices are denoted by (*italic*) capital letters with two subscripts. Several type matrices are denoted by specially reserved (*italic*) capital letters with subscripts indicative of the order of these matrices. Vectors are denoted by (*italic*) lower–case letters. Vectorizations of matrices and indexed matrices are therefore usually denoted by (*italic*) lower–case letters and lower–case letters with subscripts, respectively. Elements of multiway arrays, matrices and vectors are represented by (*italic*) lower–case letters, indexed by multiple, two or one subscript(s), respectively. Of course there are cases where these notational rules may then lead to ambiguous meanings. At these instances we hope to have clarified the context in which the symbol is used at that point well enough to the reader to circumvent any doubts. If not, one more notational rule may be of help; we denote elements of indexed matrices, of subscripted block–matrices or of both their vectorizations, by placing the symbol for the matrix or vector between round brackets. For instance, we may have matrix A, block–matrix A_{ij}, vector $a \equiv \text{vec}(A)$, element a_{ij} of A, and element $(A_{ij})_{kl}$ of A_{ij}. The vectorization of block–matrix A_{ij} will be indicated by a_{ij}, $\text{vec}(A_{ij})$, or $\text{vec } A_{ij}$. Vector $a_{\cdot j}$ denotes the j–th column of matrix A, but a_j denotes the j–th element of vector a. We also have an indexed matrix C_i, vector $c_i \equiv \text{vec } C_i$, element $(C_i)_{ij}$ of C_i and element $(c_i)_j$ of c_i. Finally, matrix $P_{m,n}$ is a matrix of order $mn \times mn$.

The (full) list of symbols, with a brief statement of their meaning, is the following.

General symbols.

■	End of proof.
□	End of example.
.	Denotes a product of a scalar with any other operand. Also indicates the end of a sentence.
$\equiv$	'Equals by definition'.

$\approx$	'Approximately equal to'.
min	Minimum, minimize.
max	Maximum, maximize.
lim	Limit.
e, exp(.)	Exponential.
$n!$	Factorial.
$\mathbb{R}$	Set of real numbers.
$\mathbb{R}^n$, $\mathbb{R}^{n\times m}$	Set of real $n\times 1$ vectors ($n\times m$ matrices).
Π	Product operator.
Σ	Summation operator.
$O(n^2)$, $O(n \log n)$, etc.	Order of complexity or convergence of process.
θ^0	True value of population parameter θ
$\hat{\theta}$	Estimator for (or estimate of) θ.

Logic.

$p \Rightarrow q$	Implication:'p implies q'. Condition p is sufficient for state q.
$p \Leftarrow q$	Implication:'q implies p'. Condition p is necessary for state q.
$p \Leftrightarrow q$	Biconditional. Condition p is necessary and sufficient.
$\forall$	'For every'.
$\exists$	'For at least one'.

Matrix operators.

$A = (a_{ij})$	Matrix A with elements a_{ij}.
A' or A^T	Transpose of the matrix A.
A^{-1}	Inverse of matrix A.
A^+	Moore–Penrose inverse of matrix A.
A^-	Generalized inverse of matrix A.
dg(A), dg A	Diagonal matrix, with the diagonal elements of A.
$A =$ diag(a_1,...,a_n)	Diagonal matrix with diagonal entries a_1,...,a_n.

$A^{1/2}$	Square root of matrix A.
$(A : B)$, (A,B), $\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$	Partitioned matrix with blocks A and B, or A_{11} to A_{22}.
A_v	$A_v=(A_1',\ldots,A_m')'$, when $A=(A_1:\ldots:A_m)$.
$\tilde{A}$	Tilde–transform of (usually partitioned) matrix A.
$\det(A)$, $\|A\|$	Determinant of A.
$\mathrm{tr}(A)$, tr A	Trace of square matrix A.
$\Delta_{(p)}[A]$, $\Delta[A]$	Block–trace of a partitioned matrix A; $\Delta_{(p)}[A] \equiv \sum_{i=1}^{p} A_{ii}$.
$\mathrm{r}(A)$, $\mathrm{rank}(A)$	Rank of matrix A.
$\|A\|$, $\mathrm{norm}(A)$	Norm of the matrix A defined by $\|A\| = (\mathrm{tr}\ A'A)^{1/2}$.
$\|a\|$	Norm of vector a.
vec A, vec(A)	Vec operator.
v(A)	The $\frac{1}{2}n(n+1)$ distinct elements of a symmetric $n\times n$ matrix A.
$\otimes$	Kronecker product operator.
π	Tracy–Singh operator.
δ_{ks}	Kronecker delta: $\delta_{ks} = 1$ for $k = s$ and $\delta_{ks} = 0$ for $k \neq s$.

Special matrices and vectors.

I_n	Identity matrix (of order $n\times n$).
$I(.)$, $I(\theta)$	Information matrix.
0	Zero, null matrix, or null vector.
$P_{m,n}$	Permutation matrix, of order $mn\times mn$.
P_n	Permutation matrix, of order $n^2\times n^2$.
C_i	Commutation matrix, of order $\ell\times\ell$.
D_i	Commutation matrix, of order $m\times m$.
C_i^j, D_i^j	Commutation matrix, of order $\ell^j\times\ell^j$, $m^j\times m^j$, respectively.
D_n	Duplication matrix, of order $n^2\times\frac{1}{2}n(n+1)$.
B_i	Tracy–Singh permutation matrix.
l, ι	Unit column vector.
e_i	Column vector with unit element in its i–th position, zeroes elsewhere.

Functions.

$f\colon S \to T$	Function defined on domain S with values in image T.
ϕ, ψ	Real-valued function.
f, g	Vector(-valued) function.
F, G	Matrix(-valued) function.
ψ^{-1}	Inverse function.
$\log x$, $\ln(x)$	Natural logarithm.
${}^{a}\log(x)$	Logarithm to the base a.
$\lvert x \rvert$	Absolute value.

Derivatives.

d	(First-order) differential.
d^2	Second-order differential.
d^n	n-th order differential.
$\partial\phi(X)/\partial X$, $\partial F(X)/\partial X$	Matrices of partial derivatives.
$\partial\phi(x)/\partial x'$	Derivative of $\phi(x)$.
$\partial f(x)/\partial x'$	Derivative (Jacobian matrix) of $f(x)$.
$\partial \mathrm{vec}\, F(X)/\partial(\mathrm{vec}\, X)'$	Derivative (Jacobian matrix) of $F(X)$.
$\mathrm{grad}(\phi)$, $\mathrm{grad}(f)$	Gradient of ϕ and f, respectively.
$\partial^2\phi(x)/\partial x\partial x'$	Second-order derivative (Hessian matrix) of $\phi(x)$.

Statistical symbols.

$\mathrm{P}(.)$, $\mathrm{Pr}(.)$, Prob	Probability.
$\mathrm{p}(.,\theta)$	The probability density function (p.d.f.) of a random variable/vector with parameter vector θ.
$\underset{\sim}{x}$	$\underset{\sim}{x}$ is a stochastic (random) variable or vector of variables.
$\sim$	'Is distributed as'.
$\overset{a}{\sim}$	'Is asymptotically distributed as'.

$\underline{x}_n \xrightarrow{a.s} \underline{x}$, $\underline{x}_n \overset{a}{=} \underline{x}$	The sequence of random variables or vectors $\{\underline{x}_n\}$ is said to converge almost surely (or with probability one) to $\underline{x}$, if given ε, $\delta > 0$, there exists a $k_o(\varepsilon,\delta)$ such that $P(\lvert\underline{x}_n-\underline{x}\rvert < \varepsilon, \forall k \geq k_o) > 1 - \delta$.
$\underline{x}_n \xrightarrow{L} \underline{x}$	'Converges in law to'. Let $\{\underline{x}_n\}$ be a sequence of random variables, $n = 1,2, \ldots$. Let $\underline{x}_n$ have distribution F_n and $\underline{x}$ distribution F. Then $\underline{x}_n \xrightarrow{L} \underline{x}$ if $\lim_{n\to\infty} F_n(.) = F(.)$ for all continuity points of F.
$\underline{x}_n \xrightarrow{P} c$, $\operatorname{plim}_{n\to\infty}(\underline{x}_n) = c$	'Convergence in probability'. $\lim_{n\to\infty} P(\lvert\underline{x}_n-c\rvert < \varepsilon) = 1$ for every $\varepsilon > 0$.
$\underline{x} \sim N(\mu, \sigma^2)$	$\underline{x}$ has a normal (Gaussian) distribution with mean μ and covariance σ^2.
$\underline{x} \sim N_n(\mu, V)$	$\underline{x}$ has a n–variate normal (Gaussian) distribution with mean μ and covariance matrix V.
χ^2_m, $\chi^2(m)$	χ^2 distribution with m degrees of freedom.
$F(m,n)$	F distribution with m and n degrees of freedom.
$W_n(f, V)$	$n \times n$ matrix variate Wishart distribution with f degrees of freedom and covariance matrix V.
$W_n(f, V; M)$	$n \times n$ matrix variate noncentral Wishart distribution with f degrees of freedom, covariance matrix V, and noncentrality matrix M, having expectation $fV + M$.
V(.), var(.), covar(.)	(Co–)variance (matrix).
ML	Maximum Likelihood.
ULS	Unweighted Least Squares.
WLS	Weighted Least Squares.
GLS	Generalized Least Squares.
OLS	Ordinary Least Squares.

INTRODUCTION

Many models and methods have been proposed to deal with multidimensional data. Overviews of several models in multidimensional data analysis are presented in De Leeuw, Heiser, Meulman and Critchley (1986), De Leeuw, Keller and Wansbeek (1983), and Law, Snyder, Hattie and McDonald (1984). De Leeuw et al. (1986) covers the areas of distance models used in multidimensional scaling, and of discrete and hierarchical models used in classification and cluster analysis. De Leeuw et al. (1983) covers two main themes, namely latent variables and discrete or categorical variables, in several papers that fall in either of three content areas. These three areas are linear structural equation modeling, exploratory multivariate analysis, and categorical data analysis. Law et al. (1984) consists of a number of papers on multidimensional data analysis and methods that are divided in two areas, factor analysis and multidimensional scaling.

The statistical methods that may be included under the domain of multidimensional data analysis (sometimes defined as multimode data analysis, see e.g., Law et al. (1984)) are characterized by the study of data that are organized in typically complex arrays. The complexity of these arrays arises from various and different levels of depth of detail.

One detail is the terminology one uses to describe the type of array in which the data set is structured. There are some definitions to better appreciate the differences between types of arrays. Mode is defined as 'a particular class of entities' such as units or objects, stimuli or variables or items, occasions or subjects or experimental conditions, replications or times. Way is defined as each elementary component of the cartesian product of the number of modes (some of which may be repeated) defining the data array. So, an array can be defined by a number of modes lower than, or equal to, the number of ways. For example, a three–way array may be constituted by $A \times B \times B$, where A identifies a set of subjects and B a set of stimuli. Therefore such an array has only two modes. Such a mode's definition in certain situations is not fully satisfactory. For instance, in the case of multiple contingency

tables, the mode 'variable' is partitioned among the various ways of the table, while the mode 'units' 'disappears' within the data transform into frequencies. It is then possible to describe this case as an array with an unique mode (i.e., the mode 'variable') or as an array with as many modes as variables. To make up for the lack of definition, it might be useful to consider a further characteristic of the data that we call 'dimension', as the support defining the data array for the study. A matrix for example, is a two–way array with two dimensions (generated by one or two modes: in the former case the mode is the same set made up of the same elements). The term 'dimension' would be useful to distinguish the analysis of a k–way array carried out at an analytical level lower than k. The term dimension is, in a different meaning, utilized also to define the cardinality of modes. For instance, a two–way array (quantitative characteristics × times) is a case with measurements on I quantitative characteristics (length, wide, height, income, capital, etc.) at J different times. Here, I and J is the dimension (i.e., the cardinality) of the modes quantitative characteristics and times, respectively.

For the sake of simplicity we will refrain from using this terminology to its full subtlety. From now on we use terms 'mode', 'way', and 'dimension' interchangeably, since most of our k–dimensional data analysis models (cf. chapters 4, 5, 6 and 7) will be meant for k–dimensional data, represented in k–way arrays that are defined by k modes. For the other k–dimensional models (cf. chapter 8), this subtlety of terminology has become of minor importance to our discussion.

Classifications of the models and related methods were introduced in different contexts as multidimensional scaling, classification analysis, cluster analysis, factor analysis, and principal component analysis, amongst others. These classifications take into account particularly the model defined in their framework by focusing on the theoretical and methodological features. The models differ with respect to the role (symmetric or asymmetric) given to the variables, to the nature (qualitative or quantitative) of the data, and to the type of array in which the data set is structured. Mainly three groups of models are described in the literature, of which in this study the first two for several of their members are discussed. The three groups of models are: the models for multivariate numerical data, the models for categorical or mixed data, and the models for proximity or preference data.

The aim of this study is threefold. First, we investigate the possibility of extending several other data analysis models (that are classified in the first two groups mentioned above), to an arbitrary number of modes. Second, we investigate the properties of several estimators, and numerical methods in solving for the estimators, using simulated and real data. Third, we investigate the possibility of developing program routines which may be basic to all our models and which result from 'lower level' calculus instead of an 'at face value' approach.

We will briefly review some earlier literature on models for multidimensional data analysis, together with a summary of the contents of the chapters that follow.

The first group of models mentioned above, considers multivariate numerical data which are the entries of a k-way array of real numbers. Two situations can occur when this type of data is considered: the ways are 'dependent' on each other (e.g., the variables of the objects are different on some occasion), or the ways are 'independent' of each other (e.g., the same objects and same variables on each occasion). We leave the case of dependent ways aside in this study and consider the situation of independent ways. In the literature the three-dimensional case of data analysis models in this category is mainly discussed, because of its main interest in the applications and the wide variety of methods proposed to deal with it.

A symmetric point of view with respect to the modes' roles is considered in the methods based on trilinear and quadrilinear models. A general view of multilinear models is provided in Kruskal (1984). In this context the most general model was proposed by Tucker (1963, 1966). This is a quadrilinear model based on a generalization of the Singular Value Decomposition applied to bilinear models. Tucker introduced a new factorial concept based on the idea that a different underlying structure is associated to each mode. The three 'observational modes' that characterize the observed data array are associated each with a 'derivational mode', that can be thought of as a set of factors or idealized categories corresponding to the observational mode. Thus we can think of the subjects, the variables and the occasions as, respectively, linear combinations of 'idealized' subjects, 'latent' variables and 'psychological' occasions. The relationships among the three sets of factors are taken into account in a three-way array, the 'core matrix', estimated by

the model. Consequently from a logical point of view Tucker's approach cannot be considered a generalization of the classical principal component analysis (PCA) for two–mode data, because in that case a single set of components was introduced for both modes. The algorithms developed by Tucker did not produce a least squares solution to the data. Kroonenberg and De Leeuw (1980) proposed alternating least squares (ALS) to fit both the trilinear and the quadrilinear models. Successively these were implemented in the programs TUCKALS2 and TUCKALS3 described by Kroonenberg (1983).

A generalization to more than three modes has been developed in the framework of Tucker's component analysis. An extension of the Tucker three–way model to the four–way case is given by Lastovicka (1981). In this paper a 'canonical' solution to a four–way model is provided without the characteristic of a least squares fit to the four–way data array. An application to viewer perceptions of repetitive advertising was provided to show the applicability of the model. Recently, Kapteyn, Neudecker and Wansbeek (1986) extended Lastovicka's results to an arbitrary number k of ways, deriving a least squares solution and comparing with the canonical solution, using the example cited above. An iterative converging procedure, similar to ALS, was used to compute the least squares solution.

Except for the last study above and a few others not cited here, just brief indications concerning the k–way generalizations are usually provided but never exploited, possibly due to the lack of data. As we have the same problem, we will also have to rely on the four–mode advertising dataset when discussing our estimation results with applying ALS and canonical estimation approaches in the k–mode PCA model. Chapter 4 is based for most part on the paper by Kapteyn, Neudecker and Wansbeek (1986), but has some additional results. First, a Maximum Likelihood (ML) approach is presented. Second, both canonical and ALS solution are compared with respect to the different interpretations of their results.

Chapters 5, 6 and 7 are devoted to other k–mode models for the analysis of numerical data. For illustrative purposes, most of the models are applied to 2–mode and 3–mode real data and subjected to simulated data with more than three modes. The k–mode data analysis models are, in the respective order, proposed to deal with factor analysis, covariance structure analysis, and interdependent regression.

Chapter 8 discusses a model for the analysis of categorical data when the qualitative variables have a symmetric role in the analysis. Existing one–way and multi–way models for data of this kind are log–linear models amongst others. When analyzing a set of categorical data it is possible to structure the k–way data array as a k–way contingency table. Both the k–mode log–linear model and our k–mode Poisson regression model are data analysis models for handling a k–way contingency table. But our model differs from the log–linear model since our specification is not based on a fixed design matrix. The second difference is that the specification of our model is based on a Kronecker product structure for the relation between the modes.

The Kronecker product and vec operator are vital tools in all of our models. Several aspects of Kronecker product and vec operator are shortly discussed in chapter 2. More on the subject can be found in Graham (1981), Neudecker (1977), Searle (1982), and Wansbeek (1988). Due to the properties of the Kronecker product and vec operator, and their close connection to each other, the necessary expressions for deriving estimates are easy to handle. Several (0,1)–matrices, discussed in chapter 3, are also very useful. A large number of papers has been published on (0,1)–matrices, such as Balestra (1976), Hartwig and Morris (1975), Henderson and Searle (1981), Magnus and Neudecker (1979, 1988), Neudecker and Wansbeek (1983), and Tracy and Singh (1972).

A third necessity is the theory and application of matrix differential calculus. Amongst others, we have consulted Balestra (1976), Bentler and Lee (1978b), Dwyer (1967), Graham (1981), Magnus and Neudecker (1988), McCulloch (1982), Neudecker(1969), Pollock (1985b), Tracy and Singh (1972, 1975), and Wiens (1985). Methods given by Tracy and Dwyer (1969) and Tracy and Jinadasa (1987) gave results in an inconvenient form for matrix manipulation. Neudecker (1969) gave results for Kronecker products, based on a sound definition of a matrix derivative, but did not develop a comprehensive system. Magnus and Neudecker (1988) produced such a system in the form of a comprehensive set of theorems and algebraic identities. From a theoretical point of view, the two concepts 'differentials' and 'derivatives', are equivalent. But the approach via differentials is superior, and when in this study derivatives are needed, as will often be the case, they will be obtained from differentials and first– and second identification theorems (cf. Magnus and Neudecker (1988)).

A final tool is the notation, which was first introduced by Kapteyn, Neudecker and Wansbeek (1986), and which combines the three previous tools as a swiss-army knife. The notation works well on paper.

But the final computer program implementation shows that efficient programming is a goal to be set since several Kronecker products, large matrices, and sparse (0,1)-matrices appear on paper. Chapter 9 discusses transparent procedures, that result from a separate 'lower-level' calculus which is not standard in existing statistical software. These procedures handle any choice of the number of modes, k, in the aforementioned models.

SOME CONVENTIONS

A distinction is made between chapters, sections, and subsections. Intra-chapter references omit the chapter number; cross-chapter references are prefixed by a chapter number. A similar convention holds regarding sections and subsections. For instance, in subsection 7.4.2 we refer to subsection 7.4.1 as 'subsection 1', to section 7.3 as 'section 3', to subsection 7.2.1 as 'subsection 2.1', to subsection 4.3.3 as 'subsection 4.3.3', and to section 3.3 as 'section 3.3'.

An analogous convention is adopted regarding formulae. Within a section, formulae are numbered consecutively. When referring to a formula in the same chapter in a different section, the section number preceeds the formula number. When referring to a formula in a different chapter, both the chapter number and the section number precede the formula number. Tables and figures are numbered consecutively within chapters. When referring to tables or figures in other chapters the chapter number is prefixed.

PART I

MATRIX ALGEBRA

1. MATRICES AND PROPERTIES

1.1. INTRODUCTION

In this chapter we discuss in short just a few well known definitions and properties of operators and operands in matrix algebra.

1.2. THE TRACE AND RANK OF A MATRIX

An $m \times n$ matrix A is a rectangular 2-dimensional array of real numbers a_{ij} positioned as

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} . \tag{1.2.1}$$

The *trace* of a square $n \times n$ matrix A, denoted by $\mathrm{tr}(A)$ or $\mathrm{tr}\ A$, is the sum of its diagonal elements:

$$\mathrm{tr}(A) = \sum_{i=1}^{n} a_{ii}. \tag{1.2.2}$$

Several rules concerning the trace are:

$$\mathrm{tr}(A + B) = \mathrm{tr}(A) + \mathrm{tr}(B), \text{ if } A \text{ and } B \text{ are of same order,} \tag{1.2.3}$$

$$\mathrm{tr}(\lambda A) = \lambda.\mathrm{tr}(A) \text{ if } \lambda \text{ is a scalar,} \tag{1.2.4}$$

$$\mathrm{tr}\ A' = \mathrm{tr}\ A, \tag{1.2.5}$$

$$\mathrm{tr}(AB) = \mathrm{tr}(BA). \tag{1.2.6}$$

We will discuss more propositions related to the trace later on.

The *rank* of A, denoted by $r(A)$ or $\text{rank}(A)$, is the maximum number of linearly independent columns or rows it contains. We mention the following results concerning the rank.

(1.2.7) $\quad r(A) = r(A') = r(A'A) = r(AA')$,

(1.2.8) $\quad r(AB) \leq \min(r(A), r(B))$,

(1.2.9) $\quad r(AB) = r(A)$ if B is square of full rank,

(1.2.10) $\quad r(A + B) \leq r(A) + r(B)$, A and B of the same order.

A generalization of the concept of rank to three–way and k–way arrays is introduced in Kruskal (1977) and further discussed in Kruskal (1988). The rank of a k–way array X is the smallest number $r(X)$ such that X can be written as a linear combination of $r(X)$ outer product arrays $a_1 \otimes a_2 \otimes \ldots \otimes a_k$ whose elements are $a_{i_1 1} \otimes a_{i_2 2} \otimes \ldots \otimes a_{i_k k}$. For $k = 2$, this reduces directly to the definition of matrix rank above. This definition relates strongly to the Parafac model (Carroll and Chang (1970)): if X is a three–way data array, then $r(X)$ is the smallest number of dimensions (or factors) with which Parafac can fit the data exactly. Kruskal (1988) also introduces a simpler generalization of matrix rank, which is called 'Tucker rank'. Let r_i be the dimensionality of the space spanned by all 1–dimensional 'fibers' of X in direction i, i.e., by all vectors which can be formed by letting the i–th subscript of X vary and fixing all the other subscripts. Then the Tucker rank of X is the set $(r_1,\ldots,r_k)$. Tucker rank is related to the Tucker (1963, 1966) three–mode PCA model. A few weak inequalities between $r(X)$ and Tucker rank (e.g., for three–way arrays) exist: $r(X) \leq \min(r_1 r_2, r_1 r_3, r_2 r_3)$ and $r(X) \geq \max(r_1, r_2, r_3)$. These relations are also proven by Wansbeek and Verhees (1989) by applying a singular value decomposition (SVD) approach (cf. subsection 5.2).

Not only in this recent literature has thought been given to several extensions of definitions in matrix algebra. For instance, Oldenburger (1940) studied higher–dimensional determinants of k–way arrays. In addition, Oldenburger (1941) studied the generalization of an eigenvalue–eigenvector

decomposition (cf. subsection 5.1) of symmetric matrices to the case for k–way symmetric arrays.

1.3. PARTITIONED MATRIX AND BLOCK-TRACE

Let A be an $pq \times pr$ matrix, partitioned into p^2 blocks A_{ij}, of order $q \times r$:

$$A = \begin{bmatrix} A_{11} & \cdots & A_{1p} \\ \vdots & & \vdots \\ A_{p1} & \cdots & A_{pp} \end{bmatrix}. \tag{1.3.1}$$

Ordinary matrix summation and multiplication generalize to partitioned matrix summation and multiplication of partitioned matrices A and B, provided each block–matrix A_{ij} and B_{ij} is of appropriate order. For instance, in case $p = 2$,

$$A + B = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} A_{11}+B_{11} & A_{12}+B_{12} \\ A_{21}+B_{21} & A_{22}+B_{22} \end{bmatrix},$$

provided each B_{ij} is also of order $p \times q$. In this case we say partitioned matrices A and B must be *conformable* for summation or multiplication, respectively.

In part of this study the notion of the *block–trace* will prove convenient. The block–trace of A, denoted by $\Delta[A]$ is defined as the $q \times r$ matrix

$$\Delta[A] = \sum_{i=1}^{p} A_{ii} , \tag{1.3.2}$$

i.e., the sum of the diagonal blocks of A. We will also use the notation $\Delta_{(p)}[A]$, where the index to the block–trace operator gives the number of blocks over which the sum is taken. Some elementary properties are:

(i) If $q = r = 1$, the block–trace reduces to the 'ordinary' trace.

(ii) For square A (i.e., $q = r$)

(1.3.3) $\quad \text{tr } A = \text{tr}(\Delta[A])$.

(iii) For square A

(1.3.4) $\quad \Delta[A \otimes B] = (\text{tr } A).B$.

(iv) If B is of order $r \times s$, then

(1.3.5) $\quad \Delta[A(I_p \otimes B)] = \Delta[A]\ B$.

(v) If matrix A is a square matrix of order $pq \times pq$, then for any $p \times r$ matrix X:

$$
\begin{aligned}
(1.3.6) \qquad \Delta_{(p)}[A(XX' \otimes I_q)] &= \Delta_{(p)}[A(X \otimes I_q)(X' \otimes I_q)] \\
&= \Delta_{(r)}[(X' \otimes I_q)A(X \otimes I_q)] \\
&= \Delta_{(p)}[(X \otimes I_q)(X' \otimes I_q)A] \\
&= \Delta_{(p)}[(XX' \otimes I_q)A].
\end{aligned}
$$

The notion of the *Kronecker product operator*, denoted by $\otimes$, is discussed in section 2.2. More properties of the block–trace are discussed in Wansbeek (1980).

1.4. THE MOORE–PENROSE INVERSE

The *inverse* of a square $n \times n$ matrix A, denoted by A^{-1}, is defined and unique when A is non–singular, that is when A is of full rank n. A generalization of the concept of invertibility to the case of singular square matrices and non–square (or, e.g., rectangular) matrices while retaining uniqueness is the *Moore–Penrose inverse* (in short, *MP inverse*).

Definition 1.4.1

An $n \times m$ matrix X is the MP inverse of a real $m \times n$ matrix A if it satisfies the four defining conditions

(1.4.1) $AXA = A,$

(1.4.2) $XAX = X,$

(1.4.3) $(AX)' = AX,$

(1.4.4) $(XA)' = XA.$

We shall denote the MP inverse of A by A^+. The existence of A^+ for any matrix A can be proven by using the singular value decomposition of A, such as was done in Magnus and Neudecker (1988). The singular value decomposition will be discussed in section 5.

Several properties of the MP inverse are the following of which an extensive list with proofs may be found in Magnus and Neudecker (1988).

(i) $A^+ = A$, if A is symmetric and idempotent.
(ii) $A^+ = (A'A)^+A' = A'(AA')^+$.
(iii) $A^+ = (A'A)^{-1}A'$, if A has full column rank.
(iv) $A^+ = A'(AA')^{-1}$, if A has full row rank.

1.5. TWO MATRIX DECOMPOSITIONS

In this section we present two decomposition schemes of a matrix. The first decomposition relates to a symmetric matrix, the second to a rectangular matrix. The decompositions will prove useful later in this study.

1.5.1 Eigenvalue-eigenvector decomposition

Let A be a symmetric $n \times n$ matrix. Then there exist an orthogonal $n \times n$ matrix S and a diagonal matrix Λ whose diagonal elements are the eigenvalues of A, such that

(1.5.1) $S'AS = \Lambda.$

1.5.2 Singular value decomposition

Let A be an $m \times n$ matrix with rank r(A)=r. Then there exists an $m \times m$ matrix $S = (S_1 : S_2)$ such that $S_1'S_1 = I_r$, an $n \times n$ matrix $T = (T_1 : T_2)$ such that $T_1'T_1 = I_r$, and an $r \times r$ diagonal matrix Λ with positive diagonal elements, such that

$$A = SDT' = S_1 \Lambda T_1' \tag{1.5.2}$$

with

$$D = \begin{bmatrix} \Lambda & 0 \\ 0 & 0 \end{bmatrix}. \tag{1.5.3}$$

2. KRONECKER PRODUCT AND VEC OPERATOR

2.1. INTRODUCTION

The *Kronecker product*, also known as the direct product or the tensor product, has numerous applications in various fields including statistics, economics, and econometrics. The development of the techniques which use the Kronecker product concept will be seen as a topic within the scope of matrix algebra. We shall see that the Kronecker product and the vec operator are closely connected. The vec operator was, amongst other others, discussed by Neudecker (1969), Vetter (1975), and Wansbeek (1988). Neudecker (1969) uses the vec operator (and differentials) to describe three different definitions of partial derivatives matrices (and shows their pros and cons). Vetter (1975) applies the vec operator to convert linear matrix equations to vector and dimension–reduced vector forms, for which he then describes an applicable solution procedure. Wansbeek (1988) mentions another application to show the usefulness of the vec operator: the variance of a stochastic matrix can be handsomely expressed in vecs. The vec operator transforms a matrix into a vector by stacking its columns one underneath another. Both the Kronecker product and vec operator, viewed as matrix tools, will be elaborately used in the rest of this study and will prove useful for deriving elegant expressions in our multidimensional data analysis models.

2.2. THE KRONECKER PRODUCT

Let A be an $m\times n$ matrix and B a $p\times q$ matrix. The Kronecker product of the two matrices, denoted by $A\otimes B$, is defined as the partitioned $mp\times nq$ matrix

$$A\otimes B \equiv \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix}. \tag{2.2.1}$$

It has mn blocks, the (i,j)-th block is the matrix $a_{ij}B$ of order $p\times q$.

Some properties and rules for the Kronecker product are the following.

(i) If α is a scalar, then

(2.2.2) $\quad A\otimes(\alpha B) = \alpha(A\otimes B).$

(ii) The product is distributive with respect to addition, that is

(2.2.3) $\quad (A + B)\otimes(C + D) = A\otimes C + A\otimes D + B\otimes C + B\otimes D.$

(iii) The product is associative, i.e.,

(2.2.4) $\quad A\otimes B\otimes C = A\otimes(B\otimes C) = (A\otimes B)\otimes C.$

(iv) The transposition rule states

(2.2.5) $\quad (A\otimes B)' = A'\otimes B'.$

(v) If A and B are non-singular matrices, then

(2.2.6) $\quad (A\otimes B)^{-1} = A^{-1}\otimes B^{-1}.$

The same rule actually also holds for the MP inverse, i.e.,

(2.2.7) $\quad (A\otimes B)^{+} = A^{+}\otimes B^{+}.$

(vi) The product rule states

(2.2.8) $\quad (A\otimes B)(C\otimes D) = AC\otimes BD,$

provided the orders of the matrices are such that the various multiplications are valid, or in short, the matrices should be conformable for the operations.

We recognize that the list of properties and rules at present is not exhaustive and we will certainly use more rules in later chapters, at which

instance we will point this out. One more matrix rule we mention is the following.

(2.2.9) $\qquad \text{vec } ABC = (C' \otimes A)\text{vec } B,$

provided the orders of the matrices are such that the matrix product ABC is defined. This last rule shows the close connection between the Kronecker product and the vec operator as presented in the next section.

2.3. THE VEC OPERATOR

The vector valued function of a matrix A, denoted by vec(A) is named the *vec operator*. If A is an $m \times n$ matrix and $a_{.j}$ its j–th column, then vec(A) is an $mn \times 1$ vector

(2.3.1) $$\text{vec } A \equiv \begin{bmatrix} a_{.1} \\ a_{.2} \\ \vdots \\ a_{.n} \end{bmatrix}.$$

The vec operator therefore transforms a matrix into a vector by stacking the columns of the matrix one after the other in one long vector. The close connection between the vec operator and Kronecker product was given by (2.9). The connection between the vec operator and the trace operator is

(2.3.2) $\qquad (\text{vec } A)'\text{vec } B = \text{tr}(A'B),$

where matrices A and B are of the same order.

3. (0,1)-MATRICES AND THEIR PROPERTIES

3.1. INTRODUCTION

In this final chapter of Part One we will discuss some more special matrices which will be extensively used in further chapters. These matrices are of the type of what we call (0,1)–*matrices*. Our extensive use of these matrices is no surprise since we follow the recent upsurge of admiration for the use of these (0,1)–matrices by theoretically oriented scientists in the fields of matrix algebra and data analysis. Specifically, in the econometric literature to these ends we recognize one (0,1)–matrix, the so–called *permutation matrix* as such in much multivariate statistical (theoretical) work. The permutation matrix is alternatively known as *permuted identity matrix* (Balestra (1976)), *vec–permutation matrix* (Henderson and Searle (1981)), *universal flip matrix* (Hartwig and Morris (1975)) or *commutation matrix* (Magnus and Neudecker (1979, 1988), Neudecker and Wansbeek (1983)). We will use the term 'permutation matrix' and will use the term 'commutation matrix' for another (0,1)–matrix. These (0,1)–matrices together with several other ones are discussed in the next sections.

3.2. THE PERMUTATION MATRIX

The operational definition of the permutation matrix $P_{n,m}$ is the basic property:

(3.2.1) $\qquad P_{n,m}\ \text{vec}\ A = \text{vec}\ A',$

where A is an $m\times n$ matrix. A descriptive definition of the permutation matrix $P_{n,m}$ of order $mn\times mn$ is as follows. It is essentially a partitioned matrix of mn blocks each of order $n\times m$, where the (i,j)–th block has an unit element in the (j,i)–th position and zeroes elsewhere. The following properties of the

matrix itself are of interest:

(3.2.2) $\qquad P'_{n,m} = P_{m,n},\ P_{n,m}P_{m,n} = I_{mn},\ P_{n,1} = P_{1,n} = I_n.$

The major properties of the permutation matrix are that it interchanges the two matrices of a Kronecker product and that it enables us to transform the vec of a kronecker product into the Kronecker product of two vecs. These properties involve the following statements.

(3.2.3) $\qquad$ (i) $\quad P_{m,p}(A\otimes B) = (B\otimes A)P_{n,q},$

(3.2.4) $\qquad$ (ii) $\quad P_{m,p}(A\otimes B)P_{q,n} = (B\otimes A),$

(3.2.5) $\qquad$ (iii) $\quad \mathrm{vec}(A\otimes B) = (I_n\otimes P_{m,q}\otimes I_p)(\mathrm{vec}\ A\otimes \mathrm{vec}\ B),$

where matrices A and B are of order $m\times n$ and $p\times q$, respectively. One more property we mention (see e.g., Balestra (1976) and Neudecker and Wansbeek (1983)) is:

(3.2.6) $\qquad (P_{m,p}\otimes I_n)(I_m\otimes P_{n,p}) = P_{mp,n}.$

Don and Van der Plas (1981) discussed the characteristic polynomial of the permutation matrix, which is a correct and simplified version of the expression for the characteristic polynomial by Hartwig and Morris (1975). Holmquist (1985) discussed several matrix expressions of permutation operators on tensor spaces, the so-called direct product permuting matrices, of which the permutation matrix is a special case. The class of direct product permuting matrices generalizes (4), in so far as any permutation of the order of the matrices in a Kronecker product of an arbitrary number of matrices results when pre- and postmultiplied by a member. In this study just one member of this class is needed, the so-called commutation matrix, which will be discussed in section 5.

3.3. THE PERMUTATION MATRIX AND THE WISHART DISTRIBUTION

The permutation matrix also plays a role in distribution theory, especially where it concerns the normal theory. We will state one particular result due to Magnus and Neudecker (1979).

Let us consider k random $n\times 1$ vectors $y_1,\ldots,y_k$, distributed independently as

(3.3.1) $\qquad y_i \sim \mathrm{N}(\mu_i,\ V),$

where V is a positive semidefinite $n\times n$ matrix. The joint distribution of the elements of the $n\times n$ matrix S, where S is defined as

(3.3.2) $\qquad S \equiv \sum_{i=1}^{k} y_i y_i'$

is said to be noncentral Wishart with k degrees of freedom, and is denoted by $\mathrm{W}_n(k,\ V,\ M)$, where M is the $k\times n$ matrix

(3.3.3) $\qquad M = (\mu_1\ \mu_2 \ldots \mu_k)'.$

If $M = 0$ the distribution is said to be central. The mean and variance of the random matrix S which is $\mathrm{W}_n(k,\ V,\ M)$ distributed are

(3.3.4) $\qquad \mathrm{E}(S) = kV + M'M,$

(3.3.5) $\qquad \mathrm{V}(\mathrm{vec}\ S) = (I_{n^2}+P_{n,n})[k(V\otimes V)+V\otimes M'M+M'M\otimes V].$

3.4. THE DUPLICATION MATRIX

For a symmetric $n\times n$ matrix A we denote by $\mathrm{v}(A)$ a $\frac{1}{2}n(n+1)\times 1$ vector which contains only the distinct elements of A, whereas vec A contains all elements of A. There exists a unique $n^2\times\frac{1}{2}n(n+1)$ matrix, denoted by D_n, which transforms, for symmetric A, $\mathrm{v}(A)$ into vec A. For symmetric A we then have the relationship:

(3.4.1) $\quad D_n \, \mathrm{v}(A) = \mathrm{vec}\, A.$

Since D_n has full column rank, the MP inverse equals

(3.4.2) $\quad D_n^+ = (D_n'D_n)^{-1}D_n',$

and $\mathrm{v}(A)$ can be uniquely solved from (1) in which case we get

(3.4.3) $\quad \mathrm{v}(A) = D_n^+ \, \mathrm{vec}\, A.$

Three properties of D_n we state are

(3.4.4) $\quad$ (i) $\quad P_{n,n}D_n = D_n,$

(3.4.5) $\quad$ (ii) $\quad D_nD_n^+ = \frac{1}{2}(I_{n^2}+P_{n,n}),$

(3.4.6) $\quad$ (iii) $\quad D_n' \, \mathrm{vec}\, A = \mathrm{v}(A+A'-\mathrm{dg}(A)),$

where $\mathrm{dg}(A)$ is a diagonal matrix with diagonal elements of A on its diagonal. The last property therefore holds for general (i.e., not specific symmetric) $n \times n$ matrices. The duplication matrix and several properties are more elaborate discussed in Magnus and Neudecker (1988). The duplication matrix was previously defined by Balestra (1976), Browne (1974), and Vetter (1975).

3.5. THE COMMUTATION MATRIX

The commutation matrix, as a generalization to the permutation matrix has recently been studied by Kapteyn, Neudecker and Wansbeek (1986). Their results are summarized below. An operational definition of the commutation matrix, denoted by C_i, is as follows. Let

(3.5.1) $\quad \ell \equiv \prod_{i=1}^{k} \ell_i,$

where ℓ_i denote integers to be of use in the sequel to specify orders of matrices and arrays. Let y be an $\ell \times 1$ vector, where each element is positioned

in the vector through k indices, with the i–th index assuming values from 1 to ℓ_i. The elements are positioned in such a way that the first index runs slowest and the last index runs fastest. For instance, in case of $k = 3$, $\ell_1 = \ell_2 = \ell_3 = 2$, vector y is of structure

$$y = (y_{111}\ y_{112}\ y_{121}\ y_{122}\ y_{211}\ y_{212}\ y_{221}\ y_{222})'.$$

When C_i is applied to y, the running order of the indices (and therefore the arrangement of the elements in y) is changed in such a way that the i–th index will now run fastest and the running order of the other indices remains unaffected. Its descriptive definition is

$$C_i = I_{\ell_1 \times \ldots \times \ell_{i-1}} \otimes P_{\ell_i, \ell_{i+1} \times \ldots \times \ell_k}, \tag{3.5.2}$$

so that $C_1 = P_{\ell_1, \ell^1}$ and $C_k = I_\ell$, where

$$\ell^i \equiv \prod_{j \neq i}^{k} \ell_j = \ell / \ell_i. \tag{3.5.3}$$

Due to the property that $P_{a,b} = P'_{b,a}$ for permutation matrices (see section 2), it follows that

$$C'_i C_i = C_i C'_i = I_\ell. \tag{3.5.4}$$

The operational definition of C_i describes the operation of C_i to a vector y such as given above. By defining the 'anti-vec' matrix Y_i of order $\ell^i \times \ell_i$, the operational definition becomes

$$\mathrm{vec}(Y_i) = C_i y. \tag{3.5.5}$$

For example,

$$y = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{bmatrix}, \; Y_1 = \begin{bmatrix} 1 & 5 \\ 2 & 6 \\ 3 & 7 \\ 4 & 8 \end{bmatrix}, \; Y_2 = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 5 & 7 \\ 6 & 8 \end{bmatrix}, \; Y_3 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{bmatrix},$$

so $\ell = 8$ and $\ell_1 = \ell_2 = \ell_3 = 2$, $\ell^1 = \ell^2 = \ell^3 = 4$. Note that

$$\text{vec } Y_1' = \begin{bmatrix} 1 \\ 5 \\ 2 \\ 6 \\ 3 \\ 7 \\ 4 \\ 8 \end{bmatrix}, \; \text{vec } Y_2' = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 4 \\ 5 \\ 7 \\ 6 \\ 8 \end{bmatrix}, \; \text{vec } Y_3' = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{bmatrix},$$

In this case, the commutation matrices of interest are:

$$C_1 = \begin{bmatrix} 1 & . & . & . & . & . & . & . \\ . & . & . & . & 1 & . & . & . \\ . & 1 & . & . & . & . & . & . \\ . & . & . & . & . & 1 & . & . \\ . & . & 1 & . & . & . & . & . \\ . & . & . & . & . & . & 1 & . \\ . & . & . & 1 & . & . & . & . \\ . & . & . & . & . & . & . & 1 \end{bmatrix}, \; C_2 = \begin{bmatrix} 1 & . & . & . & . & . & . & . \\ . & . & 1 & . & . & . & . & . \\ . & 1 & . & . & . & . & . & . \\ . & . & . & 1 & . & . & . & . \\ . & . & . & . & 1 & . & . & . \\ . & . & . & . & . & . & 1 & . \\ . & . & . & . & . & 1 & . & . \\ . & . & . & . & . & . & . & 1 \end{bmatrix}, \; C_3 = I_\ell \; .$$

There are k different matrices of order $\ell^i \times \ell_i$ of interest in our study, which can be obtained from the stacked k–dimensional array y through use of the commutation matrices C_i, $i = 1,...,k$. One more property is the following. Let A be a matrix structured as a string of Kronecker products, i.e.,

(3.5.6) $\qquad A = A_1 \otimes A_2 \otimes ... \otimes A_k,$

where each matrix A_i is of order $\ell_i \times \ell_i$. Define the $\ell^i \times \ell^i$ matrix A^i

(3.5.7) $\qquad A^i \equiv A_1 \otimes ... \otimes A_{i-1} \otimes A_{i+1} \otimes ... \otimes A_k.$

Then,

(3.5.8) $\quad C_i A C_i' = A^i \otimes A_i.$

Further on we will also use the symbol D_i, $i = 1,...,k$, for commutation matrices (from the context there should be no confusion with the duplication matrix D_n from section 4), this time of order $m \times m$. Here, m is defined as

(3.5.9) $\quad m \equiv \prod_{i=1}^{k} m_i,$

and m^i is defined likewise as ℓ^i in (2). Again, $D_i' D_i = D_i D_i' = I_m$, $i = 1,...,k$. In this case, (8) may be generalized to

(3.5.10) $\quad C_i A D_i' = A^i \otimes A_i,$

for $\ell_i \times m_i$ matrices A_i.

A final piece of notation for commutation matrices involves the matrices C_i^j and D_i^j, of order $\ell^j \times \ell^j$ and $m^j \times m^j$, respectively, which perform the operation

(3.5.11) $\quad C_i^j A^j D_i^{j\prime} = A^{ij} \otimes A_i.$

Matrix A^{ij} is of order $\ell^{ij} \times m^{ij}$ (with $\ell^{ij} \equiv \ell/(\ell_i \ell_j)$, $m^{ij} \equiv m/(m_i m_j)$):

(3.5.12) $\quad A^{ij} \equiv A_1 \otimes ... \otimes A_{i-1} \otimes A_{i+1} \otimes ... \otimes A_{j-1} \otimes A_{j+1} \otimes ... \otimes A_k.$

3.6. THE BLOCK-MUTATION MATRIX

One more (0,1)–matrix of interest is what we call the *block–mutation matrix*. The block–mutation matrix B_i is defined as:

(3.6.1) $\quad B_i \equiv I_n \otimes P_{q,m} \otimes I_p.$

Again it holds that $B_i' B_i = B_i B_i' = I_{mnpq}$. To express its operation, we need two more expressions defined earlier in the literature. The first is the so–called

Tracy – Singh block – by – block Kronecker product (Tracy and Singh (1972)), denoted by the symbol "π": if an $mp \times nq$ matrix A and $su \times tv$ matrix B are partitioned as mn blocks A_{ij}, each of order $p \times q$ and st blocks B_{ij}, each of order $u \times v$, respectively, then

$$A \ \pi \ B \equiv [C_{ij}], \tag{3.6.2}$$

where $pu \times qv$ block–matrix C_{ij} is

$$C_{ij} \equiv \begin{bmatrix} A_{ij} \otimes B_{11} & \cdots & A_{ij} \otimes B_{1t} \\ \vdots & & \vdots \\ A_{ij} \otimes B_{s1} & \cdots & A_{ij} \otimes B_{st} \end{bmatrix} . \tag{3.6.3}$$

The second matrix–expression was introduced by Wansbeek and Verhees (1986), and is denoted by a "tilde" above the matrix it operates upon, hence we say $\tilde{A}$ is the *tilde – transform of A*. The matrix $\tilde{A}$ is obtained from a partitioned $mp \times nq$ matrix A as follows. Vectorize each of the blocks A_{ij}, $i = 1,\ldots,m$, $j = 1,\ldots, n$ according to the vec operator and arrange the thus created vectors next to each other as columns of the matrix $\tilde{A}$. In formula: let $a_{ij} \equiv \text{vec } A_{ij}$. Then, the $pq \times mn$ matrix $\tilde{A}$ is defined as

$$\tilde{A} \equiv (a_{11}, a_{21}, \ldots, a_{m1}, a_{12}, a_{22}, \ldots, a_{m2}, \ldots, a_{1n}, a_{2n}, \ldots, a_{mn}). \tag{3.6.4}$$

The close connection of the block–mutation matrix B_i with the 'π'– product and the tilde–transform is stated through the following properties.

Theorem 3.6.1.

(3.6.5) (i) $B_i \text{ vec } A = \text{vec } \tilde{A}$,

where A is an $mp \times nq$ matrix, and $B_i = I_n \otimes P_{q,m} \otimes I_p$.

(3.6.6) (ii) $B_i(U \otimes V)B_i' = U \ \pi \ V$,

where U and V are $mp \times mp$ matrices, partitioned as m^2 blocks, each block of order $p \times p$, and $B_i = I_m \otimes P_{p,m} \otimes I_p$.

Proof

(i) Let a_{ijh} (p×1), denote the h-th column of A_{ij} ($h = 1,\ldots,q$). We can write

$$A_{ij} = \sum_{h=1}^{q} a_{ijh}e_h' \text{ and vec } A_{ij} = \sum_{h=1}^{q} e_h \otimes a_{ijh},$$

with e_h the h-th unit vector, and

$$A = \sum_{i=1}^{m}\sum_{j=1}^{n} e_i e_j' \otimes A_{ij} = \sum_{i}^{m}\sum_{j}^{n}\sum_{h}^{q} e_i e_j' \otimes a_{ijh} e_h' .$$

Then,

$$\begin{aligned}
(I_n \otimes P_{q,m} \otimes I_p)\text{vec } A &= \sum_{i,j,h} (I_n \otimes P_{q,m} \otimes I_p)(e_j \otimes e_h \otimes e_i \otimes a_{ijh}) \\
&= \sum_{i,j,h} (e_j \otimes e_i \otimes e_h \otimes a_{ijh}) \\
&= \sum_{i,j} (e_j \otimes e_i \otimes \sum_{h} (e_h \otimes a_{ijh})) \\
&= \sum_{i,j} (e_j \otimes e_i \otimes \text{vec } A_{ij}) \\
&= \text{vec} \sum_{i,j} (\text{vec } A_{ij})(e_j \otimes e_i)' \\
&= \text{vec } \tilde{A} .
\end{aligned}$$

(ii) Let $U = \sum_{i_1=1}^{m}\sum_{j_1=1}^{m} (e_{i_1} e_{j_1}' \otimes U_{i_1 j_1})$ and $V = \sum_{i_2=1}^{m}\sum_{j_2=1}^{m} (e_{i_2} e_{j_2}' \otimes V_{i_2 j_2})$

with e_{i_1}, e_{i_2}, e_{j_1}, and e_{j_2} all m×1 vectors.

Then,

$$B_i(U \otimes V)B_i' =$$

$$= \sum_{i_1,j_1,i_2,j_2}^{m,m,m,m} (e_{i_1}\otimes P_{p,m}(U_{i_1,j_1}\otimes e_{i_2})\otimes V_{i_2,j_2})(e'_{j_1}\otimes(I_m\otimes e'_{j_2})P_{m,p}\otimes I_p)$$

$$= \sum_{i_1,j_1,i_2,j_2}^{m,m,m,m} (e_{i_1}\otimes(e_{i_2}\otimes U_{i_1,j_1})\otimes V_{i_2,j_2})(e'_{j_1}\otimes(e'_{j_2}\otimes I_p)\otimes I_p)$$

$$= \sum_{i_1,j_1,i_2,j_2}^{m,m,m,m} (e_{i_1}e'_{j_1}\otimes(e_{i_2}e'_{j_2}\otimes U_{i_1,j_1})\otimes V_{i_2,j_2})$$

$$= \sum_{i_1,j_1,i_2,j_2}^{m,m,m,m} ((e_{i_1}\otimes e_{i_2})(e_{j_1}\otimes e_{j_2})'\otimes(U_{i_1,j_1}\otimes V_{i_2,j_2}))$$

$$\equiv U\,\pi\,V.$$

■

So the matrix $U\,\pi\,V$ may be envisioned as a matrix that consists of m^4 blocks, each of order $m^2\times m^2$. These blocks are the Kronecker products of each of m^4 pairs of the m^2 blocks of order $m\times m$ of U and V. With the last equation but one we have also given the scheme for positioning each of the m^4 possible Kronecker products in the matrix $U\,\pi\,V$; the Kronecker product of the $m\times m$ block $(i_1,\ j_1)$ in matrix U with $m\times m$ block $(i_2,\ j_2)$ in matrix V is an $m^2\times m^2$ block positioned at $((i_1-1)m+i_2,\ (j_1-1)m+j_2)$ in $U\,\pi\,V$, $i_1,j_1,i_2,j_2 = 1,\ldots,m$.

In case of a Kronecker structure of U and V, (5) and (6) will be more transparent. Define matrix $A \equiv X^i\otimes X_i$ as an $mp\times nq$ matrix, with $m\times n$ matrix X^i of structure as in (5.7) and X_i of order $p\times q$. Take matrix $B_i = I_n\otimes P_{m,q}\otimes I_p$.

Corollary 3.6.1.

(3.6.7) $\qquad B_i \text{ vec } A = \text{vec } X^i\otimes\text{vec } X_i.$

Proof

Let x^i_p be the p–th column of X^i. Then

$$\text{vec } A = \text{vec}(x^i_1\otimes X_i,\ldots,x^i_n\otimes X_i)$$

$$= \text{vec}[\text{vec}(x_1^i \otimes X_i), \ldots, \text{vec}(x_n^i \otimes X_i)]$$

$$= \text{vec}[\text{vec}(P_{p,m}(X_i \otimes x_1^i)), \ldots, \text{vec}(P_{p,m}(X_i \otimes x_n^i))]$$

$$= \text{vec}[(I_q \otimes P_{p,m})(\text{vec } X_i \otimes x_1^i), \ldots, (I_q \otimes P_{p,m})(\text{vec } X_i \otimes x_n^i)]$$

$$= \text{vec}[(P_{m,q} \otimes I_p)(x_1^i \otimes \text{vec } X_i), \ldots, (P_{m,q} \otimes I_p)(x_n^i \otimes \text{vec } X_i)]$$

$$= \text{vec}[(P_{m,q} \otimes I_p)(x_1^i \otimes \text{vec } X_i, \ldots, x_n^i \otimes \text{vec } X_i)]$$

$$= (I_n \otimes P_{m,q} \otimes I_p)\text{vec}(x_1^i \otimes \text{vec } X_i, \ldots, x_n^i \otimes \text{vec } X_i)$$

$$= (I_n \otimes P_{m,q} \otimes I_p)(\text{vec } X^i \otimes \text{vec } X_i),$$

from which (7) follows. The fifth step is due to (2.6).

■

Notice that this corollary is also a special result of (2.5), for which a proof can be found in Neudecker and Wansbeek (1983). Define $U \equiv X^i \otimes X_i$, $V \equiv X^i \otimes X_i$, with matrices X^i and X_i of order $m \times m$ and $p \times p$, respectively.

Corollary 3.6.2.

(3.6.8) $$B_i(U \otimes V)B_i' = X^i \otimes X^i \otimes X_i \otimes X_i,$$

if X^i is an $m \times m$ matrix, X_i is a $p \times p$ matrix and $B = I_m \otimes P_{m,p} \otimes I_p$.

Proof

$$B_i(U \otimes V)B_i' = (I_m \otimes P_{m,p} \otimes I_p)(X^i \otimes X_i \otimes X^i \otimes X_i)(I_m \otimes P_{p,m} \otimes I_p)$$

$$= X^i \otimes P_{m,p}(X_i \otimes X^i)P_{p,m} \otimes X_i$$

$$= X^i \otimes X^i \otimes X_i \otimes X_i,$$

where the last step is based on (2.4).

■

Theorem 3.6.2.

(3.6.9) $\quad B_i'(P_{n^i,n^i}\otimes P_{n_i,n_i})B_i = P_{n,n},$

where $n = n^i * n_i$, and $B_i = I_{n^i}\otimes P_{n_i,n^i}\otimes I_{n_i}$.

Proof

Take the Kronecker product $a\otimes b\otimes c\otimes d$, the column vectors being of order n^i, n_i, n^i, and n_i, respectively. Then,

$$B_i'(P_{n^i,n^i}\otimes P_{n_i,n_i})B_i(a\otimes b\otimes c\otimes d) =$$

$$= (I_{n^i}\otimes P_{n^i,n_i}\otimes I_{n_i})(P_{n^i,n^i}\otimes P_{n_i,n_i})(I_{n^i}\otimes P_{n_i,n^i}\otimes I_{n_i})(a\otimes b\otimes c\otimes d)$$

$$= (I_{n^i}\otimes P_{n^i,n_i}\otimes I_{n_i})(P_{n^i,n^i}\otimes P_{n_i,n_i})(a\otimes c\otimes b\otimes d)$$

$$= (I_{n^i}\otimes P_{n^i,n_i}\otimes I_{n_i})(c\otimes a\otimes d\otimes b)$$

$$= (c\otimes d\otimes a\otimes b)$$

$$= P_{n^in_i,n^in_i}(a\otimes b\otimes c\otimes d) = P_{n,n}(a\otimes b\otimes c\otimes d),$$

from which (9) follows.

■

Theorem 3.6.3.

(3.6.10) $\quad P_{n_i,n_i}\,\tilde{A}\,P_{n^i,n^i} = \tilde{A},$

where matrix A is a symmetric $n\times n$ matrix, partitioned as n^{i2} blocks A_{ij}, each

of order $n_i \times n_i$.

Proof

$$\begin{aligned}\mathrm{vec}(P_{n_i,n_i}\,\tilde{A}\,P_{n^i,n^i}) &= (P_{n^i,n^i}\otimes P_{n_i,n_i})\mathrm{vec}\,\tilde{A} = \\ &= (P_{n^i,n^i}\otimes P_{n_i,n_i})(I_{n^i}\otimes P_{n_i,n^i}\otimes I_{n_i})\mathrm{vec}\,A \\ &= (P_{n^i,n^i}\otimes P_{n_i,n_i})(I_{n^i}\otimes P_{n_i,n^i}\otimes I_{n_i})P_{n,n}\mathrm{vec}\,A \\ &= (I_{n^i}\otimes P_{n_i,n^i}\otimes I_{n_i})\mathrm{vec}\,A = \mathrm{vec}\,\tilde{A},\end{aligned}$$

where the second and last step are based on (5), the third step is induced by the symmetry of A, and the fourth step follows from (9).

■

The more general result than (6) is :

Theorem 3.6.4.

(3.6.11) $\quad (I_m\otimes P_{p,s}\otimes I_u)(U\otimes V)(I_n\otimes P_{t,q}\otimes I_v) = U\ \pi\ V,$

where U and V are $mp\times nq$ and $su\times tv$ matrices, respectively.

Proof

Let e_{i_1}, e_{i_2}, e_{j_1}, and e_{j_2} be an $m\times 1$, $s\times 1$, $n\times 1$, and $t\times 1$ vector, respectively, where e_{i_1} indicates the i_1–th unit vector, and so forth. Then,

$$(I_m\otimes P_{p,s}\otimes I_u)(U\otimes V)(I_n\otimes P_{t,q}\otimes I_v) =$$

$$= \sum_{i_1,j_1,i_2,j_2}^{m,n,s,t} (e_{i_1}\otimes P_{p,s}(U_{i_1,j_1}\otimes e_{i_2})\otimes V_{i_2,j_2})(e_{j_1}'\otimes(I_q\otimes e_{j_2}')P_{t,q}\otimes I_v)$$

$$= \sum_{i_1, j_1, i_2, j_2}^{m, n, s, t} (e_{i_1} \otimes (e_{i_2} \otimes U_{i_1, j_1}) \otimes V_{i_2, j_2})(e'_{j_1} \otimes (e'_{j_2} \otimes I_q) \otimes I_v)$$

$$= \sum_{i_1, j_1, i_2, j_2}^{m, n, s, t} (e_{i_1} e_{j_1}' \otimes (e_{i_2} e_{j_2}' \otimes U_{i_1, j_1}) \otimes V_{i_2, j_2})$$

$$= \sum_{i_1, j_1, i_2, j_2}^{m, n, s, t} ((e_{i_1} \otimes e_{i_2})(e_{j_1} \otimes e_{j_2})' \otimes (U_{i_1, j_1} \otimes V_{i_2, j_2}))$$

$$\equiv U \pi V.$$

■

A further elaboration of (1.3.6) which results from using the tilde transform, is stated in the following theorem. However, contrary to (1.3.6), A is no longer required to be a square matrix.

Theorem 3.6.5.

(3.6.12) $\quad \text{vec}(\Delta_{(p)}[A(XX' \otimes I_s)]) = ((x' \otimes I_s)P_{pr,s} \otimes I_q)(I_{ps} \otimes X' \otimes I_q)a,$

where A is a $pq \times ps$ matrix, X is of order $p \times r$, $a \equiv \text{vec } A$, and $x \equiv \text{vec } X$.

Proof

Let $T \equiv XX'$, $t \equiv \text{vec } T$ and $a_{ij} \equiv \text{vec } A_{ij}$. Then

(3.6.13) $\quad t = \text{vec } XX' = (I_p \otimes X)\text{vec } X' = (I_p \otimes X)P_{r,p}x.$

Hence

$$\text{vec}(\Delta_{(p)}[A(XX' \otimes I_s)]) = \text{vec}(\Delta_{(p)}[A(T \otimes I_s)])$$

$$= \text{vec} \sum_i^p \sum_j^p (T)_{ij} A_{ij} = \sum_i^p \sum_j^p (T)_{ij} a_{ij} =$$

$$= (a_{11}, a_{21}, \ldots, a_{pp})\text{vec } T = \tilde{A}t =$$

$$= (t'\otimes I_{qs})\text{vec } \tilde{A}$$

$$= (t'\otimes I_s\otimes I_q)(I_p\otimes P_{s,p}\otimes I_q)a$$

$$= (x'P_{p,r}(I_p\otimes X')\otimes I_s\otimes I_q)(I_p\otimes P_{s,p}\otimes I_q)a$$

$$= (x'P_{p,r}\otimes I_s\otimes I_q)(I_p\otimes X'\otimes I_s\otimes I_q)(I_p\otimes P_{s,p}\otimes I_q)a$$

$$= (x'P_{p,r}\otimes I_s\otimes I_q)(I_p\otimes P_{s,r}\otimes I_q)(I_p\otimes I_s\otimes X'\otimes I_q)a$$

$$= (x'\otimes I_s\otimes I_q)((P_{p,r}\otimes I_s)(I_p\otimes P_{s,r})\otimes I_q)$$

$$(I_p\otimes I_s\otimes X'\otimes I_q)a$$

$$= ((x'\otimes I_s)P_{pr,s}\otimes I_q)(I_{ps}\otimes X'\otimes I_q)a,$$

where the eighth step is due to (5), the nineth step is due to (13) and the last step is due to (2.6).

■

The (0,1)–matrices discussed in this chapter are applicable in what has been named multimode (or k–mode) data analysis, which incorporates k–mode principal component analysis models (Kapteyn et al. (1986)), k–mode factor analysis models (Wansbeek and Verhees (1986)), k–mode (or k–factorial) covariance structure analysis models (Verhees and Wansbeek (1988)), k–mode interdependent regression models (Wansbeek and Buyze (1981)), and k–mode Poisson regression models (Wansbeek and Verhees (1988)). In all these instances we analyze data y that are measured variables in k dimensions or modes, therefore viewed as elements of k–dimensional hyperboxes or arrays with k indices. Part Two of our study will be devoted to analyses of these multimode data analysis models.

PART II

MULTIDIMENSIONAL DATA ANALYSIS MODELS

4. k–MODE PRINCIPAL COMPONENT ANALYSIS

4.1. INTRODUCTION

Although Tucker (1963, 1966) has suggested extending *principal component analysis* (PCA) to four or more modes of classifications, the meaning and structure of a four–mode model until recently has not been articulated. But through exposition of a mathematical extension of Tucker's model and an empirical application, Lastovicka (1981) presented the structure of a four–mode principal component model, together with an extension of Tucker's algorithm to four modes. He applied standard principal component analysis on each of the four combination–mode matrices, and subsequent combination of the three solutions to form the core matrix. From hereafter we will denote this solution technique and the solution itself by the term 'canonical'. In his notation, Lastovicka's starting point consisted of the following fundamental four–mode model:

$$(4.1.1) \qquad y_{i',j',k',l'} \approx \hat{y}_{i',j',k',l'} = \sum_{m'=1}^{m} \sum_{p'=1}^{p} \sum_{q'=1}^{q} \sum_{r'=1}^{r} a_{i',m'} b_{j',p'} c_{k',q'} d_{l',r'} g_{m'p'q'r'}$$

$$i' = 1,\ldots,i,\ j' = 1,\ldots,j,\ k' = 1,\ldots,k,\ l' = 1,\ldots,l,$$

where $a_{i',m'}$, $b_{j',p'}$, $c_{k',q'}$ and $d_{l',r'}$, are elements in *component loadings* matrices ${}_iA_m$, ${}_jB_p$, ${}_kC_q$ and ${}_lD_r$, respectively. The coefficients are $g_{m'p'q'r'}$, elements of the four–mode matrix G, which, following Tucker (1966), is termed the *core matrix*.

Apart from Lastovicka's work cited above, various other related references can be mentioned. Carroll and Chang (1970) present general methods for analyzing multidimensional arrays. Carroll, Pruzansky and Kruskal (1980) extend these results to include linear constraints on the parameters. The

present chapter can be considered to be a k–mode extension of the work of Kroonenberg and De Leeuw (1980) and Sands and Young (1980) on three–mode data, which itself is a further development of Tucker's (1966) work on three–mode principal component analysis. Kroonenberg and De Leeuw (1980) give an algorithm (TUCKALS3) based on what they call *Alternating Least Squares*, which is appropriate for metric data. Sands and Young (1980) derive an algorithm (ALSCOMP) which is quite similar to ALS, but with additional optimal scaling steps. This algorithm is appropriate for data with a wide variety of measurement characteristics. The data may be defined at nominal, ordinal, interval, ratio, or mixed levels, and the measurement process may be discrete or continuous. In this study, we will only concentrate on metric data.

Ten Berge, De Leeuw and Kroonenberg (1987) show that the fitted sum of squares in TUCKALS3 can be partitioned according to elements of each mode in the three–way data array. In addition, a relationship between a special case of TUCKALS3 and the CANDECOMP/PARAFAC model (Harshman and Lundy (1984)) is shown. This special case is where one mode has only one component (i.e., m_i=1) and the two other modes have the same number of modes. When TUCKALS3 has converged to a global minimum, then the core matrix G is a diagonal matrix. It then follows (cf. Ten Berge et al. (1987)) that in this case TUCKALS3 computes a PARAFAC solution.

Tucker's (1963) three–mode PCA model is usually regarded as a model with the assumption of invariant loading patterns, but changing factor scores. There exist many methods to analyze (three–way) longitudinal data next to three–mode PCA. Tenenhaus (1988) gives a detailed example of what can be done on real data by three–mode PCA and generalized canonical analysis (GCA), which is a special case of PCA. Murakami (1979) formulates a quasi three–mode PCA model, which is a simplified version of Tucker's (1963) model, and which is also applicable to longitudinal data. Formally, it is equivalent to Kroonenberg and De Leeuw's (1980) approach, since it intends to decompose the standardized data matrix for each occasion into the product of three matrices, namely, the factor score matrix, the core matrix, and the factor loadings matrix. But it has the additional property that the patterns to which core matrices are restricted, can distinguish the change of factor loadings from the change of factor scores.

Kapteyn, Neudecker and Wansbeek (1986) extended the four-mode model to the k-mode PCA model, where in the former notation it would become increasingly more complex, if not impossible, to keep track of the summation over the proper indices. Kapteyn et al. (1986) introduce similar notation and (0,1)-matrices as discussed in the previous chapters, by which it is demonstrated that the (0,1)-matrix algebra and calculus are fit for handling multimode arrays. Kapteyn et al. (1986) also present an iteration method, denoted by Kroonenberg (1983) as a 'TUCKALSn algorithm' (n=k), leading to an Unweighted Least Squares (ULS) solution of the k-mode PCA model, and they compare it to the canonical solution. From hereafter, we speak of a 'least squares solution' as an ULS solution to the k-mode PCA model based upon Kapteyn et al. (1986).

We investigate the k-mode PCA model of Kapteyn et al. (1986) in section 2. In subsection 3.1, we present the ULS solution as obtained by their ALS approach. The ULS solution is compared with Lastovicka's canonical solution in subsection 3.2. We give a short description of the iterative algorithm to compute the ULS solution in section 4. In section 5 we prove the equivalence of the canonical and least squares solution in case $k = 2$. In section 6 we discuss varimax rotation, which is a method to present solutions in a simple structure. In section 7 we discuss statistical assumptions. In section 8 we discuss ML estimation as an alternative to ALS in case of least squares estimation for the model with a normal sampling distribution. In section 9 we state the asymptotic distribution of the ML estimator. In section 10 we state the asymptotic distribution of a goodness-of-fit test statistic. In section 11 we consider the complexity of expressions that occur in an actual computer program.

Lastovicka (1981) applied his four-mode model and canonical solution method to real data on advertising without giving a thorough interpretation of the results which may concern future advertising data research. The same may be said, to a smaller extent however, for Kapteyn et al. (1986), who next to a revisit to Lastovicka's solution also present a least squares solution to this same model and data but mention little about any equalities or differences (in meaning) of both solutions. In section 12, we intend to compare both solutions as to their ease of interpretation, overall explanation of variance, and consistency with theoretical expectations based upon earlier advertising

studies. Section 12 is for most part based on Verhees (1985). In subsection 12.1, we give a short description of the data concerning viewers' responses to television commercials. Subsection 12.2 contains the resulting component loadings matrices of both solutions, preceded by a short review of results of former advertising studies primarily concerning the possible (mis)interpretations of the components, for the elementary mode of items. Subsection 12.3 contains the core matrices of both the canonical and least squares solution. Subsection 12.4 discusses the goodness-of-fit for both solutions. Section 13 concludes.

4.2. MODEL SPECIFICATION

The k-mode PCA model, defined as a pure data analysis or data-reduction model by Kapteyn et al. (1986), states

(4.2.1) $\quad y = Ab,$

where

(4.2.2) $\quad A \equiv A_1 \otimes \ldots \otimes A_k,$

with matrix A_i an unknown $\ell_i \times m_i$ matrix, $\ell_i \geq m_i$, satisfying $A_i'A_i = I_{m_i}$, $i = 1,\ldots,k$. Let

(4.2.3) $\quad \ell \equiv \prod_{i=1}^{k} \ell_i,$

(4.2.4) $\quad m \equiv \prod_{i=1}^{k} m_i,$

(4.2.5) $\quad \ell^i \equiv \ell/\ell_i,$

and

(4.2.6) $\quad m^i \equiv m/m_i.$

So, $\ell \geq m$ and A is of order $\ell \times m$. Let y be a known $\ell \times 1$ vector, whose elements are identified by k indices, with the i–th index ($i = 1,...,k$) assuming values from 1 to ℓ_i. The elements are arranged in such a way that the first index runs slowly and the last index runs fast. Analogously, let b be an unknown $m \times 1$ vector, also with elements identified by k indices, with the i–th index running from 1 to m_i, ($i = 1,...,k$). The elements are arranged in the same way as the elements of y. The vectors y and b can be considered as vec representations of k–dimensional matrices. It is our aim to estimate A_i, $i = 1,...,k$ and b such that Ab represents y 'as well as possible', whether this is in a least squares sense or canonical sense.

It will be clear that in spite of the normalization $A_i'A_i = I_{m_i}$, the representation $(A_1 \otimes ... \otimes A_k)b$ is not unique, because if T_i, $i = 1,...,k$, are orthonormal $m_i \times m_i$ matrices there holds

$$(4.2.7) \qquad (A_1T_1 \otimes ... \otimes A_kT_k)(T_1' \otimes ... \otimes T_k')b = (A_1 \otimes ... \otimes A_k)b.$$

This lack of identification will be used to pick a convenient representation of the ULS and canonical solutions, when discussing the respective estimates.

4.3. ESTIMATION APPROACHES

4.3.1. The least squares solution

The ULS solution amounts to finding $A_1,...,A_k$, and b such that

$$(4.3.1) \qquad Q \equiv (y - Ab)'(y - Ab)$$

is minimal, subject to $A_i'A_i = I_{m_i}$ and (2.2). The solution to this problem is readily obtained. First, it follows from standard least squares theory that for any choice of $A_1,...,A_k$, the solution for b is (carets indicate ULS solutions)

$$(4.3.2) \qquad \hat{b} = A'y = (A_1 \otimes ... \otimes A_k)'y.$$

Notice that

$$(4.3.3) \qquad y'(A_1\otimes\ldots\otimes A_k)\hat{b} = y'(A_1A_1'\otimes\ldots\otimes A_kA_k')y =$$

$$= y'C_i'C_i(A_1A_1'\otimes\ldots\otimes A_kA_k')C_i'C_iy =$$

$$= (\text{vec } Y_i)'(A^iA^{i\prime}\otimes A_iA_i')\text{vec } Y_i =$$

$$= \text{tr } A_i'Y_i'A^iA^{i\prime}Y_iA_i,$$

for any one index i. It is convenient to define

$$(4.3.4) \qquad S_i \equiv Y_i'A^iA^{i\prime}Y_i.$$

Then Q is equal to

$$(4.3.5) \qquad Q = y'y - \text{tr } A_i'S_iA_i.$$

Minimization of Q subject to $A_i'A_i = I_{m_i}$, $i = 1,\ldots,k$, can be done by differentiating the Lagrangean L:

$$(4.3.6) \qquad L = y'y - \text{tr } A_i'S_iA_i + \text{tr} \sum_{i=1}^{k} F_i(A_i'A_i - I_{m_i}),$$

with F_i a symmetric $m_i\times m_i$ matrix of Lagrange multipliers, with respect to A_i and setting the result equal to zero. This yields:

$$(4.3.7) \qquad S_iA_i - A_iF_i = 0.$$

So a solution is to choose for F_i the diagonal matrix $\hat{F}_i$ containing the m_i largest eigenvalues of S_i (as tr $\hat{F}_i$ = tr $\hat{A}_i'S_i\hat{A}_i$), and for A_i the corresponding orthonormal eigenvectors $\hat{A}_i$. As $Y_i'\hat{A}^i\hat{A}^{i\prime}Y_i\hat{A}_i = \hat{A}_i\hat{F}_i$, $\hat{A}^{i\prime}Y_i\hat{A}_i$ contains the first m_i principal components of $\hat{A}^{i\prime}Y_i$, provided that $m_i \leq m^i$. This necessary condition follows from the fact that $\hat{A}^{i\prime}Y_i\hat{A}_i$ is an $m^i\times m_i$ matrix. (Note that this condition cannot be satisfied when $k = 2$, unless $m_1 = m_2$.)

An interpretation of the ULS solution of A_i from (7) is obtained by observing that $A^iA^{i\prime}$ is idempotent and hence a projection matrix. It projects

onto the space spanned by A^i. So the ULS solution of A_i amounts to a search for principal components, not of Y_i itself, but of its projection onto the space spanned by A^i.

A relative measure for the goodness of fit of the ULS solution naturally follows. Define the $\ell \times 1$ vector of residuals:

$$\hat{e} = \hat{y} - \hat{A}\hat{b}, \tag{4.3.8}$$

then, analogous to (5) we have:

$$\hat{e}'\hat{e} = y'y - \hat{b}'\hat{b} = y'y - \text{tr } \hat{A}_i'\hat{S}_i\hat{A}_i = y'y - \text{tr } \hat{F}_i, \tag{4.3.9}$$

for any one i. So the sum of the m_i largest eigenvalues in the ULS solution is the same for all i. It is obvious to define

$$R^2 \equiv \hat{b}'\hat{b}/y'y = \text{tr } \hat{F}_i/y'y \tag{4.3.10}$$

as the coefficient of determination.

4.3.2. The canonical solution

Lastovicka (1981) proposes a canonical solution $\check{A}_i$, $i = 1,...,k$, where the columns of $\check{A}_i$ are the m_i orthonormal eigenvectors corresponding to the m_i largest eigenvalues of $Y_i'Y_i$. The solution for b is

$$\check{b} = (\check{A}_1' \otimes ... \otimes \check{A}_k')y. \tag{4.3.11}$$

The ULS and the canonical solutions differ only with respect to the A_i. Given A_i, b is the same in both cases. With ULS, we have

$$\text{vec } \hat{B}_i' = D_i\hat{b} = D_i(\hat{A}_i' \otimes ... \otimes \hat{A}_k')y = D_i(\hat{A}_1' \otimes ... \otimes \hat{A}_k')C_i'C_iy = \tag{4.3.12}$$

$$= (\hat{A}^{i\prime} \otimes \hat{A}_i')C_iy = (\hat{A}^{i\prime} \otimes \hat{A}_i') \text{ vec } Y_i' = \text{vec } \hat{A}_i'Y_i'\hat{A}^i.$$

So

(4.3.13) $\hat{B}_i = \hat{A}^{i\prime} Y_i \hat{A}_i.$

By applying the same procedure to (11), the canonical solution for B_i is

(4.3.14) $\check{B}_i = \check{A}^{i\prime} Y_i \check{A}_i.$

Let us write the canonical equation as:

(4.3.15) $Y_i' Y_i \check{A}_i = \check{A}_i \check{F}_i,$

where $\check{A}_i'\check{A}_i = I_{m_i}$. Now, $\check{A}_i$ does not contain the eigenvectors of $Y_i'\check{A}^i\check{A}^{i\prime}Y_i$ but of $Y_i'Y_i$. Hence if for all i except one $A_i = I_{\ell_i}$, the two solutions coincide, as $A^i = I_{\ell^i}$ in that case. Thus, when considering index i, the canonical solution does not involve the data–reduction with respect to any other index.

4.4. AN ITERATION METHOD FOR ALTERNATING ULS

Since S_i according to (3.4) depends on the unknown parameters in A^i the solution to the minimization problem has to be obtained iteratively. An obvious iterative procedure to obtain the ULS solution is described by Kapteyn et al. (1986), and basically comprises the following. At the r–th iteration, estimates of A_1 to A_k, say $\hat{A}_{1(r)}$ to $\hat{A}_{k(r)}$, which are updates of previous estimates $\hat{A}_{1(r-1)}$ to $\hat{A}_{k(r-1)}$, are computed. This is done in an alternating sequence. To compute a new estimate $\hat{A}_{i(r)}$ of A_i, we go through the following steps.

1. Compute $\hat{A}^i_{(r-1)}$ on basis of $\hat{A}_{1(r-1)},\ldots,\hat{A}_{i-1(r-1)},\hat{A}_{i+1(r-1)},\ldots,\hat{A}_{k(r-1)}$.
2. Form matrix $Y_i'\hat{A}^i_{(r-1)}\hat{A}^{i\prime}_{(r-1)}Y_i$ and its eigenvalue–eigenvector decomposition.
3. Retain the m_i eigenvectors of $Y_i'\hat{A}^i_{(r-1)}\hat{A}^{i\prime}_{(r-1)}Y_i$, corresponding to the m_i largest eigenvalues, as columns of $\ell_i \times m_i$ matrix $\hat{A}_{i(r)}$.
4. Raise index i with one, and repeat steps 1 to 3. The process is continued until convergence.

We make three remarks to this alternating process, which is called a TUCKALS algorithm, following Kroonenberg and De leeuw (1980). First, we use previously updated values, if possible, in step 1. For instance, to compute $\hat{A}_{k(r)}$ by steps 1 to 3 we use $\hat{A}_{1(r)}$ to $\hat{A}_{k-1(r)}$ instead of $\hat{A}_{1(r-1)}$ to $\hat{A}_{k-1(r-1)}$, respectively, in step 1. Of course, for instance, for updating $\hat{A}_1$ in the r-th iteration, we will have to use $\hat{A}_{2(r-1)}$ to $\hat{A}_{k(r-1)}$ in step 1. Second, take $\check{A}_2$ to $\check{A}_k$ as starting values for A_2 to A_k at the first iteration, and then begin steps 1 to 3 with index $i = 1$. Third, convergence of the iterations follows from the following considerations. Criterion Q defined in (1) is quadratic and consequently non-negative. For each i, the solution for A_i corresponding to (7) minimizes Q, for any value of A^i. So, in the above iteration process, each newly computed A_i lowers the value of Q or leaves it unaffected. Thus, we obtain a non-increasing sequence of values of Q which is bounded from below by zero. As a result, the sequence converges.

Kroonenberg and De Leeuw (1980) use the following "fixed point" theorem to prove that the TUCKALS algorithm converges to a minimum.

Theorem 4.4.1. (d'Esopo's (1959) convergence theorem)

Let F, $p_i \equiv \mathrm{tr}(A_i'S_iA_i) = \mathrm{tr}(F_i)$, and S satisfy the following conditions.

1. a. $S \equiv \{s | s=(A_1,...,A_k),\ A_i \in \mathbb{R}^{\ell_i \times m_i},\ A_i$ columnwise orthonormal$\}$ is a compact subset of a finite dimensional space,
   b. F is a continuous transformation of S to S, where F defines a complete step of the main iteration,
   c. p_i is a real function defined and continuous for all $s \in S$,
2. $p_i(F(s)) \leq p_i(s)$,
3. if $p_i(F(s)) = p_i(s)$, then $F(s) = s$,
4. if the sequence $s_0,s_1,...$ satisfies $p_i(s_{i+1}) \geq p_i(s_i)$ with $s_i \in S$, then for every limit point $\bar{s}$ of $s_0,s_1,...$, $F(\bar{s}) = \bar{s}$.

The conditions of Theorem 1 also apply to the ALS algorithm described above, as may be seen from the discussions of these in Kroonenberg and De Leeuw (1980). As S is a bounded real subspace, any infinite sequence $s_0,s_1,...$ is bounded, and thus the sequences generated by the algorithm are bounded as well. A theorem due to Weierstrass shows that such sequences have at least one

accumulation point. It is shown in the Appendix to Kroonenberg and De Leeuw (1980) that every point $\bar{s}$ such that $F(\bar{s}) = \bar{s}$ is a stationary point of p_i and since at every step p_i increases, the stationary points will not be minima.

The question of achieving convergence numerically is a different matter. Numerically, we can consider convergence to be achieved at the t–th iteration of the ALS algorithm, when a new estimate of A_i is obtained, if

$$(4.4.1) \qquad \frac{\|\hat{A}_i^{(t)} - \hat{A}_i^{(t-1)}\|}{\|\hat{A}_i^{(t)}\|} \leq \varepsilon_1,$$

$$(4.4.2) \qquad \frac{|Q(\hat{A}_i^{(t)}) - Q(\hat{A}_{i-1}^{(t)})|}{|Q(\hat{A}_{i-1}^{(t)})|} \leq \varepsilon_2,$$

$$(4.4.3) \qquad \|y - \hat{y}^{(t)}\| \equiv \|y - \hat{A}^{(t)}\hat{b}^{(t)}\| \leq \varepsilon_3,$$

$$(4.4.4) \qquad \frac{|\mathrm{tr}(\hat{F}_i^{(t)}) - \mathrm{tr}(\hat{F}_{i-1}^{(t)})|}{|\mathrm{tr}(\hat{F}_{i-1}^{(t)})|} \leq \varepsilon_4,$$

with ε_1, ε_2, ε_3, and ε_4 small errors, preferably close to machine–precision if it is troublesome to analyze numerically the restrictions set to the limit of precision as dictated by the precision of the data. Note that in the expressions above, $A_0^{(t)} = A_k^{(t-1)}$ and $F_0^{(t)} = F_k^{(t-1)}$.

As for the empirical application, we relied merely on (4). Convergence towards the optimum for alternating ULS was rapid. The iteration was terminated as soon as the left–hand side of (19) was less than $\varepsilon_4 = 5.10^{-8}$. This occurred with $\mathrm{tr}(F_4)$ at the end of the tenth round of iterations. (Notice that $\mathrm{tr}(F_i)$ depends on all A_i. In general it will therefore only remain constant between iterations if all A_i remain constant.)

4.5. EQUIVALENCE OF CANONICAL AND LEAST SQUARES SOLUTION FOR k = 2

The canonical and least squares solution coincide for the situation of a general number of modes k if for all i except one, $A_i = I_{\ell_i}$. This was also mentioned in subsection 3.2. However, for $k = 2$ the two solutions also coincide.

Theorem 4.5.1.

For $k = 2$,

(i) $\hat{A}_i = \check{A}_i$, $i = 1$ or 2,

(ii) $\hat{b} = \check{b}$.

Proof

(i) As $k = 2$, we have $Y_2 = Y_1'$. The order condition $m_i \leq m^i$ leads to the fact that $m_1 = m_2$. Due to the canonical equations (3.15) for Y_1 and $Y_2 = Y_1'$, the singular value decomposition (see section 1.5.2) for $\ell_2 \times \ell_1$ matrix Y_1 is

$$(4.5.1) \qquad Y_1 = \begin{bmatrix} \check{A}_2 & R_2 & S_2 \end{bmatrix} \begin{bmatrix} \check{F}_1 & 0 & 0 \\ 0 & F_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \check{A}_1' \\ R_1' \\ T_2' \end{bmatrix}$$

with $\check{A}_2$, R_2, $\check{A}_1$, and R_1 of order $\ell_2 \times m_1$, $\ell_2 \times (r-m_1)$, $\ell_1 \times m_1$, and $\ell_1 \times (r-m_1)$, respectively. According to (1.5.2), for $S_1 \equiv (\check{A}_2 : R_2)$ and $T_1 \equiv (\check{A}_1 : R_1)$ it holds that $S_1'S_1 = T_1'T_1 = I_r$. Here, $r = \mathrm{r}(Y_1) = \mathrm{r}(Y_2)$. Matrix F_{22} is of order $(r-m_1) \times (r-m_1)$ and contains strictly positive values on its diagonal and zeroes elsewhere. Matrices R_1 and R_2 incorporate eigenvectors of Y_1Y_1' that correspond to the positive eigenvalues in F_{22}. From here on let us impose that $m_1 = m_2 = r$, such that we can ignore matrices R_1, R_2 and F_{22} in the further analysis.

The ULS equations (3.7) imply

$$(4.5.2) \qquad Y_2'\hat{A}_1\hat{A}_1'Y_2\hat{A}_2 = \hat{A}_2\hat{F}_2.$$

By starting the ALS algorithm with $\hat{A}_1 = \check{A}_1$, we have according to (2)

(4.5.3) $\quad Y_2'\check{A}_1\check{A}_1'Y_2A_{2(1)} = A_{2(1)}F_{2(1)}$

which leads to

(4.5.4) $\quad Y_1\check{A}_1\check{A}_1'Y_1'A_{2(1)} = A_{2(1)}F_{2(1)}.$

Substituting (1) in (4) we get

(4.5.5) $\quad \check{A}_2\check{F}_1\check{A}_1'\check{A}_1\check{A}_1'\check{A}_1\check{F}_1\check{A}_2'A_{2(1)} = \check{A}_2\check{F}_1\check{F}_1\check{A}_2'A_{2(1)} = A_{2(1)}F_{2(1)},$

whereas by substituting (1) in (3.15) for $i=2$ we get

(4.5.6) $\quad \check{A}_2\check{F}_1\check{F}_1\check{A}_2'\check{A}_2 = \check{A}_2\check{F}_2.$

Equation (5) is of the form as in (6) which postulates an eigenvalue–eigenvector decomposition of the full rank (m_1) matrix $\check{A}_2\check{F}_1\check{F}_1\check{A}_2'$. As a result, $\hat{A}_{2(1)} = \check{A}_2$, i.e., the first round estimate of $\hat{A}_2$ is $\hat{A}_{2(1)} = \check{A}_2$. As a result, $\hat{A}_{1(1)} = \check{A}_1$.

(ii) As $\hat{A}_i = \check{A}_i$, (3.13) and (3.14) lead to $\hat{B}_i = \check{B}_i$, i.e., $\hat{b} = \check{b}$.

■

4.6. VARIMAX ROTATION

As was pointed out in section 3.2, there is a freedom of rotation of the 'loading' matrices A_1 to A_k and vec of core matrix b. This lack of uniqueness of the solution provides us with the means of representing estimates $\hat{A}_1$ to $\hat{A}_k$ (and $\hat{b}$) by simple structures. One wellknown rotation method is based on Kaiser's (1958) varimax criterion. Kaiser (1958) defines the varimax criterion as the maximum contrast between the columns of A_i. Maximum contrast gives the maximum simplicity of the structure, and maybe also ease of interpretation, of a solution $\hat{A}_i$.

A matrix formulation of Kaiser's varimax criterion is given by Magnus and Neudecker (1988), Neudecker (1981a), and Sherin (1966). Nevels (1986) discusses a direct solution of the pairwise rotations that are induced in Kaiser's varimax rotation. Ten Berge (1984) shows that the maximization of the varimax function, which is the varimax rotation problem (cf. Neudecker (1982)), can be interpreted as a problem of simultaneously diagonalizing symmetric matrices in the least squares sense. Archer and Jennrich (1973) give formulas for standard errors of rotated component loadings. Clarkson (1979) discusses jackknifing for estimation of the standard errors.

We will use varimax rotation to derive simple structures of the estimates when we discuss an empirical application in section 12.

4.7. STATISTICAL ASSUMPTIONS

We describe two ways to introduce stochastic nature in the pure data–reduction model, after which we can discuss ML estimation, iterative solving techniques, and asymptotic properties of estimators. The first way is to treat one mode asymmetrically to the others by assuming that one mode of the k–mode PCA model is the 'individuals' mode which is affected by random sampling. A model which justifies the least squares or ML procedure then is the following. Replace ℓ_k by N to denote the number of individuals as observed in a sample of size N. Implicitly we assume that the k–th mode is the 'individuals' mode. Define the model

$$Y_k = A^k G_k + E_k, \tag{4.7.1}$$

where the matrices Y_k, A^k, G_k and E_k are of order $\ell^k \times N$, $\ell^k \times m^k$, $m^k \times N$, and $\ell^k \times N$, respectively. The N columns of matrix Y_k indicate observations on N individuals. Matrix G_k is structured as

$$G_k = B_k A_k', \tag{4.7.2}$$

with B_k and A_k of order $m^k \times m^k$, and $N \times m^k$, respectively. The vec representation of (1) is

(4.7.3) $\quad \text{vec } Y_k = (A_k \otimes A^k)\text{vec } B_k + \text{vec } E_k.$

Using the permutation matrices and commutation matrices we get the statistical model

(4.7.4) $\quad y = Ab + \varepsilon.$

If the columns of Y_k are i.i.d. with mean zero and covariance matrix Σ, then the error term ε has mean zero and covariance matrix $I_N \otimes \Sigma$. If we assume the elements of ε to be normally distributed, then the least squares procedure provides ML estimates of A and b.

Another way to introduce stochastics in the model (2.1) is to assume that we observe N independent replications of the $\ell \times 1$ data vector y, i.e., N replications of the data array Y_k. We then essentially assume a sample with N observations on ℓ entities, i.e., where the combination of all modes is affected by random sampling. If the sample size N becomes large enough, the experiment may also be viewed (asymptotically) as a draw from a multivariate normal distribution, i.e.,

(4.7.5) $\quad \sqrt{N}\ (\bar{y}-Ab) \xrightarrow{L} N(0,\Sigma),$

where it is now assumed that $V(\varepsilon) = \Sigma$ is known. Vector $\bar{y}$ is

(4.7.6) $\quad \bar{y} \equiv (\bar{y}_1,\ldots,\bar{y}_\ell)'$

with

(4.7.7) $\quad \bar{y}_j = \frac{1}{N} \sum_{r=1}^{N} y_{jr},$

where subscript j indicates the j-th element of $\ell \times 1$ vector y. The log-likelihood reduces to (ignoring constants and signs)

(4.7.8) $\quad L = \text{tr}(\bar{y}-Ab)'\Sigma^{-1}(\bar{y}-Ab) = (\bar{y}-Ab)'\Sigma^{-1}(\bar{y}-Ab),$

which for $\Sigma = I$ conforms to the ULS criterion.

The advantage of this second approach to stochastics is that we need not bother with treating the modes asymmetrically as imposed by the first approach. We therefore follow the second approach.

4.8. NUMERICAL SOLUTION OF LIKELIHOOD EQUATIONS

We explicitly treat the restrictions $A_i'A_i = I_{m_i}$ by using the Lagrange multiplier method. We will follow the discussion by Aitchison and Silvey (1958) of the Lagrange multiplier method for ML estimation of parameters subject to constraints.

Matrix Λ_i is a symmetric matrix of Lagrange multipliers. As the relationships $A_i'A_i = I_{m_i}$ induce symmetry, matrices Λ_i can be chosen as such. Let

$$\theta^0 = ((\text{vec } A_1^0)', (\text{vec } A_2^0)', \ldots, (\text{vec } A_k^0)', b^{0\prime})' \tag{4.8.1}$$

be the vector of true, but unknown parameters, of order $p = \sum_{i=1}^{k} \ell_i m_i + m$. We suppose that θ^0 satisfies the $r \leq p$ relationships as described by $A_i'A_i = I_{m_i}$, $i = 1,\ldots,k$. In this case, $r = \frac{1}{2}\sum_{i=1}^{k} m_i(m_i+1)$, since we have r independent restrictions of a total m_i^2 restrictions.

The constrained ML estimator of θ^0, provided it exists, is the vector $\hat{\theta}$ which minimizes $L(\theta)$ and satisfies $h(\theta) = 0$, with

$$h(\theta) = (h_1'(\theta),\ldots,h_k'(\theta))' \tag{4.8.2}$$

where h_1 to h_k describe the r independent constraints. In this case,

$$h_i(\theta) = E_i((I_{m_i} \otimes A_i')\text{vec } A_i - \text{vec } I_{m_i}) = 0, \tag{4.8.3}$$

where E_i is an $\frac{1}{2}m_i(m_i+1) \times m_i^2$ elimination matrix.

Denote by λ_i the vec of the nonredundant elements of the symmetric matrix Λ_i,

(4.8.4) $\lambda_i = E_i \text{vec } \Lambda_i,\ i = 1,...,k.$

By the first-order conditions for obtaining an optimum of the Lagrange function, there exists a vector $\hat{\lambda} = (\hat{\lambda}_1',...,\hat{\lambda}_k')'$ of Lagrange multipliers such that

(4.8.5) $$\frac{\partial L}{\partial \theta}\Big|_{\theta=\hat{\theta}} - M(\hat{\theta})\hat{\lambda} = 0,$$

(4.8.6) $$h(\hat{\theta}) = 0,$$

where $\frac{\partial L}{\partial \theta}$ is the gradient vector of the log-likelihood L, and $M(\hat{\theta})$ denotes the $p \times r$ matrix $\left[\frac{\partial h}{\partial \theta'}\right]'$, evaluated in point $\hat{\theta}$.

To solve (5) and (6) numerically we will need an iterative procedure. By taking first-order derivatives of the Lagrange function, we obtain the conditions which state the existence of matrices $\hat{\Lambda}_1$ to $\hat{\Lambda}_k$ in addition to the parameter estimates, such that

(4.8.7) $$(\bar{Y}_i - \hat{A}^i\hat{B}_i\hat{A}_i)'\hat{A}^i\hat{B}_i - 2\,\hat{A}_i\hat{\Lambda}_i = 0,$$

(4.8.8) $$\hat{A}'\Sigma^{-1}(\bar{y} - \hat{A}\hat{b}) = 0,$$

(4.8.9) $$\hat{A}_i'\hat{A}_i - I_{m_i} = 0,$$

$i = 1,...,k$. Equation (7) is equivalent to

(4.8.10) $$(\hat{B}_i'\hat{A}^{i\,\prime} \otimes I_{\ell_i})C_i\Sigma^{-1}(\bar{y} - \hat{A}\hat{b}) - 2(I_{m_i} \otimes \hat{A}_i)\text{vec } \hat{\Lambda}_i =$$

$$= \hat{Z}_i'\Sigma^{-1}(\bar{y} - \hat{A}\hat{b}) - 2(I_{m_i} \otimes \hat{A}_i)\text{vec } \hat{\Lambda}_i = 0$$

with Z_i implicitly defined. According to (5),

$$(4.8.11)\qquad \frac{\partial L}{\partial \theta}|_{\theta=\hat{\theta}} = \begin{bmatrix} \hat{Z}_1'\Sigma^{-1}(\bar{y}-\hat{A}\hat{b}) \\ \vdots \\ \hat{Z}_k'\Sigma^{-1}(\bar{y}-\hat{A}\hat{b}) \\ A'\Sigma^{-1}(\bar{y}-\hat{A}\hat{b}) \end{bmatrix} = Z'\Sigma^{-1}(\bar{y}-\hat{A}\hat{b})$$

where

$$(4.8.12)\qquad Z \equiv (Z_1,\ldots,Z_k,A).$$

Also,

$$(4.8.13)\qquad M(\hat{\theta}) = 2\begin{bmatrix} (I_{m_1}\otimes\hat{A}_1)E_1^+ & & \\ & \ddots & \\ & & (I_{m_k}\otimes\hat{A}_k)E_k^+ \end{bmatrix}$$

with $E_i^+ = (E_i'E_i)^{-1}E_i'$. Assuming that $\bar{y}$ is such that the likelihood equations (5) and (6) have a solution $(\hat{\theta}, \hat{\lambda})$ and that $\theta^{(0)}$ is an initial approximation to $\hat{\theta}$ such that $\|\theta^{(0)} - \hat{\theta}\|$ is small, a Taylor series expansion of $\frac{\partial L}{\partial \theta}|_{\theta=\hat{\theta}}$ and $h(\hat{\theta})$ up to second order is

$$(4.8.14)\qquad \frac{\partial L}{\partial \theta}|_{\theta=\hat{\theta}} = \frac{\partial L}{\partial \theta}|_{\theta=\theta^{(0)}} + \frac{\partial^2 L}{\partial\theta\partial\theta'}|_{\theta=\theta^{(0)}}(\hat{\theta}-\theta^{(0)})$$

and

$$(4.8.15)\qquad h(\hat{\theta}) = h(\theta^{(0)}) + M(\hat{\theta})(\hat{\theta}-\theta^{(0)}),$$

respectively. Then, approximately, we have

$$(4.8.16)\qquad \begin{bmatrix} -\frac{1}{N}\frac{\partial^2 L}{\partial\theta\partial\theta'}|_{\theta=\theta^{(0)}} & M(\theta^{(0)}) \\ M'(\theta^{(0)}) & 0 \end{bmatrix}\begin{bmatrix} \hat{\theta}-\theta^{(0)} \\ \frac{1}{N}\hat{\lambda} \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\frac{\partial L}{\partial\theta}|_{\theta=\theta^{(0)}} \\ h(\theta^{(0)}) \end{bmatrix}.$$

If N is large enough then $\hat{\theta}$ is near $\theta^{(0)}$. Consequently, $-\frac{1}{N}\frac{\partial^2 L}{\partial\theta\partial\theta'}|_{\theta=\theta^{(0)}}$ approximates $-\frac{1}{N}\frac{\partial^2 L}{\partial\theta\partial\theta'}|_{\theta=\theta^0}$ which in turn approximates the information matrix $I(\theta^0)$,

$$(4.8.17) \qquad I(\theta^0) \equiv \mathrm{E}\left[\frac{\partial L}{\partial \theta}\left(\frac{\partial L}{\partial \theta}\right)'\right]_{|\theta=\theta^0} = Z'\Sigma^{-1}Z$$

with Z in (12) evaluated in point θ^0. Then by similar heuristic arguments, we can replace (16) by

$$(4.8.18) \qquad \begin{bmatrix} Z'(\theta^{(0)})\Sigma^{-1}Z(\theta^{(0)}) & M(\theta^{(0)}) \\ M'(\theta^{(0)}) & 0 \end{bmatrix}\begin{bmatrix} \hat{\theta}-\theta^{(0)} \\ \frac{1}{N}\hat{\lambda} \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\frac{\partial L}{\partial \theta}|_{\theta=\theta^{(0)}} \\ h(\theta^{(0)}) \end{bmatrix},$$

i.e.,

$$(4.8.19) \qquad B^*(\theta^{(0)})\begin{bmatrix} \hat{\theta}-\theta^{(0)} \\ \frac{1}{N}\hat{\lambda} \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\frac{\partial L}{\partial \theta}|_{\theta=\theta^{(0)}} \\ h(\theta^{(0)}) \end{bmatrix},$$

where the bordered information matrix $B^*(\theta^{(0)})$ is implicitly defined.

Due to the freedom of rotation (cf. (4.2.7)), the ML estimator $\hat{\theta}$ will not be identified. This lack of identification is reflected in a loss of rank of the information matrix $I(\theta)$ in (17) which is part of the bordered information matrix $B^*(\theta^{(0)})$ in (19). To solve for the lack of identification, we impose $\frac{1}{2}m_i(m_i-1)$ restrictions on A_i, by setting $\frac{1}{2}m_i(m_i-1)$ elements of A_i equal to zero. The way to treat these restrictions is by using an elimination matrix $\bar{E}$, of order $(p+r-q)\times(p+r)$, to eliminate $q \equiv \frac{1}{2}\sum_{i=1}^{k} m_i(m_i-1)$ rows and columns of $B^*(\theta)$.

As a result, (19) should be replaced by

$$(4.8.20) \qquad \bar{E}B^*(\theta^{(0)})\bar{E}'\bar{E}\begin{bmatrix} \hat{\theta}-\theta^{(0)} \\ \frac{1}{N}\hat{\lambda} \end{bmatrix} = \bar{E}\begin{bmatrix} \frac{1}{N}\frac{\partial L}{\partial \theta}|_{\theta=\theta^{(0)}} \\ h(\theta^{(0)}) \end{bmatrix}.$$

From (20) follows that, more generally, $\theta^{(t)}$ and $\lambda^{(t)}$ are defined by

$$(4.8.21)\qquad \bar{E}\begin{bmatrix}\theta^{(t+1)}\\ \frac{1}{N}\lambda^{(t+1)}\end{bmatrix} = \bar{E}\begin{bmatrix}\theta^{(t)}\\ 0\end{bmatrix} + \left(\bar{E}B^{*}(\theta^{(t)})\bar{E}'\right)^{-1}\bar{E}\begin{bmatrix}\frac{1}{N}\frac{\partial L}{\partial\theta}|_{\theta=\theta^{(t)}}\\ h(\theta^{(t)})\end{bmatrix},$$

$t = 1,2,...,$ where $B^{*}(\theta^{(t)})$ indicates that $B^{*}(\theta)$ is evaluated at point $\theta^{(t)}$ which is the estimate of θ at the t–th iteration step. We expect the sequences $(\theta^{(t)},\lambda^{(t)})$ to converge in most practical situations. If these sequences converge, they converge to a solution of the likelihood equations since $\bar{E}(B^{*}(\theta^{(t)}))\bar{E}'$ is nonsingular.

The iterative procedure to (21) involves the inversion of a large matrix. However, we can replace the procedure by a method that involves inversion of only one matrix which is a slight modification of Newton's method (see, e.g., Bard (1974), Dahlquist, Björck and Anderson (1974)). The sequences $(\theta^{(t)},\lambda^{(t)})$ are defined by

$$(4.8.22)\qquad \bar{E}\begin{bmatrix}\theta^{(t+1)}\\ \frac{1}{N}\lambda^{(t+1)}\end{bmatrix} = \bar{E}\begin{bmatrix}\theta^{(t)}\\ \frac{1}{N}\lambda^{(t)}\end{bmatrix} + \left(\bar{E}B^{*}(\theta^{(0)})\bar{E}'\right)^{-1}\bar{E}\begin{bmatrix}\frac{1}{N}\frac{\partial L}{\partial\theta}|_{\theta=\theta^{(t)}} - M(\theta^{(t)})\frac{1}{N}\lambda^{(t)}\\ h(\theta^{(t)})\end{bmatrix}$$

$t = 0,1,...,$ where $\theta^{(0)}$ and $\lambda^{(0)}$ are $\theta^{(1)}$ and $\lambda^{(1)}$, respectively, as obtained by the first iteration of (21).

As most common iterative methods, the schemes (21) and (22) rely on approximations that are only valid in the neighbourhood of $\hat{\theta}$, and one must try to respect this condition in setting the starting values $\theta^{(0)}$. The worst may be a choice of starting values such that the parameter vector goes off in the wrong direction, as there may be parameters values θ at which $\bar{E}(B^{*}(\theta^{(t)}))\bar{E}'$ is numerically near–singular (in case of ML estimation this is accompanied by movements away from the maximand). In the k–mode PCA model with a Kronecker product structure (2.2) between the parameter matrices A_1 to A_k, and Kronecker products in first– and second–order derivatives, the choice of starting values for A_1 to A_k from a set of admissible starting values, is of importance to circumvent numerical instabilities. We will discuss this point further in chapter 8 on the k–mode Poisson regression model which will turn out to be related to the PCA model. In section 8.6, several simulations are made to

investigate the numerical instabilities that may arise for k–mode Poisson regression. A better approximation is usually obtained by reflecting on the various implications of several (starting) values of parameters, and selecting what seems plausible.

4.9. ASYMPTOTIC DISTRIBUTION OF THE ML ESTIMATOR

The joint asymptotic distribution of the ML estimators is given in the next theorem.

Theorem 4.9.1.

The joint asymptotic distribution of $\sqrt{N}\,\bar{E}\begin{bmatrix}\hat{\theta}-\theta^{(0)}\\ \hat{\lambda}\end{bmatrix}$ is multivariate normal with zero mean and covariance matrix

$$(4.9.1)\qquad \left[\bar{E}B^{*}(\theta^{(0)})\bar{E}'\right]^{-1}\bar{E}\begin{bmatrix}Z^{0\prime}\Sigma^{-1}Z^{0} & 0\\ 0 & 0\end{bmatrix}\bar{E}'\left[\bar{E}B^{*}(\theta^{(0)})\bar{E}'\right]^{-1}.$$

Proof

The proof is similar to the proofs found in Don (1982, 1985), and is therefore omitted.

■

Don (1982, 1985) uses the calculus of generalized inverses to the general restricted ML problem with a full rank bordered information matrix, and a singular bordered information matrix, respectively. Don (1985) also gives an expression for the Moore–Penrose inverse of the bordered information matrix, for which a general result is stated by Hung and Markham (1975).

4.10. GOODNESS-OF-FIT STATISTIC

As was already mentioned in section 7, a model which would justify the ULS procedure is (7.4). If we assume the elements of ε to be normally distributed, the ULS procedure provides ML estimators of A and b. Given this model and the assumption of normality, we can carry out an asymptotic F-test to see whether estimation results are consistent with the data (cf., e.g. Goldfeld and Quandt (1972)). The only ingredients we need for this test are the R^2 values or sum of squares of residuals $e'e$ for the canonical and ULS solutions. The goodness-of-fit statistic is

$$F \equiv \frac{(\check{e}'\check{e} - \hat{e}'\hat{e})/s}{\hat{e}'\hat{e}/(\ell - s)}, \tag{4.10.1}$$

where $s = \sum_{i=1}^{k} \ell_i^2 + \ell_1$, if mode 1 indicates the mode which was subjected to errors.

4.11. PROGRAMMING CONSIDERATIONS

It is in the interest of the programmer to develop the necessary procedures to handle expressions (8.11) and (8.21) for any choice of the number of modes, k. Each of these expressions depends on one or more sizable matrices. This alerts the programmer to routinely write procedures which do the work at a level of straightforward, but exhaustive matrix multiplication. For instance, an expression $E = A(B \otimes C)$ should not be solved as product $E = AD$, where first $D = B \otimes C$ is calculated and stored as intermediate result. This is an inefficient way of programming since it imposes extra restrictions on computer memory access. We call this 'programming at face value'. A more efficient approach, in terms of computer memory-access, is to construct these expressions with possible sizable matrices at a scalar level. We would then use the following scheme:

Set matrix E equal to null;

```
for i_A := 1 to row_A do
  for j_B := 1 to col_B do
    for j_C := 1 to col_C do
      for i_B := 1 to row_B do
        for i_C := 1 to row_C do
(4.11.1)      E[i_A, (j_B-1)*col_C+j_C] := E[i_A, (j_B-1)*col_C+j_C] +
                A[i_A, (i_B-1)*row_C+i_C]*B[i_B, j_B]*C[i_C, j_C].
```

What this means in case of the k–mode PCA model is more elaborately discussed in chapter 9.

4.12. EMPIRICAL APPLICATION: ADVERTISING DATA

Earlier studies in advertising research concerning the quantification of consumers' subjective reactions to television commercials (Leavitt (1970), Wells, Leavitt and McConville (1971), Schlinger (1979), and Aaker and Bruzzone (1981), are based upon classical principal components analyses in one plane of data matrices, even though many advertising data matrices can be seen as multidimensional hyper–planes. Lastovicka (1981) however, recognizes the importance of the remaining modes of advertising data, develops a model for four–mode PCA and shows that the four–mode model is applicable to real data. According to Lastovicka (1981), "... The developed model, which is an extension of Tucker's three–mode factor analysis model, allows for the simultaneous analysis of all modes of a four–mode data matrix and the consideration of relationships among the modes ..." (p. 47). This proposal of simultaneity however, is somewhat misleading since Lastovicka's canonical solution technique to four–mode data consists of four separate principal components analyses. The alternating ULS algorithm, does recognize the simultaneity in the right sense, by iterative and alternate solving of classical principal components analyses. In this section, we use Lastovicka's four–mode data to demonstrate the resulting differences and equalities of interpretation of Lastovicka's canonical solution and the least squares solution. To this end, we recompute the canonical solution.

4.12.1. Data and starting values

To illustrate the four-mode model as more than an exercise in matrix algebra, Lastovicka (1981) applied the model to data, consisting of a set of individuals' responses to repetitive advertising stimuli on a battery of items. More specifically, $\ell_1 = 27$ individuals were exposed to a set of $\ell_2 = 6$ different television advertisements, on $\ell_3 = 5$ separate occasions. After each exposure to an ad the individuals responded to a battery of $\ell_4 = 16$ items designed to measure viewer reaction to television advertising. These 16 items used 1–6 'strongly disagree – strongly agree' scales, and may be read in Table 2 below. Since we intend to revisit Lastovicka's solution, we follow his choice of the number of components to retain in each mode. We therefore choose $m_1 = 4$, corresponding to four idealized individuals; $m_2 = 3$, corresponding to three psychological exposures; $m_3 = 2$, leading to two basic types of ads, and $m_4 = 3$, leading to three latent scales of the items. This data set was standardized such that the 16×16 matrix $Y_4'Y_4$ is a correlation matrix. The factor $1/(27)^{1/2}$ mentioned in Lastovicka (1981) on page 51 should read $1/(810)^{1/2}$.

4.12.2. Review and results for the items mode

Before we may present the results concerning the elementary mode of items, we discuss some history of advertising data research in which we will stress the studies by Leavitt (1970), Wells et al. (1971), Schlinger (1979), and Aaker and Bruzzone (1981). As we will see, the components in the items matrix of the canonical solution of Lastovicka diverge from the results of these former studies, but the items matrix of the least squares solution shows more resemblance with results of these studies.

We will then also present the resulting component loadings matrices for the elementary modes of exposures of both solutions together with a comparison to an earlier 'three-exposure theory' by Krugman (1972). We will see that both solutions support this theory. Concerning the varimax rotated component loadings matrices for the elementary modes of advertisements and individuals we will merely remark that both solutions show a same picture. Both solutions suggest two advertisement types to be at hand, one type consisting of airline and automobile ads, the other consisting of a pair of ads for a single product, namely an automobile insurance. Both solutions suggest four types of

individuals to be at hand. Since the individual and advertisement mode loadings matrices of both solutions would add little to the current analysis, they will not be presented. We could say more if more was known about the 27 individuals interviewed. We would be able to typify the four idealized individuals more precisely if background information was known about these individuals such as to personal income, car–ownership, number of holiday trips a year, etc.

Before we present the results concerning the elementary mode or items, we will stress some former studies in the history of advertising data research. The Leavitt (1970) and Wells et al. (1971) studies were Leo Burnett Co. efforts (Leo Burnett Company; an advertising and communications research institute, settled in Chicago) to profile viewers' responses to television commercials, starting with 150 statements, drawn from 1000 actual words or short phrases that consumers might conceivably be able to use in giving their opinions of TV commercials. This reduction of 100 actual words or phrases to 150 statements takes place in the Leavitt (1970) and Wells et al. (1971) studies by using four and three sequential principal component analyses respectively, as filtering techniques. For instance, the final analysis in both studies, employed a new set of raters, a new set of commercials, and the words that loaded highest in the preceding principal component analysis. The raters were housewives, recruited from Chicago suburbs, or in other instances just visitors to the Prudential building, from various parts of the USA, who were persuaded to contribute a few minutes of their vacation time.

These earlier efforts, therefore, relied on forced exposure, more elaborate scales than the later study by Aaker and Bruzzone (1981) (Leavitt, for instance, used the unipolar scales instead of the bipolar format of the semantic differential), and personal interviews. The same may be said for another Leo Burnett Co. effort, results of which were presented by Schlinger (1979), which also differed from the early Wells et al. (1971) effort. It resulted in a rating instrument, the Viewer Response Profile, that included just 32 statements. By using five sequential principal components analyses with varimax rotation, the reduction of 150 surviving words and phrases of the earlier efforts to these 32 more basic themes became possible. The Aaker and Bruzzone analysis, in contrast, did not involve any Leo Burnett commercials. Their database involves a representative sample of 524 prime–time television

commercials shown under normal exposure, using a probability mail sample. As seen from Table 1, which is taken from Aaker and Bruzzone (1981), these former studies to profile viewers' responses to television commercials show a high degree of consistency concerning major factors, and it distinctively shows that the first two factors in each of the studies, 'entertainment' and 'personal relevance', are pronounced. The 'dislike' and 'warm' factors also appeared in all of these studies, although in the Schlinger (1979) study the 'dislike' ('alienation') factor did not appear in two of the five sequential principal components analyses. But the degree of consistency is remarkable, when one keeps in mind that the three sets of studies used different databases and methodologies.

Table 4.12.1. Comparing components across studies[a].

Aaker and Bruzzone (1981)		Wells et al. (1971) Leavitt (1970)		Schlinger (1979)[d]
entertaining	(37.6%)	energetic/amusing	(55%)	entertainment
personal relevance	(26.3%)	personal relevance	(22%)	relevant news
dislike	(7.2%)	dislike	(2%)	alienation[b]
warm	(4.3%)	sensual	(9%)	empathy
		novel	(3%)	confusion
		familiar[c]	(5%)	brand reinforcement
		authoritative[c]	(2%)	familiarity[b]

a) The percentage of variance explained is in parentheses.

b) Did not emerge in two of the five sequential principal components analyses, used as means of reducing the number of words or phrases from 150 to 32.

c) Considered unstable since they did not appear in several sequential principal components analyses of both Wells et al. and Leavitt studies.

d) Unfortunately, Schlinger (1979) does not present any percentages of variance explained. But all seven components in total explain 78% of the variance.

We will now present our recomputation of the canonical solution and the least squares solution concerning the elementary mode of items. We then interpret both solutions and make a comparison concerning their consistency with the results in the aforementioned studies.

Table 2 contains the varimax rotated items components as obtained by our recomputation of a canonical solution and the computation of an ULS solution. According to Lastovicka (1981), the first principal component may be interpreted as a scale measuring 'personal evaluation', component 2 as a scale measuring 'comprehension', and component 3 as a scale measuring 'emotive response'. When we follow his labelling and ordering of components, it would show a perfect fit for the classification of the statements under each of these labels. From Table 2, we then notice that items 3, 2, 8, 6, 11 and 14 load on his scale measuring 'personal evaluation', items 5, 7, 12, 16 and 15 load on his scale measuring 'comprehension', and items 1, 10, 13, 4 and 9 load on his scale measuring 'emotive response'. This does show a perfect fit when we look at the content of each statement. We may for instance point the reader to the items 1, 10 and 13 where a significant shift of loadings has arisen, indicating that these items 1, 10, and 13 will be classified for the least squares solution under the component 'entertainment' in the same way as is shown in Schlinger's study (1979; Table 1, p. 40).

However, we notice from Table 2, in contrast to Lastovicka's conclusion, the varimax rotated item components as obtained by a canonical solution are not similar to the major components as found in the former studies. We also see from Table 2 that the least squares solution shows a better similarity with these components found in the former studies, although the number of correct classifications is still not large.

Furthermore, based upon these findings and further shifts occurring at items 2, 6, 8 and 14, such that these load less on the first component, we postulate that the first component seems to tap 'entertainment' (including 'empathy'), the second component taps ' personal relevance' (which includes 'informative' and 'effective') and the third component indicates 'dislike/alienation' (which includes 'familiarity' and 'confusion'). Based upon this postulate, the correct number of classifications of both solutions may be read from Table 3, which supports our preference for the least squares solution. Note that, according to the Viewer Response Profile analysis of Schlinger (1979), item 15 should fall in the category of statements representing the dimension 'confusion' of response to television commercials, and item 8 in the category of statements representing the dimension 'alienation'.

Another interesting point is to look at the percentages of variance explained by the item components of both solutions, as these were also given in former studies (see Table 4). We make two remarks. Firstly, these percentages can be compared to those in Table 1 even though in this work we use Lastovicka's four-dimensional data which resulted from, for instance, no new set of just 27 raters, no new set of commercials, and forced exposure, after each separate occasion. Secondly, the considerably higher percentage of total variance explained and the percentage of variance explained by the item component 'entertainment' of the least squares solution in comparison to the canonical solution, is of course in part due to to the fact that the least squares method is based on maximization of the total variance explained.

Table 4.12.2. Varimax rotated component loadings matrices for the elementary mode of items[a]

Item no.	Item	Canonical[b] 1	2	3	Least squares[b] 1	2	3
1	The commercial was lots of fun to watch and listen to.	8	0	−44	33	2	−10
10	I thought the commercial was clever and quite entertaining.	6	4	−43	32	4	−5
13	The ad was not just selling– it was entertaining me. I appreciated that.	5	0	−45	33	3	−5
3	I felt as though I was right there in the commercial experiencing the same thing.	31	−4	−14	37	−2	10
2	During the commercial I thought how the product might be useful for me.	44	−4	−5	33	−8	−6
8	The ad did not have anything to do with me or my needs.	−44	−6	−21	−23	−5	−12
6	The commercial was meaningful to me.	45	4	4	32	4	10
11	The commercial gave me a good idea.	31	−1	−13	34	2	4
14	As I watched I thought of reasons why I would buy or not buy the product.	42	−4	4	26	−15	−13
5	I clearly understood the commercial.	14	37	7	12	35	13
7	The commercial was too complex. I wasn't sure what was going on.	0	−54	−1	−5	−59	1
12	I was not sure what was going on in the commercial.	−1	−53	−1	−3	−53	2
16	The commercial went by so quickly that it just did not make an impression on me.	7	−41	7	11	−34	22
15	I was so busy watching the screen, I did not listen to the talk.	8	−34	2	18	−25	36
4	I have seen this commercial before.	10	6	37	0	17	73
9	I have seen this commercial so many times that I am tired of it.	1	−5	43	−20	−5	44

a) A higher loading, in absolute sense, represents a higher correlation between item and factor.

b) Loadings have been multiplied by 100 and rounded.

Table 4.12.3. Number of correct classifications of items[a)].

	Canonical solution			Least squares solution		
Item	1	2	3	1	2	3
	1	0	2	4	0	3

a) Based upon our postulate and the table of statements mentioned in a study by Schlinger (1979; Table 1, p. 40).

Table 4.12.4. Comparing percentages of variance explained by the item components of both solutions.

Item	Canonical solution	Least squares solution
1. Entertainment	18.2%	18.7%
2. Personal relevance	6.9%	7.2%
3. Dislike/alienation	3.9%	6.4%
Total variance explained	29.0%	32.3%

We now present results for the remaining three elementary modes. Our conclusions concerning the elementary modes of advertisements and individuals are no different from Lastovicka's propositions. We only differ in respect to the computation and interpretation concerning the mode of exposures, as may be read below.

Table 5 contains the rotated exposure mode component matrices of the canonical solution and the least squares solution. The loading patterns of both solutions are consistent with Krugman's (1972) 'three exposure theory', but not in the sense as Lastovicka (1981) proposed it to be. According to Lastovicka (1981), it seemed appropriate to interpret the columns headed 'I_p^*', 'II_p^*' and 'III_p^*' in his Table 4 as Krugman's first, second and third psychological exposures, respectively. Unfortunately, our recomputation of

Lastovicka's results shows that the columns headed 'I_p^*' and 'II_p^*' in his Table 4 should be interchanged. Moreover, there are two minor points of difference. Firstly, there were differences in the last digit of various figures throughout all tables, presumably due to rounding errors. Secondly, root 2 of items mode (in Lastovicka's Table 1) should read 2.247 instead of 2.271.

Table 4.12.5. Varimax rotated component loadings matrices for the elementary mode of exposures[a].

Number of advertising exposure	Canonical solution[b]			Least squares solution[b]		
	I	II	III	I	II	III
1	–2	99	3	–1	100	2
2	84	8	–20	88	4	–14
3	53	–10	27	44	–7	35
4	7	4	64	10	2	57
5	–4	–1	69	–12	0	73

a) Loadings have been multiplied by 100 and rounded.

b) Higher loadings, in absolute sense, indicate higher 'loadings' on type of psychological exposure.

Krugman (1972) proposed a theory of advertising effectiveness implying that about three exposures are optimal, whereby he characterized the first exposure as dominated by a cognitive 'what is it' type of response, the second exposure dominated by a more personally evaluative 'what of it' type of response, and the third exposure as both a reminder and also the beginning of disengagement. Ehrenberg (1974) found similar results from other advertising data. To do more justification to later work of Krugman, we refer to Krugman's (1977) article in which he gave an enlightening presentation of the possible misunderstanding of the terms '(actual) perception', and '(psychological) exposure'. To do justice to both authors, we have followed Lastovicka's renaming of 'perception' as 'actual exposure' and 'exposure' as 'psychological exposure'.

We see that both the canonical and least squares solution support this theory when we note that the first actual exposure promotes the second type of psychological exposure (denoted by II) in which the informational role of advertising is stressed, the second and third exposure stress the persuasive role of advertising (denoted by I) and fourth and fifth actual exposures cause the third type of psychological exposure (denoted by III) whereby the reinforcing role of advertising is stressed.

4.12.3. The core matrices

Corresponding to the number of components retained for the individual, exposure, advertisement and item modes, the core matrices of both solutions are of order 4×3×2×3. The four-dimensional core matrices G, transformed to reflect varimax rotations on all four modes, are shown in Table 6. That is, the core matrix elements indicate the reactions (on three scales) of four different types of idealized viewers to two different groups of ads at three psychological exposure levels. Notice that most of the entries in Lastovicka's core matrix in Table 5 of his paper are based upon products of incorrect entries in his component loadings matrices and we therefore place some question marks to his conclusions. But we were also unable to reproduce the entries that are unaffected by the error, i.e., those in the two columns headed 'emotive response' in his Table 5. The overall magnitudes of the entries in this Table are too low to be correct. A rough check is provided by computing the sum of squares of the entries, i.e., $\check{b}'\check{b}$ in our notation. This yields 0.069 or, cf. (3.10), R^2 = 0.004, an improbably low value. In fact, $\check{b}'\check{b}$ = 4.634 according to our computations.

The entries in both core matrices and their shifts in values can be easier interpreted when displayed in two-dimensional plots, such as in Figures 1a to 3b. Inspection of Figures 1a and 1b shows a few points. Firstly, the canonical solution indicates an almost uniform change of entertainment scores of each individual to both advertisement groups, whereas the least squares solution shows a different picture. The canonical solution therefore shows uniform response of each individual to both advertisement groups. But the least squares solution clearly shows individual type 1 feels more content as to his awareness of entertainment by watching ads of groups 1, and to feel less entertained by commercials of group 2.

Table 4.12.6. Varimax transformed core matrices $_{p'q'r'}\check{G}_{m'}$ and $_{p'q'r'}\hat{G}_{m'}$ [a)]

Psyc. exp.	Adv. group	Resp. scale	Idealized individual 1st Can.	1st L.s.	2nd Can.	2nd L.s.	3rd Can.	3rd L.s.	4th Can.	4th L.s.
1st	Airl./autom.	1:Entertain	−26	−64	43	16	−15	−34	−25	5
		2:Relevance	−25	−33	25	13	−12	33	44	23
		3:Dislike	38	13	−27	−13	4	−13	10	8
	Insurance	1:Entertain	13	1	39	40	−27	−10	−3	35
		2:Relevance	−11	−18	24	9	−4	32	39	29
		3:Dislike	−14	2	−3	−13	32	−14	−18	3
2nd	Airl./autom.	1:Entertain	−24	−65	38	−11	5	−37	−20	9
		2:Relevance	−12	−14	29	8	−4	17	27	33
		3:Dislike	56	64	−41	28	−43	23	32	−3
	Insurance	1:Entertain	−5	−15	29	4	10	−18	−8	27
		2:Relevance	−4	−3	16	5	−2	21	26	29
		3:Dislike	5	29	−15	15	2	6	1	−3
3rd	Airl./autom.	1:Entertain	−26	7	−24	−67	41	10	4	3
		2:Relevance	16	26	−7	−2	−4	−38	−41	−12
		3:Dislike	−5	6	3	21	−33	27	2	−20
	Insurance	1:Entertain	−48	−54	−18	−73	44	−18	−12	−35
		2:Relevance	14	21	−8	1	−4	−25	−22	−10
		3:Dislike	38	15	−10	17	−50	24	35	−6

a) The symbols '$\check{}$' and '$\hat{}$' refer to the canonical and least squares solution, respectively. Core matrix values presented have been multiplied by 100 and rounded.

Secondly, both solutions indicate individual differences in perceptions of the two different types of advertisements. For instance, whereas individual type 2 became less amused with additional exposures to both ads, respondent type 3's reactions to the ads types show a reverse trend.

Thirdly, both solutions show a less profound change in values between first and second exposure than between second and third exposure. This may indicate that perception of (non)entertainment occurs between second and third psychological exposure. The same observation, a more profound change in reaction intensities between second and third exposures, also shows with respect to perception of personal relevance. This may be read from Figures 2a and 2b. Both the canonical and least squares solution show that in every pair of respondent and advertisement type, perception of message content relevance was most intense between second and third psychological exposures. A second observation is the same movement of reaction intensities to the relevance of both product types for all individuals. Thus, all respondents viewed both advertisement groups to be related concerning their relevance. This should come as no surprise, since two products promoted, automobiles and automobile insurance are complementary commodities.

The dislike/alienation reaction scores, as presented in Figures 3a and 3b, show more monotonically non-increasing or non-decreasing values in case of the least squares solution than in case of the canonical solution. We also see the same movement of reaction intensities in case of the least squares solution.

Both observations support the least squares solution in favour of the canonical solution, since the former conforms to Krugman's finding that the third exposure functions as both a reminder and the beginning of disengagement. Figure 3b shows this reinforcing role of advertising most clearly, and furthermore shows the high correlation between both advertisement groups. Figure 3a shows that the canonical solution is not able to live up to these theoretical expectations.

Figure 4.12.1a. Entertainment reaction intensities of the canonical solution for advertisement groups 1 and 2.

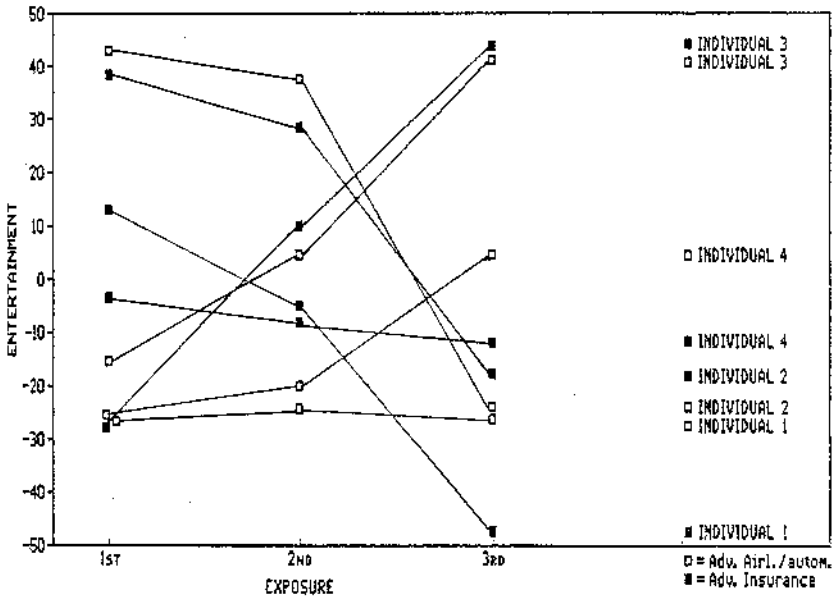


Figure 4.12.1b. Entertainment reaction intensities of the least squares solution for advertisement groups 1 and 2.

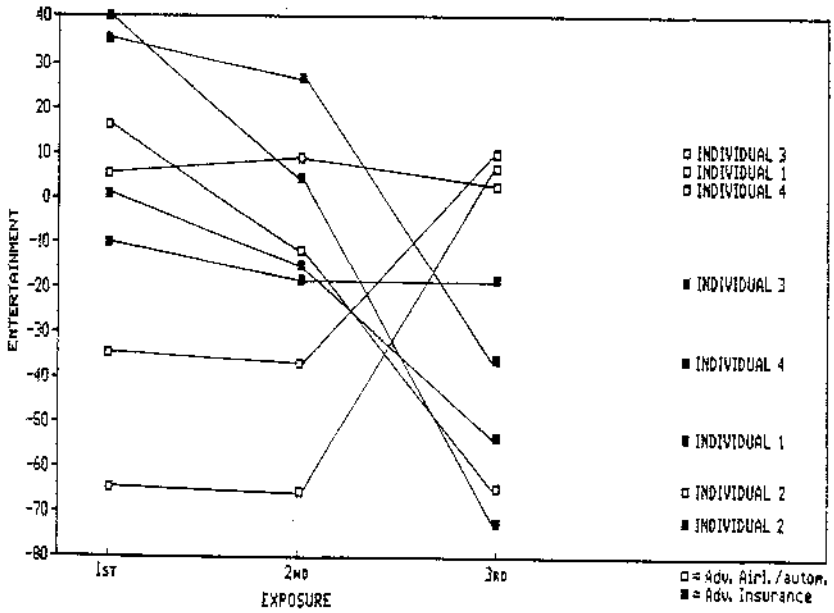


Figure 4.12.2a. Personal relevance reaction intensitics of the canonical solution for advertisement groups 1 and 2.

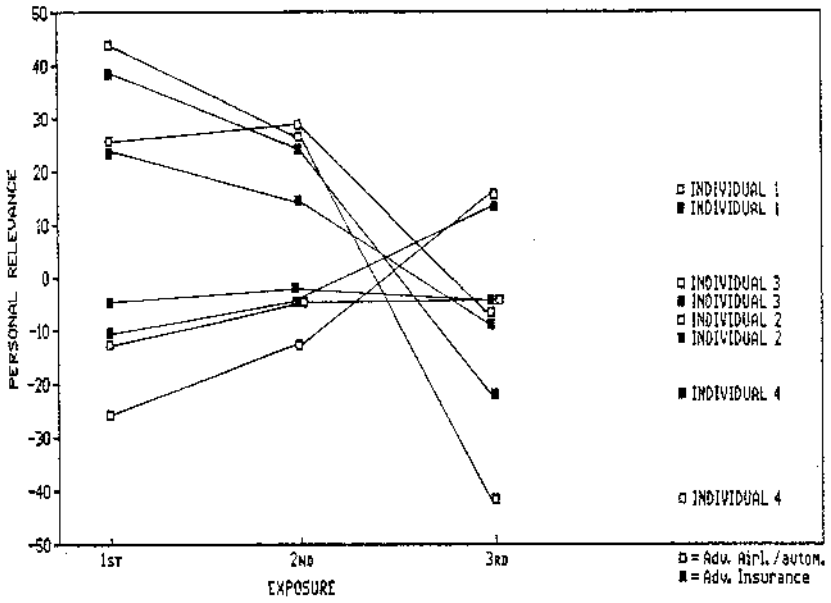


Figure 4.12.2b. Personal relevance reaction intensities of the least squares solution for advertisement groups 1 and 2.

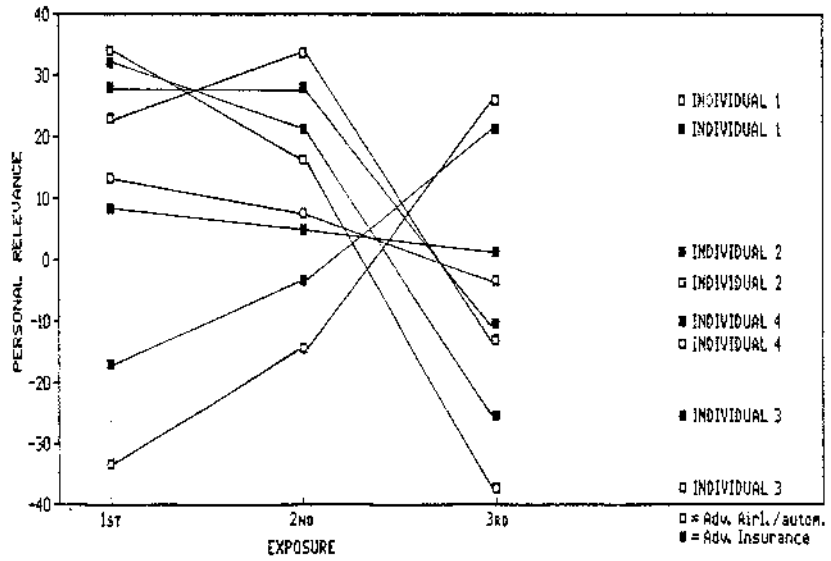


Figure 4.12.3a. Dislike/alienation reaction intensities of the canonical solution for advertisement groups 1 and 2.

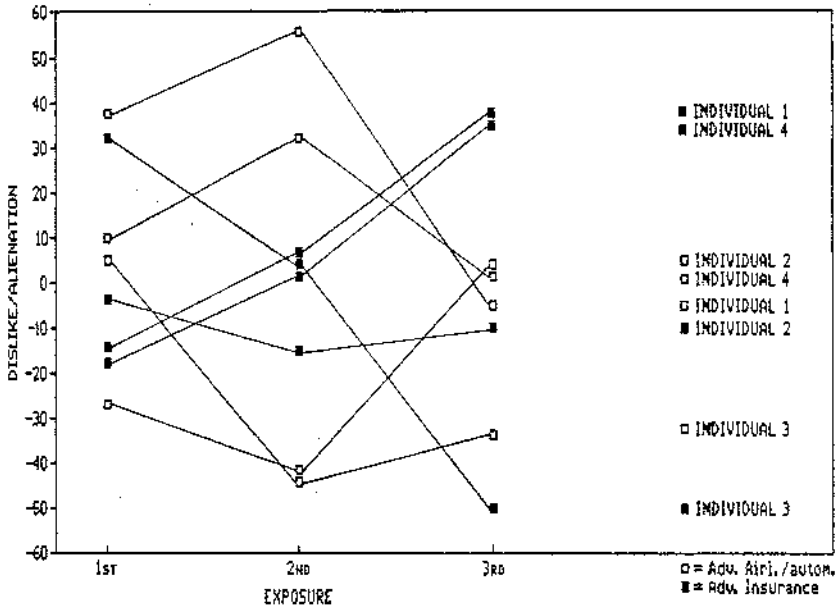
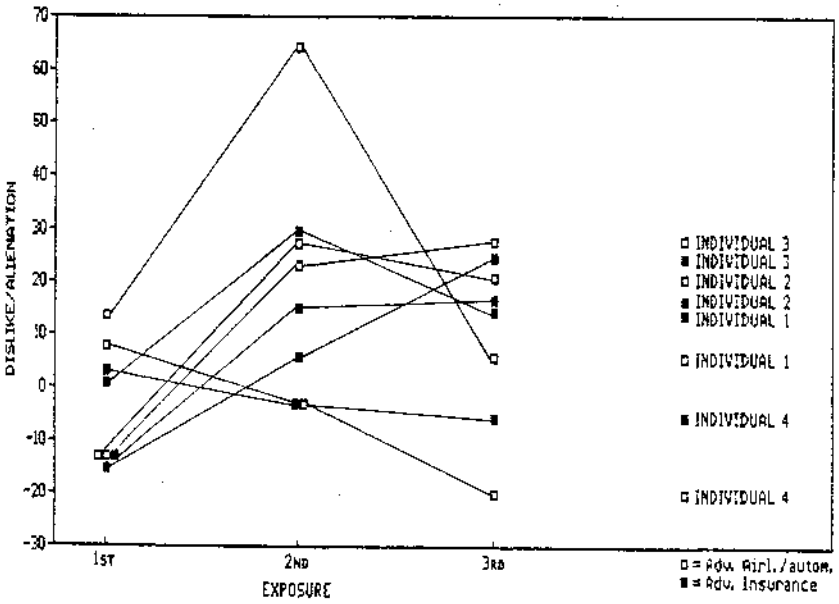


Figure 4.12.3b. Dislike/alienation reaction intensities of the least squares solution for advertisement groups 1 and 2.



4.12.4. Goodness of fit

The most interesting question is of course how the ULS method compares to the canonical one with respect to the fit; see Table 7, where we show how $y'y$ (= 16, as the 16×16 matrix $Y_4'Y_4$ is a correlation matrix, and $y'y = \text{tr } Y_i'Y_i$ for all i) is split up in an "unexplained" part $(\hat{e}'\hat{e})$ and an "explained" part $(\hat{b}'\hat{b})$, cf. (3.9). It appears that R^2 goes up from 0.290 to 0.323 when replacing the canonical method by ours. The overall level of explanation may not seem to be very high, but one has to keep in mind that the job of reproducing the 27×6×5×16 = 12960 elements of y has to borne by the 4×2×3×3 = 72 elements of the core matrix, listed in $\hat{b}$.

Table 4.12.7. ULS versus the canonical method: some results.

Method	$y'y$	$e'e$	$b'b$	R^2
ULS	16.000	10.835	5.165	0.323
Canonical	16.000	11.366	4.634	0.290

As we have ULS as our loss function, our method scores, of course, higher on this criterion then the canonical one. We compute the asymptotic F–distributed goodness–of–fit statistic (10.1) to see whether Lastovicka's results are consistent with the data. We find that $F(1073, 11887) = 0.54$, which is not significant. Thus, according to the statistical model (7.4), both the canonical and the ULS solution are compatible with the data. Yet, the results differ quite a bit between both procedures. This suggests that, at least in this data set, not all parameters in A and b can be estimated accurately. With the statistical model (7.4), and standard non–linear least squares theory, it is possible to obtain standard errors of all parameters, much in the same way as was done for the restricted ML estimator in sections 8 and 9. As a result, we then have an indication of the reliability of the parameter estimates. With the canonical solution, it is not clear how this reliability could be gauged.

4.13. CONCLUSION

In this chapter we describe an alternating ULS and canonical estimation approach to multi–mode PCA. The underlying k–mode PCA model is applied to four–mode advertising data, and both the alternating ULS and canonical solution are stated. The use of the notation based on (0,1)–matrices allows for easy handling of the algebra involved, making derivations simple and transparent.

According to Lastovicka (1981), the structure of his four–mode model is such that it allows for the simultaneous analysis of all modes of classification. Lastovicka (1981) shows the practical applicability of his four–mode model in advertising data research, using existing four–mode data based upon viewer perceptions of repetitive advertising. The result is a solution to the four–mode principal components analysis model. Lastovicka's usage of the term simultaneity, however, is somewhat unclear since Lastovicka's canonical solution technique to four–mode data consists of four separate principal components analyses. An existing alternating ULS algorithm, which is described by Kapteyn et al. (1986) for the more general k–mode PCA model, does recognize the simultaneity in the right sense, by iterative and alternate solving of ordinary principal components problems.

Next to a recomputation and interpretation of the canonical solution to the component analysis of four–mode data, an interpretation of the least squares solution is given. Our recomputation of the canonical solution using Lastovicka's data, results in a core matrix with different entries regarding Lastovicka's core matrix, so that our interpretation of entries in the loadings matrices and the core matrix is also different from Lastovicka's conclusions. For instance,

- we found that the canonical solution does not conform, in part, with theoretical expectations based upon results of earlier advertising studies mentioned above. This in contrast to Lastovicka's observation.
- the model's representation of the data is not as adequate as Lastovicka (1981) proposed. Our computations show that the sum of squares of entries in the core matrix is 4.634 which is significantly lower than Lastovicka's proposal of approximately 14.08.

The least squares solution, in contrast, does conform with our theoretical expectations. Secondly, the model's representation of the data is greater than by the canonical solution, in terms of higher percentages of (overall) explained variation of computed principal components for each of the four elementary modes. We conclude that the least squares solution is also preferable to Lastovicka's canonical solution, since the two-dimensional plots distilled from both resulting core matrices show a more defensible picture for the least squares solution.

5. k– MODE FACTOR ANALYSIS

5.1. INTRODUCTION

In its standard form, "ordinary" factor analysis (OFA) is a method that gives values to the matrix A ($\ell \times m$ with $m < \ell$), the matrix Γ ($m \times m$) and the ℓ diagonal elements of the non–negative diagonal matrix Ψ ($\ell \times \ell$), such that

(5.1.1) $$\Sigma \equiv A\Gamma A' + \Psi$$

reproduces an observed covariance matrix S ($\ell \times \ell$) in some good way. In factor analysis the matrices A, Γ and Ψ are named factor loadings matrix, core matrix and unique variances matrix, respectively. When no core matrix Γ is present, or $\Gamma = I_m$, then model (1) reduces to

(5.1.2) $$\Sigma = AA' + \Psi,$$

which is used in most early applications.

Factor analysis is discussed in the context of latent variable models in econometrics by Aigner, Hsiao, Kapteyn and Wansbeek (1984). They show factor analysis is closely related to instrument variables. Bartholomew (1985) and Bartholomew and McDonald (1986) reinterpret factor score estimates as the conditional expectations of random variables. Bekker and De Leeuw (1987) also relate factor analysis and econometrics to each other. They state that psychometricians working in factor analysis and econometricians working in regression analysis with measurement error in all variables, are both interested in the rank of the covariance matrix Σ under variation of the diagonal elements. Bentler (1976) presents a statistical model for the multivariate analysis of covariance structures that has as special feature that it does not allow the confounding of PCA and OFA. McDonald (1986) states that in principle factor analysis can be treated as either structural or functional models. In a structural model, the variables as well as the errors

are considered random. In the functional model the variables, connected by a mathematical function ('a law'), are nonrandom quantities which become incidental parameters to be estimated as well as the structural parameters describing the law. Random regressors and fixed regressors in regression theory are special cases of structural and of functional relations, respectively, in which the errors in the independent variables have zero variance. The choice between these two sets is an unpleasant task if the factor analysis model is nonlinear. If the law is nonlinear, the structural model is problematic because the resulting distributions are nonnormal, and the functional model is problematic because the structural parameters of the functional relationship may not be consistently estimated, or the likelihood function may not even have a maximum. The possibility of resolving the dilemma is illustrated by the approach of McDonald (1979) to linear and nonlinear, nonnormal factor analysis, as a fixed factors, functional-relations model. This model has been developed into a usable computational method for confirmatory nonlinear factor analysis by Etezadi-Amoli and McDonald (1983).

Etezadi-Amoli and McDonald (1983) describe methods of estimation for a special class of factor models, i.e., the nonlinear factor analysis models. They present methods of estimation for the polynomial case of the nonlinear fixed-factor-scores model, and give numerical illustrations of the usefulness of the method. In their computer program, NOFA, a version of the conjugate gradients method due to Fletcher and Reeves (1964) is used, as the number of parameters is generally too great to allow the use of a Newton or quasi-Newton method (e.g., Murray (1972)). This class of models is outside the scope of the present study.

McDonald (1986) proposes that nonnormal, linear and nonlinear models, for multivariate data, including nonlinear counterparts of linear structural relations models, can be developed as functional-relations models. This can be done by a generalization of the treatment of nonlinear factor models as fixed-factors models. Such models should then be less problematic than their counterpart nonlinear structural models. An advantage of this approach is that it will estimate a distribution of the latent variable in the model directly, avoiding the necessity of using estimates of higher moments of the observed variables as in Mooijaart (1985) or Bentler (1983).

In this chapter we will discuss some aspects of factor analysis as far as it is relevant in the context of multidimensionality. We have no pretension of coming anywhere near a complete coverage of factor analysis, which however, can be found, in for example, Lawley and Maxwell (1971), Harman (1976), or Mulaik (1972). In addition we mention the paper by Mulaik (1986) in a special 50–th anniversary edition of Psychometrika. Mulaik (1986) discusses many methodological advances in ordinary factor analysis that occurred in the fifty years since the founding of Psychometrika.

Factor analysis of the covariance structure model (1), where the parameter matrices Λ and Ψ are estimated, was explored by Bentler and Lee (1979) for the case where $\Lambda = \Lambda_1 \otimes \Lambda_2$. The *Multimode* (or *k–mode*) *Factor Analysis* (MFA) model in Wansbeek and Verhees (1986) is obtained from the OFA model by imposing a multiplicative k–mode Kronecker structure on the $\ell \times m$ parameter matrix Λ:

$$\Lambda = \Lambda_1 \otimes \ldots \otimes \Lambda_k, \tag{5.1.3}$$

where Λ_i, $i = 1,\ldots,k$ are of order $\ell_i \times m_i$, $m_i < \ell_i$. Both arisen covariance structure models may stem from the basic random vector model

$$y = \mu + \Lambda G\gamma + \xi, \tag{5.1.4}$$

where the expectations $E[\gamma\gamma']$, $E[\xi\xi']$ and $E[\gamma\xi']$ are given by Φ, Ψ and 0, respectively, where it was assumed that $G\Phi G' = I$.

The MFA model is a generalization of the following models which have been proposed for the analysis of data that can be cross–classified in terms of three modes of measurement. The pioneering and most comprehensive model is the one developed by Tucker (1966). It postulates that an observed three–way data observation can be decomposed into components attributable to each of the three modes as well as an internal core matrix. Levin (1965) states that the core matrix can be interpreted in terms of psychological types. Employing the terminology of idealized entities, the core matrix, for instance, gives the scores of each idealized concept on each idealized scale for each idealized subject. This model is most properly called a principal component analysis model, rather than a factor analysis (FA) model, because there is no specific provision for the concept of uniqueness. Bloxom (1968), however, formulated

the stochastic form of the Tucker model in a covariance structure form that includes the concept of uniqueness. Bentler and Lee (1978, 1979) developed the statistical aspects further, for the three–mode model without and with a core matrix present, respectively. The MFA model as presented here is a further development of the Bentler and Lee (1978, 1979) papers in which the structure (3) for the case $k = 2$ is explored, in which the population covariance matrix Σ became:

(5.1.5) $\qquad \Sigma = (\Lambda_1\Lambda_1'\otimes\Lambda_2\Lambda_2') + \Psi.$

They call this three–mode FA, the third mode coming from the units (individuals, say) over which the covariance matrix S has been derived. We extend their analysis to an arbitrary number of modes, k. In the terminology of Bentler and Lee (1978) this would be called $(k+1)$–mode FA, but it should be noted that the mode 'individuals' is dealt with differently from the 'k'–th mode. In this chapter we deal with the following aspects of MFA. In section 2 we specify the k–mode FA model. Section 4 gives first–order conditions for optima under Maximum Likelihood (ML), Weighted Least Squares (WLS), and for the simpler case of estimation under Unweighted Least Squares (ULS). All three estimation techniques are closely linked by a suitable discrepancy function, as shown in section 3. Since the derivatives of the discrepancy function are nonlinear with respect to the parameters, the solution cannot be represented as an explicit function of S. Thus it is usually obtained by means of an iterative procedure for which many time–efficient algorithms have been proposed. In subsection 4.2, we discuss two iterative and two noniterative estimation techniques. Section 5 is devoted to the asymptotic properties of the estimators. Section 6 is devoted to a discussion of several related subjects. Section 7 is devoted to a discussion of programming considerations for the MFA model. Section 8 concludes.

5.2. MODEL SPECIFICATION

By replacing the restriction $G\Phi G' = I$ in (1.4) by the identity

(5.2.1) $\qquad G\Phi G' = \Gamma,$

we take the core matrix Γ into account, and the full k–mode FA model is defined as

(5.2.2) $$y = \mu + \Lambda G\gamma + \xi,$$

with Λ of structure (1.3). The covariance structure analogue is

(5.2.3) $$\Sigma = \Lambda\Gamma\Lambda' + \Psi,$$

with $m \times m$ matrix Γ of structure (1). Matrix Ψ is an $\ell \times \ell$ diagonal matrix of unique factor variances. We define ℓ, m, ℓ^i, and m^i as in (4.2.3) to (4.2.6). Let $\ell^i \times m^i$ matrices Λ^i be defined as

(5.2.4) $$\Lambda^i = \Lambda_1 \otimes \ldots \otimes \Lambda_{i-1} \otimes \Lambda_{i+1} \otimes \ldots \otimes \Lambda_k.$$

5.3. ESTIMATION APPROACHES

Let us first review the relevant result in OFA. Three important and frequently used estimation methods are ML, WLS, and ULS. The first two methods, which give asymptotically efficient estimators when the underlying distribution is normal, were elegantly integrated and generalized by Swain (1975). The respective loss functions are

(5.3.1) $$\ln|\Sigma| + \mathrm{tr}(\Sigma^{-1}S),$$

(5.3.2) $$\tfrac{1}{2}\,\mathrm{tr}((\Sigma - S)S^{-1})^2,$$

and

(5.3.3) $$\tfrac{1}{2}\,\mathrm{tr}(\Sigma - S)^2.$$

It will prove useful to rewrite the ML criterion a little. If $q \equiv \ln|\Sigma| + \mathrm{tr}(S\Sigma^{-1})$,

(5.3.4) $$dq = d\,\ln|\Sigma| + d\,\mathrm{tr}(S\Sigma^{-1})$$

$$= |\Sigma|^{-1}\mathrm{d}|\Sigma| + \mathrm{tr}(S(\mathrm{d}\Sigma^{-1}))$$

$$= |\Sigma|^{-1}|\Sigma|\mathrm{tr}(\Sigma^{-1}\mathrm{d}\Sigma) - \mathrm{tr}(S\Sigma^{-1}(\mathrm{d}\Sigma)\Sigma^{-1})$$

$$= \mathrm{tr}(\Sigma^{-1}\mathrm{d}\Sigma) - \mathrm{tr}(\Sigma^{-1}S\Sigma^{-1}\mathrm{d}\Sigma)$$

$$= \mathrm{tr}(\Sigma^{-1}(\Sigma-S)\Sigma^{-1}\mathrm{d}\Sigma)$$

$$= \mathrm{tr}((S-\Sigma)\Sigma^{-1}\mathrm{d}(S-\Sigma)\Sigma^{-1}).$$

So when it comes to minimization of q, we can write an equivalent representation of the ML criterion as (see also Dijkstra (1987) and Browne (1974))

(5.3.5) $$q = \mathrm{tr}((S-\Sigma)\hat{\Sigma}^{-1})^2.$$

So the three cases can be summarized as

(5.3.6) $$\min\ \mathrm{tr}((S-\Sigma)V^{-1})^2$$

with $V = \hat{\Sigma}$ (ML), $V = S$ (WLS) or $V = I_n$ (ULS).

The first-order condition for Λ is

(5.3.7) $$W\Lambda\Gamma = 0$$

with W defined as

(5.3.8) $$W \equiv V^{-1}(\Sigma-S)V^{-1}.$$

The first-order conditions for Γ and Ψ are $\Lambda W\Lambda' = 0$ and $\mathrm{diag}(W) = 0$, respectively. As to MFA, we first note that the first-order conditions for Γ and Ψ have the same form as with OFA so they do not concern us here further. Next, as to Λ, note that we can view (6) alternatively as the first-order condition of the problem of minimizing tr $(W\Lambda\Gamma\Lambda')$ with respect to Λ' only. This simplifies the manipulations in handling MFA.

5.4. FIRST ORDER CONDITIONS AND ITERATIVE OPTIMIZATION TECHNIQUES

We will develop the necessary additional first-order derivatives and modifications to the information matrix which arise if one uses the Fisher scoring method as iterative solution technique in case of Maximum Wishart Likelihood (MWL) estimation or to the first-order derivatives for the Gauss-Newton method in case of WLS estimation.

5.4.1. First order conditions

If we define

(5.4.1) $$W_i \equiv C_i W C_i'$$

and

(5.4.2) $$\Gamma_i \equiv D_i \Gamma D_i' = D_i G \Phi G' D_i',$$

i.e., permuted versions of W and Γ (which preserve their symmetry), then

(5.4.3) $$\begin{aligned} \text{tr } W\Lambda\Gamma\Lambda' &= \text{tr } C_i W C_i' C_i \Lambda D_i' D_i \Gamma D_i' D_i \Lambda' C_i' \\ &= \text{tr } W_i(\Lambda^i \otimes \Lambda_i)\Gamma_i(\Lambda^i \otimes \Lambda_i)' \\ &= \text{tr } W_i(\Lambda^i \otimes \Lambda_i)\Gamma_i(I_{m^i} \otimes \Lambda_i)'(\Lambda^i \otimes I_{\ell_i})' \\ &= \text{tr } (\Lambda^i \otimes I_{\ell_i})' W_i(\Lambda^i \otimes I_{\ell_i})(I_{m^i} \otimes \Lambda_i)\Gamma_i(I_{m^i} \otimes \Lambda_i)' \\ &= \text{tr } (\Lambda^i \otimes I_{\ell_i})' W_i(\Lambda^i \otimes I_{\ell_i})(I_{mi} \otimes \Lambda_i)\Gamma_i(I_{m^i} \otimes \Lambda_i)' \\ &= \text{tr } (\Delta[(\Lambda^i \otimes I_{\ell_i})' W_i(\Lambda^i \otimes I_{\ell_i})(I_{mi} \otimes \Lambda_i)\Gamma_i']\Lambda_i'). \end{aligned}$$

The last line follows from properties (ii) and (iv) in section 1.3.

As was done above in the OFA case, we can obtain the first-order condition for minimizing tr $W\Lambda\Gamma\Lambda'$ by differentiating (3) with respect to Λ_i'. This yields

$$(5.4.4)\qquad \Delta[(\Lambda^i\otimes I_{\ell_i})'W_i(\Lambda^i\otimes I_{\ell_i})(I_{m^i}\otimes\Lambda_i)\Gamma_i] = 0.$$

Note that the block-trace operator performs the summation over the m^i diagonal blocks of its argument. Each block is of order $\ell_i\times m_i$, i.e., the order of Λ_i. With WLS, $(\Lambda^i\otimes I_{\ell_i})'W_i(\Lambda^i\otimes I_{\ell_i})$ is a linear function (as to Λ_i) of $\Lambda_i\Lambda_i'$ so (4) is a third-degree polynomial in the elements of Λ_i with only first-degree terms. However, it can not be written as an eigenvalue equation. With ML, (4) is a highly non-linear equation in Λ_i as Λ_i now also enters V.

When there is no core matrix, $\Gamma_i = I_m$ and (4) simplifies to

$$(5.4.5)\qquad \Delta[(\Lambda^i\otimes I_{\ell_i})'W_i(\Lambda^i\otimes I_{\ell_i})]\Lambda_i = 0$$

on applying property (iv) in section 1.3. If moreover we consider the simple ULS approach to estimation then $V = I_\ell$, $W = \Sigma-S$, hence

$$(5.4.6)\qquad W_i = C_iWC_i' = C_i(\Sigma-S)C_i'$$

$$= C_i\Lambda D_i'D_i\Lambda' C_i' + C_i(\Psi-S)C_i'$$

$$= \Lambda^i\Lambda^{i'}\otimes\Lambda_i\Lambda_i' + \Psi_i - S_i,$$

so (5) becomes

$$(5.4.7)\qquad \Delta[(\Lambda^i\otimes I_{\ell_i})'W_i(\Lambda^i\otimes I_{\ell_i})]\Lambda_i =$$

$$= \Delta[(\Lambda^{i'}\otimes I_{\ell_i})(\Lambda^i\Lambda^{i'}\otimes\Lambda_i\Lambda_i'+\Psi_i-S_i)(\Lambda^i\otimes I_{\ell_i})]\Lambda_i$$

$$= \Delta[(\Lambda^{i'}\Lambda^i)^2\otimes\Lambda_i\Lambda_i']\Lambda_i - Q_i\Lambda_i$$

$$= \gamma_i\Lambda_i\Lambda_i'\Lambda_i - Q_i\Lambda_i = 0,$$

where the last line is based on property (iii) in section 1.3, and where Q_i and γ_i are defined as

$$(5.4.8)\qquad Q_i = \Delta[(\Lambda^i\otimes I_{\ell_i})'(S_i-\Psi_i)(\Lambda^i\otimes I_{\ell_i})],$$

(5.4.9) $\gamma_i = \text{tr } (\Lambda^{i'}\Lambda^i)^2.$

The last line of (7) is an eigenvalue equation. But as Q_i depends on Λ^i, i.e., on all Λ_j with $j \neq i$, there is still need to obtain solutions (still given Ψ !) iteratively over the Λ_i's. After such a round of iterations we have to find a new value for Ψ, and go back to Λ_i's, etcetera. So the computational complexity in estimating the MFA model is somewhat relieved when we use ULS rather than ML or WLS, but it still remains sizeable.

The expression for the first-order condition (4) with respect to Λ_i can also be derived by making use of differential calculus. In our case, we get the following differential of (3.6),

$$\begin{aligned}
(5.4.10) \qquad \mathrm{d}\ \tfrac{1}{2}\mathrm{tr}((\Sigma - S)V^{-1})^2 &= \mathrm{tr}(V^{-1}(\Sigma - S)V^{-1}(\mathrm{d}\Sigma)) \\
&= \mathrm{tr}(W\mathrm{d}(\Lambda G\Phi G'\Lambda' + \Psi')) \\
&= \mathrm{tr}(W(\mathrm{d}\Lambda)G\Phi G'\Lambda' + W\Lambda G\Phi G'(\mathrm{d}\Lambda') + W\Lambda(\mathrm{d}G)\Phi G'\Lambda' + \\
&\quad W\Lambda G\Phi\mathrm{d}(G')\Lambda' + W\Lambda G(\mathrm{d}\Phi)G'\Lambda' + W(\mathrm{d}\Psi)) \\
&= \mathrm{tr}(2G\Phi G'\Lambda' W(\mathrm{d}\Lambda) + 2\Phi G'\Lambda' W\Lambda(\mathrm{d}G) + \\
&\quad G'\Lambda' W\Lambda G(\mathrm{d}\Phi) + W(\mathrm{d}\Psi)).
\end{aligned}$$

In this case further elaboration of (10), ignoring terms of no further interest, leads to

$$\begin{aligned}
(5.4.11) \qquad \mathrm{d}(\mathrm{tr}\ G\Phi G'\Lambda' W(\mathrm{d}\Lambda)) &= \mathrm{tr}(G\Phi G'\Lambda' WC_i'\mathrm{d}(\Lambda^i \otimes \Lambda_i)D_i) \\
&= \mathrm{tr}(G\Phi G'\Lambda' WC_i'(\Lambda^i \otimes (\mathrm{d}\Lambda_i))D_i) \\
&= \mathrm{tr}(D_i G\Phi G'\Lambda' WC_i'(\Lambda^i \otimes (\mathrm{d}\Lambda_i))) \\
&= (\mathrm{vec}(C_i W\Lambda G\Phi G'D_i'))'\mathrm{vec}(\Lambda^i \otimes \mathrm{d}\Lambda_i) \\
&= (\mathrm{vec}(C_i W\Lambda G\Phi G'D_i'))'B_i'(\mathrm{vec}\ \Lambda^i \otimes \mathrm{dvec}\ \Lambda_i)
\end{aligned}$$

$$= (\mathrm{vec}(C_i W \Lambda G \Phi G' D_i'))' B_i' (\lambda^{i} \otimes I_{\ell_i m_i}) \mathrm{dvec}\ \Lambda_i,$$

with $B_i = I_{m^i} \otimes P'_{\ell^i, m_i} \otimes I_{\ell_i}$, according to corollary 3.6.1. This results in

$$(5.4.12) \qquad \left[\frac{\partial \tfrac{1}{2} \mathrm{tr}\,((\Sigma - S)V^{-1})^2}{\partial (\mathrm{vec}\ \Lambda_i)'} \right] = 2(\lambda^{i'} \otimes I_{\ell_i m_i}) B_i (D_i \otimes C_i)(G \Phi G' \otimes I_\ell)(\Lambda V^{-1} \otimes V^{-1})(\sigma - s)$$

$$= Z_i' (V^{-1} \otimes V^{-1})(\sigma - s),$$

with Z_i implicitly defined. The right–hand side of (12) is a further elaboration of the left–hand side of (4).

In case there is no core matrix Γ present we have from (12) that the right–hand side expression up to a scalar becomes

$$(5.4.13) \qquad (\lambda^{i'} \otimes I_{\ell_i m_i}) B_i (D_i \otimes C_i)(\Lambda' \otimes I_\ell) \mathrm{vec}\ W =$$

$$= (\lambda^{i'} \otimes I_{\ell_i m_i})(I_{m^i} \otimes P_{\ell^i, m_i} \otimes I_{\ell_i})(D_i \Lambda' C_i' C_i \otimes C_i) \mathrm{vec}\ W$$

$$= (\lambda^{i'} \otimes I_{m_i} \otimes I_{\ell_i})(I_{m^i} \otimes P_{\ell^i, m_i} \otimes I_{\ell_i})(\Lambda^{i\,\prime} \otimes \Lambda_i' \otimes I_\ell) \mathrm{vec}\ W_i,$$

which is equal to the left–hand side of (5) as will be shown.

The left–hand side of (5) leads to

$$(5.4.14) \qquad \mathrm{vec}(\Delta_{(\ell^i)}[(\Lambda^i \otimes I_{\ell_i})' W_i (\Lambda^i \otimes I_{\ell_i})] \Lambda_i) =$$

$$= \mathrm{vec}(\Delta_{(\ell^i)}[W_i (\Lambda^{i\,\prime} \Lambda^i \otimes I_{\ell_i})] \Lambda_i)$$

$$= (\Lambda_i \otimes I_{\ell_i}) \mathrm{vec}(\Delta_{(\ell^i)}[W_i (\Lambda^{i\,\prime} \Lambda^i \otimes I_{\ell_i})])$$

$$= (\Lambda_i \otimes I_{\ell_i})((\lambda^{i'} \otimes I_{\ell_i}) P_{\ell^i m^i, \ell_i} \otimes I_{\ell_i})(I_\ell \otimes \Lambda^{i\,\prime} \otimes I_{\ell_i}) \mathrm{vec}\ W_i$$

$$= (\lambda^{i'} \otimes I_{m_i} \otimes I_{\ell_i})(I_{\ell_i m_i} \otimes \Lambda_i' \otimes I_{\ell_i})(P_{\ell^i m^i, \ell_i} \otimes I_{\ell_i})(I_\ell \otimes \Lambda^{i\,\prime} \otimes I_{\ell_i}) \mathrm{vec}\ W_i$$

$$= (\lambda^{i'} \otimes I_{m_i} \otimes I_{\ell_i})(P_{\ell^i, m^i} \otimes \Lambda_i' \otimes I_{\ell_i})(I_{\ell^i} \otimes P_{\ell_i, m^i} \otimes I_{\ell_i})(I_\ell \otimes \Lambda^{i\,\prime} \otimes I_{\ell_i}) \mathrm{vec}\ W_i$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{m_i,\ell^i}\otimes I_{\ell_i})(P_{\ell^i,m^i}\otimes\Lambda_i'\otimes I_{\ell_i})$$

$$(I_{\ell^i}\otimes P_{\ell_i,m^i}\otimes I_{\ell_i})(I_\ell\otimes\Lambda^{i\prime}\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})$$

$$((I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}\otimes\Lambda_i')(I_{\ell^i}\otimes P_{\ell_i,m^i})(I_\ell\otimes\Lambda^{i\prime})\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

where we used (3.6.12) in the fourth step. Now,

(5.4.15) $$(I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}\otimes\Lambda_i')(I_{\ell^i}\otimes P_{\ell_i,m^i})(I_\ell\otimes\Lambda^{i\prime}) =$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}\otimes\Lambda_i')(I_{\ell^i}\otimes(\Lambda^{i\prime}\otimes I_{\ell_i})P_{\ell_i,\ell^i})$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}\otimes I_{m_i})(I_{\ell^i}\otimes(\Lambda^{i\prime}\otimes\Lambda_i')P_{\ell_i,\ell^i})$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}\otimes I_{m_i})(I_{\ell^i}\otimes\Lambda^{i\prime}\otimes\Lambda_i')(I_{\ell^i}\otimes P_{\ell_i,\ell^i})$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(P_{\ell^i,m^i}(I_{\ell^i}\otimes\Lambda^{i\prime})\otimes\Lambda_i')(I_{\ell^i}\otimes P_{\ell_i,\ell^i})$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(\Lambda^{i\prime}\otimes I_{\ell^i}\otimes\Lambda_i')(P_{\ell^i,\ell^i}\otimes I_{\ell_i})(I_{\ell^i}\otimes P_{\ell_i,\ell^i})$$

$$= (I_{m^i}\otimes P_{m_i,\ell^i})(\Lambda^{i\prime}\otimes I_{\ell^i}\otimes\Lambda_i')P_{\ell,\ell^i}.$$

Substitution of (15) into (14) leads to

$$\mathrm{vec}(\Delta_{(\ell^i)}[(\Lambda^i\otimes I_{\ell_i})'W_i(\Lambda^i\otimes I_{\ell_i})]\Lambda_i) =$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{m_i,\ell^i}\otimes I_{\ell_i})(\Lambda^{i\prime}\otimes I_{\ell^i}\otimes\Lambda_i'\otimes I_{\ell_i})$$

$$(P_{\ell,\ell_i}\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})(\Lambda^{i\prime}\otimes I_{\ell^i}\otimes\Lambda_i'\otimes I_{\ell_i})(P_{\ell,\ell_i}\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})((\Lambda^{i\prime}\otimes I_{\ell^i}\otimes\Lambda_i')P_{\ell,\ell_i}\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

$$= (\lambda^{i\prime}\otimes I_{m_i}\otimes I_{\ell_i})(P_{\ell^i{}_{m_i},m^i}(I_{\ell^i}\otimes\Lambda_i'\otimes\Lambda^{i\prime})\otimes I_{\ell_i})\mathrm{vec}\ W_i$$

$$= (\lambda^{i'}\otimes I_{m_i}\otimes I_{\ell_i})(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(\Lambda_i'\otimes\Lambda^{i\,\prime}\otimes I_{\ell})\text{vec } W_i,$$

which equals (13) as it should be.

Furthermore,

$$(5.4.16)\qquad \left[\frac{\partial\tfrac{1}{2}\text{tr}((\Sigma-S)V^{-1})^2}{\partial(\text{vec } G)'}\right]' = 2(\Lambda V^{-1}\otimes\Phi G'\Lambda V^{-1})(\sigma-s) \equiv Z_G'(V^{-1}\otimes V^{-1})(\sigma-s)$$

with Z_G implicitly defined. Also,

$$(5.4.17)\qquad \left[\frac{\partial\tfrac{1}{2}\text{tr}((\Sigma-S)V^{-1})^2}{\partial(\text{d g } (\Psi))'}\right]' = H'(V^{-1}\otimes V^{-1})(\sigma-s),$$

with

$$(5.4.18)\qquad H \equiv \prod_{i=1}^{\ell}(e_ie_i'\otimes e_i),$$

where e_i, $i = 1,\ldots,\ell$, denotes the i–th unit vector of length ℓ. Furthermore,

$$(5.4.19)\qquad \left[\frac{\partial\tfrac{1}{2}\text{tr}((\Sigma-S)V^{-1})^2}{\partial(\text{vec } \Phi)'}\right]' = (G'\Lambda V^{-1}\otimes G'\Lambda V^{-1})(\sigma-s) \equiv Z_\Phi'(V^{-1}\otimes V^{-1})(\sigma-s),$$

with Z_Φ implicitly defined. Combining expressions (12), (16), (17), and (19), we have as first-order condition for a minimum of the general form of loss function (3.6) that

$$(5.4.20)\qquad F'(\theta)(V\otimes V)^{-1}(\sigma-s) = 0,$$

where

$$(5.4.21)\qquad F'(\theta) = K'(Z_1,\ldots,Z_k,Z_G,Z_\Phi,H)',$$

where vector θ represents all the unknown free elements in the parameter matrices Λ_1 to Λ_k, G, Φ and Ψ, in that order. The total number of parameters

is

$$(5.4.22) \qquad q \equiv \sum_{i=1}^{k} \ell_i m_i + m^2 + \tfrac{1}{2}m(m+1) + 1.$$

The rotation freedom in each Λ_i and the freedom to choose the multiplicative level of each but one of the Λ_i means we have to impose $r_1 = \tfrac{1}{2}\sum_{i=1}^{k} m_i(m_i-1) + (k-1)$ restrictions. In addition an extra $r_2 = \tfrac{1}{2}m(m-1)$ restrictions are needed for identification of G. This means the vector θ is of length h, where

$$(5.4.23) \qquad h \equiv \sum_{i=1}^{k} \ell_i m_i + m^2 + \tfrac{1}{2}m(m+1) + \ell - r_1 - r_2 = q - r_1 - r_2.$$

The matrix K is a unit matrix of order $q \times h$ where the columns corresponding to restricted parameters have been deleted, and so K eliminates the corresponding columns from the expression in brackets in (21).

As mentioned in section 4.8 for the PCA model, a way of dealing with the problem of under-identification is to restrict the parameter space by imposing (r_1+r_2) equality restrictions. The general treatment of estimation in case of ML by Aitchison and Silvey (1958) also applies for the MFA model. However, the introduction of precise identification constraints in this section unnecessarily complicates the expressions in this section, and would not add any new developments. Shapiro (1986) moreover, notes that the introduction of the identification constraints complicates the calculations too, and proposes an alternative approach based on techniques borrowed from the theory of generalized inverse matrices.

The derivations of the first-order derivatives are easier obtainable by the following argument. The loss function (3.6) can be written as the form

$$(5.4.24) \qquad \tfrac{1}{2}(s-\sigma)'(V \otimes V)^{-1}(s-\sigma),$$

which shows that iteratively computing estimates to Λ_1 to Λ_k, G, Φ and Ψ boils down to a WLS algorithm with weight matrix $V \otimes V$. It is known that for quadratic functions of the form above, first-order conditions for optima translate to (20) with F the Jacobian matrix, i.e.,

(5.4.25) $\quad F \equiv (F_1,\ldots,F_{k+3}),$

with $F_i \equiv \dfrac{\partial\sigma}{\partial\lambda_i'}$, $i = 1,\ldots,k$, $F_{k+1} \equiv \dfrac{\partial\sigma}{\partial g'}$, $F_{k+2} \equiv \dfrac{\partial\sigma}{\partial(\mathrm{vec}\ \Phi)'}$, and $F_{k+3} \equiv \dfrac{\partial\sigma}{\partial(\mathrm{dg}(\Psi))'}$, where $g \equiv \mathrm{vec}\ G$. In this case, for $i = 1,\ldots,k$,

$$(5.4.26) \qquad F_i \equiv \frac{\partial\sigma}{\partial\lambda_i'} = \frac{\partial \mathrm{vec}(\Lambda G\Phi G'\Lambda' + \Psi)}{\partial\lambda_i'}$$

$$= \frac{\partial(\Lambda G\Phi G'\otimes I_\ell)\lambda}{\partial\lambda_i'} + \frac{\partial(I_\ell\otimes\Lambda G\Phi G')\mathrm{vec}(\Lambda')}{\partial\lambda_i'}$$

$$= \frac{\partial(I_{\ell^2}+P_{\ell,\ell})(\Lambda G\Phi G'\otimes I_\ell)\lambda}{\partial\lambda_i'}$$

$$= \frac{\partial(I_{\ell^2}+P_{\ell,\ell})(\Lambda G\Phi G'\otimes I_\ell)\mathrm{vec}\ C_i'(\Lambda^i\otimes\Lambda_i)D_i}{\partial\lambda_i'}$$

$$= \frac{\partial(I_{\ell^2}+P_{\ell,\ell})(\Lambda G\Phi G'\otimes I_\ell)(D_i'\otimes C_i')\mathrm{vec}(\Lambda^i\otimes\Lambda_i)}{\partial\lambda_i'}$$

$$= \frac{\partial(I_{\ell^2}+P_{\ell,\ell})(\Lambda G\Phi G'\otimes I_\ell)(D_i'\otimes C_i')B_i(\lambda^i\otimes\lambda_i)}{\partial\lambda_i'}$$

$$= (I_{\ell^2}+P_{\ell,\ell})(\Lambda G\Phi G'\otimes I_\ell)(D_i\otimes C_i)'B_i(\lambda^i\otimes I_{\ell_i m_i}).$$

Now,

$$F_i'(V\otimes V)^{-1}(s-\sigma) = \tfrac{1}{2}Z_i'(I_{\ell^2}+P_{\ell,\ell})(V\otimes V)^{-1}(\sigma-s)$$

$$= \tfrac{1}{2}Z_i'(V\otimes V)^{-1}(I_{\ell^2}+P_{\ell,\ell})(\sigma-s)$$

$$= Z_i'(V\otimes V)^{-1}(\sigma-s),$$

which corresponds to (12).

$$(5.4.27) \qquad F_{k+1} = \frac{\partial\sigma}{\partial g'} = \frac{\partial \mathrm{vec}(\Lambda G\Phi G'\Lambda' + \Psi)}{\partial g'}$$

$$= \frac{\partial \mathrm{vec}(\Lambda G\Phi \otimes \Lambda)g + (\Lambda \otimes \Lambda G\Phi)\mathrm{vec}(G')}{\partial g'}$$

$$= \frac{\partial (I_{\ell^2} + P_{\ell,\ell})(\Lambda G\Phi \otimes \Lambda)g}{\partial g'}$$

$$= (I_{\ell^2} + P_{\ell,\ell})(\Lambda G\Phi \otimes \Lambda).$$

Now,

(5.4.28) $$F'_{k+1}(V \otimes V)^{-1}(\sigma - s) = Z'_G(V \otimes V)^{-1}(\sigma - s),$$

which equals (16).

(5.4.29) $$F_{k+2} = \frac{\partial \sigma}{\partial \Phi'} = \frac{\partial \mathrm{vec}(\Lambda G\Phi G'\Lambda' + \Psi)}{\partial \Phi'} = \frac{\partial (\Lambda G \otimes \Lambda G)\mathrm{vec}\ \Phi}{\partial (\mathrm{vec}\ \Phi)'} =$$

$$= (\Lambda G \otimes \Lambda G) = Z_\Phi,$$

and

(5.4.30) $$F_{k+3} = \frac{\partial \sigma}{\partial (\mathrm{dg}(\Psi))'} = \frac{\partial \mathrm{vec}(\Lambda G\Phi G'\Lambda' + \Psi)}{\partial (\mathrm{dg}(\Psi))'} = \frac{\partial \mathrm{vec}(\Psi)}{\partial (\mathrm{dg}(\Psi))'} = H,$$

which lead to (19) and (17), respectively. Here matrix H is defined by (18).

5.4.2 Iterative and noniterative techniques

For both the Gauss–Newton algorithm (e.g., Bard (1974), Jennrich (1986), Murray (1972)) in case of WLS and the Fisher scoring algorithm in case of ML (e.g., Cramer (1986)), the information matrix is needed. In order to discuss the information matrix, we again first reconsider ML in the MFA case.

Let us assume that the observed covariance matrix S is based on observations of N independently distributed $\ell \times 1$ vectors, each $N(0,\Sigma)$. So S is centrally Wishart distributed with N degrees of freedom. From (3.3.5) we know that

(5.4.31) $\Omega \equiv E(\sigma-s)(\sigma-s)' = N(I_{\ell}\otimes I_{\ell}+P_{\ell,\ell})(\Sigma\otimes\Sigma),$

so

(5.4.32) $\Omega^{+} = (4N)^{-1}(I_{\ell}\otimes I_{\ell}+P_{\ell,\ell})(\Sigma\otimes\Sigma)^{-1}.$

The information matrix then is

(5.4.33) $I(\theta) \equiv F'(\theta)(\Sigma\otimes\Sigma)^{-1}F(\theta)$

where $F(\theta)$ as in (20) is evaluated for $V = \Sigma$.

This is the main result on the information matrix. Just as with the first-order condition, it can be readily seen that $I(\theta)$ contains the OFA result as a special case, as it should. The OFA result is

(5.4.34) $\tilde{F}'(\theta)(\Sigma\otimes\Sigma)^{-1}\tilde{F}(\theta)$

with

(5.4.35) $\tilde{F}'(\theta) = \tilde{K}'(Z, Z_G, Z_{\Phi}, H)',$

where Z replaces Z_1 to Z_k in (21), and is defined as

(5.4.36) $Z \equiv 2(I_{\ell}\otimes\Lambda G\Phi G').$

The information matrix (33) can be used for a Fisher scoring approach to estimation. We do not advise procedures that use the matrix of second-order derivatives (rather than its expectation, the information matrix), such as Newton-Raphson for the simple reason that the matrix is extremely complicated. Another direct gain from using the Fisher scoring method is that the inverse of the information matrix in the last iteration step will give asymptotic standard errors of the estimates. This will be discussed in section 5.

We note again that we have ignored the matter of imposing identification constraints in this section and refer to part of subsection 1. We also mention that the loss of rank of the information matrix alone is not sufficient for

local nonidentifiability of θ, and that the condition of local regularity is essential here. An illustrative counterexample associated with the OFA model can be found in Shapiro and Browne (1983).

Almost all estimation methods for factor analysis, e.g., the ULS–, WLS–, and ML estimation methods, are based on iterative procedures as in this section. When the number of variables is large, these methods often require a lot of time to compute these estimators. Jennrich (1986) develops the Fisher scoring algorithm for the OFA case (1.1) as a Gauss–Newton algorithm, and uses the property of one–step efficiency of Fisher scoring (and Gauss–Newton) to develop an algorithm that moves from consistency to efficient estimates in one step. Methods based on instrumental variables technology (Hägglund (1982)) give consistent estimates of Σ, that can be used as starting values to obtain statistically efficient estimates in one step of Fisher scoring or Gauss–Newton. Hägglund (1982) presents three alternative estimation procedures for exploratory factor analysis based on the instrumental variables method. Bentler (1982), inspired by Hägglund's work, develops a noniterative method of estimating the parameters of both, confirmatory and exploratory factor analysis models. Ihara and Kano (1986) propose an estimator of the unique variances matrix Ψ in the OFA case (1.2), which is an explicit function of S, i.e. a closed form estimator. We have not pursued these matters any further, but these techniques, and their properties, can also be applied in the MFA case.

5.5. ASYMPTOTIC PROPERTIES

In this section we consider the covariance matrix of the asymptotic distribution of the estimators or the model parameters. We denote an estimator of θ by $\hat{\theta}$, and the true value by θ^0. We assume that there is a sufficient number of (zero) restrictions on the set of all elements of the Λ_i's, G, Φ and Ψ such that F is of full column rank in an open neighborhood of θ^0.

Browne (1974) has shown that under normality, that is when $E(S) = \Sigma$ and S is Wishart distributed, WLS estimates obtained by choosing $V = S$, are asymptotically equivalent to the MWL estimates. Lee and Jennrich (1979) have shown that setting $V = \Sigma^{-1}$ in the Gauss–Newton algorithm for WLS gives the

Fisher scoring algorithm for ML. The consequence of this is that after convergence of the Gauss–Newton algorithm and the Fisher scoring algorithm, the estimate of the asymptotic covariance matrix of the parameter estimates, which results as a by–product in both algorithms, are asymptotically equivalent. This holds under normality. For both estimation techniques the first–order derivative was of the form (4.20).

The asymptotic covariance matrix of ML, WLS, and ULS parameter estimators is then straightforwardly obtained as follows. Further remarks on asymptotic properties are presented in subsection 6.3.

Theorem 5.5.1.

(5.5.1) $\sqrt{N}\ (\hat{\theta}-\theta^0) \xrightarrow{L} N(0,U(\theta^0))$

where

(5.5.2) $U(\theta) \equiv (F'\Omega^+ F)^{-1}$

for ML and WLS, and

(5.5.3) $U(\theta) \equiv (F'F)^{-1} F'\Omega F (F'F)^{-1}$

for ULS.

Proof

Under normality, $s-\sigma$ is asymptotically normally distributed with mean zero and covariance matrix Ω. Therefore, due to application of the delta method (e.g., Rao (1973), p.31), (4.20) and (4.31) lead to the fact that parameter vector θ is asymptotically normal distributed with covariance matrix

(5.5.4) $(F'(V\otimes V)^{-1}F)^{-1}F'(V\otimes V)^{-1}\Omega(V\otimes V)^{-1}F(F'(V\otimes V)^{-1}F)^{-1}.$

In case of ML and WLS, V may be replaced by its expectation Σ. In case of ULS, $V = I$. The result follows.

■

5.6. DISCUSSION

Six other aspects of the MFA model have not been discussed at length and are briefly commented below.

5.6.1 Lisrel specification

It is in principle possible to estimate the MFA model by means of the LISREL program (Jöreskog and Sörbom (1981)). The structure on Λ is non-linear, but as the non-linearities are of a polynomial type, it is possible to translate the structure into a linear one using the concept of 'phantom variables' (Rindskopf (1984)). For most values of the number of modes, k, and size of the dimensions, ℓ_i and m_i, however, the presentation in LISREL format leads to a model that outsizes the LISREL program limits.

To specify the MFA model in terms of LISREL, we consider the k-mode FA model (2.2), with the multiplicative structure on Λ as given in (1.2). We define

$$m^{j\downarrow} \equiv \prod_{i=1}^{j-1} m_i, \tag{5.6.1}$$

$$\ell^{j\uparrow} \equiv \prod_{i=j}^{k} \ell_i. \tag{5.6.2}$$

Note that the following relations hold.

$$\ell^{k\uparrow} = \ell_k,\ \ell^{1\uparrow} = \ell,\ m^{2\downarrow} = m_1,\ m^{k\downarrow} = m^k. \tag{5.6.3}$$

We define the 'phantom variable' x_i as

$$x_i \equiv (I_{m^{k+i-1}\downarrow} \otimes \Lambda_{k+1-i} \otimes I_{\ell^{k-i+2}\uparrow}) x_{i-1} \equiv \Gamma_i x_{i-1},\ i = 2,\ldots,k-1 \tag{5.6.4a}$$

(5.6.4b) $\quad x_1 \equiv (I_{m^k} \otimes \Lambda_k)x_0 \equiv \Gamma_1 x_0,$

(5.6.4c) $\quad x_0 \equiv G\gamma$

where x_i, $i = 1,...,k-1$, are vectors of length $p_i \equiv m^{k+i-1\downarrow}\ell^{k-i+2\uparrow}$. Matrices Γ_i, $i = 1,...,k-1$, are implicitly defined. So, in terms of the general LISREL model, we have a measurement model for y

(5.6.5) $\quad y = \mu + \Lambda_y\eta + \varepsilon,$

with

(5.6.6) $\quad \eta \equiv (x_0',x_1',...,x_{k-1}')'$

as a vector of length $m + \sum_{i=1}^{k-1} p_i$, including $k-1$ vectors of 'phantom variables' and the matrix

(5.6.7) $\quad \Lambda_y = (0,...,0,\Lambda_1 \otimes I_{\ell^1})$

of order $\ell \times (m + \sum_{i=1}^{k-1} p_i)$. In addition, we have a structural equations model

(5.6.8) $$\eta = \begin{bmatrix} 0 & & \dots & 0 & 0 \\ \Gamma_1 & 0 & \dots & 0 & 0 \\ 0 & \Gamma_2 & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & \dots & \Gamma_{k-1} & 0 & \end{bmatrix} \eta + \begin{bmatrix} G \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \gamma.$$

Bentler, Poon and Lee (1988) show that several three–mode (and the multimode extensions of) models for factor analysis can be represented in LISREL format and easily implemented in the LISREL program. Essentially, Bentler, Poon and Lee (1988) translate several standard three–mode models (cf. Tucker (1966), Bloxom (1968), and Bentler and Lee (1979)) to the Bentler and Weeks (1980) structural relations model specification. Our equations (3) to (8), which we derive by using phantom variables, turn out to be derived by a similar approach as the Bentler–Weeks specification approach by Bentler, Poon and Lee (1988).

5.6.2. Ledermann bound and identification

All recent methods of factor analysis, aimed at estimating the parameters of (1.2) by least squares (e.g., Harman and Jones (1966), Jöreskog and Goldberger (1972), Okamoto and Ihara (1983), Zegers and Ten Berge (1983)), or maximum likelihood methods (Jöreskog (1972, 1975)), typically encounter estimates of Ψ that are negative. Since negative variances are troublesome to deal with, the offending elements of Ψ are usually set arbitrarily to zero (Jöreskog (1967)). Alternatively, the computing procedure yields zero uniqueness automatically (e.g., Jöreskog and Goldberger (1972)). It is not made explicit in such instances that the factor model is given up in favor of a mixed factor–components model. The Harman–Jones (1966) and Zegers–Ten Berge (1983) procedures however, recognize a solution to Heywood cases is known. Zegers and Ten Berge (1983) show empirically that the Harman–Jones procedure tends to converge to a lower minimum of the residual function, but at a slower pace than their procedure.

Furthermore, contrary to some wideheld beliefs in the past, parameter estimates obtained by various ML methods (Clarke (1970), Jennrich and Robinson (1969), Jöreskog (1967)) are nonunique as unrestricted estimates. This can be shown through an evaluation of the eigenvalues of the entire Hessian matrix obtained with a complete Newton–Raphson procedure.

In the OFA model without a core matrix Γ present (i.e., model (1.2)), a simple counting rule yields the Ledermann–bound

(5.6.9) $$(\ell-m)^2 \geq \ell+m$$

as a necessary order condition for local identification. Kapteyn and Wansbeek (1982) also stated this counting rule for local identification of model (1.2). Recently, Shapiro (1982) showed that (9) is also sufficient for almost all parameter values. McDonald (1982) also discussed the point that if a nonlinear model such as factor analysis and most models for the analysis of covariance structure), is locally identified at one point in a parameter space Ω, it does not follow that it is locally identified everywhere in Ω.

If Ψ is identified, then Λ is subject to rotational indeterminacy. Uniqueness of Λ, under orthogonal rotation, can be achieved by imposing linear restrictions on the columns of Λ. Conditions for the uniqueness of Λ in the restricted OFA model have been studied by Bekker (1986a), Jöreskog (1967), and Geweke and Singleton (1981). However, these conditions assume the prior identification of Ψ. Bekker (1986b) considers the model (1.1), where Λ, Γ, and Ψ are subject to arbitrary, possibly nonlinear, restrictions. Bekker (1986b) states a rank condition for the local identification of this more general model. Bekker (1986b) also presents a method for the computation of the exact rank, for regular points, of the resulting Jacobian matrix. This method has been implemented in a computer program (ERA: Exact Rank Analyzer) by Bekker and Merckens (1988) for identification of simultaneous equations models with measurement error. The paper by Geweke and Singleton (1981) refers to Geweke and Singleton (1980), in which it is shown (by a Monte Carlo experiment) that the asymptotic distribution theory should not be routinely employed to assess the goodness of fit of (1.2) when there is under–identification present.

Applying the same counting rule (i.e., number of independent elements in S vs. the number of elements in Λ_1 to Λ_k, G, Φ plus Ψ, minus the rotation freedom in each Λ_i and the freedom to choose the multiplicative level of each but one of the Λ_i) yields

$$\frac{\ell(\ell+1)}{2} \geq h, \tag{5.6.10}$$

where h is defined as in (4.23). Equivalent to (10) is

$$\sum_{i=1}^{k} (\ell_i - m_i)^2 + (\ell^2 - \sum_{i=1}^{k} \ell_i^2) + 2(k-1) \geq \ell + \sum_{i=1}^{k} m_i + m(m+1), \tag{5.6.11}$$

which clearly reduces to (9) on letting $k = 1$ and deleting the term $m(m+1)$ which was due to $\Gamma = G\Phi G'$. In deriving (the rather unwieldy) expression (11) it is assumed that the only rotational freedom is *within* each of the k dimensions, i.e., for each Λ_i separately, and that there are no freedoms between different dimensions. We were not yet able to prove this.

5.6.3 Further remarks on asymptotic properties

Browne (1974) and Lee (1977) have proved that in the analysis of covariance structures, the WLS estimator $\hat{\theta}$ of θ^0 possesses some attractive asymptotic properties. These were deduced under mild regularity conditions and normal theory. Bentler and Dijkstra (1985) generalize Browne's normal theory WLS results to an elliptical case. Bentler and Lee (1979) use the resulting goodness-of-fit statistic to test, both exploratory and confirmatory three-mode factor analysis models. To the same degree we rehearse Browne (1974) and Lee (1977) statements in terms of our MFA model as this model is a special case of the covariance structure model, as seen from the covariance structure analogue (2.3). The WLS estimator possesses the following asymptotic properties, some of which were mentioned in section 5.

(a) It is consistent.

(b) It is asymptotically equivalent to the ML estimator.

(c) As the ML estimator is asymptotically efficient within the class of consistent estimators, the WLS estimator is a 'best WLS' estimator.

(d)

$$\hat{\theta} \xrightarrow{L} N(\theta^0, I^{-1}(\theta^0)), \tag{5.6.12}$$

where $I(\theta^0)$ is the information matrix as in (4.33), evaluated at the point θ^0.

The estimator minimizing (3.6) is consistent for any covariance structure model in which the sample covariance matrix converges to the true covariance matrix. This is a consistency statement made by Anderson and Rubin (1956) and Browne (1974) for the OFA model with random factor γ. Anderson and Rubin (1956) stated a result which implies that, when the usual normal theory ML methods are misapplied in the situation where the common factors are fixed rather than random, the resulting estimators are still asymptotically efficient. This result was proved by Dahm and Fuller (1986) for the class of best normal theory GLS estimators. Amemiya, Fuller and Pantula (1987) obtain a stronger result to the consistency and limiting distribution of the estimators

in the one–mode case. Amemiya et al. (1987) use a technique for deriving the limiting distribution for the case with random factor γ from the result for the case of a fixed factor γ. That technique enables Amemiya et al. (1987) to show that the limiting normality and the limiting covariance matrix are valid for any random γ with second moments. Browne (1987) shows that the Anderson–Rubin, Dahm–Fuller, and Amemiya–Fuller–Pantula results apply not only to the unrestricted OFA model, but also to a wider class of models which includes some restricted OFA models and some LISREL models.

(e)

$$(5.6.13) \qquad \frac{N}{2}\mathrm{tr}((\Sigma-S)S^{-1})^2 \xrightarrow{L} \chi^2(\frac{\ell(\ell+1)}{2} - h),$$

i.e., the asymptotic distribution of the term on the left–hand side in the above expression is chi–square with degrees of freedom equal to $\frac{\ell(\ell+1)}{2} - h$.

Property (e) enables us to test the hypothesis that $\Sigma(\theta^0)$ has the structure $\Sigma = \Lambda G\Phi G'\Lambda' + \Psi$, with structure on Λ as given by (1.2), against the alternative that $\Sigma(\theta^0)$ is any symmetric positive definite matrix.

5.6.4. Incomplete data

Our general MFA model (2.2) may include the missing data case. Let us denote by y_p ($p = 1,...,N$) the p–th data vector y. We have N fixed $\ell_p \times \ell$ matrices W_p, called pattern matrices, which are formed from an $\ell \times \ell$ identity matrix by removing the rows corresponding to the missing elements of vector y_p. Since data are missing the observed parts of each vector y_p are not identically distributed, so the Wishart log–likelihood function must be modified.

Finkbeiner (1979) and Browne (1983) presented an ML method of estimating the parameters of the OFA model using the pattern matrices and assumption of randomly missing data. Completely randomly missing data implies that the distribution of random components which are observed is given by the appropriate marginal distribution of y. So each vector $W_p y_p$ is multivariate normal distributed with mean vector 0 (as $\mu = 0$) and covariance matrix $W_p \Sigma_p W_p'$. This leads to the log–likelihood function L:

$$(5.6.14) \qquad L(\Sigma;\ W_p,\ y) = -\tfrac{1}{2}\sum_{p=1}^{N}(\mathrm{tr}\ y_p' W_p'(W_p\Sigma W_p')^{-1}W_p y_p + \ln|W_p\Sigma W_p'|),$$

for which Finkbeiner (1979) stated the necessary first–order conditions for ML estimates. Brown (1983) stated an usable algorithm with first–order convergence for solving these equations which was also explored in Beale and Little (1975). It is a special case of the EM algorithm (Dempster, Laird and Rubin (1977)) and uses explicitly the regression nature of this estimate in updating fitted values to the missing data points. For completeness the algorithm is given below in the context of the MFA model.

Step 1.

Form an initial estimate of Σ; Σ should be positive definite. Then at the r–th iteration we go through steps 2 to 4, until we find no significant changes in the estimate of Σ between successive iterations.

Step 2.

Calculate updated replacement values for missing data points by regression based on the present values for each case

$$(5.6.15) \qquad \hat{y}_{ip}^{(r)} = \mathrm{E}(y_{ip} |\ W_p y_p\ ;\ \hat{\Sigma}^{(r-1)}),$$

where y_{ip} denotes variable i at observation point p. Note that $\hat{y}_{ip}^{(r)}$ assumes the value y_{ip} if y_{ip} is observed, and assumes the value of a linear combination of the observed variables at observation point p ($W_p y_p$), if y_{ip} is missing.

Step 3.

Now that the data are completed by (15), we compute total sum of squares and cross products S for the variables, i.e.,

$$(5.6.16) \qquad s_{ij} = \frac{1}{N}\sum_{p=1}^{N}\hat{y}_{ip}^{(r)}\hat{y}_{jp}^{(r)}.$$

The matrix S is adjusted by adding $\sigma_{ij.p}^{(r)}$ for every observation p to its (i,j)–th element, where

$$(5.6.17) \qquad \sigma_{ij.p}^{(r)} \equiv \mathrm{cov}(\hat{y}_{ip}^{(r-1)},\ \hat{y}_{jp}^{(r-1)}\ |\ W_p y_p\ ;\ \hat{\Sigma}^{(r-1)}).$$

Note that this adjustment is zero unless both y_{ip} and y_{jp} are missing, in which case the entry is the computed residual covariance based on the former estimate $\hat{\Sigma}^{(r-1)}$.

Step 4.

Thus, update the Σ estimate by $\hat{\Sigma}^{(r)} = \{\hat{\sigma}_{ij}^{(r)}\}$, where

$$\sigma_{ij}^{(r)} = \hat{s}_{ij} + \sigma_{ij.p}^{(r)}, \tag{5.6.18}$$

cycling back to step 2 if the criterion is not satisfied.

Rubin and Thayer (1982, 1983) and Bentler and Tanaka (1983) also discuss the application of the EM algorithm to factor analysis. Rubin and Thayer (1982) claimed that the EM algorithm had found three local maxima of the likelihood function for a certain dataset and drew certain conclusions from this. Bentler and Tanaka (1983) found that only one was in fact a local maximum, since all the eigenvalues of the Hessian matrix were negative, as required for local identification. Bentler and Tanaka (1983) establish the fact that first-order iterative methods, such as EM, cannot be relied upon alone in higher dimensional ML estimation. The step sizes become too small in the region of the maximum to achieve practical convergence. At some point, the computing algorithm has to shift to a second-order procedure such as Newton-Raphson and both first- and second-order conditions on the maximum have to be examined before the solution is accepted. Rubin and Thayer (1983) make the good point, however, that more should be known about by the curvature in the neighborhood of the global maximum.

5.6.5. Representation of the invariant factors model as a Tucker-Bloxom model

Bloxom (1968) defines the three-mode factor analysis analogue of the Tucker (1966) three-mode principal component analysis model. By writing Tucker's model in terms of random variables instead of matrices of parameters for a finite number of individuals, Bloxom postulates the factor analysis model as (1.3) with $\Lambda = \Lambda_1 \otimes \Lambda_2$. Assuming γ and ξ to be independent, Bloxom denotes the covariance structure analysis form of the factor analysis model as

(5.6.19) $\Sigma = (\Lambda_1 \otimes \Lambda_2) G \Phi G' (\Lambda_1 \otimes \Lambda_2)' + \Psi.$

McDonald (1984) proposes the invariant factors model, where the matrix Λ_2 is interpretable as a matrix of factor loadings, or in the world of regression, as a matrix of regression coefficients of the ℓ_2 observed variables (e.g., individual scores on ℓ_2 tests) on m common factors, invariant over ℓ_1 conditions (e.g., conditions of measurement under which the individual scores on each of the ℓ_2 tests). This is the result of the basic notion of the model which is that in each of the ℓ_1 conditions, the ℓ_2 tests fit the factor model, with m common factors and a factor loading matrix that is invariant over all conditions. Accordingly, the three-mode invariant factors model may be written as

(5.6.20) $y_k = Bz_k + \xi_k,$

$k = 1,...,\ell_1$. Basic to the model is also the assumption that the factor-score vector z_k in condition k is given by an $m \times m$ transformation A_k of the basic factor-score vector γ. That is,

(5.6.21) $z_k = A_k \gamma,$

$k = 1,...,\ell_1$, where $E[\gamma\gamma'] = \Phi$. McDonald (1984) defines matrices

(5.6.22) $B^* = I_{\ell_1} \otimes B,$

an $\ell \times \ell_1 m$ matrix, and

(5.6.23) $A^* = \begin{bmatrix} A_1 \\ \vdots \\ A_{\ell_1} \end{bmatrix},$

an $\ell_1 m \times m$ matrix, to yield the covariance-structure analogue of (20):

(5.6.24) $\Sigma = B^* \Lambda^* \Phi A^{*\prime} B^{*\prime} + \Psi.$

McDonald (1984) shows that his proposed invariant factors model is a special case of the three-mode data model, as described in Bloxom (1968) by rewriting

(24) as

(5.6.25) $\Sigma = (A\otimes B)F\Phi F'(A\otimes B)' + \Psi$

where B is a $\ell_2\times m$ matrix, as in (20), and

(5.6.26) $A=[\text{vec } A_1, \text{ vec } A_2, \ldots, \text{ vec } A_{\ell_1}]'$,

an $\ell_1\times m^2$ matrix. We see that matrix B plays the role of matrix A_2 which McDonald proposed. Furthermore, F, an $m^3\times m$ matrix of unities and zeroes, is just an operator enabling the rewriting of (24) as a model of the Tucker–Bloxom type as given by (19). That is,

(5.6.27) $F = I_m\otimes\text{vec } I_m$.

McDonald states that "the proof of the identity of (25) with (24) is by a straightforward but tedious scalar expansion of the two expressions, ..." (1984, p. 303). We agree that the proof is straightforward as follows but we feel that the scalar expansion is in no sense 'tedious' since a high order of structure and repetition is underlying both expressions, therefore offering the circumvention of any 'tedious' scalar building of the expressions in applicative work. The proof of equality of (24) and (25) is

(5.6.28) $$(A \otimes B)F = (I_{\ell_1}\otimes B)(A\otimes I_m)(I_m\otimes\text{vec } I_m)$$

$$= B^*\begin{bmatrix} (\text{vec } A_1)'\otimes I_m \\ \vdots \\ (\text{vec } A_{\ell_1})'\otimes I_m \end{bmatrix}(I_m\otimes\text{vec } I_m)$$

$$= B^*\begin{bmatrix} ((\text{vec } A_1)'\otimes I_m)(I_m\otimes\text{vec } I_m) \\ \vdots \\ ((\text{vec } A_{\ell_1})'\otimes I_m)(I_m\otimes\text{vec } I_m) \end{bmatrix}$$

$$= B^*\begin{bmatrix} A_1 \\ \vdots \\ A_{\ell_1} \end{bmatrix} \equiv B^*A^* \ .$$

In the last step but one we used a result by Balestra (1976) : if X is an $m \times n$ matrix, then $X=((\text{vec } X)' \otimes I_m)(I_n \otimes \text{vec } I_m)$. The last step but one may also be seen as

$$(5.6.29) \qquad B^* \begin{bmatrix} (\text{vec } A_1)' \otimes I_m \\ \vdots \\ (\text{vec } A_{\ell_1})' \otimes I_m \end{bmatrix} \begin{bmatrix} \text{vec } I_m & & \\ & \ddots & \\ & & \text{vec } I_m \end{bmatrix} =$$

$$= B^* \begin{bmatrix} A_{1.1}, A_{1.2}, \ldots, A_{1.m} \\ \vdots \quad \vdots \quad \vdots \\ A_{\ell_1.1}, A_{\ell_1.2}, \ldots, A_{\ell_1.m} \end{bmatrix} = B^* \begin{bmatrix} A_1 \\ \vdots \\ A_{\ell_1} \end{bmatrix}.$$

Note the relation between A^* and A is as follows.

$$(5.6.30) \qquad A^* = \begin{bmatrix} A_1 \\ \vdots \\ A_{\ell_1} \end{bmatrix},$$

therefore

$$(5.6.31) \qquad \text{vec}(A^{*\prime}) = \text{vec}[A_1', \ldots, A_{\ell_1}'] = \begin{bmatrix} \text{vec } A_1' \\ \vdots \\ \text{vec } A_{\ell_1}' \end{bmatrix} = \begin{bmatrix} P_{m,m}\text{vec } A_1 \\ \vdots \\ P_{m,m}\text{vec } A_{\ell_1} \end{bmatrix}$$

$$= (I_{\ell_1} \otimes P_{m,m}) \begin{bmatrix} \text{vec } A_1 \\ \vdots \\ \text{vec } A_{\ell_1} \end{bmatrix} = (I_{\ell_1} \otimes P_{m,m})\text{vec}(A')$$

$$= \text{vec}(P_{m,m}A').$$

If in applied work we were to study an invariant factors model of the form (25), we would not use a general computer program for the Tucker–Bloxom model as given by (19) while inserting the special structure for A and F, since this stresses the statement by McDonald (1984) of 'tedious' scalar building of the expression (25). The proof shows that a more direct approach would be to evaluate the expression (25) in the form (24), where the repetitive character of the expression may be further enhanced by using a single procedure for calculating the matrix products BA_i, $i = 1,\ldots,\ell_1$. This last subject is discussed in more detail for the MFA model ((2.2) and (1.3)) in section 7.

5.6.6. Ordinary factor analysis (OFA) of longitudinal data

A number of methods of factor analyzing data obtained in longitudinal studies are available, which can be thought of arising from two distinct views of the underlying statistical models. One assumes that factor scores remain invariant over occasions (see, for instance Harris (1963)) and the other assumes that the factor pattern is constant over time (e.g., Tucker (1963) and Bieber and Meredith (1986)). Both assumptions are restrictive and unrealistic when for example, evaluating and assessing the effectiveness of an educational program by studying the progress of a group of students in the program. First, the educational researcher who studies how the factors (and hence the factor scores) change over time will find his interests in conflict with the assumption of constant factor scores. Second, the assumption of an invariant factor pattern, although more realistic, takes out the possibility of possibly establishing that the tests are measuring the same factors, or abilities, or are measuring different abilities on different occasions. Swaminathan (1984) developed a factor model with the feature that both factor scores and factor pattern are permitted to vary from occasion to occasion. Swaminathan (1984) gave an estimation procedure for the ML estimation of the parameters in the model. The parameters were identified by adding enough restrictions, such as an orthogonality restriction on the factor scores and imposing a first–order autoregressive process assumption on these factor scores. But the problem of estimating the factor scores itself, and in particular, change scores, remained untackled. The same remark may also be made for the work of Jöreskog and Sörbom (1977). It addresses the notions of changing factor scores, factor pattern, and unique scores through a LISREL approach, without considering these issues simultaneously.

In this subsection we make use of the prediction theory in factor analysis, such as may be found in Lawley and Maxwell (1971), to lead us to an estimation procedure for obtaining estimates of the factor scores. Furthermore, unlike the factor scores, no autoregressive scheme was postulated for the unique scores in Swaminathan (1984), even though it may be more appropriate. We will discuss the longitudinal FA model in case of this extension.

We follow Swaminathan's (1984) notation and modeling closely by representing his factor model for T occasions, to repeat his underlying assumptions for identifying the parameters in the model and subsequently, to further elaborate on the possible estimation of the factor scores. Therefore, suppose ℓ tests are administered to the same N individuals on T separate occasions. We shall assume that the tests have the same m common factors on each occasion. We assume that for each occasion t, the observed $\ell \times 1$ random vector y_t of test scores for an individual can be modeled as a factor analysis structure, i.e.,

$$y_t = \Lambda_t x_t + \xi_t \tag{5.6.32}$$

$t = 1,...,T$, where Λ_t is the $\ell \times m$ matrix of unknown factor loadings, x_t is the $m \times 1$ random vector of factor scores, and ξ_t is the $m \times 1$ random vector of unique scores. We also make the same assumptions about first and second (cross-) moments of random variates as Swaminathan (1984) did. These are:

$$E(y_t) = 0,\ E(\xi_t) = 0,\ E(x_t) = 0, \tag{5.6.33}$$

$$E(x_{t_1}\xi_{t_1}') = 0,\ E(x_{t_1}\xi_{t_2}') = 0,\ t_1 \neq t_2, \tag{5.6.34}$$

$$E(\xi_{t_1}\xi_{t_2}') = \Psi_{t_1t_2} = \Psi_{t_2t_1},\ \text{diagonal.} \tag{5.6.35}$$

Conditions (34) indicate that the factor scores are uncorrelated with the unique scores, within as well between occasions. The diagonality of $\Psi_{t_1t_2}$ reflects the assumptions that within occasions, the unique scores are uncorrelated, and that between occasions, unique scores on the same test are correlated but unique scores on different tests are uncorrelated. In order to formulate a model for changing factor scores, without exploding the number of parameters to be estimated, thereby introducing possible non-identification, we follow Swaminathan's (1984) proposition of a first-order autoregressive process for the factor scores:

$$x_t = A_t x_{t-1} + \eta_t \ (t = 2,...,T), \tag{5.6.36}$$

where A_t is a $m \times m$ diagonal matrix. It is also assumed that the factor scores are uncorrelated within each occasion, i.e.,

(5.6.37) $\mathrm{E}(x_t x_t') = I.$

The change scores, given by η_t satisfy the following conditions

(5.6.38) $\mathrm{E}(\eta_t) = 0,\ \mathrm{E}(\eta_t x_{t-1}') = 0$

and

(5.6.39) $\mathrm{E}(\eta_{t_1}\eta_{t_2}') = 0,\ t_1 \neq t_2,\ \mathrm{E}(\eta_t\eta_t') = \phi_t^2,\ t = 2,...,T,$

where ϕ_t^2 is a $m \times m$ diagonal matrix. It immediately follows from (36) to (39) that

(5.6.40) $\phi_t^2 + A_t^2 = I,\ t = 2,...,T,$

and

(5.6.41) $\mathrm{E}(x_{t_1}x_{t_2}') = A_{t_1+1}A_{t_1+2}...A_{t_2}\ (t_1 = 1,...,t_2-1).$

Model (32) can be written compactly as

(5.6.42) $y = \Lambda x + \xi$

by defining vectors y, x and ξ of appropriate length and form. Matrix Λ has structure

(5.6.43) $$\Lambda = \begin{bmatrix} \Lambda_1 & 0 \\ & \ddots \\ 0 & \Lambda_T \end{bmatrix}.$$

It follows that the population covariance matrix Σ of the observed test scores y has structure

(5.6.44) $\Sigma \equiv \mathrm{E}(yy') = \Lambda\ \mathrm{E}(xx')\Lambda' + \mathrm{E}(\xi\xi') = \Lambda\Phi\Lambda' + \Psi$

where

$$(5.6.45) \qquad \Psi = \begin{bmatrix} \Psi_{11} & \cdots & \Psi_{1T} \\ & \vdots & \\ \Psi_{T1} & \cdots & \Psi_{TT} \end{bmatrix},$$

with T^2 matrices $\Psi_{t_1t_2}$ defined by (35), and

$$(5.6.46) \qquad \Phi = \begin{bmatrix} I & A_2 & A_2A_3 & \cdots & A_2A_3..A_T \\ A_2 & I & A_3 & \cdots & A_3..A_T \\ A_2A_3 & A_3 & I & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ A_2A_3..A_T & A_3..A_T & . & \cdots & I \end{bmatrix}$$

as follows from (37) and (41) and the diagonality of matrices A_t. Swaminathan (1984) transforms Φ to a form more amenable for treating the factor model (44) and to show which parameter matrices are to be estimated. The transformation which is used is

$$(5.6.47) \qquad \Phi = \Delta K \Delta^{-1} \Omega \; \Delta^{-1} K' \Delta,$$

where

$$(5.6.48) \qquad \Delta = \begin{bmatrix} I & & & & \\ & A_2 & & & \\ & & A_2A_3 & & \\ & & & \ddots & \\ & & & & A_2A_3..A_T \end{bmatrix},$$

K is a $Tm \times Tm$ lower-triangular matrix with $\frac{1}{2}T(T+1)$ blocks each equal to $m \times m$ identity matrices, and

$$(5.6.49) \qquad \Omega = \begin{bmatrix} I & & & \\ & \phi_1^2 & & \\ & & \ddots & \\ & & & \phi_T^2 \end{bmatrix} = \begin{bmatrix} I & & & \\ & I-A_2^2 & & \\ & & \ddots & \\ & & & I-A_T^2 \end{bmatrix},$$

due to (40). The parameters that are estimated by ML, that is, assuming the random vector of test scores y is independently and normally distributed with covariance matrix Σ, are therefore the $(T-1)m$ elements of Δ, the $T\ell m$ elements

of the factor pattern matrices Λ_1 to Λ_T, and the $\frac{1}{2}kT(T+1)$ unique parameters collected in Ψ. We refer the reader to the very enlightening discussion of the necessary derivatives to Swaminathan's (1984) paper. For now, we assume that the parameter matrices $\hat{\Lambda}_1,...,\hat{\Lambda}_T$, $\hat{\Lambda}_2,...,\hat{\Lambda}_T$, $\hat{\phi}_t^2 = I-\hat{A}_t^2$, $t = 2,...,T$ and $\hat{\Psi}$ have been estimated. We make use of the prediction theory to solve the remaining estimation problem, namely, the estimation of factor scores.

Consider again the factor analysis model (42), where the factor covariance matrix ψ was defined in (46). We seek a good predictor of x, which is a linear function of the observations y. As a measure of goodness of fit of the linear predictor $\hat{x} = Ay$, we use the variance of the predictor, i.e.,

(5.6.50) minimize $E[(Ay-x)'(Ay-x)]$.

The derivative with respect to A is

(5.6.51) $E[2y(y'A'-x')] = 2\,\Sigma A' - 2\,E[yx']$.

Now,

(5.6.52) $E[yx'] = E[(\Lambda x+\xi)x'] = \Lambda\Phi$.

Therefore, equating expression (51) to zero, we get as a result

(5.6.53) $A = \Phi\Lambda'\Sigma^{-1}$,

and $\hat{x}$ is given by

(5.6.54) $\hat{x} = \Phi\Lambda'\Sigma^{-1}y$.

We use the following rule in matrix algebra for nonsingular matrices A and D.

(5.6.55) $[BDB'+A]^{-1} = A^{-1} - A^{-1}B[B'A^{-1}B+D^{-1}]^{-1}B'A^{-1}$.

In our case,

(5.6.56) $\Sigma^{-1} = [\Lambda\Phi\Lambda'+\Psi]^{-1} = \Psi^{-1} - \Psi^{-1}\Lambda[\Lambda'\Psi^{-1}\Lambda+\Phi^{-1}]^{-1}\Lambda'\Psi^{-1}$.

Therefore,

$$(5.6.57)\qquad \Phi\Lambda'\Sigma^{-1} = \Phi\Lambda'\Psi^{-1} - \Phi\Lambda'\Psi^{-1}\Lambda[\Lambda'\Psi^{-1}\Lambda+\Phi^{-1}]^{-1}\Lambda'\Psi^{-1} =$$

$$= \Phi((\Lambda'\Psi^{-1}\Lambda+\Phi^{-1})-\Lambda'\Psi^{-1}\Lambda)[\Lambda'\Psi^{-1}\Lambda+\Phi^{-1}]^{-1}\Lambda'\Psi^{-1}$$

$$= \Phi\Phi^{-1}[\Lambda'\Psi^{-1}\Lambda+\Phi^{-1}]^{-1}\Lambda'\Psi^{-1}$$

$$= \Phi[\Lambda'\Psi^{-1}\Lambda\Phi+I]^{-1}\Lambda'\Psi^{-1}.$$

The covariance matrix for the estimators is

$$(5.6.58)\qquad \mathrm{E}(\hat{x}\hat{x}') = \Phi\Lambda'\Sigma^{-1}\Lambda\Phi = \Phi-\Phi[I+\Lambda'\Psi^{-1}\Lambda\Phi]^{-1}.$$

The covariance matrix of the errors of prediction, $\hat{x}-x$ is

$$(5.6.59)\qquad \mathrm{E}[(\hat{x}-x)(\hat{x}-x)'] = \Phi-\Phi\Lambda'\Sigma^{-1}\Lambda\Phi = \Phi[I+\Lambda'\Psi^{-1}\Lambda\Phi]^{-1}.$$

The regression estimators $\hat{x}$ are biased since their conditional expectation is not equal to x, as will be shown.

$$(5.6.60)\qquad \mathrm{E}[y|x] = \mathrm{E}[\Lambda x+\xi|x] = \Lambda x.$$

Hence

$$(5.6.61)\qquad \mathrm{E}[\hat{x}|x] = \mathrm{E}[\Phi\Lambda'\Sigma^{-1}y|x] = \Phi\Lambda'\Sigma^{-1}\Lambda x = \Phi[\Lambda'\Psi^{-1}\Lambda\Phi+I]^{-1}\Lambda'\Psi^{-1}\Lambda x$$

$$= [\Phi\Lambda'\Psi^{-1}\Lambda+I]^{-1}\Phi\Lambda'\Psi^{-1}\Lambda x = x-[\Phi\Lambda'\Psi^{-1}\Lambda+I]^{-1}x\ .$$

In the second step we used (57).

So far we assumed that within occasions, the unique scores are uncorrelated, and that between occasions, unique scores on different tests are uncorrelated, and unique scores on the same test are correlated. That is,

$$(5.6.62)\qquad \mathrm{E}(\xi_t\xi_t') = \Psi_{tt}\ ,\ \text{diagonal},$$

and

(5.6.63) $\quad E(\xi_{t_1}\xi'_{t_2}) = \Psi_{t_1 t_2}$, diagonal, $t_1 \neq t_2$.

From now on, we will assume that the unique scores are generated by a first–order autoregressive series

(5.6.64) $\quad \xi_t = B_t\xi_{t-1} + \varepsilon_t$

where B_t is an $\ell\times\ell$ diagonal matrix and with ε_t an error term satisfying the conditions

(5.6.65) $\quad E(\xi_t) = 0,$

(5.6.66) $\quad E(\varepsilon_t\xi'_{t-1}) = 0,$

and

(5.6.67) $\quad E(\varepsilon_{t_1}\varepsilon'_{t_2}) = 0,\ (t_1 \neq t_2)$

(5.6.68) $\quad E(\varepsilon_t\varepsilon'_t) = M_t^2$, diagonal.

We keep the assumption that the unique scores are uncorrelated within each occasion by requiring that

(5.6.69) $\quad E(\xi_t\xi'_t) = I,$

$t = 1,...,T$. The covariance matrix Ψ of the unique scores will now have the same structure as Φ in (46) and may be written as

(5.6.70) $\quad \Psi = \nabla K\nabla^{-1}\tilde{\Omega}\nabla^{-1}K'\nabla$

with ∇ and $\tilde{\Omega}$ of same structure as Δ in (48) and Ω in (49), respectivcly, but A_t replaced by B_t, $t = 2,...,T$. K remains the same lower–triangular matrix. Instead of estimating $\frac{1}{2}\ell T(T+1)$ uniqueness parameters of Ψ in the first model, we now need to estimate $(T-1)\ell$ elements of ∇. Assuming that

(5.6.71) $\quad E(x_{t_1}\varepsilon'_{t_2}) = 0,\ (t_1 \neq t_2)$

and

(5.6.72) $\quad E(\eta_t\varepsilon'_t) = 0,$

conditions (34) will still hold and therefore we get the covariance structure of Σ of the form :

(5.6.73) $\quad \Sigma = \Lambda\Phi\Lambda'+\Psi = \Lambda\triangle K\triangle^{-1}\Omega\triangle^{-1}K'\triangle\Lambda'+\nabla K\nabla^{-1}\bar{\Omega}\nabla^{-1}K'\nabla.$

We see that the total number of structural equations given in (73) is $\frac{1}{2}T\ell(T\ell+1)$, which is the number of independent elements in Σ. The number of parameters to be estimated is $h = Tm+(T-1)\ell+(T-1)m$. A necessary condition for the solution to be identified is

(5.6.74) $\quad \frac{1}{2}T\ell(T\ell+1)-h \geq 0.$

Once this condition is met, estimation may once again be done through ML. The ML estimates are found among the solutions of the system of likelihood equations, which are the derivatives of the log-likelihood function L,

(5.6.75) $\quad L = \log|\Sigma| + \mathrm{tr}(S\Sigma^{-1}) - \log|S| - T\ell,$

with respect to each of the parameter matrices Λ, $\triangle$ and ∇. The solution of the likelihood equations is not available in closed form and thus a numerical procedure, such as the Fletcher-Powell algorithm (Fletcher and Powell (1963)), has to be employed.

5.7. PROGRAMMING CONSIDERATIONS

When implementing the ML estimation method for the k-mode FA model by means of the Fisher scoring algorithm in Pascal source-code, one may program the relevant formulae 'at face value' as given in this chapter. It is evident that a considerable computational gain can be had when the formulae are first broken down in nested do-loops. Doing this systematically requires a separate

calculus at a 'lower' level. What this means for ML estimation of the model will be discussed next. The same reasoning also holds for implementation of the Gauss–Newton approach in case of WLS estimation of the model.

Expressions such as (1.3), (2.3), (3.6), (4.4), (4.8), (4.12), (4.16), (4.17), (4.19), (4.31), and (4.33), depend on one or a combination of (0,1)–matrices and (possible) sizable matrices, sparse or not sparse.

Use of preset routines for the operative work of (0,1)–matrices, which are efficient with respect to declaration space and computing time, is a must once dimensions of arrays and the number of modes k increases. The main modules for the (0,1)–matrices for actual program implementations in Pascal source–code are discussed in chapter 9. In the Appendix to chapter 9, a listing of the modules for the (0,1)–matrices is given. We remark that in our description of our source–code we have made some modifications for convenience of display by preliminary storing the operations of the matrices P, C_i and B_i into vectors, arranged as columns in one matrix. For instance, the operation of $B_i = I_{\ell^i} \otimes P_{\ell_i,\ell^i} \otimes I_{\ell_i}$ to a series of integer $1,\ldots,\ell^2$ is stored in row i of a matrix called *table_B*, and contains integers which state the new position of an element of vector x in vector $y = B_i x$. The matrix *table_B* has k such rows, each of length ℓ^2.

We make two remarks before we discuss the main routines for k–mode FA. First, most of the routines are recursive procedures to accommodate for any choice of the number of modes k. Second, the matrices are mainly stored in their respective vec representations. This is done to save declaration space by declaring just one vector for any type array, in this way accommodating vec representations of several matrices at once.

The main routines for k–mode FA are:

(P1). procedure Make_LAMBDA_trsp_S_inv_SminPSI_S_inv_LAMBDA,
(P2). procedure Make_Delta_of_matrix_V_i_for_ULS_no_CORE,
(P3). procedure Make_Delta_of_matrix_V_i_CORE,
(P4). procedure Make_covar_matrix_S_with_CORE,
(P5). procedure Make_covar_matrix_S_no_CORE,
(P6). procedure Make_Delta_Vi_I_kron_LAMBDA_i_CORE_i,

(P7). procedure Make_vec_of_matrix_LAMBDA_trsp_W_LAMBDA,.
(P8). procedure Make_matrix_V_i_CORE,
(P9). procedure Make_part_F_wrt_vec_LAMBDA_i_no_CORE,
(P10). procedure Make_part_F_wrt_vec_LAMBDA_i_with_CORE,
and finally,
(P11). procedure Make_part_matrix_F_wrt_vec_of_CORE.

Procedure P1 is for constructing matrix $\Lambda' S^{-1}(S-\Psi)S^{-1}\Lambda$, which is an expression that is relevant for factor analysis by WLS of the OFA model with core matrix Γ. The matrix is part of the first-order derivative of the WLS criterion with respect to Γ. The procedure is recursive and essentially consists of two parts. The first part is the repetitive call of the procedure in its own body to handle any choice of the number of modes, k, i.e., to construct a new set of four do-loops each time an internal counter is increased by one. The second part is the finishing step, i.e., when the counter reaches value k+1, then we have constructed elements of Λ and Λ', and since the indices to these elements were also updated by the same recursion, we then multiply these elements with the right elements of $S^{-1}(S-\Psi)S^{-1}$.

Procedure P2 is for constructing Q_i in (4.8), which is an expression that is relevant for estimation by ULS of the MFA model with no core matrix. The procedure is recursive and essentially consists of three parts. The first part is the repetitive call of the procedure when counter does not equal the value of index i, which is meant to construct a new set of four do-loops each time counter unequals index i and is smaller than its upper boundary value k+1. The second part is the simple call of the procedure with no update to its parameters other than increasing the value of counter by one, and appears when counter is equal to the value of index i. This is due to the appearance of matrices Λ^i and $\Lambda^{i'}$ in (4.8). The third part is the final step. Indices of elements of $S_i-\Psi_i$ are computed with the use of *table_C* and indices of elements of Λ^i and $\Lambda^{i'}$.

Procedure P3 is of the same structure as procedure P2, and is meant for constructing part of (4.4), i.e., $\Delta[V_i] = \Delta[(\Lambda^i \otimes I_{\ell_i})' W_i (\Lambda^i \otimes I_{\ell_i})]$.

Procedure P4 is for constructing part of expression (2.3), i.e., $S = \Lambda\Gamma\Lambda'$. Again, the procedure is recursive and consists of two parts, the recursive

updating part and the final computing part. Procedure P5 is meant for $S = \Lambda\Lambda'$, i.e., the MFA model with no core matrix. Both P4 and P5 should be followed by a program statement by which Ψ is added to S.

Procedure P6 is of use for constructing (4.4), where V_i is constructed according to P3. Matrix V_i is the only intermediary result being used in this non-recursive procedure P6. Procedure P7 constructs a similar expression as in (4.16) and (4.19). Expression $\mathrm{vec}(\Lambda'W\Lambda) = (\Lambda\otimes\Lambda)(V\otimes V)^{-1}(\sigma-s)$ is formed in this procedure. The recursive procedure P7 consists of the two familiar parts. Procedure P8 is nothing new in comparison to procedure P3, other than just constructing the matrix V_i instead of $\Delta[V_i]$. Procedure P9 constructs part of the right-hand side of (4.12), i.e., $(\lambda^{i\prime}\otimes I_{\ell_i m_i})B_i(D_i\otimes C_i)$. Due to the presence of λ^i, and our aim of writing procedures that work for any choice of the number of modes, k, P9 is also a recursive procedure.

The final two routines, P10 and P11, are meant for evaluating the first-order derivatives, similar to expressions (4.16), (4.27), and (4.29) for F_i, $i = 1,\ldots,k+2$, in the MFA model with or without the core matrix Γ present.

The computational gain, in terms of less computing time and less intrusion on computer memory by not having to declare many large arrays, is considerable by using recursive procedures of the kind above. The scalar level of calculus, as applied in most of the routines, asks for considerable efforts of the programmer but the reward is promising when programming data analysis techniques in multimode factor analysis models.

5.9. CONCLUSION

In this chapter we introduce the MFA model. The MFA model is, amongst other, a generalization of the Tucker (1966), Bloxom (1968), and Bentler and Lee (1978, 1979) models for three-mode factor analysis. The model is specified by using the notation of Kapteyn et al. (1986), which was introduced for their k-mode PCA model.

The three major estimation approaches (ML, WLS, and ULS) lead to similar first-order conditions for obtaining optima of their respective loss

functions. The similarity is mainly due to the integration of the three loss functions into one general loss function. Iterative optimization techniques, namely Gauss–Newton in case of WLS and Fisher scoring in case of ML, are based on the information matrix for which elegant expressions are derived. A separate section discusses the asymptotic distribution of the three estimators, and shows the reappearance of the information matrix as part of the expected covariance matrix of the estimators.

Six other aspects of the MFA model were briefly commented on in section 6. The first aspect concerned the representation of the MFA model in LISREL format by making use of the concept of 'phantom variables' (Rindskopf (1984)). The Bentler–Weeks specification approach by Bentler, Poon and Lee (1988) turns out to be quite similar to our approach. The second aspect that is shortly reviewed is the so–called Ledermann bound, which is an order condition for identification of model parameters. We give a generalization of this bound in the OFA case to the MFA case. The third aspect concerns a further discussion of the asymptotic properties of the WLS and ML estimator. The basic properties enable us to test the hypothesis of the covariance structure analogue of the model, i.e., $\Sigma = A G \Phi G' A' + \Psi$, against the alternative that Σ is any symmetric positive definite matrix. The fourth aspect concerned the missing data case of the MFA model. It turns out that a slight modification of the EM algorithm (Dempster, Laird and Rubin (1977)) is sufficient to derive an elegant procedure. Fifth, McDonald (1984) shows that the invariant factors model in covariance structure form is identical to a special case of the three–mode Tucker–Bloxom model. The proof of this identity is indeed straightforward as McDonald states, but certainly does not involve a "tedious scalar expansion of the two expressions, ..." (McDonald (1984), p. 303). The main modules for efficient programming, in terms of memory–access and processing time, are discussed in section 7.

A computer program which includes the three alternative estimators (ML, WLS and ULS) has been written for both model (1.1) and (1.2). A test run with the program has been made using three–mode data from the literature. The aim was to compare the three methods with each other. However, for several simulations we are at present unable to get satisfactory results from our program. The main reason (next to a programming error of course) is the fact that the weight matrix V (with either $V = \Sigma$ or $V = S$, in case of ML or WLS,

respectively) is not positive definite from the start of the iterative process, or became numerically nonpositive definite in later iteration steps. For now we conclude that our programming efforts for the k–mode factor analysis require additional attention in the future, and that additional test runs are needed.

The last aspect in section 6 is the applicability of factor analysis in longitudinal studies. Swaminathan (1984) developed a factor model with the feature that both factor scores and factor pattern are permitted to vary form occasion to occasion. We tackled the problem of estimating the factor scores, by using the prediction theory in factor analysis (e.g., Lawley and Maxwell (1971).) This last aspect was discussed in terms of the OFA model in contrast to most of main sections in this chapter.

6. k-MODE COVARIANCE STRUCTURE ANALYSIS

6.1. INTRODUCTION

Social, behavioral, biological and medical quantitative studies typically involve several measurements that correlate with each other in various ways. Covariance analysis is a general technique for analyzing such measurements to detect and access latent sources of variation and covariation in the observed measurements. A variety of models have been developed for testing hypotheses on variances structures (e.g., Bock and Bargmann (1966), and Browne (1974, 1982)). These structures may yield covariance matrices which exhibit special patterns or symmetries. An example of patterned covariance structure is the spherical model (Anderson (1958)), in which all the variances are equal and the covariances are zero. Another example is the circular symmetric model, of which Olkin (1973) studies an extension to block circular symmetry. Rubin and Szatrowski (1982) apply the EM algorithm to several patterned covariance matrices that do not have explicit ML estimates. The EM algorithm can be used to calculate the desired ML estimates, when the patterned covariance matrix is viewed as submatrix of a larger patterned covariance matrix, which is viewed as the covariance matrix for both observed and missing variables. An example, taken from Rubin and Szatrowski (1982), is a stationary covariance matrix, a pattern without an explicit ML estimate, which is a hypothetical missing data version of the circular symmetry pattern, which does have an explicit ML estimate.

The models for covariance structures are mainly used to reduce the degree of parameterization in order to estimate a parameter vector θ which contains all unknown parameters in a population covariance matrix $\Omega(\theta)$ of a random $\ell \times 1$ vector y, which is a variate in one dimension or mode. The models may be represented in different forms, such as the structural equations analysis form (Bock and Bargmann (1966), Bentler and Weeks (1980)), factor analysis form (Lawley and Maxwell (1974)), or handled by more general methods of multivariate analysis of variances (Browne (1974, 1982)).

Lee (1985) defined a model meant for analyzing covariance or correlation structures with parameters subject to general nonlinear constraints, while general linear constraints were considered by Bentler and Weeks (1980). General constraints that include nonlinear functions are also studied by Lee and Bentler (1980). Lee (1980) has implemented techniques for more general constraints, including inequalities, but these techniques were not adopted into computer programs. Rindskopf (1983) shows that certain kinds of inequality constraints can be introduced into linear structural models, and estimated by computer programs, e.g., LISREL (Jöreskog and Sörbom (1981)), which were not designed primarily to handle such constraints.

At the same time it was recognized that data may vary random in more than one way of classification, subsequently leading to what was dubbed 'three–mode' covariance analysis. Again different forms were proposed, such as the three–mode factor analysis model (Bentler and Lee (1978)) and general three–mode covariance structure models (see e.g., Jöreskog (1973)). Swain (1980) pointed out that, even though these three–mode models differ from each in form and assumption, they all boil down to the three–mode factor analysis type, leading to an additive structure for Ω:

$$\Omega = L\Phi L' + H^2, \tag{6.1.1}$$

where matrix L has Kronecker structure. Swain's (1980) alternative proposal for modeling the structure of Ω avoids the additive assumption by imposing a general type of nonlinear model with multiplicative structure for Ω:

$$\Omega = \Sigma_1 \otimes \Sigma_2. \tag{6.1.2}$$

In the psychometric literature, also there is evidence that the modes (facets, dimensions, etc.) in multimode data interact multiplicatively. A basic expression of this idea is that a covariance matrix may then be written as repeated Kronecker product of k, say, parameter matrices, where k is the number of modes.

In this chapter, we investigate the resulting *k–mode covariance structure analysis (CSA) model* in k dimensions or modes, such as presented in Wansbeek and Verhees (1986). In the next section it is shown how we structure the factorially structured covariance matrix Ω, and the corresponding data vector

y in the k–mode CSA model. For such a k–mode or 'k–factorial' covariance structure model we give an integrated treatment of ML, WLS and ULS estimators. Swain (1980) discussed the ML estimator for the 2–mode CSA model (2) and applied the model to biometric data. We discuss the two other major estimation criteria, namely WLS and ULS, all for the general k–mode CSA model. In section 3 we discuss the respective loss functions for these estimation approaches. In section 4 we give further notation, which will come in handy when we describe in sections 5 and 6 each of the resulting three estimators. The ML and WLS estimators (in section 5) require iterative methods. In section 6 we discuss the iterative ULS estimator, but we also give a modification of ULS that is noniterative. Section 7 is devoted to a discussion of the asymptotic properties of the three estimators. The k–mode CSA model is clearly under identified as we can multiply all Σ_i's but one by arbitrary scaling factors. This problem is discussed in section 8. Section 9 is devoted to a discussion of programming considerations. A discussion of results for several datasets with empirical and simulated data is presented in section 10. The k–mode CSA model is applied to 2–mode biometric data in Swain (1980), 2–mode psychometric data in Campbell and Fiske (1959), and 3–mode psychometric data in Cudeck (1988). Section 11 concludes.

6.2. MODEL SPECIFICATION

Let Y be an k–dimensional matrix with elements that are jointly normally distributed with zero expectation. Let the order of Y be $\ell_1 \times \ell_2 \times \ldots \times \ell_k$ and let ℓ be the total number of elements of Y. Obviously, ℓ is defined as

$$\ell = \prod_{i=1}^{k} \ell_i. \tag{6.2.1}$$

For $k = 3$, a multi–occasion×multitrait×multimethod (MOMTMM) matrix is one example of interest. For purpose of analysis it is convenient to stack the elements of Y one under each other to form the $\ell \times 1$ vector y. There are many ways of doing so, and a convenient one is to order the $y_{i_1,\ldots,i_k}$ such that the first index (i_1) runs slow, the second index (i_2) runs faster, etcetera, and the last index (i_k) runs fastest. In case of covariance structure analysis of a MOMTMM matrix, indices i_1, i_2, i_3 to an element of the vector y are indicative of which occasion (i_1), trait (i_2), and method (i_3) the observation $y_{i_1 i_2 i_3}$ relates to.

Independent observations on y from a random sample of size N are summarized in their $\ell\times\ell$ covariance matrix S, a sufficient statistic. The expectation of S is denoted by Ω (also $\ell\times\ell$). We say that y has a k-factorial covariance structure when Ω has the form

$$(6.2.2) \qquad \Omega = \Sigma_1 \otimes \Sigma_2 \otimes \ldots \otimes \Sigma_k \ ,$$

where Σ_i $(i=1,..,k)$ is of order $\ell_i\times\ell_i$.

For now, let us give an example to illustrate where (2) can be applied. Let there be a set of N married couples in our random sample. Let y be a vector containing answers to questions concerning the needed amount of expenditures on 'food', 'housing', 'travel' and 'other' (ℓ_1=4) for three different hypothesized levels of net income (ℓ_2=3), for each male and female separately (ℓ_3=2). If it is expected that the relationship between any possible combination of the category of expenditures and level of net income could be the same for both sexes and the relationship between any category of expenditure should be the same for any level of net income, and that the correlation between both sexes, for each possible combination of the categories of expenditures and level of net income could be the same, then (2.2) is an appropriate model-specification. But not only in the economic theory can applications be found. In biometric and psychometric literature there are several instances where the k-mode CSA model is conjectured, but this for $k \leq 3$. We will discuss, in section 10, one dataset from the biometric theory and two sets from the psychometric theory.

6.3. ESTIMATION APPROACHES

The purpose of factorial CSA is to estimate the Σ_i's from S. From the literature of covariance structure analysis (e.g., Bentler (1983), Browne (1982)) it is well-known that three major criteria are available when estimating covariance structure parameters Σ_i, bundled in a parameter vector θ. These criteria are:

$$(6.3.1) \qquad \text{Maximum Likelihood (ML)} \qquad : \min_{\theta}(\ln|\Omega| + \mathrm{tr}S\Omega^{-1}),$$

$$(6.3.2) \qquad \text{Weighted Least Squares (WLS)} \ : \min_{\theta} \mathrm{tr}((S-\Omega)S^{-1})^2,$$

(6.3.3) Unweighted Least Squares (ULS): $\min_{\theta} \mathrm{tr}(S-\Omega)^2$.

The three estimation criteria can be summarized as

(6.3.4) $$\min_{\theta} \mathrm{tr}((S-\Omega)W^{-1})^2$$

with $W = \Omega$ (ML), $W = S$ (WLS) or $W = I_{\ell}$ (ULS). Details to this argument are stated in section 5.3.

ML is of course statistically efficient, and so is WLS as it leads to estimates with the same asymptotic distribution. The ULS estimators are usually computationally less burdensome than both ML and WLS, but are in general statistically not efficient. These statements make sense if the underlying distribution is normal with covariance matrix (2.2). For ease of discussion we indeed assume this to hold, although the estimators to be given below can of course equally well be used in a purely data-analytic, non-model-based analysis of Ω.

Bentler and Dijkstra (1985) derive asymptotically distribution-free efficient estimates for a large class of models, including covariance structures. Browne (1984) also makes some modifications to methods for the analysis of covariance structures, by making them applicable to the wider class of elliptical distributions.

6.4. CONVENIENT NOTATION

The permuted versions of W, S and Ω are denoted by W_i, S_i and Ω_i:

(6.4.1)
$$W_i \equiv C_i W C_i',$$
$$S_i \equiv C_i S C_i',$$
$$\Omega_i \equiv C_i \Omega C_i' = \Sigma^i \otimes \Sigma_i,$$

where the $\ell^i \times \ell^i$ matrix Σ^i is defined as

(6.4.2) $$\Sigma^i \equiv \Sigma_1 \otimes \ldots \otimes \Sigma_{i-1} \otimes \Sigma_{i+1} \otimes \ldots \otimes \Sigma_k \ .$$

It is also convenient to have separate notation for "vectorized" matrices:

(6.4.3) $\sigma_i \equiv \text{vec } \Sigma_i$,

(6.4.4) $\sigma^i \equiv \text{vec } \Sigma^i$.

In estimation, the interest typically centers around the parameter matrix Σ_i. Then it is convenient to rewrite the goodness-of-fit criterion as

(6.4.5) $\min \text{tr}((S_i-\Omega_i)W_i^{-1})^2$.

One final result we need is the following. Define X_i as

(6.4.6) $X_i \equiv (\text{vec } \Sigma^i)\otimes I_{\ell_i}\otimes I_{\ell_i} = \sigma^i\otimes I_{\ell_i}\otimes I_{\ell_i}$,

then (cf. (3.6.7))

(6.4.7) $B_i\text{vec } \Omega_i = B_i\text{vec}(\Sigma^i\otimes\Sigma_i) = \sigma^i\otimes\sigma_i =$

$$= (\sigma^i\otimes I_{\ell_i}\otimes I_{\ell_i})\sigma_i = X_i\sigma_i$$

with X_i implicitly defined. Here,

(6.4.8) $B_i = I_{\ell^i}\otimes P_{\ell_i,\ell^i}\otimes I_{\ell_i}$.

So far for the notation.

6.5. THE ML AND WLS ESTIMATORS

Given the notation, deriving estimators is straightforward as

(6.5.1) $\text{tr}((S_i-\Omega_i)W_i^{-1})^2 = (\text{vec}(S_i-\Omega_i))'(W_i\otimes W_i)^{-1}\text{vec}(S_i-\Omega_i) =$

$$= (\text{vec}(S_i-\Omega_i))'B_i'B_i(W_i\otimes W_i)^{-1}B_i'B_i\text{vec}(S_i-\Omega_i)$$

$$= (\tilde{s}_i-X_i\sigma_i)'(W_i\ \pi W_i)^{-1}(\tilde{s}_i-X_i\sigma_i)$$

where the last step is based on (3.6.5) and (3.6.6). So for ML, WLS and ULS the estimator $\hat{\sigma}_i$ for σ_i is

$$(6.5.2) \qquad \hat{\sigma}_i = \left[\hat{X}_i'(\hat{W}_i \pi \hat{W}_i)^{-1}\hat{X}_i\right]^{-1}\hat{X}_i'(\hat{W}_i \pi \hat{W}_i)^{-1}\tilde{s}_i$$

where $W_i=\hat{\Omega}_i$ (ML), $W_i=S_i$ (WLS) or $W_i=I_{\ell}$ (ULS), and hats on X_i and W_i indicate their possible dependence on unknown parameters that also have to be estimated. So in all three cases the estimators have the same structure, that of regressing $\tilde{s}_i$ on $\hat{X}_i$. To see this from another angle, let $U\equiv S-\Omega$ and define $\tilde{u}_i$ analogous to $\tilde{s}_i$. Then we directly obtain the regression equation

$$(6.5.3) \qquad \tilde{s}_i=X_i\sigma_i+\tilde{u}_i.$$

As $N.S\sim W_{\ell}(N,\Omega)$, the variances and covariances of the elements of the stochastic sample covariance matrix S are represented by (cf. Magnus and Neudecker (1979)):

$$(6.5.4) \qquad \mathrm{Var}(s) = \frac{1}{N}(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega).$$

Then,

$$(6.5.5) \qquad \mathrm{Var}(\tilde{s}_i) = \frac{1}{N}(I_{\ell^2}+P_{\ell,\ell})(\Omega_i \pi \Omega_i).$$

Ignoring the matrix factor $I+P$ (the justification for this is straightforward), we see that ML corresponds to the regression with the true weight matrix, WLS uses a consistent estimator, and ULS ignores the weight structure altogether.

Although the estimator formula (2) is general as it covers the three estimation criteria, it can be elaborated for each of these. In case of ML estimation, a simplification of (2) is

$$(6.5.6) \qquad \begin{aligned}\hat{\sigma}_i &= \frac{1}{\ell^i}(\hat{\Sigma}_i\otimes\hat{\Sigma}_i)\hat{X}_i'(\hat{\Omega}_i \pi \hat{\Omega}_i)^{-1}\tilde{s}_i \\ &= \frac{1}{\ell^i}(\hat{\sigma}^i\otimes\hat{\Sigma}_i\otimes\hat{\Sigma}_i)'(\hat{\Sigma}^i\otimes\hat{\Sigma}^i\otimes\hat{\Sigma}_i\otimes\hat{\Sigma}_i)^{-1}\tilde{s}_i \\ &= \frac{1}{\ell^i}(\mathrm{vec}(\hat{\Sigma}^i)^{-1}\otimes I_{\ell_i}\otimes I_{\ell_i})'\mathrm{vec}\ \tilde{S}_i \\ &= \frac{1}{\ell^i}\ \tilde{S}_i\mathrm{vec}(\hat{\Sigma}^i)^{-1}\end{aligned}$$

where again $\tilde{S}_i$ is the $\ell_i^2 \times (\ell^i)^2$ matrix of which $\tilde{s}_i$ is the "vec". Evidently, ML estimation requires iteration over the k dimensions.

The WLS estimator can be simplified by noting that the second part of the estimator formula (2) is in this case

$$
\begin{aligned}
(6.5.7) \qquad X_i'(S_i \,\pi\, S_i)^{-1}\tilde{s}_i &= X_i'B_iB_i'(S_i \,\pi\, S_i)^{-1}B_iB_i'\mathrm{vec}\,\tilde{S}_i \\
&= X_i'B_i(S_i \otimes S_i)^{-1}\mathrm{vec}\; S_i \\
&= X_i'B_i\mathrm{vec}\; S_i^{-1} \\
&= X_i'\mathrm{vec}(\tilde{S}_i^{-1}) \\
&= (\tilde{S}_i^{-1})\sigma^i
\end{aligned}
$$

where the notation $(\tilde{S}_i^{-1})$ denotes the inverse of S_i, subjected to the "tilde" transform. Also with WLS, iteration is needed.

6.6. THE ULS ESTIMATORS

As $\hat{X}_i'\hat{X}_i = \hat{\sigma}^{i\prime}\hat{\sigma}^i \otimes I_{\ell_i} \otimes I_{\ell_i}$, we have in case of ULS estimation the following elaboration:

$$
(6.6.1) \qquad \hat{\sigma}_i = (\hat{\sigma}^{i\prime}\hat{\sigma}^i)^{-1}((\mathrm{vec}\;\hat{\Sigma}^i) \otimes I_{\ell_i} \otimes I_{\ell_i})'\mathrm{vec}\;\tilde{S}_i = (\hat{\sigma}^{i\prime}\hat{\sigma}^i)^{-1}\tilde{S}_i\hat{\sigma}^i.
$$

This estimator also requires an iterative procedure, but a modification is possible that allows for a noniterative estimator. To that end, we neglect the structure in Σ^i for the moment and consider it a single parameter matrix with no dependence on other parameter matrices (through its Kronecker structure whenever $k > 2$). Since the ULS criterion is ($a \equiv \mathrm{tr}\, S_i^2$)

$$
\begin{aligned}
(6.6.2) \qquad \mathrm{tr}(S_i - \Omega_i)^2 &= a - 2\mathrm{tr}\; S_i(\Sigma^i \otimes \Sigma_i) + \mathrm{tr}(\Sigma^i \otimes \Sigma_i)^2 \\
&= a - 2(\mathrm{vec}\; S_i)'B_iB_i'\mathrm{vec}(\Sigma^i \otimes \Sigma_i) + \sigma^{i\prime}\sigma^i.\sigma_i'\sigma_i \\
&= a - 2\sigma_i'\tilde{S}_i\sigma^i + \sigma^{i\prime}\sigma^i.\sigma_i'\sigma_i,
\end{aligned}
$$

the ULS estimator for σ^i is

(6.6.3) $\qquad \hat{\sigma}^i = (\hat{\sigma}_i'\hat{\sigma}_i)^{-1}\tilde{S}_i'\hat{\sigma}_i.$

(This expression is analogous to (1).) Substitution of (2) into (1) yields

(6.6.4) $\qquad (\tilde{S}_i\tilde{S}_i' - \hat{\lambda}I)\hat{\sigma}_i = 0$

with $\hat{\lambda}$ defined as

(6.6.5) $\qquad \hat{\lambda} \equiv \hat{\sigma}_i'\hat{\sigma}_i.\hat{\sigma}^{i\prime}\hat{\sigma}^i = \hat{\sigma}^{i\prime}\hat{\sigma}^i,$

imposing the normalization that $\hat{\sigma}_i'\hat{\sigma}_i = 1$. So $\hat{\sigma}_i$ is an eigenvector of the $\ell_i^2 \times \ell_i^2$ matrix $\tilde{S}_i\tilde{S}_i'$. On premultiplying both sides of (3) with $\hat{\sigma}^{i\prime}$ we have

(6.6.6) $\qquad \hat{\lambda} = \hat{\sigma}_i'\tilde{S}_i\hat{\sigma}^i$

which means that the value of the criterion in the optimum equals $a - \hat{\lambda}$, hence in order to render this minimal the largest eigenvalue in (3) should be chosen. So there is a noniterative ULS estimator for σ_i (and hence of Ω) that is consistent but of course not asymptotically efficient. (Again, within the assumed statistical context.) Note that, when $k = 2$, $\hat{\sigma}_i$ in (3) is exactly the ULS estimator as Σ^i then is a non–Kronecker–product–structured matrix, so no structure is neglected in following this route. It is of interest to note that, then, $\hat{\sigma}_1$ can be seen as the first left singular vector of $\tilde{S}_1$ and $\hat{\sigma}_2$ as the first right singular vector of this same matrix $\tilde{S}_1$. This approach has the advantage that the (minor) problem of sign indeterminacy of eigenvectors vanishes at the same time. In general, we can view $\hat{\sigma}_i$ as the first left singular vector of $\tilde{S}_i$.

The computational burden of calculating the singular value decomposition (SVD) of $\tilde{S}_1$ can be reduced by noting that σ_1 and σ_2 are stacked versions of symmetric matrices. This spawns the question whether there is a matrix of smaller size whose first left and right singular vectors contain the non–redundant elements of $\hat{\sigma}_1$ and $\hat{\sigma}_2$. Such a matrix is of order $\frac{1}{2}\ell_1(\ell_1+1) \times \frac{1}{2}\ell_2(\ell_2+1)$ instead of $\ell_1^2 \times \ell_2^2$, the order of $\tilde{S}_1$.

The answer is affirmative. In order to present it we need some symbols borrowed from Magnus and Neudecker (1980). For index ℓ_i, elimination matrix L_{ℓ_i}, of order $\frac{1}{2}\ell_i(\ell_i+1)\times\ell_i^2$, eliminates the supradiagonal elements of an $\ell_i^2\times 1$ vector. (The elimination matrix was also mentioned by Balestra (1976) and Vetter (1975).) The $\ell_i^2\times\frac{1}{2}\ell_i(\ell_i+1)$ duplication matrix D_{ℓ_i} has the reverse effect of stretching an $\frac{1}{2}\ell_i(\ell_i+1)\times 1$ vector into the vec of a symmetric $\ell_i\times\ell_i$ matrix (see section 3.4). For convenience we also define $\Delta_{\ell_i} \equiv D_{\ell_i}'D_{\ell_i}$. By a straightforward application of the calculus developed by Magnus and Neudecker (1980) one can show that

$$(\tilde{S}_1\tilde{S}_1' - \hat{\lambda}I)\hat{\sigma}_1 = 0 \tag{6.6.7}$$

is equivalent to

$$(AA' - \hat{\lambda}I)x = 0, \tag{6.6.8}$$

where

$$x \equiv \Delta_{\ell_1}^{\frac{1}{2}} L_{\ell_1}\hat{\sigma}_1. \tag{6.6.9}$$

Matrix A is defined as

$$A \equiv \Delta_{\ell_1}^{-\frac{1}{2}} D_{\ell_1}'\tilde{S}_1 D_{\ell_2}\Delta_{\ell_2}^{-\frac{1}{2}}, \tag{6.6.10}$$

which is the matrix sought after. The operation $\Delta_{\ell_1}^{-\frac{1}{2}}D_{\ell_1}'$ means that one of each two rows with row number $(i-1)\ell_1 + j$ and $(j-1)\ell_1 + i$, $i \neq j$, in $\tilde{S}_1$ is replaced by their average, and that the other is omitted. Likewise, $D_{\ell_2}\Delta_{\ell_2}^{-\frac{1}{2}}$ operates on the columns of $\tilde{S}_1$. The resulting vector x should be scaled (cf. $\Delta_{\ell_1}^{\frac{1}{2}}$ in (9), diagonal with elements 1 or $\sqrt{2}$) and duplicated where necessary (cf. L_{ℓ_1} in (9)) to obtain $\hat{\sigma}_1$. The scaling of $\hat{\sigma}_2$ is analogous.

We define *vec–symmetry* of a vector as follows. A vector is vec–symmetric if it is the vec of a symmetric matrix. Here $\hat{\sigma}_1$ is vec–symmetric as was noted above. This vec–symmetry also holds for the other eigenvector x_i which correspond to eigenvalues $\lambda_i > 0$ of $\tilde{S}_i\tilde{S}_i'$. This is easily seen from (4) as follows.

An important property which is due to (3.6.10) in Theorem 3.6.3 is

(6.6.11) $$P_{\ell_i,\ell_i}\tilde{S}_iP_{\ell^i,\ell^i} = \tilde{S}_i.$$

Then (4) is equal to

$$\begin{aligned}(\tilde{S}_i\tilde{S}_i'-\lambda_iI)x_i &= (P_{\ell_i,\ell_i}\tilde{S}_iP_{\ell^i,\ell^i}P_{\ell^i,\ell^i}\tilde{S}_iP_{\ell_i,\ell_i}-\lambda_iI)x_i\\ &= P_{\ell_i,\ell_i}(\tilde{S}_i\tilde{S}_i'-\lambda_iI)P_{\ell_i,\ell_i}x_i\\ &= (\tilde{S}_i\tilde{S}_i'-\lambda_iI)P_{\ell_i,\ell_i}x_i\\ &= 0,\end{aligned}$$

which means that $x_i = P_{\ell_i,\ell_i}x_i$, henceforth x_i is vec–symmetric.

6.7. ASYMPTOTIC PROPERTIES

If we draw a sample of N observations for each of ℓ variables in our experiment from a multivariate normal distribution to compute the $(\ell\times\ell)$ sample covariance matrix S and if N is large enough, an appropriate approximate distribution to the exact Wishart distribution would be the normal distribution, i.e.,

(6.7.1) $$\sqrt{N}(s-\omega) \xrightarrow{L} \mathrm{N}(0,U),$$

where

(6.7.2) $$U=(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)$$

due to (5.4), and

(6.7.3) $$\omega \equiv \text{vec } \Omega.$$

Since Σ_1 to Σ_k are unknown an iterative solution method has to be used to obtain estimates $\hat{\Sigma}_1$ to $\hat{\Sigma}_k$ for all three estimators, as presented by equation

(5.2). Therefore, finding the exact distribution of these estimators looks like a hopeless problem and we resort to asymptotic distributions instead. To derive the limiting distributions we proceed with (5.2) which shows that all three estimators are of the form

$$\hat{\sigma}_i = \hat{Q}_i s, \tag{6.7.4}$$

where

$$\hat{Q}_i \equiv \left[\hat{X}_i'(\hat{W}_i \pi \hat{W}_i)^{-1}\hat{X}_i\right]^{-1}\hat{X}_i'(\hat{W}_i \pi \hat{W}_i)^{-1}B_i(C_i \otimes C_i) \tag{6.7.5}$$

is a stochastic matrix that converges in probability limit to a nonstochastic matrix, denoted by Q_i, which is the same matrix both for WLS and ML, but differs for ULS. We now use the delta method (e.g., Rao (1973, p. 321)) to derive

$$\sqrt{N}\begin{bmatrix}\hat{\sigma}_1 - \sigma_1 \\ \vdots \\ \hat{\sigma}_k - \sigma_k\end{bmatrix} \xrightarrow{L} N(0,V) , \tag{6.7.6}$$

where

$$V = \begin{bmatrix}Q_1 \\ \vdots \\ Q_k\end{bmatrix} U(Q_1', \ldots, Q_k') . \tag{6.7.7}$$

We will look at the expressions for the three estimators more closely.

6.7.1. Asymptotic distribution of the ML estimator

In case of ML estimation $W_i = \Omega_i = \Sigma^i \otimes \Sigma_i$ leads to

$$\begin{aligned} X_i'[\Omega_i \pi \Omega_i]^{-1}X_i &= (\sigma^i \otimes I_{\ell_i} \otimes I_{\ell_i})'[\Sigma^i \otimes \Sigma^i \otimes \Sigma_i \otimes \Sigma_i]^{-1}(\sigma^i \otimes I_{\ell_i} \otimes I_{\ell_i}) \\ &= \sigma^{i\prime}(\Sigma^i \otimes \Sigma^i)^{-1}\sigma^i \otimes (\Sigma_i \otimes \Sigma_i)^{-1} \\ &= (\mathrm{vec}(\Sigma^i)^{-1})'\mathrm{vec}\ \Sigma^i \otimes (\Sigma_i \otimes \Sigma_i)^{-1} \\ &= \ell^i(\Sigma_i \otimes \Sigma_i)^{-1}. \end{aligned} \tag{6.7.8}$$

Therefore,

$$(6.7.9) \qquad V_{ii} \equiv Q_i U Q_i'$$

$$= \left[X_i'(\Omega_i \pi \Omega_i)^{-1}X_i\right]^{-1} X_i'(\Omega_i \pi \Omega_i)^{-1}B_i(C_i \otimes C_i)(I_{\ell^2}+P_{\ell,\ell})(\Omega \otimes \Omega)$$

$$(C_i' \otimes C_i')B_i'(\Omega_i \pi \Omega_i)^{-1}X_i\left[X_i'(\Omega_i \pi \Omega_i)^{-1}X_i\right]^{-1}$$

$$= \left(\frac{1}{\ell^i}\right)^2(\Sigma_i \otimes \Sigma_i)X_i'(\Omega_i \pi \Omega_i)^{-1}B_i(I_{\ell^2}+P_{\ell,\ell})(C_i \Omega C_i' \otimes C_i \Omega C_i')$$

$$B_i'(\Omega_i \pi \Omega_i)^{-1}X_i(\Sigma_i \otimes \Sigma_i)$$

$$= \left(\frac{1}{\ell^i}\right)^2(\Sigma_i \otimes \Sigma_i)X_i'(\Omega_i \pi \Omega_i)^{-1}(I_{\ell^2}+(P_{\ell^i,\ell^i} \otimes P_{\ell_i,\ell_i}))B_i(\Omega_i \otimes \Omega_i)B_i'$$

$$(\Omega_i \pi \Omega_i)^{-1}X_i(\Sigma_i \otimes \Sigma_i)$$

$$= \left(\frac{1}{\ell^i}\right)^2(\Sigma_i \otimes \Sigma_i)X_i'(\Omega_i \pi \Omega_i)^{-1}(I_{\ell^2}+(P_{\ell^i,\ell^i} \otimes P_{\ell_i,\ell_i}))X_i(\Sigma_i \otimes \Sigma_i)$$

$$= \frac{1}{\ell^i}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\Sigma_i \otimes \Sigma_i) \ ,$$

where we used in the last equation but two property (3.6.9). In the last equation but one we used the fact that $P_{\ell^i,\ell^i}\sigma^i=\sigma^i$, due to the fact that Σ^i is symmetric.

The matrix $\frac{1}{N}V_{ii}$ is the covariance matrix of a stochast vec(T), where

$$(6.7.10) \qquad T \equiv \frac{1}{N}\sum_{p=1}^{N} y_p y_p' \ .$$

Here, the $\ell_i \times 1$ vectors y_p result from a transformation of the $\ell \times 1$ vectors x_p (p=1,...,N), where each vector x_p comprises the p-th observation of ℓ variables in our experiment. Assumptions underlying the experiment would however change. Let us clarify this from a viewpoint of a three-mode CSA model. First, the ℓ variables in x are indexed according to three indices (i.e., $x_{i_1 i_2 i_3}$), where two indices (i_2 and i_3) each indicate a mode with levels of a label for a variable measured at, at least, a nominal scale, and

the third index (i_1) indicates levels of a label for a variable measured at interval scale. For example, index i_2 indicates 'post–war period' or 'pre–war period' (i.e., i_2=1 or 2, ℓ_2=2), index i_3 indicates 'European–', 'Asian–', or 'American–' region (i.e., i_3=1, 2 or 3, ℓ_3=3), and index i_1 indicates 'net income', 'food expenditure', or 'housing expenditure' figures (i.e., i_1=1, 2 or 3, ℓ_1=3) for one chosen household type. For this kind of a dataset it is then justified to average observations over the two modes indicated by indices i_2 and i_3. Second, once we have knowledge of the variance matrices Σ_2 and Σ_3, we should be able to reason that we can restrict $\Sigma^i = \Sigma_2 \otimes \Sigma_3$ such that $\iota'\Sigma^i\iota = (\ell^i)^{1/2}$. Then, once we postulate that each $x_p \sim N(0,\Omega)$ we can form each random vector y_p:

$$(6.7.11) \qquad y_p = (\ell^i)^{-1/2}(\iota' \otimes I_{\ell_i})C_i x_p \ , \ p = 1,\ldots,N.$$

The distribution of y_p is $N(0,(\ell^i)^{-1/2}\Sigma_i)$. Then $E(T)=(\ell^i)^{-1/2}\Sigma_i$, and V(vec T) will be equal to $\frac{1}{N} V_{ii}$ with V_{ii} defined as the expression in (9).

Furthermore, the off–diagonal block–matrices V_{ij} are, in case of ML,

$$(6.7.12) \qquad V_{ij} = Q_i U Q_j'$$

$$= \frac{1}{\ell^i \ell^j}(\Sigma_i \otimes \Sigma_i)X_i'(\Omega_i \pi \Omega_i)^{-1}(I_{\ell^2} + (P_{\ell^i,\ell^i} \otimes P_{\ell_i,\ell_i}))$$
$$B_i(C_i \otimes C_i)(\Omega \otimes \Omega)(C_j \otimes C_j)' B_j'(\Omega_j \pi \Omega_j)^{-1} X_j(\Sigma_j \otimes \Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(\Sigma_i \otimes \Sigma_i)(I_{\ell_i^2} + P_{\ell_i,\ell_i})X_i' B_i(C_i C_j' \otimes C_i C_j')B_j'$$
$$(\Omega_j \pi \Omega_j)^{-1} X_j(\Sigma_j \otimes \Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})(\sigma^i \otimes \Sigma_i \otimes \Sigma_i)' B_i(C_i \otimes C_i)(\Omega \otimes \Omega)^{-1}(C_j \otimes C_j)' B_j'$$
$$(\sigma^j \otimes \Sigma_j \otimes \Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})(\sigma^i \otimes \Sigma_i \otimes \Sigma_i)' B_i(\Sigma^i \otimes \Sigma_i \otimes I_\ell)^{-1}(C_i \otimes C_i)(C_j \otimes C_j)'$$
$$(I_\ell \otimes \Sigma^j \otimes \Sigma_j)^{-1} B_j'(\sigma^j \otimes \Sigma_j \otimes \Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})(\sigma^i \otimes \Sigma_i \otimes \Sigma_i)' B_i (\Sigma^i \otimes \Sigma_i \otimes I_{\ell^i} \otimes I_{\ell_i})^{-1}$$

$$(C_i \otimes C_i)(C_j \otimes C_j)' (I_{\ell^j} \otimes I_{\ell_j} \otimes \Sigma^j \otimes \Sigma_j)^{-1} B_j' (\sigma^j \otimes \Sigma_j \otimes \Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})((\text{vec } I_{\ell^i}) \otimes I_{\ell_i} \otimes \Sigma_i)' B_i$$

$$(C_i \otimes C_i)(C_j \otimes C_j)' B_j' ((\text{vec } I_{\ell^j}) \otimes \Sigma_j \otimes I_{\ell_j}).$$

Premultiplication of V_{ij} with (vec L)', where L is an arbitrary $\ell_i \times \ell_i$ matrix and postmultiplication with vec R, R an arbitrary $\ell_j \times \ell_j$ matrix leads to

(6.7.13) $$(\text{vec } L)' V_{ij} \text{vec } R = \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})((\text{vec } I_{\ell^i}) \otimes (\text{vec } \Sigma_i L))' B_i$$

$$(C_i \otimes C_i)(C_j \otimes C_j)' B_j' ((\text{vec } I_{\ell^j}) \otimes (\text{vec } R\Sigma_j))$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})(\text{vec}(I_{\ell^i} \otimes \Sigma_i L))'$$

$$(C_i \otimes C_i)(C_j \otimes C_j)' \text{vec}(I_{\ell^j} \otimes R\Sigma_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i})(\text{vec}(C_i'(I_{\ell^i} \otimes \Sigma_i L) C_i)'$$

$$\text{vec}(C_j'(I_{\ell^j} \otimes R\Sigma_j) C_j)$$

$$= \frac{1}{\ell^i \ell^j}(I_{\ell_i^2} + P_{\ell_i,\ell_i}) \text{tr}(I_{\ell_i} \otimes \ldots \otimes L' \Sigma_i \otimes \ldots \otimes R\Sigma_j \otimes \ldots \Sigma_k)$$

$$= (I_{\ell_i^2} + P_{\ell_i,\ell_i}) \text{tr}(L' \Sigma_i) . \text{tr}(R\Sigma_j)$$

$$= (I_{\ell_i^2} + P_{\ell_i,\ell_i})(\text{vec } L)' \sigma_i \sigma_j' (\text{vec } R).$$

The off–diagonal blocks V_{ij} in matrix V are therefore

(6.7.14) $$V_{ij} = 2\sigma_i \sigma_j' ,$$

which are rank one matrices. The factor 2 arises because $P_{\ell_i,\ell_i}\sigma_i = \sigma_i$, due to the symmetry of Σ_i.

Anderson (1973) has proposed an iterative process for ML estimation of the parameters of linear covariance structures, and has shown that the estimates obtained in a single iteration are asymptotically efficient, provided that the initial estimate is consistent.

6.7.2. Asymptotic distribution of the WLS estimator

In case of WLS estimation, replacement of the stochastic weight matrix $\hat{Q}_i$ by its probability limit Q_i will lead to the same asymptotic variances and covariances as in case of ML, because the stochastic matrix $W_i = S_i$ is replaced by its probability limit Ω_i, since S_i is a consistent estimator of Ω_i.

6.7.3. Asymptotic distribution of the ULS estimators

In case of ULS estimation, $W_i = I_\ell$ in (5) leads to

$$
\begin{aligned}
(6.7.15) \qquad V_{ii} &= Q_i U Q_i' \\
&= (X_i'X_i)^{-1}X_i'B_i(C_i \otimes C_i)(I_{\ell^2}+P_{\ell,\ell})(\Omega \otimes \Omega)(C_i \otimes C_i)'B_i'X_i(X_i'X_i)^{-1} \\
&= (X_i'X_i)^{-1}X_i'(I_{\ell^2}+(P_{\ell^i,\ell^i} \otimes P_{\ell_i,\ell_i}))(\Omega_i \pi \Omega_i)X_i(X_i'X_i)^{-1} \\
&= \frac{1}{(\sigma^{i\prime}\sigma^i)^2}(I_{\ell_i^2}+P_{\ell_i,\ell_i})X_i'(\Omega_i \pi \Omega_i)X_i \\
&= \frac{\sigma^{i\prime}(\Sigma^i \otimes \Sigma^i)\sigma^i}{(\sigma^{i\prime}\sigma^i)^2}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\Sigma_i \otimes \Sigma_i) \\
&= \frac{\mathrm{tr}(\Sigma^i)^4}{(\mathrm{tr}(\Sigma^i)^2)^2}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\Sigma_i \otimes \Sigma_i).
\end{aligned}
$$

The scalar ratio in (15) is larger than $\frac{1}{\ell^i}$ in view of Schwarz's inequality. Notice that the diagonal blocks V_{ii} for ULS are proportional to V_{ii} for ML in (9).

The covariance matrices of ULS estimators $\hat{\sigma}_i$ and $\hat{\sigma}_j$ are V_{ij},

(6.7.16) $$V_{ij} \equiv Q_i U Q_j'$$

$$= (X_i'X_i)^{-1}X_i'B_i(C_i\otimes C_i)(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)(C_j\otimes C_j)'B_j'X_j(X_j'X_j)^{-1}$$

$$= \frac{1}{(\sigma^{i\prime}\sigma^i)(\sigma^{j\prime}\sigma^j)}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\sigma^i\otimes\Sigma_i\otimes\Sigma_i)'B_i(C_i\otimes C_i)(\Omega\otimes\Omega)$$

$$(C_j\otimes C_j)'B_j'(\sigma^j\otimes\Sigma_j\otimes\Sigma_j\}.$$

Then

(6.7.17) $$(\text{vec } L)'V_{ij}(\text{vec } R) =$$

$$= \frac{1}{(\sigma^{i\prime}\sigma^i)(\sigma^{j\prime}\sigma^j)}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\text{vec}(\Sigma_1\otimes\ldots\otimes L\otimes\ldots\otimes\Sigma_k))'$$

$$(\Omega\otimes\Omega)\text{vec}(\Sigma_1\otimes\ldots\otimes R\otimes\ldots\otimes\Sigma_k)$$

$$= \frac{2}{\text{tr}(\Sigma^i)^2}\frac{1}{\text{tr}(\Sigma^j)^2}\frac{\text{tr}(\Omega^4)}{\text{tr}(\Sigma_i)^4}\frac{1}{\text{tr}(\Sigma_j)^4}(I_{\ell_i^2}+P_{\ell_i,\ell_i})$$

$$(\text{vec } L)'(\text{vec}(\Sigma_i^3))(\text{vec}(\Sigma_j^3))'(\text{vec } R).$$

Therefore

(6.7.18) $$V_{ij} = \frac{2}{\text{tr}(\Sigma^i)^2}\frac{1}{\text{tr}(\Sigma^j)^2}\frac{\text{tr}(\Omega^4)}{\text{tr}(\Sigma_i)^4}\frac{1}{\text{tr}(\Sigma_j)^4}(\text{vec}(\Sigma_i^3))(\text{vec}(\Sigma_j^3))',$$

which are rank one matrices. Note that V_{ij} for ULS is not proportional to V_{ij} for ML in (14).

In case of ULS estimation we also had the alternative of a noniterative approach to the problem of estimation. This resulted in an eigenvalue–eigenvector decomposition of the matrices $\tilde{S}_i\tilde{S}_i'$, where we then had to find the largest eigenvalue ($\hat{\lambda}_i$) and its corresponding eigenvector $\hat{\sigma}_i$ to solve the problem. We use the following reasoning to evaluate the necessary derivative

$$(6.7.19) \qquad Q_i = \text{plim}\ \frac{\partial \hat{\sigma}_i}{\partial s'}\ .$$

First,

$$(6.7.20) \qquad \frac{\partial \hat{\sigma}_i}{\partial s'} = \frac{\partial \hat{\sigma}_i}{\partial (\text{vec}\ \tilde{S}_i\tilde{S}_i')'} \cdot \frac{\partial\ \text{vec}\ \tilde{S}_i\tilde{S}_i'}{\partial (\text{vec}\ \tilde{S}_i)'} \cdot \frac{\partial\ \text{vec}\ \tilde{S}_i}{\partial s'}.$$

Second, if $\tilde{S}_i\tilde{S}_i'$ is a real symmetric matrix with a simple largest eigenvalue $\hat{\lambda}_i$, we can then use a well–known result concerning the derivative of an eigenvector to the elements of a matrix (e.g., Magnus (1985)). Basically, the conditions of the implicit function theorem are satisfied to deduce from the proof of this result that in our case

$$(6.7.21) \qquad \frac{\partial \hat{\sigma}_i}{\partial (\text{vec}\ \tilde{S}_i\tilde{S}_i')'} = \hat{\sigma}_i' \otimes (\hat{\lambda}_i I - \tilde{S}_i\tilde{S}_i')^+,$$

where $\hat{\sigma}_i$ is the eigenvector of $\tilde{S}_i\tilde{S}_i'$, normalized such that $\hat{\sigma}_i'\hat{\sigma}_i = 1$, with corresponding eigenvalue $\hat{\lambda}_i$, which in probability limit must converge to $\sigma_i'\sigma_i.\sigma^{i\prime}\sigma^i$, for all $i = 1,\ldots,k$. Then, according to (20)

$$(6.7.22) \qquad Q_i = \text{plim}\ \frac{\partial \hat{\sigma}_i}{\partial s'}$$

$$= \frac{1}{\sigma^{i\prime}\sigma^i}(\sigma_i' \otimes (I - \sigma_i\sigma_i')^+)(I_{\ell_i^4} + P_{\ell_i^2,\ell_i^2})(\sigma_i\sigma^{i\prime} \otimes I_{\ell_i^2})B_i(C_i \otimes C_i)$$

where we replaced $\tilde{S}_i$ with its probability limit $\sigma_i\sigma^{i\prime}$. The Moore–Penrose inverse of $I - xx'$ is $I - xx'$ if $x'x = 1$, since this projector matrix fulfills the four properties of a Moore–Penrose inverse A^+ of A (cf. (1.4.1) to (1.4.4)). Matrix Q_i may then be further simplified to

$$(6.7.23) \qquad Q_i = \frac{1}{\sigma^{i\prime}\sigma^i}(\sigma^{i\prime} \otimes (I_{\ell_i^2} - \sigma_i\sigma_i'))B_i(C_i \otimes C_i).$$

Then,

$$(6.7.24) \qquad V_{ii} \equiv Q_i U Q_i'$$

$$= \frac{1}{(\sigma^{i\prime}\sigma^{i})^{2}}(\sigma^{i\prime}\otimes(I-\sigma_i\sigma_i'))B_i(C_i\otimes C_i)(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)$$

$$(C_i\otimes C_i)'B_i'(\sigma^{i}\otimes(I-\sigma_i\sigma_i'))$$

$$= \frac{1}{(\sigma^{i\prime}\sigma^{i})^{2}}(\sigma^{i\prime}\otimes(I-\sigma_i\sigma_i'))(I_{\ell^2}+(P_{\ell^i,\ell^i}\otimes P_{\ell_i,\ell_i}))(\Sigma^{i}\otimes\Sigma^{i}\otimes\Sigma_i\otimes\Sigma_i)$$

$$(\sigma^{i}\otimes(I-\sigma_i\sigma_i'))$$

$$= \frac{\mathrm{tr}(\Sigma^{i})^{4}}{(\mathrm{tr}(\Sigma^{i})^{2})^{2}}(I_{\ell_i^2}+P_{\ell_i,\ell_i})((I-\sigma_i\sigma_i')(\Sigma_i\otimes\Sigma_i)(I-\sigma_i\sigma_i')).$$

(6.7.25) $V_{ij} \equiv Q_iUQ_j'$

$$= \frac{1}{(\sigma^{i\prime}\sigma^{i})(\sigma^{j\prime}\sigma^{j})}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\sigma^{i\prime}\otimes(I_{\ell_i^2}-\sigma_i\sigma_i'))$$

$$B_i(C_i\otimes C_i)(\Omega\otimes\Omega)(C_j\otimes C_j)'B_j'(\sigma^{j}\otimes(I_{\ell_j^2}-\sigma_j\sigma_j'))$$

$$= \frac{1}{\mathrm{tr}(\Sigma^{i})^{2}}\cdot\frac{1}{\mathrm{tr}(\Sigma^{j})^{2}}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\sigma^{i\prime}\otimes(I_{\ell_i^2}-\sigma_i\sigma_i'))$$

$$(\Sigma^{i}\otimes\Sigma_i\otimes I_{\ell^i}\otimes I_{\ell_i})^{-1}(C_i\otimes C_i)(C_j\otimes C_j)'(I_{\ell^j}\otimes I_{\ell_j}\otimes\Sigma^{j}\otimes\Sigma_j)^{-1}$$

$$B_j'(\sigma^{j}\otimes(I_{\ell_j^2}-\sigma_j\sigma_j'))$$

$$= \frac{1}{\mathrm{tr}(\Sigma^{i})^{2}}\cdot\frac{1}{\mathrm{tr}(\Sigma^{j})^{2}}(I_{\ell_i^2}+P_{\ell_i,\ell_i})$$

$$(\mathrm{vec}(\Sigma^{i})^{2}\otimes((\Sigma_i\otimes I_{\ell_i})-\mathrm{vec}(\Sigma_i)^{2}\sigma_i'))'$$

$$B_i(C_i\otimes C_i)(C_j\otimes C_j)'B_j'(\mathrm{vec}(\Sigma^{j})^{2}\otimes((I_{\ell_j}\otimes\Sigma_j)-\mathrm{vec}(\Sigma_j)^{2}\sigma_j')).$$

Then,

(6.7.26) $(\mathrm{vec}\ L)V_{ij}(\mathrm{vec}\ R) =$

$$= \frac{1}{\mathrm{tr}(\Sigma^i)^2}.\frac{1}{\mathrm{tr}(\Sigma^j)^2}\,(I_{\ell_i^2}+P_{\ell_i,\ell_i})$$

$$(\mathrm{vec}(C_i'((\Sigma^i)^2\otimes L'\Sigma_i)C_i)-\mathrm{tr}(L'\Sigma_i).\mathrm{vec}(C_i'((\Sigma^i)^2\otimes(\Sigma_i)^2)C_i))'$$

$$(\mathrm{vec}(C_j'((\Sigma^j)^2\otimes\Sigma_jR)C_j)-\mathrm{tr}(\Sigma_jR).\mathrm{vec}(C_j'((\Sigma^j)^2\otimes(\Sigma_j)^2)C_j))$$

$$= \frac{1}{\mathrm{tr}(\Sigma^i)^2}.\frac{1}{\mathrm{tr}(\Sigma^j)^2}\,(I_{\ell_i^2}+P_{\ell_i,\ell_i})$$

$$\Big[\mathrm{tr}(\Sigma_1^4\otimes\ldots\otimes L'\Sigma_i^3\otimes\ldots\otimes\Sigma_j^3R\otimes\ldots\otimes\Sigma_k^4)\ -$$

$$\mathrm{tr}(L'\Sigma_i).\mathrm{tr}(\Sigma_1^4\otimes\ldots\otimes\Sigma_i^4\otimes\ldots\otimes\Sigma_j^3R\otimes\ldots\otimes\Sigma_k^4)\ -$$

$$\mathrm{tr}(\Sigma_jR).\mathrm{tr}(\Sigma_1^4\otimes\ldots\otimes L'\Sigma_i^3\otimes\ldots\otimes\Sigma_j^4\otimes\ldots\otimes\Sigma_k^4)\ +$$

$$\mathrm{tr}(L'\Sigma_i).\mathrm{tr}(\Sigma_jR).\mathrm{tr}(\Omega^4)\Big]$$

$$= \frac{1}{\mathrm{tr}(\Sigma^i)^2}.\frac{1}{\mathrm{tr}(\Sigma^j)^2}.\frac{\mathrm{tr}(\Omega^4)}{\mathrm{tr}(\Sigma_i)^4}.\frac{1}{\mathrm{tr}(\Sigma_j)^4}\,(I_{\ell_i^2}+P_{\ell_i,\ell_i})$$

$$\Big[\mathrm{vec}\ L)'(\mathrm{vec}(\Sigma_i^3))(\mathrm{vec}(\Sigma_j^3))'(\mathrm{vec}\ R)\ -$$

$$\mathrm{tr}(\Sigma_i^4).(\mathrm{vec}\ L)'\mathrm{vec}(\Sigma_i)(\mathrm{vec}(\Sigma_j^3))'(\mathrm{vec}\ R)\ -$$

$$\mathrm{tr}(\Sigma_j^4).(\mathrm{vec}\ L)'\mathrm{vec}(\Sigma_i^3)(\mathrm{vec}(\Sigma_j))'(\mathrm{vec}\ R)\ +$$

$$\mathrm{tr}(\Sigma_i^4)\mathrm{tr}(\Sigma_j^4)(\mathrm{vec}\ L)'\mathrm{vec}(\Sigma_i)(\mathrm{vec}(\Sigma_j))'(\mathrm{vec}\ R)\Big].$$

As a result,

$$(6.7.27)\qquad V_{ij} = \frac{2}{\mathrm{tr}(\Sigma^i)^2}.\frac{1}{\mathrm{tr}(\Sigma^j)^2}.\frac{\mathrm{tr}(\Omega^4)}{\mathrm{tr}(\Sigma_i)^4}.\frac{1}{\mathrm{tr}(\Sigma_j)^4}$$
$$\Big[(\mathrm{vec}(\Sigma_i^3)-\mathrm{tr}(\Sigma_i^4).\mathrm{vec}(\Sigma_i))(\mathrm{vec}(\Sigma_j^3)-\mathrm{tr}(\Sigma_j^4).\mathrm{vec}(\Sigma_j)')\Big].$$

Again, matrices V_{ij} are rank one matrices.

Notice the striking similarity of expressions (15) and (24) of V_{ii} for iterative and noniterative ULS estimators, respectively. The exact similarity of both expressions is due to the additional restrictions that have to be imposed to make the model identifiable. This will be discussed in the next section.

6.8. IDENTIFICATION

Once a solution $\hat{\Omega} = \hat{\Sigma}_1 \otimes ... \otimes \hat{\Sigma}_k$ has been obtained, we know that, e.g., $\hat{\Omega} = a_1\hat{\Sigma}_1 \otimes ... \otimes a_k\hat{\Sigma}_k$ with $\prod_{i=1}^{k} a_i = 1$, is also a solution. This means the parameter estimates $\hat{\sigma}_1$ to $\hat{\sigma}_k$ are under–identified and we need to impose k–1 additional restrictions to make the model identifiable. This can be done by setting the upper left element of k–1 arbitrary chosen Σ_i's equal to a fixed value, preferably one. A nice symmetric alternative to this method is to replace Ω by $\gamma\Omega$, γ a parameter to be estimated, and to scale each Σ_i by the upper left element of each Σ_i, as a point of reference for the other elements. So instead of estimation with respect to all elements of σ_i we invoke one restriction on each $\hat{\sigma}_i$. For instance

$$e_1'\sigma_i = e_1'\hat{\sigma}_i = 1 \qquad (6.8.1)$$

where e_1 is an $\ell_i \times 1$ vector with unit in first element and zeroes elsewhere.We then estimate the last $\ell_i^2 - 1$ elements of σ_i. The appropriate $\ell_i^2 \times \ell_i^2$ matrices N_i, that show this operation to the estimates and the blocks of the estimated covariance matrix can be obtained from a differentiation of each vector σ_i, scaled to its first element, to its corresponding unscaled version. Matrix N_i is expressed as follows.

$$N_i = I_{\ell_i^2} - (\frac{1}{\sigma_{i(1)}})\sigma_i(e_1 \otimes e_1)' \qquad (6.8.2)$$

where $\sigma_{i(1)}$ indicates the first element of σ_i, i.e., element $(\sigma_i)_{11}$ in Σ_i.

Together with the estimation of each $N_i\sigma_i$ (by this notation we indicate that the first element of σ_i should be ignored, since it is set to one due to the scaling) we estimate a scaling factor γ. We will show what this means for the covariance matrix V of each estimator. First of all, for each estimator we should actually ignore the first row and column of each V_{ii} and V_{ij} discussed

in the previous section, and therefore replace these blocks in the partitioned covariance matrices V by $N_iV_{ii}N_i'$ and $N_iV_{ij}N_j'$, respectively, which will lead to several rows and columns with zero elements. We will look at the estimators separately. In case of ULS estimation, we minimize $\mathrm{tr}(S-\gamma\Omega)^2$, where a priori $\sigma_{i(1)} = 1$ for all $i = 1,\ldots,k$. Setting the first-order derivative with respect to γ equal to zero leads to

$$(6.8.3) \qquad \hat{\gamma} = \frac{\mathrm{tr}(\hat{\Omega}S)}{\mathrm{tr}(\hat{\Omega})^2} = (\hat{\omega}'\hat{\omega})^{-1}\hat{\omega}'s = \hat{Q}_\gamma s,$$

with $\hat{Q}_\gamma$ implicitly defined. We define $Q_\gamma = \mathrm{plim}(\hat{Q}_\gamma) = (\omega'\omega)^{-1}\omega'$. The asymptotic variance of $\hat{\gamma}$ is

$$(6.8.4) \qquad V_\gamma = Q_\gamma(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)Q_\gamma'$$

$$= 2\frac{\mathrm{tr}(\Omega)^4}{(\mathrm{tr}(\Omega)^2)^2}$$

$$= 2\frac{\mathrm{tr}(\Sigma^i)^4}{(\mathrm{tr}(\Sigma^i)^2)^2}\cdot\frac{\mathrm{tr}(\Sigma_i)^4}{(\mathrm{tr}(\Sigma_i)^2)^2}, \quad i = 1,\ldots,k.$$

The scalar ratio in (4) is larger than $\frac{1}{\ell}$ in view of Schwarz's inequality.

The covariances between $\hat{\gamma}$ and each $N_i\hat{\sigma}_i$ are asymptotically equal to

$$(6.8.5) \qquad V_{i\gamma} = \frac{1}{\mathrm{tr}(\Omega)^2}N_i(X_i'X_i)^{-1}X_i'B_i(C_i\otimes C_i)(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)\omega$$

$$= 2\frac{\mathrm{tr}(\Sigma^i)^4}{(\mathrm{tr}(\Sigma^i)^2)^2}\cdot\frac{1}{\mathrm{tr}(\Sigma_i)^2}N_i\,\mathrm{vec}(\Sigma_i^3).$$

The scalar ratio $\frac{\mathrm{tr}(\Sigma^i)^4}{(\mathrm{tr}(\Sigma^i)^2)^2}$ is larger than $\frac{1}{\ell^i}$.

For noniterative ULS estimation we get the same expressions for $\hat{\gamma}$, V_γ, and $V_{i\gamma}$ as in (3), (4), and (5), respectively.

We will now show that the (rest of the) asymptotic covariance matrix V in (7.6) are equal for the iterative and noniterative ULS estimators. In case of iterative ULS estimation we get from (7.15)

$$(6.8.6)\qquad N_iV_{ii}N_i' = \frac{\mathrm{tr}(\Sigma^i)^4}{(\mathrm{tr}(\Sigma^i)^2)^2}(I_{\ell_i^2}+P_{\ell_i,\ell_i})(I_{\ell_i^2}-(\frac{1}{\sigma_{i(1)}})\sigma_i(e_1\otimes e_1)')(\Sigma_i\otimes\Sigma_i)$$

$$(I_{\ell_i^2}-(\frac{1}{\sigma_{i(1)}})\sigma_i(e_1\otimes e_1)')',$$

and $N_iV_{ij}N_j'$ accordingly. In case of noniterative ULS estimation we get from (7.24) a similar expression as (6). Equation (6) for noniterative ULS estimation incorporates the restrictions $e_1'\hat{\sigma}_i = 1$, whereas the iterative ULS estimators fulfill $\hat{\sigma}_i'\hat{\sigma}_i = 1$. Both expressions are however exactly similar, i.e., the same asymptotic results $N_iV_{ii}N_i'$ hold for both estimators. The exact similarity is due to the fact that $N_i(I-\sigma_i\sigma_i') = N_i$. The same remarks follow for comparisons of (7.18) and (7.27).

In case of ML estimation, we minimize $\mathrm{tr}((S-\gamma\Omega)\frac{1}{\gamma}\Omega^{-1})^2$, where we set $\sigma_{i(1)}$ equal to one for all $i = 1,...,k$. We then get

$$(6.8.7)\qquad \hat{\gamma} = \frac{\mathrm{tr}(\hat{\Omega}^{-1}S)}{\ell} = \frac{1}{\ell}(\mathrm{vec}\ \hat{\Omega}^{-1})'s = \hat{R}_\gamma s$$

with $\hat{R}_\gamma$ implicitly defined. We define $R_\gamma \equiv \mathrm{plim}(\hat{R}_\gamma) = \frac{1}{\ell}(\mathrm{vec}\ \Omega^{-1})'$. The asymptotic variance of $\hat{\gamma}$ is

$$(6.8.8)\qquad V_\gamma = R_\gamma(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)R_\gamma' = \frac{2}{\ell}$$

which points out the statistical efficiency of the ML estimator. The covariances between $\hat{\gamma}$ and each $N_i\hat{\sigma}_i$ are asymptotically equal to

$$(6.8.9)\qquad V_{i\gamma} = \frac{1}{\ell\ \ell^i}N_i(\sigma^i\otimes\Sigma_i\otimes\Sigma_i)'(\Sigma^i\otimes\Sigma^i\otimes\Sigma_i\otimes\Sigma_i)^{-1}B_i(C_i\otimes C_i)$$
$$(I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)\mathrm{vec}(\Omega^{-1})$$
$$= \frac{1}{\ell\ \ell^i}N_i(\mathrm{vec}((\Sigma^i)^{-1})\otimes I_{\ell_i}\otimes I_{\ell_i})'(I_{\ell^2}+(P_{\ell^i,\ell^i}\otimes P_{\ell_i,\ell_i}))$$
$$B_i(C_i\otimes C_i)(\Omega\otimes\Omega)\mathrm{vec}(\Omega^{-1})$$
$$= \frac{1}{\ell\ \ell^i}N_i(I_{\ell_i^2}+P_{\ell_i,\ell_i})(\mathrm{vec}((\Sigma^i)^{-1})\otimes I_{\ell_i}\otimes I_{\ell_i})'(\Sigma^i\otimes\Sigma^i\otimes\Sigma_i\otimes\Sigma_i)$$
$$B_i\mathrm{vec}(\Sigma^i\otimes\Sigma_i)^{-1}$$
$$= \frac{2}{\ell}N_i\sigma_i.$$

In case of WLS estimation we minimize $\mathrm{tr}((S-\gamma\Omega)S^{-1})^2$, where we set $\sigma_{i(1)}$ equal to one for all $i = 1,...,k$. Setting the first–order derivative with respect to $\hat{\gamma}$ equal to zero leads to

$$\hat{\gamma} = \frac{\mathrm{tr}(\hat{\Omega}S^{-1})}{\mathrm{tr}(\hat{\Omega}S^{-1})^2} . \tag{6.8.10}$$

Asymptotically, we get the same results as with ML.

The basic properties for unconstrained ML and GLS estimation are well known (see, e.g., Browne (1974) and Lee (1977)). These asymptotic statistical theorems for multivariate analysis with latent variables in general, or covariance structure models in particular, are also stated in Bentler and Weeks (1982). More general statistical results deal with models whose parameters are subject to constraints. Although Aitchison and Silvey (1958) provided the statistical basis for obtaining and evaluating constrained ML estimates, their methods were not applied to covariance structure models (or to latent variable models) until the work of Lee (1980), Lee and Bentler (1980), and Shapiro (1983). Their GLS results extend Aitchison and Silvey's ML results.

6.9. PROGRAMMING CONSIDERATIONS

Expressions (2.2), (5.2), (5.6), (5.7), (6.1), (7.2), (7.5), and (7.7) should be evaluated by procedures that work at a level of scalar multiplication because of the appearance of large sparse (0,1)–matrices. The main modules for the (0,1)–matrices for actual program implementations are discussed in chapter 9. In this section we discuss several programming considerations that are of particular interest for k–mode CSA. We remark that in our description of our source–code we have made some modifications for convenience of display by preliminary storing the operations of the matrices P, C_i and B_i into vectors arranged as columns in a matrix. For instance, the operation of C_i to a series of integers $1,...,\ell$ is stored in row i of a matrix called *table_C*, and contains integers which state the new position of an element of vector x in vector $y \equiv C_i x$. The matrix *table_C* has k such rows, each of length ℓ.

This means that the operation $s_i = \text{vec}(C_i S C_i')$ can be added to the source–code as follows (for any choice of k).

(6.9.1)
- as assignments:
  $\ell := \ell[1]\times\ldots\times\ell[k]$.
- as loops:
  for $i_1 := 1$ to ℓ do
    for $i_2 := 1$ to ℓ do
      $s_i[(i_1-1)\ell+i_2] := S[table\_C[i,i_1], table\_C[i,i_2]]$.

The operation $\tilde{s}_i = B_i s_i$ can be added to the source–code as follows.

(6.9.2)
- as assignments:
  $\ell := \ell[1]\times\ldots\times\ell[k]$.
- as loops:
  for $i_1 := 1$ to ℓ do
    for $i_2 := 1$ to ℓ do
      $\tilde{s}_i[(i_1-1)\ell+i_2] := s_i[table\_B[i,(i_1-1)\ell+i_2]]$.

Let us now first look back at the iterative ULS estimator $\hat{\sigma}_i$ in (6.1). This equation is equivalent to

(6.9.3)
$$\hat{\sigma}_i = (\hat{\sigma}^{i\,\prime}\hat{\sigma}^i)^{-1}\tilde{S}_i\hat{\sigma}^i$$
$$= (\text{vec}\ \hat{\Sigma}^i)'(\text{vec}\ \hat{\Sigma}^i)\tilde{S}_i\text{vec}\ \hat{\Sigma}^i.$$

In our analysis of (3) we will focus on the term $\tilde{S}_i\text{vec}\ \hat{\Sigma}^i$ and stepwise show our constructive view of this world.

Step 1.

Suppose we have computed new estimates $\hat{\Sigma}_1$ to $\hat{\Sigma}_k$ in one round of iteration, each "vectorized" to a vector $\hat{\sigma}_1$ to $\hat{\sigma}_k$, respectively. In concordance with our actual program implementations we assume we have stored $\hat{\Sigma}_i$ in $\hat{\sigma}_i$ by a row–vectorization, i.e., $\hat{\sigma}_i = \text{vec}(\hat{\Sigma}_i')$ instead of (4.3). Due to symmetry of matrices Σ_i, $i = 1,\ldots,k$, this is of no further consequence. We ignore the hats for now.

Step 2.

Assume we want to Kronecker-multiply element (i_1,j_1) of matrix Σ_1 with element (i_2,j_2) of matrix Σ_2. We know we must then multiply $\sigma_1[(i_1-1)\ell[1]+j_1]$ with $\sigma_2[(i_2-1)\ell[2]+j_2]$ and would place the result in matrix $\Sigma_1\otimes\Sigma_2$ at position $((i_1-1)\ell[1]+j_1,(i_2-1)\ell[2]+j_2)$. We then also know where any elements of $\Sigma_1\otimes\Sigma_2\otimes\Sigma_3$ would have to placed in this matrix from the positions of elements in Σ_1 to Σ_3. And so forth.

Step 3.

Of course we want to handle the choice of any value of k. This is achieved by a recursive call of the routine that structures matrix Σ^i. The recursive call of the routine in its own body will enable this choice of any value of k, since after each call a new set of two necessary do-loops is created with which the two indices for positioning an element of Σ^i, and the value of the intermediary product to form an element of Σ^i, will be updated. This position of elements in Σ^i is indicated in the routine by two indices, named 'first_index_sigma_but_i' and 'second_index_sigma_but_i'. The two indices are each ℓ^i times updated, according to a test to see whether a counter, to keep track of the number of recursive calls, still differs from the value index i chosen.

Step 4.

Each time we have reached for counter its upper boundary value $k+1$, we have computed an element of Σ^i, at position (first_index_sigma_but_i, second_index_sigma_but_i). We then also know its position 'total_index_sigma_but_i' in the $(\ell^i)^2\times 1$ vector σ^i. We can then retrieve the correct ℓ_i^2 elements in S_i, which each have to be post-multiplied with this element of σ^i, namely $\tilde{S}_i[i_1$, total_index_sigma_but_i], where i_1 runs from 1 to ℓ_i^2. These same elements can also be retrieved from $\tilde{s}_i$, and due to a scheme similar to (2) we can likewise take the correct elements from s_i. Eventually we then have computed a term which is a product of an element of $\tilde{S}_i$ and one of σ^i, and which should be added to the prior value of $\tilde{S}_i\sigma^i[i_1]$ if this has not occurred $(\ell^i)^2$ times already. Each time counter reaches its upper boundary value $k+1$, we also update the prior value (or starting value zero at the first occurrence) of scalar $\sigma^{i'}\sigma^i$.

The noniterative ULS estimator, induced by (6.4), also causes no extra problems for our modules. Once we have computed s_i and know that $\text{vec}(\tilde{S}_i) = \tilde{s}_i$

$= B_i s_i = B_i \mathrm{vec}(S_i)$, we can compute $\tilde{S}_i \tilde{S}_i'$ from elements of S_i.

The iterative ML estimator in (5.6) closely resembles the iterative ULS estimator as to its structure, and similar steps as steps 1 to 4 are taken. The point to make is that due to the Kronecker product structure of Σ^i in (4.3), we position elements in $(\Sigma^i)^{-1}$ merely with the help of $(\Sigma_1)^{-1}$ to $(\Sigma_k)^{-1}$, as in step 2.

The iterative WLS estimator closely resembles the iterative ULS estimator as to the part in (5.7), and as such the prior information in steps 1 to 4 holds once again. We shed some more light on part of its first term in (5.2), namely

$$(6.9.4) \qquad X_i'(S_i \,\pi\, S_i)^{-1} X_i = (\sigma^i \otimes I_{\ell_i} \otimes I_{\ell_i})' B_i' (S_i \otimes S_i)^{-1} B_i (\sigma^i \otimes I_{\ell_i} \otimes I_{\ell_i}).$$

Again we want to handle the choice of any value for k. This is achieved by a recursive call of the routine that calculates elements of matrix Σ^i. The same idea as in step 3 applies. However, since we now construct the right-hand side in (4) instead of $\tilde{S}_i \sigma^i$, a few minor changes to the description in step 4 have to be made. First, we need two pairs of indices, (first_index_sigma_but_i_1, second_index_sigma_but_i_1) and (first_index_sigma_but_i_2, second_index_sigma_but_i_2), both of which indicate the position of an element in Σ^i. We then also know their positions, denoted by 'total_index_sigma_but_i_1' and 'total_index_sigma_but_i_2', in σ^i. These two indices determine which pair of $\ell_i \times \ell_i$ blocks of S_i^{-1} have to be Kronecker-multiplied with each other. The two pairs of indices (index1_first_block_of_Si, index2_first_block_of_Si) $(=(p_1,p_2))$ and (index1_second_block_of_Si, index2_second_block_of_Si) $(=(q_1,q_2))$ indicate which Kronecker product $[S_i^{-1}]_{p_1,q_1} \otimes [S_i^{-1}]_{p_2,q_2}$ would be formed, i.e., which two $\ell_i \times \ell_i$ blocks of S_i^{-1} are needed each time counter reaches its upper boundary value $k+1$. These two blocks are easier to recover from $\ell_i^2 \times \ell_i^2$ matrix $\tilde{S}_i^{-1}$, since the vectorizations of the two needed blocks are two respective columns of $\tilde{S}_i^{-1}$. We have not stored $\tilde{S}_i^{-1}$ or $\tilde{s}_i^{-1}$ but we know that $\tilde{s}_i = B_i s_i$. This means that if we know which two columns of $\tilde{S}_i^{-1}$ are needed each time to use the right respective Kronecker operands $[S_i^{-1}]_{p_1,q_1}$ and $[S_i^{-1}]_{p_2,q_2}$, we then know which two pairs of ℓ_i^2 elements of $\tilde{s}_i^{-1}$ are needed, henceforth which two pairs of ℓ_i^2 elements of s_i^{-1} are needed.

The covariance matrices of the estimates according to the four different estimators also show their dependence upon the (0,1)–matrices, in particular the permutation matrix P_{ℓ_i,ℓ_i}. The necessary block–matrices V_{ii} and V_{ij} for each estimator are decomposed to such a structure that matrix multiplication is unnecessary and that operative work at a scalar level is no laborious task. We note that these block–matrices are not actually created, but will be directly placed at correct block positions in matrices V. We will discuss the necessary expressions separately.

From (7.9) and (7.14) it is clear that the block–matrices V_{ii} and V_{ij}, respectively, for the composition of V for ML and WLS estimates, can be straightforward calculated from elements of $\hat{\sigma}_i$ and $\hat{\sigma}_j$. Expressions (7.15) and (7.18) for the block–matrices V_{ii} and V_{ij}, respectively, concerning the iterative ULS estimator raise no new problems. Two remarks to this end are the following. First, the scalar $\mathrm{tr}(\Sigma^i)^4$ is calculated efficiently as a consequence of the matrix rule

$$\mathrm{tr}(\Sigma^i)^4 = \prod_{\substack{j=1 \\ j\neq i}}^{k} \mathrm{tr}(\Sigma_j)^4. \tag{6.9.5}$$

This may be further decomposed to

$$\mathrm{tr}(\Sigma^i)^4 = \prod_{\substack{j=1 \\ j\neq i}}^{k} \sigma_j'(\Sigma_j \otimes \Sigma_j)\sigma_j \tag{6.9.6}$$

which eases the understanding of the order of necessary do–loops. The same goes for the scalar $\mathrm{tr}(\Sigma^i)^2$. Secondly, $\mathrm{vec}(\Sigma_i{}^3)(\mathrm{vec}(\Sigma_j{}^3))'$ in expression (7.18) is calculated (i.e., iteratively updated at a scalar level) by using the matrix rule

$$\mathrm{vec}(\Sigma_i{}^3) = (\Sigma_i \otimes \Sigma_i)\sigma_i \tag{6.9.7}$$

which also eases and enlightens the order of necessary do–loops. Due to the close similarity of expressions (7.24) and (7.27) to the expressions (7.15) and (7.18), respectively, the formation of the block–matrices V_{ii} and V_{ij} for the noniterative ULS estimator poses no extra problems.

6.10. EMPIRICAL APPLICATIONS AND SIMULATIONS

6.10.1. Empirical applications

6.10.1.1. 2-mode biometric data

We used the (8×8) covariance matrix in the numerical example in Swain's (1980) article for illustration purpose and to test the capabilities of our four algorithms. The (8×8) covariance matrix presents the covariances between measurements of weights for four muscles from both left and right hind legs, and was based upon a sample of 56 calves. It is expected that the relationship between the muscles should be the same on both sides of the animals, and that the correlation between sides for each muscle could be the same. We therefore follow Swain's proposal for modeling the structure of Ω as $\Omega = \Sigma_1 \otimes \Sigma_2$. We estimated the ML estimates and a scaling factor γ when imposing k restrictions, such as discussed in section 8. We did the same for ULS (noniterative and iterative) and WLS. The (8×8) covariance matrix of input is mentioned in Table 1. The original covariance matrix was rescaled by Swain and we cannot track the original measurements of the muscle weights. However, in the sequel we assume that the (8×8) covariance matrix in Table 1 is the true one of interest.

Table 6.10.1. Rescaled covariance matrix S for hind leg muscles of calves.

1.000							
0.599	1.000						
0.611	0.480	1.000					
0.581	0.513	0.601	1.000				
1.109	0.664	0.790	0.718	1.426			
0.687	0.977	0.541	0.619	0.847	1.129		
0.639	0.446	0.971	0.632	0.878	0.573	1.052	
0.662	0.662	0.630	0.949	0.833	0.765	0.705	1.061

We note in addition that Brien, James and Venables (1988) performed a procedure, similar to an analysis of variance, on the Fisher z–transforms of the elements of the correlation matrix. They tested the same relationships mentioned above, leading to a null hypothesis of symmetry of the pattern of the population correlation matrix.

The estimates with estimated standard errors in parentheses, are mentioned in Table 2. Before we discuss the results we will define the meaning of several statistics in Table 2. Firstly, the values of the four different criteria are calculated for each estimator. The ULS criterion is

(6.10.1) $\quad \mathrm{tr}(S-\hat{\gamma}\hat{\Omega})^2.$

The value of the so–called WMLS (Weighted Maximum Likelihood Least Squares) criterion for the ML estimator is that of

(6.10.2) $\quad \mathrm{tr}((S-\hat{\gamma}\hat{\Omega})\frac{1}{\hat{\gamma}}\hat{\Omega}^{-1})^2 \;=\; \mathrm{tr}(I-\frac{1}{\hat{\gamma}}S\hat{\Omega}^{-1})^2.$

The corresponding value of the ML criterion is that of

(6.10.3) $\quad \ln|\hat{\gamma}\hat{\Omega}| \;+\; \frac{1}{\hat{\gamma}}\mathrm{tr}\; S\hat{\Omega}^{-1}.$

The WLS criterion is

(6.10.4) $\quad \mathrm{tr}(I-\hat{\gamma}S^{-1}\hat{\Omega})^2.$

Secondly, the number of iterations indicates the sum of the number of computations of a new estimate $\hat{\Sigma}_i$ for any i. For example, one full round of new updates $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$ to their former estimates means an extra 2 iterations. In case of noniterative ULS we always have merely k iterations, so in this case 2 iterations. Thirdly, the variable 'CPU–time' stands for the used up computing time at a PC (an Olivetti M24, with clock speed 8Mhz and with Intel 8087 math co–processor).

Table 6.10.2. Estimates $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$ for Swain's (1980) covariance matrix.

	$\hat{\Sigma}_1$(iterative ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iterative ULS)	$\hat{\Sigma}_1$(iterative WLS)
11	1.0000	1.0000	1.0000	1.0000
21	0.9033 (0.0293)	1.0725 (0.0451)	1.0725 (0.0451)	0.9042 (0.0289)
22	1.0083 (0.0589)	1.2762 (0.1020)	1.2762 (0.1020)	1.0052 (0.0580)
	$\hat{\Sigma}_2$(iterative ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iterative ULS)	$\hat{\Sigma}_2$(iterative WLS)
11	1.0000	1.0000	1.0000	1.0000
21	0.5375 (0.0718)	0.6017 (0.0954)	0.6017 (0.0954)	0.4358 (0.0858)
31	0.4717 (0.0661)	0.6280 (0.0903)	0.6280 (0.0903)	0.3928 (0.0776)
41	0.4009 (0.0769)	0.6010 (0.0928)	0.6010 (0.0928)	0.3621 (0.0872)
22	0.8657 (0.1336)	0.8723 (0.1782)	0.8723 (0.1782)	1.0135 (0.1727)
32	0.3839 (0.0773)	0.4363 (0.1080)	0.4363 (0.1080)	0.3363 (0.0920)
42	0.3733 (0.0849)	0.5425 (0.1220)	0.5425 (0.1220)	0.4804 (0.1112)
33	0.7115 (0.1115)	0.8513 (0.1666)	0.8513 (0.1666)	0.8281 (0.1412)
43	0.3958 (0.0821)	0.5489 (0.1190)	0.5489 (0.1190)	0.4649 (0.1034)
44	0.8222 (0.1394)	0.8445 (0.1706)	0.8445 (0.1706)	0.9832 (0.1730)
$\hat{\gamma}$	1.1197 (0.0668)	1.0547 (0.1797)	1.0547 (0.1797)	0.6366 (0.0668)
ULS–crit	2.7693	0.1263	0.1263	11.8779
WLS–crit	3.1748	1.9910	1.9910	1.1523
ML–crit	–1.5226	1.8261	1.8261	–0.6171
WMLS–crit	1.7824	25.2868	25.2867	7.4165
iterations	10	2	6	31
CPU–time	9	1	4	154

Our computing time (expressed in seconds; we have ignored fractionals) is the time between the start of a first estimate and a final estimate, whether this is of Σ_1 or Σ_2. This final estimate is a new update of the former estimate of Σ_1 or Σ_2, depending upon the relative difference between two successive values of the criterion at hand. We establish convergence if this relative difference is smaller than 10^{-6}. Lastly, the estimate $\hat{\gamma}$ is the estimate of the multiplicative scaling factor γ after each estimate $(\hat{\Sigma}_1, \hat{\Sigma}_2)$ of (Σ_1, Σ_2) is scaled to its first upper left element.

Our ML estimates correspond to Swain's ML estimates. Swain incorporates one restriction by scaling the matrix $\hat{\Sigma}_1$ such that the first element is set to one, and multiplies his estimate $\hat{\Sigma}_2$ with a factor 0.903272, which is the value of the first element of $\hat{\Sigma}_1$ before $\hat{\Sigma}_1$ was scaled. The first element of his estimate $\hat{\Sigma}_2$ became 1.119715, which is the value of our estimate $\hat{\gamma}$. This should come as to surprise, since due to our nice symmetric alternative of incorporating one restriction on each matrix, we divided matrix $\hat{\Sigma}_1$ by 0.903272 and matrix $\hat{\Sigma}_2$ by 1.239621, and estimated γ according to (8.7) which shows $\hat{\gamma}$ is linear in each Σ_i.

Notice also the asymptotic efficiency of the ML estimator. The standard errors of the ML estimates are smaller than the corresponding values by the ULS and WLS approaches, except for the standard errors of the elements of $\hat{\Sigma}_1$. Due to the asymptotic normality of each estimator, the significance of the difference of an estimate to its hypothesized value can be tested by using asymptotic t–values. The small standard errors and the t–value of approximately 2.70 for a t–distribution with 44 degrees at a significance level of 0.01 lead to confidence intervals with small type–I error (0.01). The noniterative and iterative ULS estimates closely resemble each other. Differences between these two estimates started in the 7th decimal which we have not reported in Table 2. For example, the noniterative ULS estimate of γ was approximately 1.0547410 whereas its iterative ULS estimate was 1.0547413. The noniterative ULS approach could be preferred to the iterative approach on grounds of less used up computing time and easier (insight in) program necessities. The WLS estimation results show that the standard errors are close to the ML standard errors. The WLS standard errors are less than the ULS standard errors which indicates the asymptotic efficiency of the WLS estimator to the ULS estimator. The WLS estimator is the most costly approach in terms of used up computing time per iteration and the number of iterations. This should come as no surprise considering the fact that the computation of matrices $\left[X_i'(S_i \ \pi \ S_i)^{-1}X_i\right]^{-1}$ requires $O((\ell^i)^4)$ operations per iteration, whereas the calculation of the iterative ULS and ML estimates requires $O((\ell^i)^2)$ operations per iteration. This should signal the analyst to prefer to report ML estimates whenever ML estimation is possible, i.e., whenever the normality assumption holds in the true world. Notice the by far lower WLS estimate of $\hat{\gamma}$ as compared to the others. Our explanation for this is that the sample covariance matrix S has such an influence on the path of convergence that another local optimum for the WLS estimator is searched and found.

6.10.1.2. 2-mode psychometric data

The second example to illustrate the theory discussed in the previous sections, is the set of 2-mode psychometric data reported by Campbell and Fiske (1959). The dataset concerns an artificial MTMM matrix, presented in Table 3, which involves three different traits (A, B and C), each measured by three methods (1, 2 and 3). Labels have been added for different regions of the matrix, and such have been provided in Table 3.

Table 6.10.3. Campbell and Fiske (1959) MTMM covariance matrix[a].

		Method 1			Method 2			Method 3		
	Trait	A_1	B_1	C_1	A_2	B_2	C_2	A_3	B_3	C_3
	A_1	(0.89)								
Method 1	B_1	0.51	(0.89)							
	C_1	0.38	0.37	(0.76)						
	A_2	0.57	0.22	0.09	(0.93)					
Method 2	B_2	0.22	0.57	0.10	0.68	(0.94)				
	C_2	0.11	0.11	0.46	0.59	0.58	(0.84)			
	A_3	0.56	0.22	0.11	0.67	0.42	0.33	(0.94)		
Method 3	B_3	0.23	0.58	0.12	0.43	0.66	0.34	0.67	(0.92)	
	C_3	0.11	0.11	0.45	0.34	0.32	0.58	0.58	0.60	(0.85)

[a] The validity diagonals are the three sets of underlined values. The reliability diagonals are the three sets of values in parentheses.

This MTMM matrix has been chosen by Campbell and Fiske (1959) such as to illustrate their four conditions which should be considered in a validation process of tests for individuals. In the first place, the entries in the validity diagonals (which could also be designated as monotrait-heteromethod values) should be significantly different from zero and sufficiently large to encourage further examination of validity. Second, a validity diagonal value should be higher than the values lying in its column and row in the

heterotrait–heteromethod triangles. Third, values in the validity diagonals should be higher than values lying in its column and/or row in the heterotrait–monomethod triangles. Fourth, the same pattern of trait interrelationship should be shown in all of the heterotrait triangles of both the monomethod and heteromethod blocks. The artificial data in Table 3 meet these conditions to a high degree. The third condition however is met to some degree for the variables A_1, B_1, and C_1 but for the other variables, A_2, B_2, etc., it is not met.

Our results in Table 4 indicate the following. First, the iterative ULS estimator equals the noniterative estimator due to convergence in the first iteration of the iterative procedure with noniterative ULS estimates as starting values. Second, ML estimates are unobtainable due to singularity of estimate $\hat{\Sigma}_1$ in the second iteration of the iterative procedure. Third, WLS estimates are unobtainable due to (near–) singularity of matrix S. Fourth, the standard errors are reported with too high values. These have to be divided by $N^{1/2}$, where we took $N = 1$. As for the fulfillment of the four so–called Campbell–Fiske conditions by the ULS estimators we note the following. The entries in the three validity diagonals range from 0.40 to 0.68, different from zero. All validity diagonal values are approximately 1½ to 2 times larger than values lying in its column and row in the heterotrait–heteromethod diagonals. Third, values of the validity diagonals for most variables are not large enough to satisfy the third Campbell–Fiske condition, which indicates that this condition is not satisfied by the original MTMM matrix. The last condition however is clearly fulfilled since the estimated covariance matrix shows the same pattern in each of the heterotrait triangles as in the original MTMM matrix.

Flamer (1983) also used the Campbell–Fiske conditions to assess the effects on Likert scale MTMM matrix validity, when the number of response alternatives for each Likert type item are varied. He used data on attitudes of adults toward discipline of children, mathematics and the law. The scores were given by three measurement techniques.

Table 6.10.4. Estimates $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$ for Campbell and Fiske (1959) MTMM covariance matrix[a)].

	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_2$(noniter. ULS)
11	1.0000	1.0000
21	0.5310 (0.8781)	0.6070 (0.7296)
31	0.5353 (0.8754)	0.4702 (0.7381)
22	1.2023 (1.9256)	0.9988 (1.4584)
32	0.7903 (1.4238)	0.4711 (0.9158)
33	1.2012 (1.9189)	0.8661 (1.3738)

$\hat{\gamma}$	0.8608 (1.1896)
ULS–crit	0.4743
WLS–crit	.
ML–crit	–0.3310
WMLS–crit	4.9191
R^2	0.7634
iterations	2
CPU–time	2

a) Dots indicate estimates or values were unobtainable.

6.10.1.3. 3-mode psychometric data

The third application of the model to data is the analysis of a three–mode MOMTMM matrix of Cudeck (1988).

Cudeck (1988) used data extracted from what is known as the Growth Study. The full version of this study followed nearly 46000 school–aged students in four cohorts. Cudeck (1988) confined his attention to female students in the third cohort who had complete data for three testings. These testings took place at two–year intervals, beginning in grade 7. Further confinement was made to just four tests that were part of the total battery of tests that were administrated in the Growth Study. The four tests constitute a generic MTMM design, in which methods are the Sequential Test of Educational Progress (STEP) and the School and College Ability Test (SCAT), while traits are quantitative and verbal abilities as measured by test subsections of STEP and SCAT. The resulting subsample's data (from N = 2163 girls) are reproduced (from Cudeck (1988)) in Table 5.

Cudeck (1988) fitted a three-way composite direct product model (cf. Browne (1984a)) to these data by means of WLS. Several studies, (e.g., Browne (1974) and Lee (1977)) discuss a property that provides a statistical basis for evaluating a model. This property relates the minimum function values (3.1) (up to a scalar) and (3.4) to a chi-square statistic. We recognize that more refined approaches to the question of choosing among competing acceptable models have recently been suggested (e.g., Bentler and Bonett (1980), Cudeck and Browne (1983)). Bentler and Bonett (1980) propose a general null model, based on modified independence among variables, that can be used in a procedure that sequentially evaluates the statistical necessity of various sets of parameters. Cudeck and Browne (1983) propose a cross-validation procedure, using part of data samples as a calibration sample and the second part as the validation sample. These refinements are not pursued here, since we use the data merely as illustrative examples for outcomes of the model (2.2).

The normal theory WLS test statistic for Cudeck's (1988) estimates was found to be $\chi^2 = \frac{1}{2}N \operatorname{tr}((\Sigma - S)S^{-1})^2 = 359.2$ with degrees of freedom 49, which indicates that the model should be rejected. We have fitted our model (2.2) to these same data by means of WLS, ML, iterative ULS, and noniterative ULS. Our results are mentioned in Table 6. It turns out that our model should also be rejected on basis of the WLS estimates. The test statistic value is $\chi^2 = 2348.65$ with 68 degrees of freedom, which is by far larger than the critical value at the upper tail chi-square probability $p = 0.05$. Furthermore, the MOMTMM matrix in Table 5 does not satisfy the second, third, and fourth Campbell-Fiske conditions. The noniterative and iterative estimates in Table 6 lead to an estimate of the MOMTMM matrix that satisfies all four Campbell-Fiske conditions. The ML estimator produces validity diagonal elements ranging in value from 74.1 to 162.7, whereas elements of the heterotrait triangles range in value from 13.8 to 90.2. The consequence is that the ML estimate of the MOMTMM matrix satisfies the first, second, and fourth conditions. The same holds for the WLS estimator with the understanding that validity diagonal elements range in value from 5.2 to 18.2, whereas elements of the heterotrait triangles range in value from 0.18 to 13.4.

Table 6.10.5. Cudeck (1988) MOMTMM covariance matrix[a].

Occasion		1961				1963				1965			
	Method	STEP		SCAT		STEP		SCAT		STEP		SCAT	
	Trait	M[b]	R[c]	Q[d]	V[e]	M	R	Q	V	M	R	Q	V
1961	STEP M	1704											
	STEP R	1464	2537										
	SCAT Q	1202	1341	1579									
	SCAT V	1101	1618	1048	1518								
1963	STEP M	1286	1470	1236	1129	1799							
	STEP R	1309	1923	1222	1430	1492	2421						
	SCAT Q	1482	1658	1512	1243	1682	1641	2472					
	SCAT V	1271	1806	1169	1524	1378	1729	1508	1933				
1965	STEP M	1379	1574	1305	1214	1526	1574	1734	1437	2532			
	STEP R	1414	2090	1306	1604	1564	2050	1699	1876	1774	2810		
	SCAT Q	1585	1751	1613	1333	1767	1721	2166	1621	1930	1898	2843	
	SCAT V	1343	1893	1222	1604	1431	1788	1563	1879	1568	2057	1762	2255

a) From "Stability and instability in academic growth: a compilation of longitudinal data" (final report, USOE, research no. 0-0140), by Hilton, T.L., Beaton, A.E. and C.P. Bower (1971), Princeton, N.J.
Covariances have been multiplied by 10.

b) M = STEP Mathematics.

c) R = STEP Reading.

d) Q = SCAT Quantitative.

e) V = SCAT Verbal.

Table 6.10.6. Estimates $\hat{\Sigma}_1$, $\hat{\Sigma}_2$, and $\hat{\Sigma}_3$ for Cudeck (1988) MOMTMM covariance matrix.

	$\hat{\Sigma}_1$(iter. ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iter. ULS)	$\hat{\Sigma}_1$(iter. WLS)
11	1.0000	1.0000	1.0000	1.0000
21	0.7205 (0.0098)	0.9761 (0.0104)	0.9785 (0.0103)	0.1842 (0.0099)
31	0.7671 (0.0099)	1.0440 (0.0121)	1.0464 (0.0120)	0.2275 (0.0127)
22	1.1887 (0.0192)	1.1913 (0.0227)	1.1905 (0.0224)	0.8770 (0.0185)
32	0.8982 (0.0158)	1.1757 (0.0217)	1.1781 (0.0215)	0.2753 (0.0129)
33	1.4476 (0.0240)	1.4126 (0.0287)	1.4132 (0.0285)	1.4366 (0.0303)
	$\hat{\Sigma}_2$(iter. ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iter. ULS)	$\hat{\Sigma}_2$(iter. WLS)
11	1.0000	1.0000	1.0000	1.0000
21	0.3438 (0.0068)	0.8993 (0.0079)	0.9008 (0.0079)	0.1405 (0.0067)
22	0.7130 (0.0114)	0.9475 (0.0154)	0.9482 (0.0153)	1.0030 (0.0103)
	$\hat{\Sigma}_3$(iter. ML)	$\hat{\Sigma}_3$(noniter. ULS)	$\hat{\Sigma}_3$(iter. ULS)	$\hat{\Sigma}_3$(iter. WLS)
11	1.0000	1.0000	1.0000	1.0000
21	0.3077 (0.0074)	0.8649 (0.0130)	0.8665 (0.0129)	0.1019 (0.0074)
22	0.7965 (0.0131)	1.1154 (0.0274)	1.1162 (0.0273)	0.7229 (0.0126)
$\hat{\gamma}$ [a)]	181.1615 (0.0088)	160.7507 (0.0299)	160.2265 (0.0299)	66.2138 (0.0088)
ULS–crit	$1.6*10^6$	55777.8901	55768.7142	$3.4*10^6$
WLS–crit	30.2734	6.3942	6.3624	2.1717
ML–crit	66.0093	358.3727	373.2153	83.9943
WMLS–crit	7.2062	38481.5953	42373.1523	525.2828
R^2	0.9999	0.9901	0.9891	0.9998
iterations	15	3	6	46
CPU–time	23	2	6	1138

[a)] Value of γ is divided by 10.

We notice that the noniterative and iterative ULS estimators are close to each other, even though more than a full round of new estimates is produced in the iterative approach. We also notice the small standard errors of the parameter estimates, which means we have significant parameter estimates. One riddle left however is the significant difference between the value of the WLS estimator of γ and its value by the other estimators. One possible explanation may be is the influence of the inverse of S in the selected path of convergence in the WLS algorithm.

6.10.2. Simulations

We did four extra simulations with artificial data to test convergence, speed, precision and stability properties of the algorithms and their solutions. We selected different values of k, i.e., the number of modes, and different values of ℓ_1 to ℓ_k. For each of these datasets we selected values for the elements of the positive definite matrices Σ_i, $i = 1,...,k$. The input for these matrices is presented in the first column of Tables 7 to 10. After these choices of matrices Σ_i, we then calculated for each dataset the 'true' sample covariance matrix according to $S=\Sigma_1\otimes...\otimes\Sigma_k$. To these sample covariance matrices S, a disturbance ε_{ij} was added to each element S_{ij} without destroying symmetry, i.e.,

$$\varepsilon_{ij} = \varepsilon_{ji} = \frac{\text{Random}}{5} - \frac{\text{Random}}{5},\ i,j = 1,...,\ell. \tag{6.10.5}$$

Here 'Random' stands for the random number generator which creates random numbers that lie within the interval [0,1]. We noticed that each element of S was disturbed with an error ε (negative or positive) of, on average, approximately $5*10^{-2}$. For each dataset we then estimated Σ_i, $i=1,...,k$, with each of the estimation techniques discussed in the prior sections. Before we discuss the results we make a few preliminary remarks. Firstly, each of the standard errors presented in the Tables should be divided by $(N)^{\frac{1}{2}}$, where N denotes the number of observations upon which the 'true' sample covariance matrix S was based. In our case we have not bothered with the more elaborate approach of generating random data x from a normal distribution, preferably $N(0,I)$, to obtain more handsome simulation experiments. We would then have gone through the following steps.

1. Generate ℓ vectors x, each of length N, where N is large, say, $N = 1000$.

2. Compute a sample product moment matrix

$$S_o = \frac{1}{N}\sum_{p=1}^{N} x_p x_p', \tag{6.10.6}$$

which would be a Wishart distributed stochast.

3. Select $\ell_i \times \ell_i$ matrices A_i, $i = 1,...,k$.

4. Compute

$$S = (A_1 \otimes ... \otimes A_k) S_o (A_1 \otimes ... \otimes A_k)'. \tag{6.10.7}$$

In this case, S_o would be approximately an $(\ell \times \ell)$ identity matrix, so that matrix $A_i A_i'$ approximates Σ_i, for $i = 1,...,k$.

5. And finally, disturb matrix S with random errors ε_{ij}, preferably from a Wishart distribution.

Secondly, the extra statistic $\|\Sigma_i - \hat{\Sigma}_i\|^2$ we mention in each Table expresses the absolute difference between two vectors in the 2-norm, i.e.,

$$\|\Sigma_i - \hat{\Sigma}_i\|^2 = (\sigma_i - \hat{\sigma}_i)'(\sigma_i - \hat{\sigma}_i). \tag{6.10.8}$$

These values thus present an absolute measure of (the effect of the disturbance of the 'true' covariance matrix S for) the difference between the 'true' matrices Σ_i of input and the estimates. Thirdly, the number of iterations we report for iterative estimates should actually be risen with k if we are to compare this number with the number of iterations for noniterative ULS estimation. Why ? Because we chose as starting values for each iterative estimator not some random matrices Σ_i, but chose the noniterative ULS estimates as starting values. However, we have also used identity matrices as starting values for each iterative estimator in each dataset. The results showed no significant differences in the number of iterations (and CPU-time), but then, the number of iterations is not large for each iterative estimator on beforehand. For instance, for the WLS estimator (which is the most costly in terms of CPU-time) we have the following results:

Dataset:	(noniter. ULS as start)		(identity matrices as start)	
	no. of iterations	CPU–time	no. of iterations	CPU–time
1	10+3=13	46+1=47	13	59
2	12+4=16	763+2=765	12	755
3	5+2=7	35+2=37	7	49
4	9+4=13	22576	15	37522

For iterative WLS it is therefore not uniformly true that it is worthwhile, in terms of reduction of computing time, to use the noniterative ULS estimates as starting values for the iterative process. For the iterative ULS and ML estimators it is worth the effort to do this, although for most datasets this made no real difference. One important result in this respect however is that for both approaches (i.e., noniterative ULS estimates or identity matrices as starting values for the iterative processes of ULS, ML, and WLS) we achieve the same estimates $\hat{\Sigma}_1$ to $\hat{\Sigma}_k$ for all four datasets. This shows that our estimates are truly the optima for each estimator. We will now discuss the results in Tables 7 to 10 separately.

For dataset 1 (see Table 7) it shows that the final noniterative and iterative ULS estimates congruent in close fashion. Both show final estimates $\hat{\Sigma}_i$ that are close to the original input Σ_i which the measures $\|\Sigma_i - \hat{\Sigma}_i\|^2$ show. The WLS estimates resemble the ML estimates. We expect a value for $\hat{\gamma}$ close to 3 for the four estimators. Both ML and WLS estimates of γ do not differ significantly from 3, at a significance level of $\alpha = 0.01$, when N is approximately 120. The standard errors for all four estimators are not large if N is large enough. The standard errors of the ML and WLS estimates $\hat{\gamma}$ are the lowest which indicates the asymptotic efficiency of these estimators to their ULS counterparts.

Table 6.10.7. Simulation 1: $k = 3$, $n_1 = n_2 = n_3 = 2$.

Σ_1	$\hat{\Sigma}_1$(iter. ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iter. ULS)	$\hat{\Sigma}_1$(iter. WLS)
11 2	1.0000	1.0000	1.0000	1.0000
21 1	0.4982 (0.4450)	0.5231 (0.7340)	0.5229 (0.7340)	0.5031 (0.4390)
22 2	1.0403 (0.9077)	0.9918 (1.4620)	0.9914 (1.4616)	1.0240 (0.8885)
$\|\Sigma_1-\hat{\Sigma}_1\|^2$	$1.63*10^{-3}$	$1.13*10^{-3}$	$1.12*10^{-3}$	$5.94*10^{-4}$
Σ_2	$\hat{\Sigma}_2$(iter. ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iter. ULS)	$\hat{\Sigma}_2$(iter. WLS)
11 3	1.0000	1.0000	1.0000	1.0000
21 2	0.6551 (0.3751)	0.6672 (0.6170)	0.6673 (0.6169)	0.6673 (0.3677)
22 3	0.9921 (0.7472)	1.0049 (1.2370)	1.0048 (1.2367)	0.9860 (0.7302)
$\|\Sigma_2-\hat{\Sigma}_2\|^2$	$3.31*10^{-4}$	$2.51*10^{-5}$	$2.39*10^{-5}$	$1.97*10^{-4}$
Σ_3	$\hat{\Sigma}_3$(iter. ML)	$\hat{\Sigma}_3$(noniter. ULS)	$\hat{\Sigma}_3$(iter. ULS)	$\hat{\Sigma}_3$(iter. WLS)
11 0.50	1.0000	1.0000	1.0000	1.0000
21 0.25	0.4445 (0.4475)	0.4852 (0.7650)	0.4852 (0.7648)	0.4509 (0.4447)
22 0.50	0.9986 (0.8944)	0.9891 (1.5217)	0.9888 (1.5211)	0.9942 (0.8868)
$\|\Sigma_3-\hat{\Sigma}_3\|^2$	$6.17*10^{-3}$	$5.54*10^{-4}$	$5.63*10^{-4}$	$4.86*10^{-3}$
$\hat{\gamma}$	2.8865 (0.5000)	3.0166 (1.1220)	3.0179 (1.1220)	2.8607 (0.5000)
ULS–crit	1.1686	0.2692	0.2692	1.1877
WLS–crit	0.2219	0.2250	0.2250	0.1953
ML–crit	12.3620	12.3972	12.3973	12.3760
WMLS–crit	0.1820	0.2930	0.2932	0.2386
iterations	6	3	4	10
CPU–time	4	1	2	46

For dataset 2 (Table 8) it shows that the final estimates for all four estimators are close to the original input. The relative small disturbance of elements of S has no significant effect for the difference between input Σ_i and estimate $\hat{\Sigma}_i$ for each estimator as is shown by the values of the measure $\|\Sigma_i - \hat{\Sigma}_i\|^2$. We expect a value for $\hat{\gamma}$ close to 1 for the estimators. The WLS estimate of $\hat{\gamma}$ however is approximately 0.84, which differs significant from 1 at a significance level of α=0.01 if one has drawn a sample of $N = 129$. For a choice of $N = 129$, several covariance estimates for all four estimators are also not significant different from 0 at a significance level of $\alpha = 0.1$. The critical absolute t–value is then 1.658 for 120 degrees of freedom.

The standard errors are again small enough if N is large enough, and the asymptotic efficiency of the ML estimator is not contradicted, although several standard errors of the WLS estimates are lower. The large value of 763 seconds computing time for WLS estimation is due to the larger number of iterations and the laborious task of computing a $\ell_i \times \ell_i$ matrix $\left[X_i'(S_i \pi S_i)^{-1} X_i\right]^{-1}$ at each iterationstep, in contrast to a scalar $(\ell_i)^{-1}$ or $(\sigma_i' \sigma_i)^{-1}$ for ML or WLS estimation, respectively. In this case, at each iteration a newly computed 4×4 matrix $\left[X_i'(S_i \pi S_i)^{-1} X_i\right]^{-1}$ requires $O((\ell^i)^4)$ operations.

Notice that the values of the criteria are the true local optima for each corresponding estimator, which also holds in the previous experiment. For instance, the value of the ML criterion is naturally the lowest for the ML estimator, the value of the WLS criterion is the lowest for the WLS estimates, and so forth.

The result for dataset 3 (Table 9) is such that for $N \approx 100$, we would have nice small standard errors for all parameter estimates. The values of the disturbances ε_{ij} to the elements of S are relatively small, relative to the values of the original input for Σ_1 and Σ_2 which would lead us to hypothesize that $\hat{\gamma}$ should come close to 110. The results show this is true for the ULS estimates but not for the WLS and ML estimates, which overestimate γ slightly.

Table 6.10.8. Simulation 2: $k = 4$, $n_1 = n_2 = n_3 = n_4 = 2$.

	Σ_1	$\hat{\Sigma}_1$(iter. ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iter. ULS)	$\hat{\Sigma}_1$(iter. WLS)
11	1.0414	1.0000	1.0000	1.0000	1.0000
21	0.0213	0.0271 (0.3515)	0.0323 (0.3684)	0.0270 (0.3690)	0.0201 (0.3539)
22	1.0000	0.9890 (0.6991)	0.9696 (0.7255)	0.9716 (0.7274)	1.0025 (0.7087)
$\|\Sigma_1-\hat{\Sigma}_1\|^2$		$9.17*10^{-4}$	$3.70*10^{-4}$	$2.17*10^{-4}$	$5.08*10^{-3}$
	Σ_2	$\hat{\Sigma}_2$(iter. ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iter. ULS)	$\hat{\Sigma}_2$(iter. WLS)
11	0.9792	1.0000	1.0000	1.0000	1.0000
21	0.1135	0.1538 (0.3588)	0.1489 (0.3665)	0.1524 (0.3656)	0.1873 (0.3641)
22	1.0000	1.0536 (0.7366)	1.0553 (0.7530)	1.0554 (0.7513)	1.0954 (0.7621)
$\|\Sigma_2-\hat{\Sigma}_2\|^2$		$3.92*10^{-3}$	$3.34*10^{-3}$	$3.83*10^{-3}$	$1.57*10^{-2}$
	Σ_3	$\hat{\Sigma}_3$(iter. ML)	$\hat{\Sigma}_3$(noniter. ULS)	$\hat{\Sigma}_3$(iter. ULS)	$\hat{\Sigma}_3$(iter. WLS)
11	1.0835	1.0000	1.0000	1.0000	1.0000
21	0.0682	0.0445 (0.3414)	0.0493 (0.3648)	0.0364 (0.3656)	0.0295 (0.3413)
22	1.0000	0.9345 (0.6601)	0.9552 (0.7130)	0.9577 (0.7155)	0.9330 (0.6594)
$\|\Sigma_3-\hat{\Sigma}_3\|^2$		$8.14*10^{-4}$	$1.98*10^{-3}$	$2.62*10^{-3}$	$2.35*10^{-3}$
	Σ_4	$\hat{\Sigma}_4$(iter. ML)	$\hat{\Sigma}_4$(noniter. ULS)	$\hat{\Sigma}_4$(iter. ULS)	$\hat{\Sigma}_4$(iter. WLS)
11	0.9907	1.0000	1.0000	1.0000	1.0000
21	0.0802	0.0699 (0.3461)	0.0792 (0.3666)	0.0765 (0.3643)	0.0743 (0.3309)
22	1.0000	0.9631 (0.6793)	0.9847 (0.7276)	0.9721 (0.7184)	0.8817 (0.6215)
$\|\Sigma_4-\hat{\Sigma}_4\|^2$		$2.93*10^{-3}$	$6.16*10^{-4}$	$1.43*10^{-3}$	$1.64*10^{-2}$
$\hat{\gamma}$		1.0522 (0.3536)	1.0389 (0.3753)	1.0432 (0.3752)	0.8395 (0.3536)
ULS–crit		1.6687	1.6642	1.6623	2.4987
WLS–crit		3.1535	3.2201	3.1858	1.9070
ML–crit		16.0565	16.0604	16.0595	16.6581
WMLS–crit		1.7007	1.6986	1.7047	4.4076
iterations		5	4	6	12
CPU–time		14	2	4	763

Table 6.10.9. Simulation 3: $k = 2$, $n_1 = n_2 = 3$.

	Σ_1	$\hat{\Sigma}_1$(iter. ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iter. ULS)	$\hat{\Sigma}_1$(iter. WLS)
11	10	1.0000	1.0000	1.0000	1.0000
21	−9	−0.8998 (0.2500)	−0.8994 (0.4323)	−0.8994 (0.4323)	−0.8999 (0.2496)
31	5	0.4989 (0.4993)	0.4997 (0.8561)	0.4997 (0.8561)	0.4991 (0.4991)
22	10	0.9971 (0.4993)	1.0002 (0.8647)	1.0002 (0.8647)	0.9968 (0.4984)
32	−5	−0.4955 (0.5294)	−0.4998 (0.9118)	−0.4998 (0.9118)	−0.4955 (0.5290)
33	10	0.9996 (0.9970)	0.9993 (1.7116)	0.9993 (1.7116)	0.9963 (0.9963)
$\|\Sigma_1 - \hat{\Sigma}_1\|^2$		$6.21 * 10^{-5}$	$1.43 * 10^{-6}$	$1.43 * 10^{-6}$	$6.61 * 10^{-5}$
	Σ_2	$\hat{\Sigma}_2$(iter. ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iter. ULS)	$\hat{\Sigma}_2$(iter. WLS)
11	11	1.0000	1.0000	1.0000	1.0000
21	10	0.9107 (0.2374)	0.9086 (0.3896)	0.9094 (0.3896)	0.9109 (0.2369)
31	5	0.4537 (0.2898)	0.4541 (0.4653)	0.4541 (0.4653)	0.4539 (0.2895)
22	11	0.9984 (0.4744)	0.9996 (0.7791)	0.9996 (0.7791)	0.9981 (0.4734)
32	5	0.4529 (0.3286)	0.4538 (0.5308)	0.4538 (0.5308)	0.4531 (0.3282)
33	5	0.4578 (0.3922)	0.4543 (0.6272)	0.4543 (0.6272)	0.4574 (0.3915)
$\|\Sigma_2 - \hat{\Sigma}_2\|^2$		$2.50 * 10^{-5}$	$2.24 * 10^{-6}$	$2.24 * 10^{-6}$	$2.31 * 10^{-5}$
$\hat{\gamma}$		110.7450(0.4714)	110.0970 (1.3059)	110.0970(1.3059)	110.8500 (0.4714)
ULS–crit		10.9911	0.2743	0.2743	14.1206
WLS–crit		0.0113	0.0173	0.0173	0.0112
ML–crit		35.8441	35.8464	35.8464	35.8441
WMLS–crit		0.0180	0.0149	0.0149	0.0110
iterations		3	2	2	5
CPU–time		4	2	1	35

Dataset 4 (Table 10) shows that a choice of a large value for k and one or several large ℓ_i's will lead to explosive high cost of computing time for the WLS estimates. The iterative ULS estimates $\hat{\Sigma}_3$ and $\hat{\Sigma}_4$ are exactly the same as the corresponding noniterative ULS estimates for the obvious reason that the ULS iterative process is stopped after iterationstep 2 as convergence was achieved. Again, the standard errors of the ML estimates have nice figures and indicate significance of the parameter estimates to their corresponding hypothesized values.

Table 6.10.10. Simulation 4: $k = 4$, $n_1 = n_2 = 2$, $n_3 = 3$, $n_4 = 4$.

	Σ_1	$\hat{\Sigma}_1$(iter. ML)	$\hat{\Sigma}_1$(noniter. ULS)	$\hat{\Sigma}_1$(iter. ULS)	$\hat{\Sigma}_1$(iter. WLS)
11	2.4	1.0000	1.0000	1.0000	1.0000
21	1.4	0.5838 (0.1640)	0.5838 (0.3850)	0.5838 (0.3850)	0.5945 (0.1623)
22	2.4	0.9866 (0.3259)	1.0011 (0.7704)	1.0011 (0.7704)	0.9856 (0.3222)
$\|\Sigma_1-\hat{\Sigma}_1\|^2$		$1.80*10^{-4}$	$1.60*10^{-4}$	$1.61*10^{-6}$	$4.58*10^{-4}$
	Σ_2	$\hat{\Sigma}_2$(iter. ML)	$\hat{\Sigma}_2$(noniter. ULS)	$\hat{\Sigma}_2$(iter. ULS)	$\hat{\Sigma}_2$(iter. WLS)
11	2.4	1.0000	1.0000	1.0000	1.0000
21	−1.4	−0.5874 (0.1662)	−0.5836 (0.3847)	−0.5836 (0.3848)	−0.5980 (0.1640)
22	2.4	1.0076 (0.3336)	0.9999 (0.7695)	0.9999 (0.7695)	1.0030 (0.3285)
$\|\Sigma_2-\hat{\Sigma}_2\|^2$		$9.17*10^{-5}$	$1.16*10^{-7}$	$1.12*10^{-7}$	$4.38*10^{-4}$
	Σ_3	$\hat{\Sigma}_3$(iter. ML)	$\hat{\Sigma}_3$(noniter. ULS)	$\hat{\Sigma}_3$(iter. ULS)	$\hat{\Sigma}_3$(iter. WLS)
11	2	1.0000	1.0000	1.0000	1.0000
21	−1	−0.5003 (0.2162)	−0.4991 (0.4992)	−0.4991 (0.4992)	−0.5113 (0.2150)
31	0	0.0068 (0.2513)	−0.0002 (0.5763)	−0.0002 (0.5763)	0.0050 (0.2515)
22	2	0.9979 (0.4318)	0.9985 (0.9976)	0.9985 (0.9976)	1.0009 (0.4302)
32	−1	−0.5007 (0.3311)	−0.4996 (0.7620)	−0.4996 (0.7620)	−0.5105 (0.3345)
33	2	1.0105 (0.5052)	0.9987 (1.1518)	0.9987 (1.1518)	1.0123 (0.5061)
$\|\Sigma_3-\hat{\Sigma}_3\|^2$		$2.08*10^{-4}$	$6.22*10^{-6}$	$6.22*10^{-6}$	$6.77*10^{-4}$

Table 6.10.10. Continued.

	Σ_4	$\hat{\Sigma}_4$(iter. ML)	$\hat{\Sigma}_4$(noniter. ULS)	$\hat{\Sigma}_4$(iter. ULS)	$\hat{\Sigma}_4$(iter. WLS)
11	2	1.0000	1.0000	1.0000	1.0000
21	1	0.4941 (0.1946)	0.4992 (0.4544)	0.4992 (0.4544)	0.5050 (0.1923)
31	0	−0.0045 (0.2874)	−0.0010 (0.6766)	−0.0010 (0.6766)	−0.0053 (0.2873)
41	0	−0.0002 (0.2883)	0.0006 (0.6764)	0.0006 (0.6764)	−0.0022 (0.2883)
22	1.4	0.6986 (0.3253)	0.6996 (0.7602)	0.6996 (0.7602)	0.6986 (0.3214)
32	0	−0.0049 (0.2402)	−0.0002 (0.5659)	−0.0002 (0.5659)	−0.0046 (0.2402)
42	0	−0.0067 (0.2410)	0.0001 (0.5658)	0.0001 (0.5658)	−0.0059 (0.2410)
33	2	0.9911 (0.5722)	0.9984 (1.3521)	0.9984 (1.3521)	0.9906 (0.5719)
43	−1	−0.5060 (0.3826)	−0.4991 (0.8941)	−0.4991 (0.8941)	−0.5183 (0.3866)
44	2	0.9972 (0.5758)	0.9979 (1.3515)	0.9979 (1.3515)	0.9975 (0.5759)
$\|\Sigma_4-\hat{\Sigma}_4\|^2$		$4.09*10^{-4}$	$1.29*10^{-5}$	$1.29*10^{-5}$	$9.93*10^{-4}$
$\hat{\gamma}$		23.1425 (0.2041)	23.0828 (0.6282)	23.0828 (0.6282)	23.3501 (0.2041)
ULS–crit		29.0022	14.9711	14.9711	126.6674
WLS–crit		1.0313	1.0805	1.0805	0.8771
ML–crit		154.3858	154.3999	154.3999	154.4593
WMLS–crit		0.8602	0.8892	0.8892	1.1737
iterations		5	4	2	9
CPU–time		164	26	8	22550 (!)

These simulation experiments show that the iterative ML and ULS algorithms compare favorably to the WLS algorithm, and the noniterative ULS approach may be preferred to the iterative ULS approach if one stresses computing time. The ML approach is the most favorable technique if asymptotic normality is stressed as assumption, because the asymptotic efficiency of the ML estimator is a property to consider if the analyst strives for small standard errors of estimates.

6.11. CONCLUSION

The analysis of covariance structures constitutes an important type of multivariate methods. To an increasing extent, this type of analysis is brought to bear upon data of multiple dimension. Such 'multimode' data may come from a multitrait–multimethod (MTMM) context, or from repeated measurements at several occasions, or both combined. One may agree with Cudeck (1988, p. 141) that "(i)t is possible that one reason for the paucity of such data is the relative scarcity of methods for describing them." The availability of multimode data analysis methods hence becomes more and more relevant. The present chapter intends to add to such methods by considering a general structure for multimode covariance matrices that is multiplicative. In particular, if Ω is a covariance matrix, we consider estimation in the model (2.2) for any value of k, where $\Sigma_1,\ldots,\Sigma_k$ are symmetric positive definite matrices to be determined. We call this a 'factorial covariance structure'. In practical cases, $k = 2$ or $k = 3$ will be most relevant, but we prefer to deal with the general case since it involves, once a proper notation having been introduced, no additional complications and indeed makes things more transparent over a treatment for $k = 2$ or $k = 3$ separately.

The motivation for considering the k–mode covariance structure is that in a number of contexts multiplicative effects may be a valid description of relationships, especially in the MTMM context. See e.g., Campbell and O'Connell (1967), Browne (1984a), and Cudeck (1988). When there are repeated measurements, and the occasions also interact multiplicatively, model (2.2) with $k = 3$ may be a valid description. Such a structure is discussed by Cudeck (1988), the difference being that he does not impose this structure on the *covariance* matrix of the *observed* scores, but on the *correlation* matrix of the *true* scores. Whatever the difference, we feel that it may be useful to consider the 'benchmark' model (2.2) first and postpone additional structure to later work. Model (2.2) extends work by Swain (1980), who considered $k = 2$, to an arbitrary number of modes. Another generalization is that, besides the ML estimation method employed by Swain (1980), we also discuss the WLS and ULS approach. It appears that estimation along the three lines (ML, WLS, and ULS) can be integrated to a certain extent. The general expression (5.2) for the three different estimators was derived from a definition of an overlapping generalized least squares criterion $\mathrm{tr}\{(S-\Omega)W^{-1}\}^2$, where weight matrix W is an appropriate weight matrix for each estimator. Since expression (5.2) can be

envisioned as a solution to the normal equations of a slightly modified version of the classical regression model, several asymptotic statistical properties of the estimators are fairly easy to deduce. Firstly, ML corresponds to the regression with the true weight matrix, WLS uses a consistent estimator, and ULS ignores the weight matrix altogether. Secondly, estimates according to all four estimation techniques are asymptotically normal distributed and ML is the asymptotically efficient estimator. Another finding is that, despite the non-linear structure at hand, there is a modification of the ULS estimator that is non-iterative in the sense that it is the solution of an eigenvalue equation. So there is a 'good' solution (i.e., a consistent estimate if (2.2) would be the true model) that can be computed in a straightforward way.

A computer program which includes the four alternative estimators described in the previous sections, was applied to the various 2-mode and 3-mode datasets from the literature (Swain (1980), Campbell and Fiske (1959), and Cudeck (1988)). In addition, the program was applied to various test runs for artificial data sets with number of modes k larger than three. A diskette containing the estimators and their variances is available upon request from the author.

The empirical applications for $k = 2$ and $k = 3$, together with our simulation experiments for different sizes of k, illustrate that the k-mode CSA model can give significant parameter estimates. The simulation experiments show the asymptotic efficiency property of the ML and WLS estimators. The hypothesized k-mode covariance structure (2.2) may be a worthwhile case to consider for empirical data if one puzzles in the world of the multimode extension of covariance analysis.

We note however that the direct product model (2.2) is not the only choice of interest. There are several extensions of the direct product model that are worth consideration. Such extensions are, for instance, the composite direct product models (cf. Browne (1984a)) with and without additional restrictions to their form. Rather than assessing a single model, a cross-validation procedure can be used to compare competing models (Cudeck and Browne (1983)).

The involvement of several large but sparse matrices in factorial covariance structure analysis to derive the separate estimators, alerts the

programmer of such noniterative and iterative algorithms to build the necessary operands at a scalar level. Constructing the matrices $\left[X_i'(S_i \pi S_i)^{-1} X_i\right]^{-1}$ at 'face value' would lead to large memory–assessment of computers' memory, which can be avoided by spending more attention to the operative work of (0,1)–matrices P, C_i, and B_i, before actually storing these matrices. The necessary programming modules for k–mode CSA were discussed in a separate section (9).

Unfortunately, the computation of a new WLS estimate still involves $O((\ell^i)^4)$ operations per iteration, which means that for several choices of values for k and ℓ_1 to ℓ_k, computing time rises rapidly. The 'Full Information Maximum Likelihood' property and asymptotic efficiency of the iterative ML estimator and the consistency of the ULS estimators strengthen us in our conclusion to use the ML or ULS approach in future practice.

As to the practical usefulness of the methods introduced in this chapter we remark the following. At present we are unaware of a substantial context where Ω exactly as in (2.2) represents a meaningful 'true' model. However, as indicated in section 1, there is a growing attention for Kronecker multiplicative structures, most notably for MTMM, and our approach (especially the noniterative ULS estimator) is useful for providing solutions in an ALS approach of the model $\bar{\Omega} \equiv \Omega + D$, D diagonal, Ω being the covariance of the latent underlying factors. An adaptation of (2.2) to correlation analysis is straightforward although not fully elaborated here. Also, our extension for dealing with an arbitrary number of modes seems useful in the light of the rapidly growing attention in psychometrics to multimode covariance structure analysis.

7. k-MODE INTERDEPENDENT REGRESSION

7.1. INTRODUCTION

In this chapter we focus on what is termed the *k-mode* or *multimode Interdependent Regression* (MIR) model. Simply put, the model states that the parameters in the model are dependent upon a smaller number of unknown parameters in a multiplicative manner. The model is stated by Wansbeek and Buyze (1981). They give as an example of when the model may be applied, a particular form of Kapteyn's (1977) income satisfaction model with preference interdependence. The example essentially boils down to the following.

Let y be a $N\times 1$ vector containing values of a measure on income satisfaction of N individuals and let the $N\times 1$ vector x contain their respective incomes. Let k denote the number of characteristics of individuals and, and let ℓ_i be the number of categories that are distinguished for the i-th characteristic. Then

$$\ell \equiv \prod_{i=1}^{k} \ell_i \tag{7.1.1}$$

is the number of types of individuals we distinguish, where a 'type' is defined by any possible combination of categories of the k characteristics. Each of the N individuals belongs to a certain type. Whether the case of 'balanced' data, in which case the number m of individuals per type is equal, i.e., $m = N/\ell$, or 'unbalanced' data is at hand is irrelevant for now. Both cases will be discussed later on. Now, it is stated that the income satisfaction of a certain individual is determined by the income of all other individuals, including himself, but in such a way that the impact of the income of the individuals of the same type is the same, and in reverse, that the income satisfaction of two individuals of the same type is influenced in the same way by all incomes in the set.

The parameters in the model are indicators of these impacts, i.e., these weights between any two types. Let these parameters be summarized in the parameter matrix B. The interdependent regression assumption then states that these 'weights between types' are products of k separate 'weights between categories'. The weights between categories are weights that apply between any two categories of a characteristic. For instance, for each individual it is known to which age group he belongs, what his educational level is and what his place of residence is. Assume there are three (ℓ_1=3) age classes, four (ℓ_2=4) educational levels and five (ℓ_3=5) categories of residence, in which case there are 60 types of individuals. Let G_1 (3×3) be the matrix of weights that apply between the three age categories, let G_2 (4×4) be the matrix of weights that apply between the four educational categories and G_3 (5×5) be defined likewise. Then the assumption of multiplicative interdependence translates into the formal specification

$$B = G_1 \otimes G_2 \otimes G_3. \qquad (7.1.2)$$

In this chapter we will discuss several versions (balanced, unbalanced, additional regressors) of a model for multiplicative interdependence in regression contexts. The formal model specification of Wansbeek and Buyze (1981) is discussed in section 2. Section 3 is concerned with the issue of estimation. Sections 4 and 5 are devoted to first– and second–order derivatives to obtain optima, (i.e., parameter estimators which tend to optima of the estimation criteria functions), and to subsequent identification considerations, respectively. The first– and second–order derivatives are obtained by using differentials and the so–called first and second identification theorems (Magnus and Neudecker (1988)). Section 6 is devoted to several iterative solving techniques for obtaining parameter estimates. Sections 7 and 8 are devoted to asymptotic properties of the estimators. Section 9 is devoted to the final computational aspects for obtaining ML estimates. In section 10, results for several simulated data are discussed. Section 11 concludes.

7.2. MODEL SPECIFICATION

The k–mode interdependent regression model of Wansbeek and Buyze (1981) is formalized as

$$y = TBT'x + \varepsilon, \tag{7.2.1}$$

with

$$B \equiv G_1 \otimes G_2 \otimes \ldots \otimes G_k \tag{7.2.2}$$

where y and x are known $N\times 1$ vectors, and ε is an $N\times 1$ vector of independently distributed error terms with zero expectation and known covariance matrix W. Matrix T is a known $N\times\ell$ matrix with a single one in each row and zeroes elsewhere, indicating to which type a certain object in the vector y belongs. The $\ell\times\ell$ matrix B contains the unknown weights that apply between any two types, and is, through relation (2), dependent upon a smaller number of unknown parameters. The smaller number of unknown parameters are contained in parameter matrices G_1 to G_k. Matrix G_i is of order $\ell_i\times\ell_i$, $i = 1,\ldots,k$. Furthermore we define

$$B^i \equiv G_1 \otimes \ldots \otimes G_{i-1} \otimes G_{i+1} \otimes \ldots \otimes G_k, \tag{7.2.3}$$

$$\text{vec } Y_i' \equiv C_i y, \tag{7.2.4}$$

$$\text{vec } X_i' \equiv C_i x, \tag{7.2.5}$$

$$\text{vec } \mathcal{E}_i' \equiv C_i \varepsilon. \tag{7.2.6}$$

So in a sense of multidimensional data analysis we again have a model to be interpreted as a way to describe the connection between two vectors y and x which are 'stacked' vec–representations of k–dimensional data arrays Y and X, respectively.

7.2.1. Balanced data

The case where there is an equal number (m) of objects per type (i.e., $m = N/\ell$) is called the 'balanced data' case. We may arrange y and x such that

(7.2.7) $\quad T = I_\ell \otimes \iota_m,$

with ι_m an $m\times 1$ vector of ones. Define $\overline{y}$ and $\overline{x}$ as the $\ell\times 1$ vectors of type–totals of the objects in y and x.

(7.2.8) $\quad \overline{y} = (I_\ell \otimes \iota_m')y,$

(7.2.9) $\quad \overline{x} = (I_\ell \otimes \iota_m')x,$

in which case (1) can be written as

(7.2.10) $\quad y = TBT'x + \varepsilon = (I_\ell \otimes \iota_m)B(I_\ell \otimes \iota_m')x + \varepsilon = (B \otimes \iota_m)\overline{x} + \varepsilon.$

7.2.2. Unbalanced data

In case of an unequal number of objects per type, matrix T has columns with an unequal number of ones, in which case it is impossible to get the neat pattern for T as in the case of balanced data. In the case of unbalanced data the matrix $T'T$ will therefore not be of the form $m.I_\ell$, but a diagonal $\ell\times\ell$ matrix with diagonal elements that indicate the number of objects per type, i.e.,

(7.2.11) $\quad \Delta \equiv T'T.$

Also define $\overline{y}$ and $\overline{x}$, in this case as

(7.2.12) $\quad \overline{y} = T'y,$

(7.2.13) $\quad \overline{x} = T'x.$

7.2.3. Additional regressors

As a further generalization to the unbalanced (and balanced) data case, we introduce additional regressors in the interdependent regression model (1). The general format then becomes:

(7.2.14) $\quad y = TBT'x + Z\beta + \epsilon,$

with an $N \times p$ matrix Z of N observations on p known regressors, and with β as a $p \times 1$ vector of parameters to be estimated in addition to the elements of G_1 to G_k. We collect the parameters in a vector θ consisting of the non-restricted elements of β and G_1 to G_k.

7.3. ESTIMATION APPROACH

The WLS estimators of the parameters in model (2.14) are determined by minimizing

(7.3.1) $\quad Q(\theta) \equiv (y - TBT'x - Z\beta)'W^{-1}(y - TBT'x - Z\beta)$

$$\equiv f'W^{-1}f$$

with f implicitly defined. Here W is the covariance matrix of the error terms. In case $W = \sigma^2.I$, expression (1) for Q is an OLS criterion. In case the error terms are normally distributed with mean vector zero and covariance matrix W; the ML estimators may also be obtained by evaluating (1). To solve for the parameter estimators of β and G_1 to G_k, first-order derivatives of (1) will need to be obtained. The next section gives an elaborate derivation of these derivatives, together with the derivation of second-order derivatives. We use differentials and the so-called first and second identification theorems (Magnus and Neudecker (1988)).

7.4. FIRST- AND SECOND-ORDER DERIVATIVES

First of all, we define $\overline{Y}_i$ and $\overline{X}_i$ according to:

$$\text{(7.4.1)} \qquad \text{vec } \overline{Y}_i' \equiv C_i\overline{y},$$

$$\text{(7.4.2)} \qquad \text{vec } \overline{X}_i' \equiv C_i\overline{x},$$

cf. (2.4) and (2.5). We then get the first differential

$$\begin{aligned}
\text{(7.4.3)} \qquad \mathrm{d}Q &= 2f'W^{-1}\mathrm{d}(y-TBT'x-Z\beta) \\
&= 2f'W^{-1}(T(\mathrm{d}B)T'x+Z\mathrm{d}\beta) \\
&= -2f'W^{-1}(\sum_{i=1}^{k} TC_i'(B^i\otimes \mathrm{d}G_i)C_iT'x+Z\mathrm{d}\beta) \\
&= -2f'W^{-1}(\sum_{i=1}^{k} (x'TC_i'\otimes TC_i')\mathrm{vec}(B^i\otimes \mathrm{d}G_i)+Z\mathrm{d}\beta) \\
&= -2f'W^{-1}(\sum_{i=1}^{k} (x'TC_i'\otimes TC_i')(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\mathrm{vec}\ B^i\otimes \mathrm{vec}\ \mathrm{d}G_i)+Z\mathrm{d}\beta) \\
&= -2f'W^{-1}(\sum_{i=1}^{k} (x'TC_i'\otimes TC_i')(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\mathrm{vec}\ B^i\otimes I_{\ell_i^2})\mathrm{vec}\ \mathrm{d}G_i+Z\mathrm{d}\beta).
\end{aligned}$$

7.4.1. First-order derivatives

Then, due to the first identification theorem, the first-order derivatives are

$$\begin{aligned}
\text{(7.4.4)} \qquad \frac{\partial Q}{\partial(\mathrm{vec}\ G_i)'} &= -2f'W^{-1}(C_i\overline{x}\otimes C_iT')'(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\mathrm{vec}\ B^i\otimes I_{\ell_i^2}) \\
&= -2f'W^{-1}(\overline{x}\otimes T')'(C_i\otimes C_i)'(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\mathrm{vec}\ B^i\otimes I_{\ell_i^2}) \\
&= -2f'W^{-1}M_i,
\end{aligned}$$

with M_i implicitly defined, and

$$(7.4.5)\qquad \frac{\partial Q}{\partial \beta'} = -2f'W^{-1}Z.$$

Equating both expressions to zero to obtain a minimum of Q leads to the following expressions:

$$(7.4.6)\qquad \text{vec } \hat{G}_i = (M_i'W^{-1}M_i)^{-1}M_i'W^{-1}(y-Z\hat{\beta}),$$

$$(7.4.7)\qquad \hat{\beta} = (Z'W^{-1}Z)^{-1}Z'W^{-1}(y-TB\bar{x}).$$

In case of OLS, $W = \sigma^2.I$ leads to

$$\begin{aligned}(7.4.8)\qquad M_i'M_i &= ((\text{vec } B^i)'\otimes I_{\ell_i^2})(I_{\ell^i}\otimes P_{\ell_i,\ell^i}\otimes I_{\ell_i})(C_i\otimes C_i)(\bar{x}\otimes T')(\bar{x}\otimes T')'(C_i\otimes C_i)'\\ &\quad (I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\text{vec } B^i\otimes I_{\ell_i^2})\\ &= (\text{vec } B^i)'\otimes I_{\ell_i^2})(I_{\ell^i}\otimes P_{\ell_i,\ell^i}\otimes I_{\ell_i})(C_i\bar{x}\bar{x}'C_i'\otimes C_i\Delta C_i')\\ &\quad (I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\text{vec } B^i\otimes I_{\ell_i^2})\\ &= ((\text{vec } B^i)'\otimes I_{\ell_i^2})(I_{\ell^i}\otimes P_{\ell_i,\ell^i}\otimes I_{\ell_i})(\text{vec } \bar{X}_i'(\text{vec } \bar{X}_i')'\otimes C_i\Delta C_i')\\ &\quad (I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\text{vec } B^i\otimes I_{\ell_i^2})\\ &= ((\text{vec } B^i)'\otimes I_{\ell_i^2})(\text{vec } \bar{X}_i'(\text{vec } \bar{X}_i')' \pi C_i\Delta C_i')(\text{vec } B^i\otimes I_{\ell_i^2})\end{aligned}$$

according to (3.6.6). If, momentarily, both $\ell\times\ell$ matrices vec $\bar{X}_i'(\text{vec } \bar{X}_i')'$ and $C_i\Delta C_i'$ are considered (viewed) as partitioned matrices with ℓ^{i^2} blocks, each of order $\ell_i\times\ell_i$, then we easily see that

$$\begin{aligned}(7.4.9)\qquad M_i'M_i &= ((\text{vec } B^i)'\otimes I_{\ell_i^2})(\text{vec } \bar{X}_i' \pi C_i\Delta^{1/2}C_i')((\text{vec } \bar{X}_i')' \pi C_i\Delta^{1/2}C_i')\\ &\quad (\text{vec } B^i\otimes I_{\ell_i^2})\\ &= (\bar{X}_i'B^{i'}\otimes I_{\ell_i})C_i\Delta^{1/2}C_i'C_i\Delta^{1/2}C_i'(B^i\bar{X}_i\otimes I_{\ell_i})\\ &= (\bar{X}_i'B^{i'}\otimes I_{\ell_i})C_i\Delta C_i'(B^i\bar{X}_i\otimes I_{\ell_i}).\end{aligned}$$

In case of balanced data, i.e., $\Delta = mI_\ell$, $M_i'M_i$ reduces to the simple form

$$M_i'M_i = m(\bar{X}_i'B^{i\prime}B^i\bar{X}_i \otimes I_{\ell_i}). \tag{7.4.10}$$

7.4.2. Second-order derivatives

Since $d^2G_i = d^2\beta = 0$, the second differential d^2Q is:

$$\begin{aligned} d^2Q = {} & -2f'W^{-1}\sum_{i=1}^{k}(x'TC_i'\otimes TC_i')(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\text{vec } dB^i\otimes I_{\ell_i^2})\text{vec } dG_i + \\ & 2(x'T(dB)T'+(d\beta)'Z')W^{-1}M_i\text{vec } dG_i + \\ & 2(x'T(dB)T'+(d\beta)'Z')W^{-1}Zd\beta \\ = {} & -2f'W^{-1}\sum_{i=1}^{k}(x'TC_i'\otimes TC_i')(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i}) \\ & ((C_j^i\otimes C_j^i)'(I_{\ell^{ij}}\otimes P_{\ell^{ij},\ell_j}\otimes I_{\ell_j})(\text{vec } B^{ij}\otimes\text{vec } dG_j)\otimes I_{\ell_i^2})\text{vec } dG_i + \\ & 2(x'T(dB)T'+(d\beta)'Z')W^{-1}M_i\text{vec } dG_i + \\ & 2(x'T(dB)T'+(d\beta)'Z')W^{-1}Zd\beta \\ = {} & -2f'W^{-1}\sum_{i=1}^{k}(x'TC_i'\otimes TC_i')(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i}) \\ & ((C_j^i\otimes C_j^i)'(I_{\ell^{ij}}\otimes P_{\ell^{ij},\ell_j}\otimes I_{\ell_j})(\text{vec } B^{ij}\otimes I_{\ell_j^2})\otimes I_{\ell_i^2}) \\ & (\text{vec } dG_j\otimes\text{vec } dG_i) + 2\sum_{i=1}^{k}(\text{vec } dG_i)'M_i'W^{-1}M_i\text{vec } dG_i + \\ & 2\sum_{j\neq i}^{k}(\text{vec } dG_j)'M_j'W^{-1}M_i\text{vec } dG_i + 2\sum_{i=1}^{k}(d\beta)'Z'W^{-1}M_i\text{vec } dG_i + \\ & 2\sum_{i=1}^{k}(\text{vec } dG_i)'M_i'W^{-1}Zd\beta + 2(d\beta)'Z'W^{-1}Zd\beta. \end{aligned} \tag{7.4.11}$$

Here, $\ell^{ij}\times\ell^{ij}$ matrix B^{ij} results from B by deleting matrices G_i and G_j from the Kronecker product string (2.2). Matrix C_j^i, of order $\ell^i\times\ell^i$, is of the same

structure as C_j, but dimension ℓ_i is deleted (cf. (3.5.11)).

Due to the second identification theorem, we get from (11)

$$(7.4.12) \qquad \frac{\partial^2 Q}{\partial \operatorname{vec} G_i \; \partial (\operatorname{vec} G_i)'} = 2M_i' W^{-1} M_i,$$

$$(7.4.13) \qquad \frac{\partial^2 Q}{\partial \operatorname{vec} G_i \; \partial (\operatorname{vec} G_j)'} = 2M_i' W^{-1} M_j - R,$$

where R is not further specified (but with $E[R] = 0$),

$$(7.4.14) \qquad \frac{\partial^2 Q}{\partial \operatorname{vec} G_i \; \partial \beta'} = 2M_i' W^{-1} Z,$$

and

$$(7.4.15) \qquad \frac{\partial^2 Q}{\partial \beta \; \partial \beta'} = 2Z' W^{-1} Z.$$

7.5. IDENTIFICATION

The next step is to derive the information matrix $I(\theta)$ which is necessary to obtain for three reasons. The information matrix will be necessary for implementation of a second-order iterative algorithm, it contains the approximate standard errors of the estimators, and it is useful in order to investigate the identification of the parameters. The information matrix derives easily from (4.12) to (4.15), and is of structure

$$(7.5.1) \qquad I(\theta) = 2\begin{bmatrix} M'W^{-1}M & M'W^{-1}Z \\ Z'W^{-1}M & Z'W^{-1}Z \end{bmatrix} = 2\begin{bmatrix} M'W^{-1/2} \\ Z'W^{-1/2} \end{bmatrix} \begin{bmatrix} W^{-1/2}M, W^{-1/2}Z \end{bmatrix}$$

with $M \equiv (M_1, M_2, \ldots, M_k)$.

In case of OLS estimation ($W = \sigma^2 I$) of the parameters the simple structure in (4.10) for $M_i'M_i$ appears, and $M_i'M_j$ is of similar structure

(7.5.2) $$M_i'M_j = (\bar{X}_i'B^{i'}\otimes I_{\ell_i})C_i\Delta C_j'(B^j\bar{X}_j\otimes I_{\ell_j}).$$

If no additional regressors are present (i.e., β=0 is fixed) in the model, the information matrix reduces further to

(7.5.3) $$I(\theta) = 2\sigma^{-2}M'M \equiv 2\sigma^{-2}L'\Delta L,$$

with $L \equiv (L_1, L_2, \ldots, L_k)$ where

(7.5.4) $$L_i' \equiv (\bar{X}_i'B^{i'}\otimes I_{\ell_i})C_i.$$

We assume then Δ is of full rank, i.e., all types are represented in the sample. The rank of $I(\theta)$ then is equal to the row-rank of L'. Let $\ell_i\times\ell_i$ matrix E_i, for $i = 1,\ldots,k$, be a matrix with arbitrary elements, and define

(7.5.5) $$e \equiv ((\text{vec } E_1)', \ldots, (\text{vec } E_k)')',$$

then L' is of incomplete row-rank if the equation $Le = 0$ admits a non-trivial solution (i.e., $e \neq 0$):

(7.5.6) $$Le = \sum_{i=1}^{k} C_i'(B^i\bar{X}_i\otimes I_{\ell_i})\text{vec } E_i =$$

$$= \sum_{i=1}^{k} C_i'\text{vec } (E_i\bar{X}_i'B^{i'}) = \sum_{i=1}^{k} C_i'(B^i\otimes E_i)\text{vec } \bar{X}_i'$$

$$= \sum_{i=1}^{k} C_i'(B^i\otimes E_i)C_iC_i'\text{vec } \bar{X}_i' = \Big[\sum_{i=1}^{k}(G_1\otimes\ldots\otimes G_{i-1}\otimes E_i\otimes G_{i+1}\otimes\ldots\otimes G_k)\bar{x}\Big]$$

The expression in brackets is a square $\ell\times\ell$ matrix, which should be zero for a non-trivial solution. In general this happens only if $E_i = \alpha_iG_i$ with $\sum_{i=1}^{k}\alpha_i = 0$, since then the matrix in brackets becomes:

(7.5.7) $$\sum_{i=1}^{k}\alpha_iB = \Big(\sum_{i=1}^{k}\alpha_i\Big).B = 0.$$

There are $k-1$ degrees of freedom in the choice of α's such that $\sum_{i=1}^{k}\alpha_i = 0$, and so there exist $k-1$ linear interdependencies between the rows of L'. So a necessary rank condition for identification is satisfied if, in each of $k-1$ arbitrarily chosen parameter matrices G_1 to G_k, one elements is fixed. Of

course, one may also restrict one element of each parameter matrix G_1 to G_k to be a fixed value and include a multiplicative factor γ as extra parameter to be estimated, where each G_i is scaled by its fixed element.

In addition, the necessary order condition for identification states that the number of observations must at least equal the number of free parameters in θ, i.e., in G_1 to G_k. So,

$$(7.5.8) \qquad \ell \geq \sum_{i=1}^{k} \ell_i^2 - (k-1).$$

Notice that for $k = 2$, this order condition will never be satisfied unless more than $k-1$ restrictions are imposed. If additional regressors are present (and still $W = \sigma^2 I$), the information matrix in (1) reduces to

$$(7.5.9) \qquad I(\theta) = 2\sigma^{-2}\begin{bmatrix} M' \\ Z' \end{bmatrix}[M,Z] = 2\sigma^{-2}\begin{bmatrix} L'\Delta^{1/2} \\ Z' \end{bmatrix}[\Delta^{1/2}L,Z],$$

which means $I(\theta)$ has rank equal to the column–rank of $[\Delta^{1/2}L,Z]$. Define e_0 as an arbitrary p–vector, and replace (6) by

$$(7.5.10) \qquad [\Delta^{1/2}L,Z]\begin{pmatrix} e \\ e_0 \end{pmatrix} = \left(\sum_{i=1}^{k} \alpha_i\right)\Delta^{1/2}B\bar{x} + Ze_0 = 0.$$

So $k-1$ restrictions on G_1 to G_k should be enough to solve identification problems, unless $\Delta^{1/2}B\bar{x} = Z\delta$ with $\delta \neq 0$, i.e., unless $\Delta^{1/2}B\bar{x}$ is contained in the column–space of Z. By the symbol $I_r(\theta)$ we indicate the 'reduced information matrix' which results from $I(\theta)$ in (1) by deleting the rows and columns that correspond to the positions of the fixed elements of the parameter vector θ.

7.6. ITERATIVE SOLVING TECHNIQUES

The equations (4.6) and (4.7) will need to be solved in an iterative manner. The reason for this is that each matrix G_i in (4.6) is linearly dependent on the parameter vector β, but dependent on G_j $(j \neq i)$ in the third–degree. Vector β in (4.7) is linear dependent on G_1 to G_k. One iterative

solution method of these equations is to solve (4.7) for β, substitute the result into (4.6), to obtain an equation in the G's only. The new estimates of G_1 to G_k are then substituted in (4.7) and the routine is repeated. This algorithm, together with several alternatives, is discussed in the next subsections.

7.6.1. ALS algorithm

Obviously we would like to construct an algorithm for the minimization of the fitted sum of squares, Q, that converges to a global minimum. Unfortunately, Q is the cross-product term of a multivariate polynomial of the $(2k)$-th degree, and in general it is not possible to prove that methods to solve such nonlinear problems attain a global optimum. In the present case this also seems to be true. We will have to be satisfied with proving that the algorithm outlined below will converge to some stationary point, which may not be a global minimum.

The method to be described utilizes the so-called Alternating Least Squares (ALS) technique. The essential feature of the ALS algorithm is that in solving optimization problems with more than one set of parameters, each set is estimated in turn by applying conditional least squares procedures holding the other sets fixed. After all sets have been estimated once, the procedure is repeated again and again until convergence. From the above discussion a rough outline for an algorithm is readily deduced. First, choose an arbitrary $G_2^{(0)}$ to $G_k^{(0)}$ and maximize over G_1 to get a new estimate G_1, $G_1^{(1)}$ say. Maximize subsequently over G_2 with the just computed $G_1^{(1)}$ and $G_3^{(0)}$ to $G_k^{(0)}$ fixed to get a new estimate G_2, $G_2^{(1)}$ say, and so forth, and iterate the procedure until – one hopes – convergence. According to (4.6) and (4.7), the minimization of Q consists of an iterative process, in which at each step, k systems of linear equations have to be solved.

One would like to be able to prove that the method one has selected converges to the true minimum of the objective function. Unfortunately convergence proofs usually require that certain assumptions be made concerning the nature of the objective function, and the validity of these assumptions is difficult to verify on any given problem. Even more significantly, the existence of a convergence proof is no guarantee of reasonable performance in practice. A method may converge in theory, yet take an excessive number of

iterations, or require computations to be carried out with an unreasonable number of significant digits.

Monte Carlo simulation of the model without the additional regressors Z for $k = 3$ showed that convergence of the ALS algorithm was slowly achieved once starting values $G_1^{(0)}$ to $G_3^{(0)}$ of G_1 to G_3, respectively, were chosen close to the true values (G_1^0, G_2^0, G_3^0) which had been set beforehand. For example, we chose values for all elements of G_i^0 (i=1,2,3) in the interval [-10,10] and chose starting values of 3×3 matrices $G_i^{(0)}$ as

$$G_i^{(0)} = G_i^0 + \mathcal{E}_i, \tag{7.6.4}$$

where

$$\mathcal{E}_i = (\varepsilon_{kl}),\ \varepsilon_{kl} \text{ i.i.d. } N(0,1), \tag{7.6.5}$$

$k,l = 1,\ldots,\ell_i,\ \ell_i = 3,\ i = 1,2,3.$

In this case, convergence would be achieved after approximately 1000 iterations whereas the value of Q was approximately 7.728 at that point and lowering at an average of about $3*10^{-5}$per iteration, so in no sense anywhere close to the lower bound of zero at the optimal solution. We therefore direct our attention to iterative methods based upon second-order derivatives of Q in the next subsection.

7.6.2. Second-order derivative based methods

A solution method is an implementation of the Fisher scoring method. Define the score vector q as the $\left[\sum_{i=1}^{k} \ell_i^2 + p\right] \times 1$ vector

$$q(\theta) \equiv \left[\frac{\partial Q}{\partial (\text{vec } G_1)'}, \ldots, \frac{\partial Q}{\partial (\text{vec } G_k)'}, \frac{\partial Q}{\partial \beta'} \right]' \tag{7.6.6}$$

and let $q_r(\theta)$ denote the 'reduced score vector', i.e., the reduced vector of elements of $q(\theta)$ that correspond with identified parameters in θ. Fisher scoring then boils down to the scheme:

1. Choose (admissible approximate) starting values.

2. Compute

$$\theta_r^{(t+1)} = \theta_r^{(t)} + I_r^{-1}(\theta^{(t)})q_r(\theta^{(t)}). \tag{7.6.7}$$

3. Since $Q^{(t+1)} \leq Q^{(t)}$, check if the standards of the convergence criteria are met. If not, repeat steps 2 to 4.

If we want to use the exact second-order derivatives, instead of their expectations as in Fisher scoring, then we need the expressions (4.12) to (4.15). The Hessian matrix $H(\theta)$ of the function Q in (3.1) is the matrix of second partial derivatives, as given by (4.12) to (4.15), i.e.,

$$H_{ij}(\theta) \equiv \partial^2 Q \; / \; \partial\theta_i \partial\theta_j' \; , \tag{7.6.8}$$

where θ denotes the vector of parameters of the model that are unrestricted. Here, θ_i and θ_j denote the i-th and j-th element of θ, respectively. When Δ is of full rank, θ would be of length $\sum_i^k \ell_i^2 - (k-1)$ if no other restrictions are made other than necessary for identification. Let $H_r^{(t)}$ be the Hessian matrix of Q evaluated at the t-th iteration, i.e., at $\theta_r = \theta_r^{(t)}$. The matrix H_r results from H by taking notice of the restrictions for identification. The scheme (7) is replaced by

$$\theta_r^{(t+1)} = \theta_r^{(t)} - (H_r^{(t)})^{-1} q_r^{(t)}. \tag{7.6.9}$$

Equation (9) defines the t-th iteration of the Newton (also known as Newton-Raphson) method. The method is acceptable as long as $H_r^{(t)}$ is positive definite, as it should be at least in some neighborhood of the minimum. In this neighborhood, convergence is quadratic. This means that the number of correct digits in θ is approximately doubled by each iteration, until further improvement is barred by the rounding errors in the calculations. Outside this neighborhood, convergence cannot be guaranteed. Various methods have been devised for overcoming this difficulty, while retaining the advantage of the method. One of these is the Marquardt method for converting an arbitrary matrix into a positive definite one. The method rests on the observation that if $P_r^{(t)}$ is any positive definite matrix, then $H_r^{(t)} + \lambda_r^{(t)} P_r^{(t)}$ is positive definite for sufficiently large $\lambda_r^{(t)}$ no matter what $H_r^{(t)}$. Marquardt (1963)

suggests the choice of $P_r^{(t)}$ as a diagonal matrix whose elements coincide with the absolute values of the diagonal elements of $H_r^{(t)}$ (with, say, zero elements replaced by ones).

We have implemented the Marquardt method for our problem of least squares estimation of the multiplicative interdependent regression model without additional regressors. Before considering the artificial data in section 10, we point out that it is not enough to compute a vector $\hat{\theta}_r$ and to state that this is the estimated value of the unknown parameters θ_r. We must also investigate the sensitivity and precision of our estimates. Even more important than the question of the sensitivity of the estimate is that of the sensitivity of the model itself. This question is handled by the use of goodness–of–fit criteria. We cover these topics only briefly here in section 7 since extensive treatments can be found in the literature (e.g., Anderson (1958)). Here, we only consider the case of OLS estimation of the model with no additional regressors present. Similar results would follow for the more general case.

7.7. SENSITIVITY AND PRECISION OF ML ESTIMATES AND MODEL

Suppose our restricted (i.e., the estimator satisfies the identification restrictions) estimate $\hat{\theta}_r$, say, leads to an optimum of the objective function $Q(\theta)$ in (3.1). $Q(\theta)$ also depends on the data, in particular, it depends on the measured values y_m, say, of the random variables y. We indicate this dependence by writing $Q(\theta,y_m)$ in place of $Q(\theta)$. At the minimum we have

$$(7.7.1) \qquad \partial Q(\theta_r,y_m) \;/\; \partial\theta_r\big|_{\theta_r=\hat{\theta}_r} = 0 \;.$$

Suppose we varied the data slightly, replacing y_m by $y_m+\delta y_m$. This would cause our estimate to shift from $\hat{\theta}_r$ to $\hat{\theta}_r + \delta\hat{\theta}_r$, where we must have

$$(7.7.2) \qquad \partial Q(\theta_r+\delta\theta_r,\; y_m+\delta y_m) \;/\; \partial\theta_r\big|_{\theta_r=\hat{\theta}_r+\delta\hat{\theta}_r} = 0 \;.$$

Expanding (2) in Taylor series and retaining only terms up to second order, we find after subtracting (1)

(7.7.3) $\quad (\partial^2 Q \,/\, \partial\theta_r \partial\theta_r')\ \delta\hat{\theta}_r + (\partial^2 Q \,/\, \partial\theta_r \partial y_m')\ \delta y_m \approx 0$

so that approximately

(7.7.4) $\quad \delta\hat{\theta}_r = -\hat{H}_r^{-1}(\partial^2 Q \,/\, \partial\theta_r \partial y_m')\ \delta y_m$

where $\hat{H}_r = \partial^2 Q \,/\, \partial\theta_r \partial\theta_r'$, evaluated at $\theta_r = \hat{\theta}_r$. The desired covariance matrix of parameter estimates is defined by

(7.7.5) $\quad V(\theta_r) \equiv \mathrm{E}[\delta\hat{\theta}_r \delta\hat{\theta}_r']$

so that

(7.7.6) $\quad V(\theta_r) \approx \mathrm{E}\left[\hat{H}_r^{-1} \dfrac{\partial^2 Q}{\partial\theta_r \partial y_m'} \delta y_m \delta y_m' \left(\dfrac{\partial^2 Q}{\partial\theta_r \partial y_m'}\right)' \hat{H}_r^{-1}\right]$

The quantities H_r^{-1} and $\partial^2 Q \,/\, \partial\theta_r \partial y_m'$ are evaluated at $\theta_r = \hat{\theta}_r$ and at the actual sample y_m. They can be taken outside the expectation sign in equation (6)

(7.7.7) $\quad V(\theta_r) \approx \hat{H}_r^{-1}(\partial^2 Q \,/\, \partial\theta_r \partial y_m')\ V(y)\ (\partial^2 Q \,/\, \partial\theta_r \partial y_m')' \hat{H}_r^{-1}$

where $V(y)$ is the covariance matrix of the data. Equation (7) applies to any objective function, but a more specific result can be obtained for our least squares function Q since it depends only on the second moments matrix of the residuals. This class of functions admits the Gauss approximation for H :

(7.7.8) $\quad H_r \approx -2\sigma^{-2}(L\Delta L')_r,$

when $W = \sigma^2 I$. Furthermore, it can be easily shown that

(7.7.9) $\quad \partial^2 Q \,/\, \partial\theta_r \partial y_m' \approx -2\sigma^{-2}(L\Delta)_r^{V_2}.$

Substituting equations (8) and (9) in (7), and postulating that $V(y) = \sigma^2 I$, we get

(7.7.10) $\quad V(\theta_r) \approx \sigma^2 (L\Delta L')_r^{-1}.$

Comparing this to (5.3) shows that

(7.7.11) $\quad V(\theta_r) \approx \tfrac{1}{2} I_r^{-1}.$

When σ is not known we replace it with its estimate,

(7.7.12) $\quad \dfrac{1}{N-(\sum_{i}^{k} \ell_i^2-(k-1))}\, Q(\hat{\theta}_r).$

Knowledge of the covariance matrix of an estimator as in (11) gives an intuitive feeling of the degree to which the various parameters are well determined. At the same level, comparison of the product moment matrix of the residuals $\hat{V} \equiv V(\hat{f})$ with a given (postulated) covariance matrix $V(\varepsilon)$ of the errors, establishes the goodness of fit of the model itself to the data. An appropriate statistic is given by

(7.7.13) $\quad \lambda = p/2 \ \mathrm{tr}((\hat{V} - V(\varepsilon))\hat{V}^{-1})^2,$

with p the number of times of producing a set of errors in a Monte Carlo simulation. For the case where the distribution of the errors is normal, λ is chi-square distributed with degrees of freedom $\ell(\ell+1)/2 - q$, with q the number of unknown parameters in $V(\varepsilon)$. In case $V(\varepsilon) = \sigma^2 I$, q would be equal to 1.

7.8. ASYMPTOTIC PROPERTIES

7.8.1. Asymptotic distribution of the MLE

The restricted Maximum Likelihood estimator (MLE) $\hat{\theta}_r$ of the true value θ_r^0 of θ_r, satisfies the condition

(7.8.1) $\quad q_r(\hat{\theta}_r) = 0$

By an expansion of a first-order Taylor series of $q_r(\hat{\theta}_r)$ we therefore get,

(7.8.2) $\quad \hat{\theta}_r - \theta_r^0 \overset{a}{=} - \left[\dfrac{\partial^2 Q}{\partial\theta_r \partial\theta_r'} \,\middle|_{\theta_r^0} \right]_r^{-1} q_r(\theta_r^0),$

or equivalently

$$(7.8.3) \qquad \sqrt{N}(\hat{\theta}_r - \theta_r^0) \overset{a}{=} - \left[\frac{1}{N} \frac{\partial^2 Q}{\partial \theta_r \partial \theta_r'} \bigg|_{\theta_r^0} \right]_r^{-1} \frac{1}{\sqrt{N}} q_r(\theta_r^0).$$

The increase of the sample size N leads to three things. First, the first–order Taylor series expansion as an approximation becomes better since $\hat{\theta}_r$ is a consistent estimator. Second, Tchebychev's version of the weak law of large numbers leads to the result that

$$(7.8.4) \qquad - \frac{1}{N} \frac{\partial^2 Q}{\partial \theta_r \partial \theta_r'} \bigg|_{\theta_r^0} \overset{P}{\to} \frac{1}{N} I_r(\theta_r^0).$$

Third, by Slutsky's theorem,

$$(7.8.5) \qquad - \left[\frac{1}{N} \frac{\partial^2 Q}{\partial \theta_r \partial \theta_r'} \bigg|_{\theta_r^0} \right]_r^{-1} \overset{P}{\to} \left[\frac{1}{N} I_r(\theta_r^0) \right]^{-1}.$$

Application of the Central Limit Theorem implicates that $\frac{1}{\sqrt{N}} q_r(\theta_r^0)$ is asymptotically normal distributed. Its expectation is zero and its variance is

$$(7.8.6) \qquad \frac{1}{N} \mathrm{E}(q_r(\theta_r^0) q_r'(\theta_r^0)) = \frac{1}{N} I_r(\theta_r^0),$$

so that

$$(7.8.7) \qquad \frac{1}{\sqrt{N}} q_r(\theta_r^0) \overset{L}{\to} \mathrm{N}(0, \frac{1}{N} I_r(\theta_r^0)).$$

The combination of (3), (5) and (7) leads to

$$(7.8.8) \qquad \sqrt{N}(\hat{\theta}_r - \theta_r^0) \overset{L}{\to} \mathrm{N}(0, N. I_r^{-1}(\theta_r^0)).$$

This result establishes several major properties of the MLE next to its consistency. The MLE is asymptotically normal, it is asymptotically unbiased and it is asymptotically efficient, since the covariance matrix of its limiting distribution equals the Cramér–Rao lower bound. Notice that for estimates under a given sample size the asymptotic covariance matrix of these estimates is, by (8), given by

(7.8.9) $\qquad V(\hat{\theta}_r) \equiv I_r^{-1}(\hat{\theta}_r).$

The reason we may evaluate the expression for $V(\theta_r)$ at $\hat{\theta}_r$, so as to take $V(\hat{\theta}_r)$ as an estimate of $V(\theta_r^0)$, is that just mild conditions are necessary for a function of MLE to be the MLE of the same function of the parameters.

7.8.2. Asymptotic distribution of the GLSE

In case the error terms ε are identically distributed, but not necessarily normal, with zero expectation and covariance matrix W, the asymptotic efficiency of the GLS estimator (GLSE) cannot be assessed, since there is no such thing as an information matrix for this model (2.14). But under fairly weak conditions the GLSE of model (2.14) with the above assumptions, are consistent and asymptotically normal distributed with the same covariance matrix as the MLE.

7.9. PROGRAMMING CONSIDERATIONS

As in the previous three chapters, computational gain can be had when the formulae are computed at a scalar level instead of 'at face value'. By the same reasoning as in these previous chapters, the variable choice of the number of modes, k, is handled by recursive calls of procedures, thereby providing an unlimited number of nested do-loops.

Procedure Make_Y is a recursive procedure for computation of (2.1). Procedure Calculate_scalars_xTBTBx_and_yTBx is meant for computing part of the OLS criterion (3.1), with $W = \sigma^2 I$ substituted. Procedure Create_vecBiXi creates, again in a recursive manner, $\text{vec}(B^i X_i)$, which is used in part in constructing the Hessian and information matrix. Procedure Make_vec_Bx_from_G_and_x creates Bx, which is part of expressions (2.1), (2.14), (3.1), (4.7), etc. Procedure Make_Gradient creates the gradient in case of ML estimation, but without the term due to the additional regressors. However, the case for additional regressors can easily be incorporated.

Procedure Make_information_matrix constructs the information matrix $I(\theta)$ for the balanced data case with $W = \sigma^2 I$, and no additional regressors present.

The information matrix $I(\theta)$ consists of several blocks similar to matrices in (4.12) to (4.15), and (5.1). Notice that, due to our identification restrictions, the first row and column are second–order derivatives with respect to the multiplicative factor γ, and that several rows and columns of (4.12) to (4.15) are deleted. Once more, generalization to the unbalanced data case, with or without additional regressors, is easily induced. Actually, the information matrix follows as expectation of the result of procedure Make_Hessian. This procedure is one of the more spectacular routines with several (internal declared) recursive sub–procedures that build various parts of the Hessian according to expressions (4.12) to (4.15). The block–matrices on the diagonal of the Hessian are computed through use of the (internal declared) recursive routines:

Make_part_non_and_diagonal_blocks_of_Hessian_from_BiXi, and
Make_rest_of_part_non_diagonal_blocks_of_Hessian.

After this, the Hessian is overwritten in 'top–left' direction, so as to delete $k-1$ rows and columns that concern the last elements of the first $k-1$ parameter matrices G_i, which were restricted. The blocks off the diagonal of the Hessian make the Hessian different from the information matrix, so we concentrate on these. Once more, inclusion of unbalanced data with additional regressors causes no revolutionary new statements and changes for the procedures.

The off–diagonal blocks in (4.13) consist of the familiar part $M_i'WM_j$ in addition to R. It is this last matrix that takes up most of the program body of procedure Make_Hessian. The construction of element $R((m-1)*\ell_i+n,\ (p-1)*\ell_j+q)$ in block [*index_i*, *index_j*] is done according to the scheme:

```
Set i := index_i ; j := index_j ;
For i1 := 1 to n do
begin
  set em[i1] := 1 ; rest zero ;
  for j1 := 1 to m do
  begin
    set em[j1] := 1 ; rest zero ;
    for i2 := 1 to q do
    begin
```

```
            set e_q[i_2] := 1 ; rest zero ;
            for j_2 := 1 to p do
            begin
              set e_p[j_2] := 1 ; rest zero ;
              begin
(7.9.1)         R[(m-1)*ℓ_i+n, (p-1)*ℓ_j+q] := 2*x'(C_i'(G_1⊗...⊗G_{i-1}⊗G_{i+1}⊗...⊗G_{j-1}⊗
                                                 e_n e_m'⊗G_{j+1}⊗...⊗G_k⊗e_q e_p')'C_j)
                                               *(Bx-y) ;
              end
            end
          end
        end
      end ;
```

By using recursive procedures of the kind above, with as few internally declared arrays or intermediary results stored in matrices as possible, we develop a separate level of calculus. The calculus makes the pure theoretical analyst aware of the many difficulties (memory availability, ease of insight into the program source–code, available computing time, etc.) which an eventual programmer has with expressions in this and previous chapters.

7.10. ESTIMATION RESULTS

A computer program for WLS estimation of the simplified version (2.1) of model (2.14) has been implemented on a PC. Several test runs with artificial data were made to test the program. In addition, model (2.1) was applied to income satisfaction data of the 'inkomenswaarderingspanel 1980–1983'. This application was presented in the introduction to this chapter.

We did several test runs with different choices of the number of modes, k, (mostly, k=3 to k=6), different values for ℓ_i (ℓ_i=2 to ℓ_i=4, i=1,...,k), and different values for y, x, and G_1 to G_k (i.e., θ^0). We found results that are puzzling. First, we found that when starting values $\theta_r^{(0)}$ for the iterative process were close to the true value θ^0 (for instance, a relative difference in the 2–norm, between each $G_i^{(0)}$ and G_i^0 of, on average, 0.16), then convergence was rapid, but a slightly higher difference between θ^0 and $\theta_r^{(0)}$

(on average, a relative difference of 1 between each $G_i^{(0)}$ and $G_i^{\square}$) meant no convergence at all. This observation was not uniformly true for all datasets. Second, several simulations ended up in final estimation points which were saddlepoints, where the Hessian in scheme (6.9) was not positive definite. Third, several simulations led to essentially different optima when slight changes in starting values were made.

Evidently, for most of our artificial data there are several local optima possible. One is a broad but low minimum of the WLS criterion when even ad hoc starting values (such as identity matrices) converge to this solution. Another is a narrow minimum in the sense that we must start near it to reach it, and a third is a saddlepoint (which looks like an optimum in one direction but is not an optimum in other directions of the multidimensional space).

These observations make our application to income satisfaction data at present doubtful. We took 60 households from the income satisfaction panel. We considered these to be 60 different types, according to 5 educational categories (the original 8 were grouped to 5), 3 occupational categories (self-employed, white collar, or blue collar), and 4 (chosen) age-groups. The dependent variable y was the reported estimate of income satisfaction (as based on the income evaluation question), and the independent variable x was the natural logarithm of the income of each household.

Our starting values ($G_1^{(0)} = I_5$, $G_2^{(0)} = I_3$, $G_3^{(0)} = I_4$) lead to a starting value 6.98 of the WLS criterion, which at iteration step 1093 was reduced to a value of 0.78459..., at which point the iteration was halted. At this point, however, the largest value in the gradient was $2.71*10^{-3}$, and the final estimates of γ and G_1 to G_3 suggest that an optimum was still not reached. The estimate of the multiplicative constant γ was $\hat{\gamma} \cong 2.975$. The largest and smallest element of scaled matrix $\hat{G}_1$ was 18.828 and -18.370, respectively. For $\hat{G}_2$, these values were 2.888 and -2.707, respectively, and for $\hat{G}_3$, 0.28 and -1.236. These values show to us too much variability, when one recognizes that $x \approx y$ and both x and y have no extreme values, but range mostly in the interval [8,11]. We note that the more general model (2.14) was not used. In this model, care can be taken of additional information in the data panel, such as to include additional regressors (income of the household itself, family size, etc.). But we did not use this model mainly because of our computational difficulties mentioned here. The estimation results above lead

us to conclude that computation with the MIR model is a difficult task, which is not solved by us for now.

7.11. CONCLUSION

In this chapter we specified a model for multiplicative interdependence in a regression context. We discussed several versions of this k–mode interdependent regression model, by taking into account the cases of balanced and unbalanced data, additional regressors or none, and normal distribution as underlying distribution or not.

The iterative solving techniques, based on first– and second–order derivatives, for finding parameter estimates, are versions of an ALS, Fisher scoring, and Marquardt algorithm, respectively. The first– and second–order derivatives consist of several (0,1)–matrices and Kronecker products. Sparseness (of the vec representations) of the derivatives leads us to develop routines and additional programming code which rely on scalar level programming. The matter of this type of programming is discussed in section 7, and will be more thoroughly presented in chapter 9. Our simulations suggest that still a lot needs to be done on a theoretical level and at the stage of generating program source–code that is efficient and flexible enough to adjust to the researcher's wish of performing confirmatory data analysis in addition to exploratory data analysis of interdependent data.

7.A. APPENDIX

In case of OLS estimation (i.e., $W = \sigma^2 I$) of the balanced data model, the expectation of (4.13) reduces to

$$M_i'M_j = m(\bar{X}_i'B^{i\prime}\otimes I_{\ell_i})C_i'C_j(B^j\bar{X}_j\otimes I_{\ell_j}), \tag{7.A.1}$$

for which particular derivatives are of related interest. These are stated in the following theorems.

Theorem 7.A.1.

For $i \neq j$,

$$(7.A.2) \qquad C_i'(I_{\ell^i} \otimes G_i \bar{X}_i') \frac{\partial \text{ vec } B^{i\prime}}{\partial g_j'} = C_j'(B^j \bar{X}_j \otimes I_{\ell_j}),$$

where $g_j \equiv \text{vec } G_j$.

Proof

$$(7.A.3) \qquad C_i'(I_{\ell^i} \otimes G_i \bar{X}_i') \frac{\partial \text{ vec } B^{i\prime}}{\partial g_j'} = C_i' \frac{\partial \text{ vec } (G_i \bar{X}_i' B^{i\prime})'}{\partial g_j'} =$$

$$= C_j' C_j C_i' \frac{\partial (B^i \otimes G_i) C_i C_j' C_j C_i' \text{vec } \bar{X}_i'}{\partial g_j'}$$

$$= C_j' \frac{\partial C_i B C_i'}{\partial g_j'} (C_j C_i' \text{vec } \bar{X}_i' \otimes I_{\ell_j} \otimes I_{\ell_j})$$

$$= C_j' (B^j \otimes \frac{\partial G_j}{\partial g_j'}) (\text{vec } \bar{X}_j' \otimes I_{\ell_j} \otimes I_{\ell_j})$$

$$= C_j' (B^j \otimes I_{\ell_j}) (I_{\ell^j} \otimes (\text{vec } I_{\ell_j})' \otimes I_{\ell_j}) (\text{vec } \bar{X}_j' \otimes I_{\ell_j} \otimes I_{\ell_j})$$

$$= C_j' (B^j \otimes I_{\ell_j}) (\bar{X}_j' \otimes I_{\ell_j}) = C_j' (B^j \bar{X}_j \otimes I_{\ell_j}).$$

The last step but one (recomposition of $\bar{X}_j$) is based on Balestra (1976, (2.2.13)), and the last step but three ($\partial G_j / \partial g_j' = (\text{vec } I_{\ell_j})' \otimes I_{\ell_j}$) on Balestra (1976, (4.2.21) and (4.2.36)).

Theorem 7.A.2.

For $i \neq j$,

$$(7.A.4) \qquad (\bar{X}_i' \otimes \bar{Y}_i') \frac{\partial \text{ vec } B^i}{\partial g_j'} = (\bar{X}_i' \otimes I_{\ell_i}) C_i B(ij)' C_j' (\bar{Y}_j \otimes I_{\ell_j}) P_{\ell_j, \ell_j},$$

with $B(ij) = B$ where I_{ℓ_i} and I_{ℓ_j} replace G_i and G_j, respectively.
Proof

(7.A.5) $$(\bar{X}_i'\otimes\bar{Y}_i')\frac{\partial \operatorname{vec} B^i}{\partial g_j'} = (\bar{X}_i'\otimes I_{\ell_i})C_iC_i'\frac{\partial \operatorname{vec}(\bar{Y}_i')B^i}{\partial g_j'} =$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'\frac{\partial\, C_jC_i'(B^i{}'\otimes I_{\ell_i})C_iC_j'}{\partial g_j'}(C_jC_i'\operatorname{vec}\bar{Y}_i'\otimes I_{\ell_j}\otimes I_{\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'\frac{\partial\, C_jB(i)'C_i'}{\partial g_j'}(\operatorname{vec}\bar{Y}_j'\otimes I_{\ell_j}\otimes I_{\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'(B^j(i)'\otimes\frac{\partial G_i'}{\partial g_j'})(\operatorname{vec}\bar{Y}_j'\otimes I_{\ell_j}\otimes I_{\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'\Big[B^j(i)'\otimes((\operatorname{vec} I_{\ell_j})'\otimes I_{\ell_j})(I_{\ell_j}\otimes P'_{\ell_j,\ell_j})\Big](\operatorname{vec}\bar{Y}_j'\otimes I_{\ell_j}\otimes I_{\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'(B^j(i)'\otimes I_{\ell_j})(I_{\ell^j}\otimes(\operatorname{vec} I_{\ell_j})'\otimes I_{\ell_j})(I_{\ell^j}\otimes I_{\ell_j}\otimes P'_{\ell_j,\ell_j})$$
$$(\operatorname{vec}\bar{Y}_j'\otimes I_{\ell_j}\otimes I_{\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'(B^j(i)'\otimes I_{\ell_j})(I_{\ell^j}\otimes(\operatorname{vec} I_{\ell_j})'\otimes I_{\ell_j})$$
$$(\operatorname{vec}\bar{Y}_j'\otimes(I_{\ell_j}\otimes I_{\ell_j})P_{\ell_j,\ell_j})$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'(B^j(i)'\otimes I_{\ell_j})((I_\ell\otimes(\operatorname{vec} I_{\ell_j})')(\operatorname{vec}\bar{Y}_j'\otimes I_{\ell_j})\otimes I_{\ell_j})P_{\ell_j,\ell_j}$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iC_j'\ (B^j(i)'\otimes I_{\ell_j})(\bar{Y}_j'\otimes I_{\ell_j})P_{\ell_j,\ell_j}$$

$$= (\bar{X}_i'\otimes I_{\ell_i})C_iB(ij)'C_j'\ (\bar{Y}_j'\otimes I_{\ell_j})P_{\ell_j,\ell_j}.$$

Here, matrix $B(i)$, of order $\ell\times\ell$, is of same structure as B, but I_{ℓ_i} replaces G_i. As mentioned above, matrix $B(ij)$, of order $\ell\times\ell$, is also of the same structure as B, but I_{ℓ_i} and I_{ℓ_j} replace G_i and G_j, respectively. Matrix $B^j(i)$, of order $\ell^j\times\ell^j$, is of same structure as B^j, but matrix I_{ℓ_i} replaces G_i. The last step but four of the proof is based on Balestra (1976, (2.2.13)) and the last step but eight is based on Balestra (1976, (2.3.22), (4.2.17), and (4.2.36)).

■

8. k-MODE POISSON REGRESSION

8.1. INTRODUCTION

For qualitative or categorical data, empirical and theoretical analyses in the literature have been primarily oriented on the log-linear model specification for the analysis of contingency tables (e.g., Fienberg and Meyer (1983), Goodman (1970, 1978), Haberman (1974a)). Next to the log-linear model approach, Fienberg and Meyer (1983) outline two different ways that correspondence analysis can be used to analyze two-way contingency table data. Correspondence analysis can be thought of as a special case of canonical correlation analysis, which is used in general to study relationships between two sets of variables. Keller and Wansbeek (1983) set correspondence analysis in the context of an errors-in-variables model. A generalization of correspondence analysis to multivariate categorical data is proposed by Greenacre (1988). Greenacre (1988) proposes a technique, which simultaneously fits all distinct two-way contingency tables between pairs of a set of categorical variables by weighted least squares.

The analysis of contingency tables containing cells with structural zeros have also received attention (e.g., Fienberg (1970b, 1972)). These incomplete tables are described by quasi-log-linear models, which are analogous to log-linear models. Fienberg (1970b) discusses aspects of ML estimation in incomplete models under the multiplicative model of quasi-independence. Fienberg (1972) extends the discussion to the incomplete multiway case. Both correspondence analysis and incomplete data will not be pursued any further in this study.

In this chapter we will discuss a *k-mode Poisson regression model*, i.e., a k-mode data analysis model for data that are Poisson distributed that was introduced by Wansbeek and Verhees (1988a). In contrast to k-mode quantitative data analysis models the k-mode Poisson regression model is not a mere extension of existing two-, three- or four-mode models for qualitative or categorical data analysis. For instance, the k-mode model to be proposed here

differs from the usual log–linear model such as discussed by Andersen (1980), Bishop, Fienberg and Holland (1975), Goodman (1970, 1978), McCullagh and Nelder (1983), and O'Muircheartaigh and Payne (1977), and the more general case (considered by Bonett, Bentler and Woodward (1989) and Haberman (1974a)), which is the conditional Poisson log–linear model. The difference is that the expected frequencies of the independent Poisson variates are log–linearly dependent on two sets of parameters with no specific nomenclature, instead of dependent on 'μ terms' (in the nomenclature associated with their ANOVA counterparts) and a preset design matrix as occurring in classical log–linear models. The first set of parameters, which replace the μ terms, are essentially elements of a vector which is a vec representation of a core matrix. These parameters are attached to a second set of parameters (factors, i.e., factor scores) that replace the elements of the design matrix. As a result, several derivations for the log–linear model cannot be directly applied to the k–mode Poisson regression model. For instance, the fact that the second set of parameters replaces a full–rank design matrix with known constants, makes the derivation of conditional distributions of goodness–of–fit statistics by McCullagh (1986) bothersome.

The classical log–linear model is much advocated (cf. the references above) as a means of describing the structure of k–way contingency tables. For any multiway table there are a number of possible hypotheses about the relationship among the data modes. For example, that all the modes are independent, or that each pair of modes is associated but the association is independent of the other modes. Such hypotheses are also stated in log–linear models for contingency tables with ordered classifications of the modes (e.g., Goodman (1979b, 1981) and Haberman (1974b)).

In the log–linear model such hypotheses are specified by models fitted to the observed cell frequencies which are decomposed into a number of multiplicative components. (e.g., Goodman (1979a) for a study of multiplicative models for the analysis of occupational mobility tables). By taking the logarithm of the cell frequencies, the components of the model are transformed to linear form. The resulting expressions may then be related to the *Generalized Linear Model* (GLM), for which Nelder and Wedderburn (1972) presented a computational procedure (GLIM). The log–linear model, in terms of the GLM specification is

(8.1.1) $\quad Y = X\beta + \varepsilon$

where Y represents the log expected frequencies, β represents the vector of constrained parameters, present as specified by the design matrix X, and ε represents the vector of disturbances. The parameters in β of the log–linear model are analogous to the effects in the analysis of variance of multiway data, when the expectation of a multidimensional dependent variable is split into additive components representing an overall mean, main effects and higher order effects. The entries in the design matrix X are coded; 1 if a parameter, indicative of a particular effect, is present in the model for the respective cell, and 0 otherwise.

Here, the term 'generalized' in the GLM stems from the fact that no restrictive assumption to the distribution of the disturbances ε is been made. In the case of log–linear models for count data (Cameron and Trivedi (1986), Frome (1983), Hausman, Hall and Griliches (1984)), a Poisson sampling scheme is common chosen. In that case it is assumed that for a fixed period of time a set of independent Poisson processes is observed, one for each cell in the multiway table. Each Poisson process yields a count for a cell. One important result in the literature is that a number of different sampling schemes (Poisson, multinomial, and product–multinomial) result in the same ML estimates of the expected cell frequencies for the log–linear model, provided constraints imposed by the particular scheme are taken into account.

Our k–mode Poisson regression model slightly differs from the classical log–linear model and their GLM specification in the ways mentioned above. In section 2, the model is specified and in section 3 an ML and IWLS estimation procedure is discussed. Section 4 discusses the problem of identification. Section 5 concentrates on asymptotic properties of both estimators. Section 6 is devoted to an analysis of several simulations. Section 7 discusses some related issues to and further extensions of the model. Section 8 concludes.

8.2. MODEL SPECIFICATION

Let Y be a k–way contingency table (or k–way integer data array), where way i ($i = 1,...,k$) is indicative of mode i, which has ℓ_i levels. Define ℓ, m, m^i, and ℓ^i as in (4.2.3) to (4.2.6). Let the $\ell \times 1$ vector y be the "stacked"

version of the k–way table Y, with elements arranged such that the index for the first mode runs slow and the index for the last mode runs fast.

The assumption of independent Poisson variates as the sampling scheme for the k–way contingency table means that the probability density function of the stochastic variate $\underline{y}_j$ (element of the $\ell \times 1$ stochastic vector y) is

(8.2.1) $$p_j = \text{Prob}\ (\underline{y}_j = y_j) = e^{-\lambda_j}\ \lambda_j{}^{y_j}/y_j\ !$$

In (1) we have underlined the stochast to differentiate it from its observed value.

Furthermore, the assumption is made that the parameters λ_j are log–linearly dependent on unknown parameters A and b:

(8.2.2) $$\ln \lambda_j = (Ab)_j + \varepsilon_j.$$

Here, the $\ell \times m$ matrix A is Kronecker structured according to

(8.2.3) $$A = A_1 \otimes \ldots \otimes A_k,$$

where A_i is of order $\ell_i \times m_i$.

The connection with the "classical" log–linear model approach (by the classical approach is meant the approach in the literature mentioned in section 1) is as follows. For the classical approach, the log–linear model is in matrix terms,

(8.2.4) $$\ln \lambda = X\mu + \varepsilon,$$

with $\lambda \equiv (\lambda_1, \ldots, \lambda_\ell)'$ and where X is an $\ell \times m$ design matrix with fixed elements, determined by the parameter μ required. Each row of the design matrix corresponds to the log–linear model for a particular cell in the table, i.e., to the μ terms which contribute to the respective log–linear model. The task is to reduce the saturated model (i.e., the model that includes all μ terms), by setting some μ terms to zero, to a model whose expected frequencies λ agree satisfactorily with the observed frequencies.

Our model is an even further generalization of the classical log–linear model. Essentially we do not assume the matrix X to be a fixed design matrix, but assume it equal to an unknown matrix A with weights attached, which are the elements of vector b. If b would again be interpreted as μ terms in an ANOVA context, then by the use of an unknown matrix A we have essentially proposed that presence of low– or high–order interactive effects may have a different size of impact for the model at hand. The assumption of a Kronecker structure of A states that the size of impact (i.e., the weight) of the high–order interactive effect at hand is dependent upon the link between the modes that make up the respective high–order effect and is a mere product of unknown but fewer factors A_i, $i = 1,...,k$. By imposing a Kronecker structure on A, the restrictions of totals for μ terms such as in the classical log–linear approach are compensated for, and with an additional number of restrictions the model can be made estimated and parameters are identified.

8.3. ESTIMATION APPROACHES

Two different numerical optimization procedures, namely the iterative proportional scaling procedure (ISP) and iterative weighted least squares (IWLS) procedure, have been widely used in the literature. Essentially, to obtain linear model parameter estimators by IWLS no distribution needs to be specified. This in contrast to ISP (e.g., Fienberg (1970a)) which is based on normality. For IWLS, it is sufficient to express the variance of the dependent variable as a function of the mean E(y) in the form of the so–called variance function

$$\text{var}(y) = c.g(\text{E}(y)) \tag{8.3.1}$$

where g is some known positive function and c is some unknown positive constant. In IWLS, $(\text{var}(y))^{-1}$ is used as a weight function. This is the idea behind the concept of quasi–likelihood, see also McCullagh and Nelder (1983). Procedure IWLS (Nelder and Wedderburn (1972)) is equivalent to WLS with a modified dependent variable f_j,

$$f_j = (k+1)\ln \lambda_j + \frac{y_j - \lambda_j}{\lambda_j}, \tag{8.3.2}$$

and diagonal weight matrix W with elements λ_j. Both f_j and W depend on the

unknown fitted (vector of means) λ_j, so that the fitting process must be iterative.

Grizzle and Williams (1972) also discuss an ML procedure, next to a minimum modified chi-square (MMCS) method, for the traditional log-linear models (i.e., where X is a known design matrix). They prefer the MMCS method because the computations necessary for the MMCS method are identical to those involved in WLS for continuous data.

We will only discuss two estimation approaches, namely an ML procedure comparable to ISP, and an WLS procedure comparable to IWLS.

8.3.1. ML estimation

Maximum Likelihood estimation is the first obvious estimation method. The log-likelihood is

$$L \equiv y'Ab - \lambda's, \tag{8.3.3}$$

with $s = (1,...,1)'$, and constant deleted. Define $\tilde{\lambda} \equiv \mathrm{dg}(\lambda)$ so that tr $\tilde{\lambda} = s'\lambda$. Define the $\ell^i \times \ell_i$ matrix A_i such that vec $A_i' = C_i\lambda$, and the $m^i \times m_i$ matrix B_i such that vec $B_i' = D_ib$. We note that

$$\begin{aligned} y'Ab &= y'C_i'C_iAD_i'D_ib = (\text{vec } Y_i')'(A^i \otimes A_i)\text{vec } B_i' \\ &= (\text{vec } Y_i')'\text{vec } A_iB_i'A^{i'} \\ &= \text{tr } Y_iA_iB_i'A^{i'} = \text{tr } B_i'A^{i'}Y_iA_i \\ &= (\text{vec } Y_i')'(A^iB_i \otimes I_{\ell_i})\text{vec } A_i. \end{aligned} \tag{8.3.4}$$

Furthermore, using matrix differential calculus (Magnus and Neudecker (1988)),

$$\mathrm{d}\,\lambda_j = \mathrm{d}\, e^{a_j'b} = e^{a_j'b}\,(a_j'\mathrm{d}b + (\mathrm{d}a_j)'b) = \lambda_j\,(a_j'\,\mathrm{d}b + (\mathrm{d}a_j)'b), \tag{8.3.5}$$

where a_j' is the j-th row of matrix A, i.e., $A = \begin{bmatrix} a_1' \\ \vdots \\ a_\ell' \end{bmatrix}$. Then ,

$$(8.3.6)\qquad d\lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_\ell \end{bmatrix} \begin{bmatrix} a_1' db + (da_1)'b \\ \vdots \\ a_\ell' db + (da_\ell)'b \end{bmatrix}$$

$$= \tilde{\Lambda}\left\{ \begin{bmatrix} a_1' \\ \vdots \\ a_\ell' \end{bmatrix} db + \begin{bmatrix} (da_1)' \\ \vdots \\ (da_\ell)' \end{bmatrix} b \right\}$$

$$= \tilde{\Lambda}(Adb+(dA)b).$$

Consequently,

$$(8.3.7)\qquad d\ s'\lambda = s'\tilde{\Lambda}\ (Adb+(dA)b) = \operatorname{tr}\ \lambda'(dA)b + \operatorname{tr}\ \lambda' Adb =$$

$$= \sum_i \operatorname{tr}\ \lambda' C_i' C_i (dA) D_i' D_i b + \sum_i \operatorname{tr}\ \lambda' C_i' C_i A D_i' D_i db$$

$$= \sum_i (\operatorname{vec}\ \Lambda_i')'(A^i \otimes dA_i)\operatorname{vec}\ B_i' + \sum_i (\operatorname{vec}\ \Lambda_i')'(A^i \otimes A_i)\operatorname{dvec}\ B_i'$$

$$= \sum_i (\operatorname{vec}\ \Lambda_i')'\operatorname{vec}((dA_i)B_i' A^{i'}) + \sum_i (\operatorname{vec}\ \Lambda_i')'(\operatorname{vec}\ A_i(dB_i')A^{i'})$$

$$= \sum_i \operatorname{tr}\ \Lambda_i (dA_i) B_i' A^{i'} + \sum_i \operatorname{tr}\ \Lambda_i A_i (dB_i') A^{i'}$$

$$= \sum_i \operatorname{tr}\ B_i' A^{i'} \Lambda_i (dA_i) + \sum_i \operatorname{tr}\ A^{i'} \Lambda_i A_i (dB_i')$$

$$= \sum_i (\operatorname{vec}\ \Lambda_i')'(A^i B_i \otimes I_{\ell_i})\operatorname{dvec}\ A_i + \sum_i \operatorname{tr}\ A^{i'} \Lambda_i A_i (dB_i').$$

Now, the first–order conditions can be presented in two ways. The first representation is:

$$(8.3.8)\qquad \frac{\partial L}{\partial A_i} = (Y_i - \Lambda_i)' A^i B_i,$$

$$(8.3.9)\qquad \frac{\partial L}{\partial B_i} = A^{i'}(Y_i - \Lambda_i) A_i \text{ (for any } i).$$

The representation in vec notation is:

$$\frac{\partial L}{\partial \text{ vec } A_i} = (B_i' A^{i'} \otimes I_{\ell_i}) C_i (y-\lambda) \equiv Z_i' (y-\lambda), \tag{8.3.10}$$

$$\frac{\partial L}{\partial b} = A'(y-\lambda). \tag{8.3.11}$$

The second representation is preferred due to its ease of applicability when implementing an iterative numerical optimization procedure.

Let $Z \equiv (Z_1,...,Z_k,A)$, then the first–order condition can be written as

$$Z'(y-\lambda) = 0, \tag{8.3.12}$$

and the information matrix as

$$I(\theta) \equiv Z'\tilde{A}Z. \tag{8.3.13}$$

We prefer to use an iterative numerical optimization approach that is based on the information matrix, for two reasons. First, the inverse of (the part that corresponds to the identified parameters) the information matrix at the final iteration will be an estimate of the covariance matrix of the ML estimators of the parameters in the model. This will be more elaborately discussed in sections 4 and 5. Second, methods based on the information matrix are preferred to iterative methods such as Newton–Raphson, that are based on second–order derivatives. Especially off–diagonal blocks in the Hessian cause the heaviest part of computation involved in these methods, and will usually be reduced by considering the information matrix. The (off–diagonal) blocks of the Hessian are discussed in section 7.

8.3.2. IWLS estimation

Let

$$\theta = ((\text{vec } A_1)', (\text{vec } A_2)', ..., (\text{vec } A_k)', b')' \tag{8.3.14}$$

be the vector of parameters. The likelihood equations for A_i and b are given by

(8.3.15) $Z_i'(y-\lambda) = 0$

and

(8.3.16) $A'(y-\lambda) = 0,$

respectively (cf. (10) and (11)). Thus given ML estimates $\theta^{(t)}$ in the t-th iteration, the Fisher scoring method gives adjustments $\delta\theta^{(t)}$, defined by

(8.3.17) $(Z'\tilde{\Lambda}Z)\delta\theta^{(t)} = Z'^{(t)}(y-\lambda^{(t)}).$

Now,

(8.3.18) $(Z'^{(t)}\tilde{\Lambda}^{(t)}Z^{(t)})\theta^{(t)} =$

$$= Z'^{(t)}\tilde{\Lambda}^{(t)}(Z_1^{(t)},\ldots,Z_k^{(t)},A^{(t)})\begin{bmatrix}\operatorname{vec} A_1^{(t)}\\ \vdots\\ \operatorname{vec} A_k^{(t)}\\ b^{(t)}\end{bmatrix}$$

$$= Z'^{(t)}\tilde{\Lambda}^{(t)}(Z_1^{(t)}\operatorname{vec} A_1^{(t)} + \ldots + Z_k^{(t)}\operatorname{vec} A_k^{(t)} + A^{(t)}b^{(t)})$$

$$= (k+1)Z'^{(t)}\tilde{\Lambda}^{(t)}A^{(t)}b^{(t)} = (k+1)Z'^{(t)}\tilde{\Lambda}^{(t)}(\ln \lambda^{(t)}),$$

since

(8.3.19) $Z_i^{(t)}\operatorname{vec} A_i^{(t)} = C_i'(A^iB_i\otimes I_{\ell_i})\operatorname{vec} A_i = Ab.$

Therefore, new estimates $\theta^{(t+1)} = \theta^{(t)}+\delta\theta^{(t)}$ satisfy the equations

(8.3.20) $(Z'^{(t)}\tilde{\Lambda}^{(t)}Z^{(t)})\theta^{(t+1)} = (Z'^{(t)}\tilde{\Lambda}^{(t)}Z^{(t)})(\theta^{(t)}+\delta\theta^{(t)})$

$$= Z'^{(t)}\tilde{\Lambda}^{(t)}((k+1)\ln \lambda^{(t)} + \tilde{\Lambda}^{(t)-1}(y-\lambda^{(t)})),$$

and these have the form of WLS equations with weight

(8.3.21) $W = \tilde{\Lambda}$

and dependent variable

(8.3.22) $\quad f = (k+1)\ln \lambda + \tilde{\Lambda}^{-1}(y-\lambda),$

cf. (3.2). So the ML estimates of the parameters θ can be obtained by iterative WLS. In this regression the dependent variable f is a linearized form of the link function $\eta = \ln \lambda$ applied to y, and the weights are functions of the fitted values $\hat{\lambda}$. The process is iterative because both the adjusted dependent variable f and the weight W depend on the fitted values $\hat{\lambda}$. The IWLS procedure underlying the iteration is as follows. Let $\hat{\eta}^{(t)}$ be the estimate of the linear predictor η in the t–th iteration, with corresponding fitted values $\hat{\lambda}^{(t)}$ derived from the link function $\eta = \ln \lambda$. Form the adjusted dependent variable with typical value

(8.3.23) $\quad f^{(t)} = (k+1)\hat{\eta}^{(t)} + \tilde{\Lambda}^{(t)-1}(y-\hat{\lambda}^{(t)})$

where $\tilde{\Lambda}$ is evaluated at $\hat{\eta}^{(t)}$, and the weight defined by

(8.3.24) $\quad W^{(t)} = \tilde{\Lambda}^{(t)}.$

Now regress $f^{(t)}$ on $Z^{(t)}$ with weight $W^{(t)}$ to give new estimates $\hat{\theta}^{(t+1)}$ of the parameters; from these form a new estimate $\hat{\eta}^{(t+1)}$, of the linear predictor. Repeat until changes in successive parameter estimates are sufficiently small.

From this section it clearly shows that when the linear predictor η has a known parametric form (e.g., when the Poisson distribution applies to the data), there are powerful techniques available, such as ISP and IWLS, to estimate the model. If the form of the dependence of the predictor on the covariates is not well known (i.e., the exact distribution of the observations y is uncertain), then a nonparametric regression technique, such as presented by O'Sullivan, Yandell and Raynor (1986), is useful. Basically, they use a generalized likelihood approach for estimating the nonparametric model.

8.4. RESTRICTIONS AND IDENTIFICATION

If we make the orthonormality assumptions $A_i'A_i = I_{m_i}$, $i = 1,...,k$, we replace the Fisher scoring method by a similar Lagrange multiplier approach as in section 4.8. Based upon the sample, the Lagrange function with respect to the nonlinear constraints on the parameters is

$$(8.4.1)\qquad L_g = L - \sum_{i=1}^{k} \text{tr } K_i(A_i'A_i - I_{m_i})$$

$$= \sum_{j=1}^{N} L_j - \sum_{i=1}^{k} \text{tr } K_i(A_i'A_i - I_{m_i})$$

$$= \tilde{y}'Ab - \lambda's - \sum_{i=1}^{k} \text{tr } K_i(A_i'A_i - I_{m_i}),$$

where

$$(8.4.2)\qquad L_j \equiv \bar{y}_j(Ab)_j - \frac{1}{N}\lambda_j,$$

$j = 1,...,N$. The $\ell \times 1$ vector s is defined as $s \equiv (\frac{1}{N},...,\frac{1}{N})'$. Matrix K_i is a symmetric matrix of Lagrange multipliers. The bordered information matrix that is used in the numerical solution method is

$$(8.4.3)\qquad B^*(\theta^{(0)}) = \begin{bmatrix} I(\theta) & M(\theta) \\ M'(\theta) & 0 \end{bmatrix},$$

where the matrix $M(\theta)$ denotes the Jacobian, of similar structure as in (4.8.13), and $I(\theta)$ is stated in (3.13).

However, as in the k–mode PCA model, the information matrix $I(\theta)$ shows loss of rank. This loss of rank reflects the lack of identification, due to freedom of rotation of A_i, $i = 1,...,k$, and b. By a similar reasoning as in section 4.8, we get the scheme

$$(8.4.4)\qquad \bar{E}\begin{bmatrix} \theta^{(t+1)} \\ \frac{1}{N}\tilde{k}^{(t+1)} \end{bmatrix} = \bar{E}\begin{bmatrix} \theta^{(t)} \\ 0 \end{bmatrix} + \left(\bar{E}B^*(\theta^{(t)})\bar{E}'\right)^{-1}\bar{E}\begin{bmatrix} \frac{1}{N}\frac{\partial L}{\partial \theta}\big|_{\theta=\theta^{(t)}} \\ h(\theta^{(t)}) \end{bmatrix}.$$

Here $\tilde{k} \equiv (k_1',...,k_k')'$, where k_i is the vec of the nonredundant elements of the symmetric matrix K_i.

Similar considerations also hold in case of IWLS.

8.5. ASYMPTOTIC PROPERTIES

We assume that for each cell in the multiway table an independent Poisson process is observed, i.e.,

(8.5.1) $\quad y_j \sim \text{Poisson}(\lambda_j),$

and p_j is defined according to (2.1). Subscript j indicates the j–th element of vector y, i.e., of a certain cell in the k–way contingency table. If we observe N independent replications of the multiway table, we have essentially created a sample with N observations for each of the ℓ separate cells. An unbiased estimate of the expected frequency λ_j in cell j is then

(8.5.2) $\quad \bar{y}_j = \frac{1}{N} \sum_{r=1}^{N} y_{jr},$

such that $E(\bar{y}_j) = \lambda_j$ and $\text{var}(\bar{y}_j) = \frac{1}{N}\lambda_j$.

8.5.1. Asymptotic distribution of the ML estimator

If the sample size N becomes large enough, the experiment may also be viewed (asymptotically) as a draw from a multivariate normal distribution, i.e.,

(8.5.3) $\quad \sqrt{N}\ (\bar{y}_j - \lambda_j) \xrightarrow{L} N(0,\ \lambda_j),$

or equivalently,

(8.5.4) $\quad \sqrt{N}\ (\bar{y} - \lambda) \xrightarrow{L} N(0,\ \tilde{\Lambda}).$

The joint asymptotic distribution of the ML estimators is stated in the next theorem, which directly follows from theorem 4.9.1.

Theorem 8.5.1.

The joint asymptotic distribution of the random variables $\sqrt{N}\begin{bmatrix} \hat{\theta} - \theta^0 \\ \hat{\tilde{\kappa}} \end{bmatrix}$ is multivariate normal with zero mean and covariance matrix

$$(8.5.5) \qquad V \equiv \left[\bar{E}B^{*}(\theta^{(t)})\bar{E}'\right]^{-1}\bar{E}\begin{bmatrix} Z^{0}\tilde{\Lambda}Z^{0} & 0 \\ 0 & 0 \end{bmatrix}\bar{E}'\left[\bar{E}B^{*}(\theta^{(t)})\bar{E}'\right]^{-1}.$$

Proof

The proof is similar to the proof of theorem 4.9.1, where Σ is replaced by $\tilde{\Lambda}$.

■

8.5.2. Asymptotic distribution of the IWLS estimator

The asymptotic distribution of $\bar{f}$ is given by

$$(8.5.6) \qquad \sqrt{N}\ (\bar{f}_j - (k+1)\ln \lambda_j) \xrightarrow{L} N(0,\ \lambda_j^{-1})$$

due to (3.21), (3.22), and (3). Equivalently,

$$(8.5.7) \qquad \sqrt{N}\ (\bar{f} - (k+1)\ln \lambda) \xrightarrow{L} N(0,\ \tilde{\Lambda}^{-1}).$$

As a result, by taking the identification restrictions into account we get the same joint asymptotic distribution of the parameter estimators and Lagrange multipliers as for the ML estimator in the previous subsection.

8.6. SIMULATIONS

We used several real and artificial datasets to test the program for the model and estimation methods. One real data set consists of coronary heart disease data with four ways of classification. The data were presented in Grizzle and Williams (1972). The data are classified by two types of lesion (infarct and myocardial scars), age, and the combination location of residence and race. Infarct and myocardial scar are binary variables, four age groups are distinguished, and three location/race groups are distinguished. When we choose the model (2.2) with $A_i = I_{\ell_i}$ (i.e., $\ell_i = m_i$), then we essentially have a model with fixed matrix A and parameters solely present in b. This model is

similar to the classical log–linear model (2.4). The outcome of the calculations is that convergence was rapid: the ULS criterion $(y-\hat{\lambda})'(y-\hat{\lambda})$ converges to zero in 15 iterations, the ML criterion converges to 5.73 in 15 iterations. But a choice of $\ell_i \neq m_i$ $(i=1,\ldots,k)$ leads to several problems, both for the real data set and the simulated data. First, several choices of starting values for A_1 to A_k and b lead to values of $A^{(t)}b^{(t)}$ at iteration step t which became too large to calculate values of $\lambda^{(t)} = \exp(A^{(t)}b^{(t)})$. Second, in several simulations, the choices of starting values $\theta^{(0)}$ had to be very close to θ^0 to guarantee any convergence at all. Usually, the iterative process, after a few iterations, started to bounce back and forth between two values of the ML criterion. This leads us to conclude that the computations with the k–mode Poisson regression model are bothersome and that further attention in these matters in needed in future.

8.7. DISCUSSION

8.7.1. Exact second-order derivatives

In this subsection we ignore the identification restrictions which were discussed in section 4 and explicitly treated in section 5. Our aim is to show that the expressions for the exact second–order derivatives are simple extensions of the expressions in section 3 for the components of the information matrix.

The so–called second identification theorem (Magnus and Neudecker (1988)) is the main tool to obtain the Hessian matrix of a real–valued function (such as the log–likelihood) from its second differential. In our case,

$$(8.7.1) \qquad dL = \operatorname{tr} y'(dA)b + \operatorname{tr} y'A(db) - \operatorname{tr} \lambda'(dA)b - \operatorname{tr} \lambda'A(db).$$

Hence, the second differential is

$$\begin{aligned}(8.7.2) \qquad d^2L = {} & \operatorname{tr} y'(d^2A)b + \operatorname{tr} y'(dA)(db) + \operatorname{tr} y'(dA)(db) + \\ & \operatorname{tr} y'A(d^2b) - \operatorname{tr} (d\lambda)'(dA)b - \operatorname{tr} \lambda'(d^2A)b - \operatorname{tr} \lambda'(dA)(db) - \\ & \operatorname{tr} (d\lambda)'A(db) - \operatorname{tr} \lambda'(dA)(db) - \operatorname{tr} \lambda'A(d^2b)\end{aligned}$$

$$= \text{tr } y'(d^2A)b + \text{tr } y'A(d^2b) + 2 \text{ tr } y'(dA)(db) -$$
$$\text{tr } \lambda'(d^2A)b - \text{tr } \lambda'A(d^2b) - \text{tr } (dAb)'\tilde{\Lambda}(dA)b -$$
$$\text{tr } (dAb)'\tilde{\Lambda}A(db) - 2 \text{ tr } \lambda'(dA)(db)$$

$$= \text{tr } y'(d^2A)b + \text{tr } y'A(d^2b) + 2 \text{ tr } y'(dA)(db) -$$
$$\text{tr } \lambda'(d^2A)b - \text{tr } \lambda'A(d^2b) - \text{tr } b'(dA)'\tilde{\Lambda}(dA)b -$$
$$2 \text{ tr } (db)'A'\tilde{\Lambda}(dA)b - \text{tr } (db)'A'\tilde{\Lambda}Adb - 2 \text{ tr } \lambda'(dA)(db).$$

Now, $d^2b = 0$, since b is linear in its parameters (b itself). Therefore,

(8.7.3) $$d^2L = 2 \text{ tr } (y-\lambda)'(dA)(db) - \text{tr } b'(dA)'\tilde{\Lambda}(dA)b + \text{tr } (y-\lambda)'(d^2A)b -$$
$$2 \text{ tr } (db)'A'\tilde{\Lambda}(dA)b - \text{tr } (db)'A'\tilde{\Lambda}A(db).$$

The last term shows

(8.7.4) $$\frac{\partial^2 L}{\partial b \, \partial b'} = - A'\tilde{\Lambda}A.$$

The second term in (3) shows

(8.7.5) $$\text{tr } b'(dA)'\tilde{\Lambda}(dA)b = \sum_i \sum_j \text{tr } b'D_i'D_i(dA)'C_i'C_i\tilde{\Lambda}C_j'C_j(dA)D_j'D_jb =$$

$$= \sum_i \sum_j (\text{vec } B_i')'(A^{i'} \otimes dA_i')C_i\tilde{\Lambda}C_j'(A^j \otimes dA_j)\text{vec } B_j'$$

$$= \sum_i \sum_j (\text{vec } (dA_i)B_i'A^{i'})'C_i\tilde{\Lambda}C_j'\text{vec}((dA_j)B_j'A^{j'})$$

$$= \sum_i \sum_j (\text{vec } dA_i)'(B_i'A^{i'} \otimes I_{\ell_i})C_i\tilde{\Lambda}C_j'(A^jB_j \otimes I_{\ell_j})\text{vec } dA_j.$$

The third term in (3) shows

(8.7.6) $$\text{tr } (y-\lambda)'(d^2A)b = \text{tr } (y-\lambda)'C_i'(I_{\ell^i} \otimes dA_i)(dA^i \otimes I_{m_i})D_ib =$$

$$= (\text{vec}((dA_i)'(Y_i-\Lambda_i)'))'\text{vec}(B_i'(dA^i)')$$

$$= (\text{vec}(dA_i)')'((Y_i-\Lambda_i)' \otimes I_{m_i})(I_{\ell^i} \otimes B_i')\text{vec}(dA^{i'})$$

$$= (\text{dvec } A_i)' P_{\ell_i, m_i} ((Y_i - \Lambda_i)' \otimes I_{m_i})(I_{\ell^i} \otimes B_i') P_{m^i, \ell^i} \text{vec } dA^i$$

$$= (\text{dvec } A_i)' (I_{m_i} \otimes (Y_i - \Lambda_i)') P_{\ell^i, m_i} (I_{\ell^i} \otimes B_i') P_{m^i, \ell^i} \text{vec } dA^i$$

$$= d(\text{vec } A_i)' (I_{m_i} \otimes (Y_i - \Lambda_i)')(B_i' \otimes I_{\ell^i}) \text{vec } dA^i$$

$$= d(\text{vec } A_i)' (I_{m_i} \otimes (Y_i - \Lambda_i)')(B_i' \otimes I_{\ell^i}) \text{vec}(C_j^i{}' (A^{ij} \otimes d\Lambda_j) D_j^i)$$

$$= d(\text{vec } A_i)' (I_{m_i} \otimes (Y_i - \Lambda_i)')(B_i' \otimes I_{\ell^i})(D_j^i \otimes C_j^i)' (I_{m^{ij}} \otimes P_{\ell^{ij}, m_j} \otimes I_{\ell_j})$$
$$(\text{vec } A^{ij} \otimes \text{dvec } A_j)$$

$$= d(\text{vec } A_i)' (I_{m_i} \otimes (Y_i - \Lambda_i)')(B_i' \otimes I_{\ell^i})(D_j^i \otimes C_j^i)' (I_{m^{ij}} \otimes P_{\ell^{ij}, m_j} \otimes I_{\ell_j})$$
$$(\text{vec } A^{ij} \otimes I_{\ell_j m_j}) \text{dvec } A_j.$$

Here, the matrices C_j^i and D_j^i are commutation matrices of order $\ell^i \times \ell^i$ and $m^i \times m^i$, respectively (cf. (3.5.11)). Matrix A^{ij} (cf. (3.5.12)) is of order $\ell^{ij} \times m^{ij}$ (with $\ell^{ij} \equiv \ell/(\ell_i \ell_j)$, $m^{ij} \equiv m/(m_i m_j)$):

$$(8.7.7) \qquad A^{ij} \equiv A_1 \otimes \ldots \otimes A_{i-1} \otimes A_{i+1} \otimes \ldots \otimes A_{j-1} \otimes A_{j+1} \otimes \ldots \otimes A_k.$$

The combined result from (6) and (7) is that

$$(8.7.8) \qquad \frac{\partial^2 L}{\partial \text{vec } A_i \; \partial (\text{vec } A_i)'} = -(B_i A^{i'} \otimes I_{\ell_i}) C_i \tilde{\Lambda} C_i' (A^i B_i \otimes I_{\ell_i}),$$

and

$$(8.7.9) \qquad \frac{\partial^2 L}{\partial \text{vec } A_i \; \partial (\text{vec } A_j)'} =$$

$$= -(B_i' A^{i'} \otimes \mathrm{I}_{\ell_i}) C_i \tilde{\Lambda} C_j' (A^j \mathrm{B}_j \otimes \mathrm{I}_{\ell_j}) +$$

$$(I_{m_i} \otimes (Y_i - \Lambda_i)')(B_i' \otimes I_{\ell^i})(D_j^i \otimes C_j^i)' (I_{m^{ij}} \otimes P_{\ell^{ij}, m_j} \otimes I_{\ell_j})(\text{vec } A^{ij} \otimes I_{\ell_j m_j}).$$

The first and fourth terms in (3) show

$$(8.7.10) \qquad 2 \text{ tr } (y - \lambda)' (dA)(db) - 2 \text{ tr } (db)' A' \tilde{\Lambda} (dA) b =$$

$$= 2 \sum_i^k \text{tr } (y-\lambda)' C_i' C_i (\mathrm{d}A) D_i' D_i (\mathrm{d}b) - 2 \sum_i^k \text{tr } (\mathrm{d}b)' A' \tilde{\Lambda} C_i' C_i (\mathrm{d}A) D_i' D_i b$$

$$= 2 \sum_i^k (\text{vec}(Y_i - A_i)')' (A^i \otimes \mathrm{d}A_i) D_i (\mathrm{d}b) - 2 \sum_i^k (\mathrm{d}b)' A' \tilde{\Lambda} C_i' (A^i \otimes \mathrm{d}A_i) \text{vec } B_i'$$

$$= 2 \sum_i^k (\text{vec}(\mathrm{d}A_i' (Y_i' - A_i') A^i))' (\mathrm{d}b) - 2 \sum_i^k (\mathrm{d}b)' A' \tilde{\Lambda} C_i' \text{vec}((\mathrm{d}A_i) B_i' A^{i'})$$

$$= 2 \sum_i^k (\text{dvec } A_i')' ((Y_i - A_i)' A^i \otimes I_{m_i}) D_i (\mathrm{d}b) -$$

$$2 \sum_i^k (\mathrm{d}b)' A' \tilde{\Lambda} C_i' (A^i B_i \otimes I_{\ell_i}) \text{dvec } A_i$$

$$= 2 \sum_i^k (\text{dvec } A_i)' P_{\ell_i, m_i} ((Y_i - A_i)' A^i \otimes I_{m_i}) D_i (\mathrm{d}b) -$$

$$2 \sum_i^k (\mathrm{d}b)' A' \tilde{\Lambda} C_i' (A^i B_i \otimes I_{\ell_i}) \text{dvec } A_i,$$

such that

$$\frac{\partial^2 L}{\partial \text{ vec } A_i \; \partial b'} = 2 P_{\ell_i, m_i} ((Y_i - A_i)' A^i \otimes I_{m_i}) D_i - 2 (B_i' A^{i'} \otimes I_{\ell_i}) C_i' \tilde{\Lambda} A. \tag{8.7.11}$$

These four second-order derivative expressions are the main elements of the Hessian matrix. By taking the minus expectation of the Hessian matrix, we get the information matrix $I(\theta) \equiv Z' \tilde{\Lambda} Z$ of (3.13).

8.7.2. Associations with k-mode principal component analysis

Essentially, the two main similarities between the ML estimators of the PCA and Poisson regression model (their numerical solution and asymptotic properties), hold due to the following parallels. Ignoring the identification and orthonormality restrictions for now, we have in the k-mode PCA model (with $\Sigma = I$) as first-order derivatives for the ML estimator (cf. (4.8.7) and (4.8.8)),

$$(\bar{Y}_i - A^i B_i A_i)' A^i B_i = 0 \tag{8.7.12}$$

and

(8.7.13) $A'(y-Ab) = 0.$

The first-order derivatives for the ML estimator in the k-mode Poisson regression model are

(8.7.14) $(Y_i-\Lambda_i)'A^{i}B_i = 0$

and

(8.7.15) $A'(y-\lambda) = 0.$

From (13) and (15) follows

(8.7.16) $A^{i'}Y_i\Lambda_i - B_i = 0,$

and

(8.7.17) $A^{i'}(Y_i-\Lambda_i)A_i = 0,$

respectively, from which B_i can be solved. Substituting the result for B_i from (16) in (12) leads to

(8.7.18) $(I_{\ell_i}-A_iA_i')Y_i'A^{i}B_i = 0,$

whereas pre-multiplication of (14) with $(I_{\ell_i}-A_iA_i')$ leads to

(8.7.19) $(I_{\ell_i}-A_iA_i')(Y_i-\Lambda_i)'A^{i}B_i = 0.$

The analogue of (18) to (19) would be an identity if

(8.7.20) $(I_{\ell_i}-A_iA_i')\Lambda_i' = 0.$

The PCA analogue of Λ_i' is $A_iB_i'A^{i'}$ (cf. (12) and (14)) and then equation (20) clearly holds due to the restriction $A_i'A_i = I_{m_i}$.

With respect to the first-order derivatives (3.10) in the Poisson regression model (and, likewise, part of expression (4.8.10) for the PCA model), we show another typical result of applying (0,1)-matrices.

Define $a_i \equiv \text{vec } A_i$ and $a^i \equiv \text{vec } A^i$. Then,

$$(8.7.21)\qquad \frac{\partial\, \Lambda b}{\partial\, a_i'} = \frac{\partial}{\partial\, a_i'}(b\otimes I_\ell)'\text{vec } A$$

$$= \frac{\partial}{\partial\, a_i'}(b\otimes I_\ell)'\text{vec}(C_i'C_iAD_i'D_i)$$

$$= \frac{\partial}{\partial\, a_i'}(b\otimes I_\ell)'(D_i\otimes C_i)'\text{vec}(A^i\otimes A_i)$$

$$= \frac{\partial}{\partial\, a_i'}(D_ib\otimes C_i)'(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(a^i\otimes a_i)$$

$$= C_i'(D_ib\otimes C_i)'(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(a^i\otimes I_{\ell_im_i}),$$

where the last step but one is based on corollary 3.6.1. The right–hand side of (21) equals $C_i'(A^iB_i\otimes I_{\ell_i})$ due to the following result.

Theorem 8.7.1.

Let $x_i = \text{vec}(X_i')$, where X_i is of order $m^i\times m_i$, and let $y^i = \text{vec}(Y^{i'})$, where Y^i is of order $\ell^i\times m^i$. Then,

$$(8.7.22)\qquad (x_i'\otimes I_\ell)(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(y^i\otimes I_{\ell_im_i}) = Y^iX_i\otimes I_{\ell_i}.$$

Proof

$$(8.7.23)\qquad (x_i'\otimes I_\ell)(I_{m^i}\otimes P_{\ell^i,m_i}\otimes I_{\ell_i})(y^i\otimes I_{\ell_im_i}) =$$

$$= ((x_i'\otimes I_{\ell^i})(I_{m^i}\otimes P_{\ell^i,m_i})(y^i\otimes I_{m_i}))\otimes I_{\ell_i}.$$

Inserting (e.g., Balestra (1976, (2.3.13)))

(8.7.24) $$I_{m^i}\otimes P_{\ell^i,m_i} = (I_{m_i}\otimes P_{\ell^i,m^i})(P_{\ell^i,m_i}\otimes I_{m^i})(P_{m^i,\ell^i}\otimes I_{m_i})$$

in the right-hand side of (23) leads to

(8.7.25) $$((x_i'\otimes I_{\ell^i})(I_{m^i}\otimes P_{\ell^i,m_i})(y^i\otimes I_{m_i}))\otimes I_{\ell_i} =$$

$$= ((x_i'\otimes I_{\ell^i})(I_{m_i}\otimes P_{\ell^i,m^i})(P_{\ell^i,m_i}\otimes I_{m^i})(\text{vec } Y^i\otimes I_{m_i}))\otimes I_{\ell_i}$$

$$= ((x_i'\otimes I_{\ell^i})P_{\ell^i,m}(\text{vec } Y^i\otimes I_{m_i}))\otimes I_{\ell_i}$$

$$= ((I_{\ell^i}\otimes x_i')(\text{vec } Y^i\otimes I_{m_i}))\otimes I_{\ell_i}$$

$$= Y^iX_i\otimes I_{\ell_i},$$

where the last step is based on (2.2.17) in Balestra (1976).

■

We see the close congruence of expressions in the k–mode PCA and the k–mode Poisson regression models. This result will be picked up in chapter 9 when the implications of our theoretical derivations for the models, and their similarities with each other, will be translated to final program routines.

8.7.3. Extension of the model

The present model can be extended in such a way, that the rather restrictive assumption of the Poisson model of no dispersion is avoided. We can avoid this assumption of Poisson variation by assuming that

(8.7.26) $$\text{var}(y_j) = \sigma^2\mathbb{E}(y_j) = \sigma^2\lambda_j,$$

where the dispersion parameter σ^2 is assumed constant over the cells, i.e., over the data. Replicated measurements of separate cell–entries will typically show that inter–cell variability leads to over–dispersion (i.e., $\sigma^2 > 1$) relative to the Poisson model. Under–dispersion, a phenomenon less common in

Poisson–like processes, is included here by putting $\sigma^2 < 1$. In both these instances, the asymptotic covariance matrix will become

(8.7.27) $\quad \hat{\sigma}^2 \hat{V}$

with V defined by (5.5). The estimate $\hat{\sigma}^2$ of the dispersion parameter is

$$(8.7.28) \qquad \hat{\sigma}^2 \cong \sum_{j=1}^{\ell} \frac{(y_j - \lambda_j)^2}{\lambda_j} \Big/ (N-s),$$

where $s \equiv m + \sum_{i=1}^{k} m_i(\ell_i - \frac{1}{2}(m_i+1))$.

8.8. CONCLUSION

The k–mode Poisson regression model as proposed in this chapter is not a mere extension of the classical log–linear models such as discussed in the earlier literature on multiway qualitative data analysis. Basically, the interpretation of underlying parameters that identify the expected frequencies of the independent Poisson distributions is different from the approach by ANOVA analysts. However, the same assumption of independent Poisson distributions underlying the observed data table is postulated in this chapter and in ANOVA literature. Once the population distribution is known, ML estimation is a customary approach to use for estimation of the unknown parameters. The vec algebra and matrix differential calculus are the main tools to derive elegant expressions of first– and second–order derivatives of the ML criterion. For WLS estimation similar handsome expressions follow, and an iterative procedure (IWLS) is discussed.

9. FROM FORMULAS TO PROGRAMS

9.1. INTRODUCTION

In the previous chapters we have discussed several data analysis models. For each of these models, we have derived and discussed many formulas, especially formulas for first– and second–order derivatives. Once we bit through the sour apple for deriving these formulas we found that by using notation of Kapteyn et al. (1986), the formulas can be represented by elegant expressions. For instance, we were then able to present asymptotic covariances of estimators in a handsome way.

However, to program these elegant formulas is a totally different matter. We have already discussed separate computational aspects in several chapters (cf. sections 4.11, 5.7, 6.9, and 7.9). But now we can consider computational aspects that are generally relevant to all chapters, i.e., to all five data analysis models. Most of these computational aspects are also discussed by Merckens, Verhees and Wansbeek (1988).

Although we try to keep matters simple, it is not easy to discuss these computational aspects. For this reason we make one simplification, which is as follows. In our discussion we ignore the complications that arise of having to restrict parameters due to identification (cf. sections 4.8, 5.6.2, 6.8, 7.5, and 8.4). We therefore ignore the complications that arise of having to neglect elements in first– and second–order derivatives.

In section 2 we will review all five models, and especially summarize the first– and second–order derivatives. As to the second–order derivatives, we will concentrate on the information matrices only. A discussion in terms of second–order derivatives or Hessian matrices is quite similar, but would involve some additional explanation due to the additional terms. This additional explanation would blur the main aspects.

The first point that we notice is the close similarity between the first–order derivatives, and likewise, between the information matrices. It is then of interest to us, as programmers, to develop general procedures such that these handle several of the derivatives.

A second point of interest is the presence of structure within first–order derivatives and information matrices. For example, the presence of similar blocks $Z_i'\Sigma Z_j$ and $Z_i'\Sigma(y-\lambda)$ in an information matrix $Z'\Sigma Z$ and first–order derivative $Z'\Sigma(y-\lambda)$, respectively, indicates much structure within $Z'\Sigma Z$ and $Z'\Sigma(y-\lambda)$. This point should also be exploited by writing general procedures.

A third point of interest is the presence of possibly large matrices, sparse matrices, and Kronecker products in the expressions for the first–order derivatives and information matrices. This alerts us to avoid using procedures that work at a level of simple matrix multiplication. We better make use of routines that work at a level of scalar multiplication, i.e., by multiplying relevant elements of matrices with one another if these matrices have to be multiplied. The main advantage of this approach is a reduction of memory–assessment. A second advantage however occurs when the matrices are sparse. In this case, the scalar level approach reduces the number of operations (i.e., multiplications and additions), by ignoring zero elements of matrices that are multiplied or added.

A fourth point of interest is the presence of sparse (0,1)–matrices. To retain a pure scalar level approach of expressions with (0,1)–matrices present, we should not declare or use (0,1)–matrices 'at face value'. There are three different approaches to this. The first is to use procedures that describe the operation of a (0,1)–matrix by a mere use of permutations of do–loops. But if expressions are cumbersome of structure, the use of permuted do–loops may lead to less transparent procedures.

Two alternatives to the use of permuted do–loops are then of interest. The first alternative is to record the operative work of (0,1)–matrices by vectors of tables. It leads to more memory–assessment than by using permuted do–loops. But a smaller number of operations is needed by this approach when the (0,1)–matrices are used several times in an expression for a first–order derivative or information matrix. The second alternative is to record the

operative work of (0,1)–matrices by functions. It leads to a similar reduction of memory–assessment but a larger number of operations than the mere use of permuted do–loops. Both alternatives however, retain the transparency of a procedure body.

As is seen from the discussion above, we base the choice of which approach to use on three criteria: transparency of a procedure body, memory–assessment, and number of operations. In particular, we might be able to attach equal weights to these criteria. In practice, the criteria will appear not to have equal weights.

A final point of interest is the number of modes, k. It may be the most important point of interest, for two reasons. First, we want procedures that evaluate expressions for any number of modes, k. Second, we want these procedures to be of a transparent form, and efficient in dealing with memory and number of operations. As a result, it affects the weight attached to each of the three criteria mentioned above, to our choice of recording approach of (0,1)–matrices and to our choice of degree of programming at a scalar level.

The three approaches of recording the operative work of (0,1)–matrices will be discussed in subsection 3.1. In subsection 3.2 we clarify the operations of the three approaches by two examples. These examples concern expressions in which Kronecker products and sparse (0,1)–matrices appear. The examples show a strong case for our preference to use functions or vectors of tables instead of permuted do–loops. Even for expressions, which appear to be of simple structure on paper, the necessary programming effort to evaluate these expressions in permuted do–loops is by far too large. Thus, in practice we are likely to accept more memory–assessment and a larger number of operations. Therefore, in section 4 we concentrate on procedures which apply the tables and functions for recording (0,1)–matrices, and set the use of permuted do–loops aside.

These procedures therefore combine, to some degree, a scalar level programming approach with the second or third approach of recording (0,1)–matrices. We describe procedures that evaluate the first–order derivative and information matrix of the k–mode PCA model. Due to the close resemblance of the derivatives in this model with corresponding derivatives in

the k–mode Poisson regression model, a discussion for the latter model is unnecessary. The close resemblance of the ML estimates in the PCA model and the Poisson regression models was discussed in section 8.7.2. A discussion of computational aspects for the k–mode CSA and MJR model is found in chapters 6 and 7, respectively. Section 5 concludes.

9.2. SUMMARY OF DERIVATIVES IN THE FIVE MODELS

i) k–mode principal component analysis.

The first–order derivatives of the OLS criterion $(y-Ab)'(y-Ab)$ in the k–mode PCA model are

$$q_i = (B_i'A^{i'}\otimes I_{\ell_i})C_i\Sigma^{-1}(y-Ab) = Z_i'\Sigma^{-1}(y-Ab), \tag{9.2.1}$$

$$q_b = A'\Sigma^{-1}(y-Ab), \tag{9.2.2}$$

with Z_i', $i = 1,\ldots,k$, implicitly defined. The information matrix is

$$I(\theta) = Z'\Sigma^{-1}Z, \tag{9.2.3}$$

with $Z \equiv (Z_1,\ldots,Z_k,A)$.

ii) k–mode factor analysis.

Maximum Likelihood estimation of the parameters (combined in a vector variable, denoted by θ) in the k–mode FA model (cf. (5.1.3) and (5.1.4)) leads to the following first–order derivatives (cf. (5.4.12), (5.4.16), (5.4.17) and (5.4.19))

$$\left[\frac{\partial\tfrac{1}{2}\mathrm{tr}\left((\Sigma-S)V^{-1}\right)^2}{\partial(\mathrm{vec}\ A_i)'}\right]' = 2(\lambda^{i'}\otimes I_{\ell_i m_i})B_i(D_i\otimes C_i)(G\Phi G'\otimes I_{\ell})(\Lambda V^{-1}\otimes V^{-1})(\sigma-s) \tag{9.2.4}$$

$$= Z_i'(V^{-1}\otimes V^{-1})(\sigma-s),$$

$$(9.2.5)\qquad \left[\frac{\partial \tfrac{1}{2}\mathrm{tr}((\Sigma-S)V^{-1})^2}{\partial(\mathrm{vec}\ G)'}\right]' = 2(\Lambda V^{-1}\otimes\Phi G'\Lambda V^{-1})(\sigma-s) \equiv Z_G'(V^{-1}\otimes V^{-1})(\sigma-s),$$

$$(9.2.6)\qquad \left[\frac{\partial \tfrac{1}{2}\mathrm{tr}((\Sigma-S)V^{-1})^2}{\partial(\mathrm{dg}\ (\Psi))'}\right]' = H'(V^{-1}\otimes V^{-1})(\sigma-s),$$

and

$$(9.2.7)\qquad \left[\frac{\partial \tfrac{1}{2}\mathrm{tr}((\Sigma-S)V^{-1})^2}{\partial(\mathrm{vec}\ \Phi)'}\right]' = (G'\Lambda V^{-1}\otimes G'\Lambda V^{-1})(\sigma-s) \equiv Z_\Phi'(V^{-1}\otimes V^{-1})(\sigma-s),$$

where $\lambda^i \equiv \mathrm{vec}\ \Lambda^i$. The $\ell^2\times\ell$ matrix H is defined as in (cf. (5.4.18))

$$(9.2.8)\qquad H \equiv \prod_{i=1}^{\ell}(e_ie_i'\otimes e_i).$$

The information matrix, denoted by $I(\theta)$ to indicate its dependence upon all parameters in the model, is in this case (cf. (5.4.33))

$$(9.2.9)\qquad I(\theta) = F'(\theta)(\Sigma\otimes\Sigma)^{-1}F(\theta)$$

with matrices $F'(\theta)$ (cf. (5.4.21)) defined as

$$(9.2.10)\qquad F'(\theta) \equiv K'(Z_1,\ldots,Z_k,Z_G,Z_\Phi,H)'.$$

iii) k-mode covariance analysis.

The first-order derivatives with respect to elements of Σ_i due to ML estimation in the k-mode CSA model (6.2.2) are

$$(9.2.11)\qquad q_i = \ell^i(\Sigma_i\otimes\Sigma_i)^{-1}\sigma_i - (\sigma^{i'}(\Sigma^i\otimes\Sigma^i)^{-1}\otimes(\Sigma_i\otimes\Sigma_i)^{-1})B_i\mathrm{vec}(C_iSC_i')$$

$$= \ell^i(\Sigma_i\otimes\Sigma_i)^{-1}\sigma_i - Z_i'\mathrm{vec}\ S,$$

with Z_i implicitly defined. The information matrix of the ML estimators is (cf. (6.7.5) and (6.7.7))

$$(9.2.12) \qquad I(\theta) = (Q'UQ)^{-1},$$

with $Q = (Q_1,...,Q_k)$, and (cf. (6.7.2))

$$(9.2.13) \qquad U = (I_{\ell^2}+P_{\ell,\ell})(\Omega\otimes\Omega)$$

with (cf. (6.2.2))

$$(9.2.14) \qquad \Omega = \Sigma_1\otimes...\otimes\Sigma_k.$$

iv) k-mode interdependent regression.

The first-order derivatives of a WLS criterion $Q(\theta)$ as in (7.3.1) in the k-mode interdependent regression model (cf. (7.2.2) and (7.2.14)), with respect to the parameters, combined in the vector θ are (cf. (7.4.4) and (7.4.5))

$$(9.2.15) \qquad q_i = -2(y-TBT'x-Z\beta)'W^{-1}(\bar{x}\otimes T')(C_i\otimes C_i)'(I_{\ell^i}\otimes P_{\ell^i,\ell_i}\otimes I_{\ell_i})(\text{vec } B^i\otimes I_{\ell_i^2})$$

$$\equiv -2f'W^{-1}M_i,$$

and

$$(9.2.16) \qquad q_\beta = -2f'W^{-1}Z$$

with f and M_i, $i = 1,...,k$, implicitly defined. In case of ML estimation, the information matrix is (cf. (7.5.1))

$$(9.2.17) \qquad I(\theta) = 2\begin{bmatrix} M'W^{-1/2} \\ Z'W^{-1/2} \end{bmatrix}\begin{bmatrix} W^{-1/2}M, & W^{-1/2}Z \end{bmatrix}$$

with $M \equiv (M_1,...,M_k)$.

v) k–mode Poisson regression.

The first–order derivatives in case of ML estimation of the k–mode Poisson regression model are (cf. (8.3.10) and (8.3.11))

(9.2.18)) $$q_i = (B_i'A^{i\prime}\otimes I_{\ell_i})C_i(y-\lambda)$$
$$\equiv Z_i'(y-\lambda),$$

and

(9.2.19) $$q_b = A'(y-\lambda)$$

with $\lambda \equiv (\lambda_1,...,\lambda_\ell)'$, with Z_i implicitly defined. The information matrix is (cf. (8.3.13))

(9.2.20) $$I(\theta) = Z'\tilde{\Lambda}Z$$

with $\tilde{\Lambda} = \mathrm{dg}(\lambda_1,...,\lambda_\ell)$ and $Z \equiv (Z_1,...,Z_k,A)$.

We see that in all five models, the first–order derivatives and information matrices are similar. They have much structure in common. In addition, there is much structure within each first–order derivative and information matrix. Furthermore, there is a presence of one or a combination of possibly large matrices, sparse matrices, and Kronecker products in these expressions. We therefore avoid writing procedures that work at a level of simple matrix multiplication. We write procedures that work at a level of scalar multiplication. By a scalar level multiplication we mean that relevant scalars (e.g., elements of matrices) are multiplied with one another. The result of each multiplication is part of the repetitive update of an element in a first–order derivative or information matrix.

For instance, take the expression $E = A(B\otimes C)$. It should not be solved as product $E = AD$, where first $D = B\otimes C$ is calculated and stored as intermediate result. This is an inefficient way of programming since it imposes extra restrictions on computer memory–access. We call this 'programming at face valuc'. More efficient is to use the following scheme, where no intermediary results (i.e., matrix D) are stored and used:

(9.2.21) Set matrix E equal to null;
for i_A := 1 to row_A do
for j_B := 1 to col_B do
for j_C := 1 to col_C do
for i_B := 1 to row_B do
for i_C := 1 to row_C do
$E[i_A, (j_B-1)*\text{col\_C}+j_C] := E[i_A, (j_B-1)*\text{col\_C}+j_C] +$
$A[i_A, (i_B-1)*\text{row\_C}+i_C] * B[i_B, j_B] * C[i_C, j_C]$.

If matrix B and C (or both) is a sparse matrix, then matrix D is also a sparse matrix. Especially, when D is a large matrix, it is clear that the scheme above is more efficient than 'at face value' matrix multiplication with matrix D as intermediary result. The scheme above is an illustration of what we call 'programming at scalar level'.

The (0,1)–matrices P, C_i, and B_i are examples of sparse, possibly large, matrices which occur at several instances in our first–order derivatives and information matrices. The (0,1)–matrices should be treated as matrices B and C in scheme (21). The first–order derivatives and information matrices should be treated as matrix E in scheme (21).

Next, the allowance for a choice of any number of modes, k, is of importance when we want to develop procedures that work at a scalar level such as in scheme (21). A difficulty appears, which is best illustrated by an example. Suppose we want to evaluate $E = \Sigma(A_1 \otimes ... \otimes A_k)$, where E, Σ, and A_i are of order $\ell \times m$, $\ell \times \ell$, and $\ell_i \times m_i$, respectively. According to the scheme above we need $2k+1$ do–loops in sequence. If we want to handle any choice of value k, the difficulty is that the number of sequential do–loops should be variable in a procedure. However, the solution is simple. The recursive calls of a procedure in its own body enable this choice of any value of k, since at each call a new set of do–loops is created. The pseudo–Pascal source–code for the procedure is the following.

```
procedure Make_E(E, SIGMA: matrix; counter, row_index_A, col_index_A:
                    integer; elem_A: real) ;
var
  p, q, r: integer ;
begin (* Make_E *)
  if counter <= k then
  begin
    row_index_A := row_index_A*ℓ[counter] ;
    col_index_A := col_index_A*ℓ[counter] ;
    for p := 1 to ℓ[counter] do
      for q := 1 to m[counter] do
        Make_E(E, SIGMA, counter+1, row_index_A+p-1,
              col_index_A+q-1, elem_A*A[counter]^[(q-1)*ℓ[counter]+p])
  end
  else
    for r := 1 to ℓ do
      E[r, col_index_A+1] := E[r, col_index_A+1] +
                              SIGMA[r, row_index_A+1]*elem_A
end ; (* Make_E *)
```

The call of the procedure in the main program is Make_E(E, *SIGMA*, 1, 0, 0, 1).

We see from the declaration of this procedure that at each call a new set of two do–loops is created to handle the Kronecker product string $A_1 \otimes ... \otimes A_k$. Once *counter* reaches value k+1, the $2k$ do–loops for $A_1 \otimes ... \otimes A_k$ are finished, and one do–loop each time finishes the update of elements of a column of E. Which column of matrix E this is, is each time set by the value *col_index_A*+1. Which column of matrix Σ is multiplied with an element of $A_1 \otimes ... \otimes A_k$, is each time set by the value *row_index_A*+1. We note that in the procedure we use *A*[*counter*]^[(*q*–1)*ℓ[*counter*]+*p*] to indicate that matrix A_i is stored as vec(A_i) in its array *A*[*i*].

We conclude that the difficulty with allowing a choice of any number of modes is easily solved for any expression, with a similar structure and appearance of matrices as in the example. But a more serious difficulty appears when (0,1)–matrices are present in the expressions. As an example, take $E = \Sigma C_i(A_1 \otimes ... A_k)$. This is similar to the former example, but with a

commutation matrix present. If a property of the procedure Make_E, its pure scalar level approach by do–loops, is to be kept, then a permutation of do–loops, due to the commutation matrix, will need to fit in the body of the procedure. We recognize that procedure make_E is not easily modified to include the permutation of do–loops for the commutation matrix, without a loss of transparency of the body of the procedure. Two alternatives to the use of permuted do–loops, for recording the operative work of an (0,1)–matrix, are then of interest. We will discuss these in the next section.

9.3. RECORDING THE OPERATIVE WORK OF (0,1)-MATRICES BY THREE APPROACHES

In this section, we discuss three types of routines which define the operative work of the (0,1)–matrices P, C_i, and B_i. The three types of routines for each (0,1)–matrix differ according to the way how the operation of a (0,1)–matrix is recorded. We call the mere use of do–loops, the first approach of recording the operative work of (0,1)–matrices (for short: OP.1). We call the use of vectors of tables, the second approach of recording the operative work of (0,1)–matrices (OP.2). The third approach (OP.3) we present is a compromise between OP.1 and OP.2 and is based on successive calls of functions which compute the indices such as stored in the tables, one at a time. We clarify the definitions of OP.1, OP.2, and OP.3 in the next subsection.

9.3.1. The operative work of (0,1)–matrices

We present the operative work of the (0,1)–matrices P, C_i, and B_i according to each of the three approaches (OP.1, OP.2 and OP.3) in pseudo–Pascal source–code. But first, let us recall what, for instance, the (0,1)–matrix P_{ℓ_i,ℓ^i} does for operative work. As an example, we describe the operation of P_{ℓ_i,ℓ^i} to an ℓ×1 vector x with elements 1 to ℓ, in that order. Define the vector x as: $x(p) = p$, with $p = (i_1-1)\ell^i+j_1$, $i_1 = 1,\ldots,\ell_i$, $j_1 = 1,\ldots,\ell^i$. Then $y = P_{\ell_i,\ell^i}x$ is an ℓ×1 vector defined as $y(p') = x(p) = p$, where $p' = (j_1-1)\ell_i+i_1$. Example: $k = 2$, $\ell_1 = 3$, $\ell_2 = 2$,

$$(9.3.1)\qquad x = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix},\ P_{3,2} = \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1 \end{bmatrix},\ y = P_{3,2}x = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 2 \\ 4 \\ 6 \end{bmatrix}.$$

Now, let vectors s_r and s_c be two $\ell\times 1$ vectors that are similar to vector x. That is, $s_r(p_r) = p_r$ with $p_r = (i_1-1)\ell^i+j_1$, and $s_c(p_c) = p_c$ with $p_c = (i_2-1)\ell^i+j_2$, $i_1,i_2 = 1,...,\ell_i$, $j_1,\ j_2 = 1,...,\ell^i$. Let t_r and t_c be two vectors that are formed as the vector y. The elements of an $\ell\times\ell$ matrix A are indexed by a pair of indices $(s_r,\ s_c)$, where $s_r(p_r)$ gives the row–position of an element and $s_c(p_c)$ gives its column–position. This means for operation $P_{\ell_i,\ell^i}A$, the operation $t_r = P_{\ell_i,\ell^i}s_r$ ensures the rows of the matrix A are reordered, for AP_{ℓ_i,ℓ^i} the operation $t_c = P_{\ell_i,\ell^i}s_c$ ensures the columns are likewise ordered, and in $P_{\ell_i,\ell^i}AP_{\ell^i,\ell_i}$ the combination of operations ($t_r = Ps_r$, $t_c = Ps_c$) ensures the elements of A are reshuffled. To the same extent this can be done for the (0,1)–matrices C_i and B_i.

The basic pseudo–Pascal source–code for the operative work of the (0,1)–matrices P, C_i, and B_i according to each of the three approaches is as follows. In terms of OP.1, the source–code for the operation $y = P_{a,b}x$, concerning two ab–vectors x and y would have to include mainly the following lines:

```
(9.3.2)   for i1 := 1 to a do for j1 := 1 to b do
            y[(j1–1)*a+i1] := x[(i1–1)*b+j1].
```

In terms of OP.2 we have the following. If we set $x[(i_1-1)*b+j_1]:=(i_1-1)*b+j_1$, then the vector y indicates that integers 1 to ab are reshuffled as in the example (1) above. In the sequel we will denote the result of this operation by *table_P*[], which is a vector of integers. For instance, if $a = 3$ and $b = 2$ then $y' =$ *table_P*[] $=$ (1 3 5 2 4 6)'. If x is again a vector with nonspecific elements, then under OP.2, module (2) becomes

```
(9.3.3)   for i1 := 1 to a*b do
            y[i1] := x[table_P[i1]].
```

In terms of the third approach (OP.3), we would declare a function *table_P*() instead of a vector *table_P*[]. The pseudo-Pascal declaration of function *table_P*() is stated in the appendix. Under OP.3 the module (2) becomes

(9.3.4) for i_1 := 1 to $a*b$ do
$y[i_1]$:= x[*table_P*(a, b, i_1)].

The relationship between OP.2 and OP.3 then is clear from the following scheme:

(9.3.5) for *index* := 1 to $a*b$ do
table_P[*index*] := *table_P*(a, b, *index*).

The distinction between OP.2 and OP.3 is clear from (3), (4), and (5). In evaluating, (possibly several), expressions involving $P_{a,b}$, OP.2 uses a once defined vector of integers *table_P*[]. Approach OP.3 calculates individual integers (i.e., elements of the tables) over and over again, for each of the expressions.

The operation $Y = P_{a,b}XP_{c,d}$, concerning two matrices X and Y of appropriate order, is described by (6), (7) and (8) in case of OP.1, OP.2 and OP.3, respectively.

(9.3.6) for i_1 := 1 to a do for j_1 := 1 to b do
for i_2 := 1 to c do for j_2 := 1 to d do
$Y[(j_1-1)*a+i_1, (i_2-1)*d+j_2]$:= $X[(i_1-1)*b+j_1, (j_2-1)*c+i_2]$.

(9.3.7) for i_1 := 1 to $a*b$ do for i_2 := 1 to $c*d$ do
$Y[i_1, i_2]$:= X[*table_P*[i_1], *table_P*[i_2]],

where the two vectors *table_P* are of length ab and cd, respectively.

(9.3.8) for i_1 := 1 to $a*b$ do for i_2 := 1 to $c*d$ do
$Y[i_1, i_2]$:= X[*table_P*(a, b, i_1), *table_P*(c, d, i_2)].

Under OP.1, the source-code for the operation $y = C_i x \equiv (I_{\ell_1} \otimes \ldots \otimes I_{\ell_{i-1}} \otimes P_{\ell_i, \ell_{i+1} \times \ldots \times \ell_k})x$, includes as basics, for any choice of $i = 1,\ldots,k$, the lines:

(9.3.9) –as assignments:

$rep_1 := \ell[1]*\ell[2]*\ldots*\ell[i-1]$; $a := \ell[i]$; $b := \ell[i+1]*\ldots*\ell[k]$.

–as loops:

for $k_1 := 1$ to rep_1 do for $i_1 := 1$ to a do for $j_1 := 1$ to b do

$y[(k_1-1)*a*b+(j_1-1)*a+i_1] := x[(k_1-1)*a*b+(i_1-1)*b+j_1]$.

Under OP.2, we define a vector, denoted by *table_C*[]. Vector *table_C*[] contains integers 1 to ℓ, reshuffled according to the operation of C_i. A pseudo-Pascal declaration of a procedure to create vector *table_C*[] for each c_i, is stated in the appendix. Then (9) is equal to:

(9.3.10) for $i_1 := 1$ to ℓ do

$y[i_1] := x[table\_C[i_1]]$,

where *table_C*[i_1] indicates the i_1-th element of vector *table_C*[]. For the case of OP.3 we have the module

(9.3.11) for $i_1 := 1$ to ℓ do

$y[i_1] := x[table\_C(i,\ i_1)]$,

where function *table_C*() is declared in the appendix.

The relationship between OP.2 and OP.3 is clear from the scheme

(9.3.12) for $i_1 := 1$ to ℓ do

$table\_C[i_1] := table\_C(i,\ i_1)$.

Under OP.1, the source-code for $Y = C_i X C_j'$ would include primarily, for any choice of k and $i = 1,\ldots,k$, the lines:

(9.3.13) –as assignments:

$rep_1 := \ell[1]*\ell[2]*\ldots*\ell[i-1]$; $a := \ell[i]$; $b := \ell[i+1]*\ldots*\ell[k]$.

$rep_2 := \ell[1]*\ell[2]*\ldots*\ell[j-1]$; $c := \ell[j]$; $d := \ell[j+1]*\ldots*\ell[k]$.

–as loops:

for k_1 := 1 to rep_1 do for i_1 := 1 to a do for j_1 := 1 to b do
for k_2 := 1 to rep_2 do for i_2 := 1 to c do for j_2 := 1 to d do
$Y[(k_1-1)*a*b+(j_1-1)*a+i_1,\ (k_2-1)*a*b+(j_2-1)*c+i_2]$:=
$X[(k_1-1)*a*b+(i_1-1)*b+j_1,\ (k_2-1)*a*b+(i_2-1)*d+j_2]$.

In case of OP.2 module (13) transforms to:

(9.3.14) for i_1 := 1 to ℓ do for i_2 := 1 to ℓ do
$Y[i_1,\ i_2]$:= $X[table\_C[i_1],\ table\_C[i_2]]$,

where it is understood that the first and second vector $table\_C[]$ concern C_i and C_j, respectively.

In case of OP.3 module (13) transforms to:

(9.3.15) for i_1 := 1 to ℓ do for i_2 := 1 to ℓ do
$Y[i_1,\ i_2]$:= $X[table\_C(i,\ i_1),\ table\_C(j,\ i_2)]$.

Under OP.1 the source-code for $y = B_i x \equiv (I_{\ell^i} \otimes P_{\ell_i, \ell^i} \otimes I_{\ell_i})x$ would include primarily, for any choice of $i = 1,...,k$:

(9.3.16) -as assignments:
rep_1 := $\ell[1]*\ell[2]*...*\ell[i-1]*\ell[i+1]*...*\ell[k]$;
a := $\ell[i]$; b := rep_1; q := a.
-as loops:
for k_1 := 1 to rep_1 do for i_1 := 1 to a do
for j_1 := 1 to b do for q_1 := 1 to q do
$y[(k_1-1)*a*b*q+(j_1-1)*a*q+(i_1-1)*q+q_1]$:=
$x[(k_1-1)*a*b*q+(i_1-1)*b*q+(j_1-1)*q+q_1]$.

In case of approach OP.2, module (16) is replaced by

(9.3.17) for i_1 := 1 to $\ell*\ell$ do
$y[i_1]$:= $x[table\_B[i_1]]$.

The procedure to create vector $table\_B[]$ for each B_i is declared in the appendix.

For the case of OP.3, module (16) is replaced by

(9.3.18) for i_1 := 1 to $\ell*\ell$ do
$y[i_1]$:= $x[table\_B(i,\ i_1)]$,

where function *table_B*() is declared in the appendix. Once again, the relationship between OP.2 and OP.3 is clear from the scheme

(9.3.19) for i_1 := 1 to $\ell*\ell$ do
table_B$[i_1]$:= *table_B*$(i,\ i_1)$.

Under OP.1, the source-code for $Y = B_iXB_j'$ would contain as basics, for any choice of i, j and k:

(9.3.20) –as assignments:
rep_1 := $\ell[1]*\ell[2]*...*\ell[i-1]*\ell[i+1]*...*\ell[k]$;
a := $\ell[i]$; b := rep_1; q := a.
rep_2 := $\ell[1]*\ell[2]*...*\ell[j-1]*\ell[j+1]*...*\ell[k]$;
c := $\ell[j]$; d := rep_2; r := d.
–as loops:
for k_1 := 1 to rep_1 do for i_1 := 1 to a do
for j_1 := 1 to b do for q_1 := 1 to q do
for k_2 := 1 to rep_2 do for i_2 := 1 to c do
for j_2 := 1 to d do for q_2 := 1 to r do
$Y[(k_1-1)*a*b*q+(j_1-1)*a*q+(i_1-1)*q+q_1,$
$(k_2-1)*c*d*r+(j_2-1)*d*r+(i_2-1)*r+q_2]$:=
$X[(k_1-1)*a*b*q+(i_1-1)*b*q+(j_1-1)*q+q_1,$
$(k_2-1)*c*d*r+(j_2-1)*c*r+(i_2-1)*r+q_2]$.

Under OP.2, module (20) is replaced by

(9.3.21) for i_1 := 1 to $\ell*\ell$ do for i_2 := 1 to $\ell*\ell$ do
$Y[i_1,\ i_2]$:= $X[table\_B[i_1],\ table\_B[i_2]]$,

where it is understood that the first and second vector *table_B*[] concern B_i and B_j, respectively.

Under OP.3, module (20) is replaced by

(9.3.22) for $i_1 := 1$ to $\ell*\ell$ do for $i_2 := 1$ to $\ell*\ell$ do
$Y[i_1, i_2] := X[table\_B(i, i_1), table\_B(j, i_2)]$.

9.3.2. Discussion

The choice of which approach (OP.1, OP.2, or OP.3) to use in actual implementations is based on three criteria: ease of insight into (i.e., transparency of) a procedure body, memory–assessment and computing time. There are a number of theoretically possible situations to distinguish:

i) If the expressions of interest are of simple structure but involve large matrices, and memory–assessment is of primary concern, then OP.1 is preferred to OP.2 and OP.3.

ii) If the expressions to be evaluated are cumbersome of structure, even on paper, then OP.2 and OP.3 is to OP.1.

iii) If the expressions are cumbersome of structure and do not involve large matrices, but have to be computed several times, for instance in iterative schemes in the main program, then OP.2 is preferred to OP.3. The argument is that the declaration and assignment of vectors of tables at one time in the program is probably less than the cost of computing time that is involved with the repetitive computation of individual elements of the same vectors of tables at each iteration in iterative schemes in the program.

iv) If the expressions are cumbersome of structure and do involve large matrices, then transparency and memory–assessment are of primary concern, in that order. In this case, OP.3 could be preferred to OP.1 and OP.2.

In practice however, we are likely to accept more memory–assessment and a larger number of operations in favor of more transparency of procedures. As it turns out below, the goal of more transparency is a difficult goal to accomplish, especially when analyzing the models and expressions involving (0,1)–matrices. For these cases, the functions or vectors of tables are easier to use than a mere use of do–loops that have to be permuted. These functions or tables are merely 'plugged' in a program. We therefore set situation *i*), mentioned above, aside and will not prefer OP.1 to OP.2 and OP.3.

To illustrate our reasoning, we will discuss two examples. The first example concerns the expression $E = \Sigma C_i(A_1 \otimes ... \otimes A_k)$ of section 2. Under OP.2 and OP.3, a modification of procedure Make_E in section 2 is straightforward as follows. The second part of procedure Make_E is replaced by:

```
(9.3.23)    for r := 1 to ℓ do
              begin
                new_index := table_C[row_index_A+1] ;
                E[r, col_index_A+1] := E[r, col_index_A+1] +
                                       SIGMA[r, new_index] * elem_A
              end
```

and

```
(9.3.24)    for r := 1 to ℓ do
              begin
                new_index := table_C(i, row_index_A+1) ;
                E[r, col_index_A+1] := E[r, col_index_A+1] +
                                       SIGMA[r, new_index] * elem_A
              end,
```

respectively. Comparison of the two schemes again shows the close resemblance in form of procedures when OP.2 and OP.3 are used. The procedure Make_E is not easily modified under OP.1, such that it would merely include (permuted) do-loops due to the commutation matrix.

The second example is the operation $P_{\ell,\ell}B_i(C_i \otimes C_i)x$, where x is an $\ell^2 \times 1$ vector. The operative work of matrices $P_{\ell,\ell}$, C_i, and B_i is stored in *table_P*[], *table_C*[], and *table_B*[], respectively, or is described by using the corresponding functions *table_P*(), *table_C*(i), and *table_B*(i), respectively. Then $y = P_{\ell,\ell}B_i(C_i \otimes C_i)x$ is indicated for OP.2 and OP.3 by the following:

```
(9.3.25)    for i1 := 1 to ℓ do for i2 := 1 to ℓ do
              y[(i1-1)*ℓ+i2] :=
                x[table_P[table_B[(table_C[i1]-1)*ℓ+table_C[i2]]]],
```

(9.3.26) for i_1 := 1 to ℓ do for i_2 := 1 to ℓ do
$y[(i_1-1)*\ell+i_2]$:=
$x[table\_P(\ell,\ell,table\_B(i,(table\_C(i,i_1)-1)*\ell+table\_C(i,i_2)))]$.

To come to an equivalent expression of the operation in terms of OP.1, we need a combination of (2), (9) and (16). This alternative approach would however mean that several intermediary results would be needed, since the indices of the operands are to be known fully before the next operator in sequence is activated. The do–loops scheme would in this case be the following.

For $y = (C_i \otimes C_i)x$ we need:

(9.3.27) rep_1 := $\ell[1]*\ell[2]*...*\ell[i-1]$; a := $\ell[i]$; b := $\ell[i+1]*...*\ell[k]$;
for k_1 := 1 to rep_1 do for i_1 := 1 to a do for j_1 := 1 to b do
for k_2 := 1 to rep_1 do for i_2 := 1 to a do for j_2 := 1 to b do
$y[((k_2-1)*a*b+(j_2-1)*a+i_2-1)*\ell+(k_1-1)*a*b+(j_1-1)*a+i_1]$:=
$x[((k_2-1)*a*b+(i_2-1)*b+j_2-1)*\ell+(k_1-1)*a*b+(i_1-1)*b+j_1]$.

Then for $x = B_i y$ we need:

(9.3.28) rep_1 := $\ell[1]*\ell[2]*...*\ell[i-1]*\ell[i+1]*...*\ell[k]$; a := $\ell[i]$;
for k_1 := 1 to rep_1 do for i_1 := 1 to a do
for j_1 := 1 to rep_1 do for q_1 := 1 to a do
$x[(k_1-1)*a*a*rep_1+(j_1-1)*a*a+(i_1-1)*a+q_1]$:=
$y[(k_1-1)*a*a*rep_1+(i_1-1)*a*rep_1+(j_1-1)*a+q_1]$.

Then finally for $y = P_{\ell,\ell}x$ we need:

(9.3.29) for i_1 := 1 to ℓ do for j_1 := 1 to ℓ do
$y[(j_1-1)*\ell+i_1]$:= $x[(i_1-1)*\ell+j_1]$.

The advantage of using OP.2 or OP.3 is clear from these two examples. Even with simple expressions, such as $\Sigma C_i(A_1 \otimes ... A_k)$ and $P_{\ell,\ell}B_i(C_i \otimes C_i)x$, the mere use of (a permutation of) do–loops (OP.1) results in less transparency of procedures than the use of vectors of tables (OP.2) or the use of functions (OP.3). So, approaches OP.2 and OP.3 to (0,1)–matrices are our main tools for

computationally efficient programming. They are used for each expression in the models discussed in section 2.

In the next section we discuss the implementation of procedures that use OP.2 and OP.3. However, we will only discuss their use for the k–mode PCA model. The reasons are the following. Due to the close resemblance of expressions in this model with corresponding expressions in the k–mode Poisson regression model, a discussion for the last model is unnecessary. A discussion of computational aspects for the k–mode FA, CSA, and MIR model is found in chapters 5, 6, and 7, respectively.

9.4. APPLICATION OF THE MODULES FOR THE k–MODE PCA MODEL

If we first concentrate ourselves at constructing $Z_i \equiv C_i(A^iB_i \otimes I_{\ell_i})$ in (2.1), we see that we need one $\ell^i \times m^i$ matrix A^i, one $m^i \times m^i$ matrix B_i and the operative work of $\ell \times \ell$ matrix C_i to construct an $\ell \times \ell_i m_i$ matrix Z_i. We will not construct matrix A^i 'at face value' but will sequentially compute its elements, and the positions they would have in matrix A^i, when necessary for further elaborations of expressions. Apart from efficiency, generality is also desirable in the sense that the routines should work for any choice of the number of modes, k, and any choice of $i = 1,...,k$. We will go through the steps to construct Z_i.

(1) Suppose we have computed an $m \times 1$ vector b. We then know that element $B_{i(p,q)}$, for $p = 1,...,m^i$ and $q = 1,...,m_i$, in matrix B_i is the element with index $(p-1)m_i+q$ in vector b. This is due to definition of the vec operator. So, we do not need to declare and form matrix B_i itself.

(2) Suppose we Kronecker–multiply element (i_1, j_1) of matrix A_1 with element (i_2, j_2) of A_2, and so forth, while ignoring elements of A_i. We then know that the result will give the value for the element (n, p) of $\ell^i \times m^i$ matrix A^i with indices n and p equal to:

$$(9.4.1) \qquad n = (i_1-1)*\prod_{\substack{j=2 \\ j \neq i}}^{k} \ell_j + (i_2-1)*\prod_{\substack{j=3 \\ j \neq i}}^{k} \ell_j + ... + (i_{(k-1)}-1)*\ell_k + i_k,$$

and

$$(9.4.2) \qquad p = (j_1-1)*\prod_{\substack{j=2\\ j\neq i}}^{k} m_j+(j_2-1)*\prod_{\substack{j=3\\ j\neq i}}^{k} m_j+\ldots+(j_{(k-1)}-1)*m_k+j_k,$$

due to the relation $A = A_1\otimes\ldots\otimes A_k$. Here we have $i_1 = 1,\ldots,\ell_1$, $j_1 = 1,\ldots,m_1$, $i_2 = 1,\ldots,\ell_2$, $j_2 = 1,\ldots,m_2$, and so forth. Subsequently, $n = 1,\ldots,\ell^i$ and $p = 1,\ldots,m^i$.

(3) We want to handle the choice of any value for k. As discussed at the end of section 2, this is achieved by a recursive call of the routine that would handle the structure of A^i (and subsequently, Z_i). The recursive call of the routine in its own body will enable this choice of any value of k, since after each call a new set of necessary do-loops is created.

(4) Suppose we have computed element $A^i_{(n,p)}$. We then know that element (n, q) in $\ell^i\times m_i$ matrix A^iB_i is the product of $A^i_{(n,p)}$ and $B_{i(p,q)}$. Likewise, due to step (1),

$$(9.4.3) \qquad A^iB_{i(n,q)} := A^i_{(n,p)}*b[(p-1)*m_i+q],$$

for $n = 1,\ldots,\ell^i$, $p = 1,\ldots,m^i$ and $q = 1,\ldots,m_i$.

(5) We then know that the operation $(A^iB_i\otimes I_{\ell_i})$ means a diagonal-wise ℓ_i-time replication of each element of matrix A^iB_i, i.e., element (n,q) of A^iB_i would appear in $\ell\times\ell_i m_i$ matrix $A^iB_i\otimes I_{\ell_i}$ at positions $((n-1)*\ell_i+rep,\ (q-1)*\ell_i+rep)$, for $n = 1,\ldots,\ell^i$, $q = 1,\ldots,m_i$ and $rep = 1,\ldots,\ell_i$. The rest of the elements of $A^iB_i\otimes I_{\ell_i}$ would not be formed since they are known to be zero. Due to the sparseness and replicative character of matrices $A^iB_i\otimes I_{\ell_i}$, in particular with higher ℓ_i, we will not construct these matrices 'at face value' (and therefore do not actually place elements in matrix $A^iB_i\otimes I_{\ell_i}$), but merely form the nonnull elements and their positions in matrix $A^iB_i\otimes I_{\ell_i}$ by running the indices n, p, and q.

(6) The operative work of C_i to a vector was discussed in section 3, and subsequently will not pose any new problems for the description of its work to a matrix. To reason this for C_i in this case will not be any more difficult.

The basic source–code for the operation $C_i'(A^iB_i\otimes I_{\ell_i})$ will indicate the following. Set n according to (1) and set p according to (2). Let $c \equiv A^i_{(n,p)}$. Then $Z_i = C_i'(A^iB_i\otimes I_{\ell_i})$ is constructed through the loops:

(9.4.4) for q := 1 to $m[i]$ do for rep := 1 to $\ell[i]$ do
$Z_i[table\_C[(n-1)*\ell[i]+rep],\ (q-1)*\ell[i]+rep] :=$
$c*b[(p-1)*m[i]+q]$,

in case of use of OP.2, or

(9.4.5) for q := 1 to $m[i]$ do for rep := 1 to $\ell[i]$ do
$Z_i[table\_C(i,\ (n-1)*\ell[i]+rep),\ (q-1)*\ell[i]+rep] :=$
$c*b[(p-1)*m[i]+q]$,

in case of use of OP.3.

The final basic procedure to compute q_i and q_b are then easily formed. The information matrix $I(\theta)$ in (2.3) will present no further difficulties now that we know how to form elements of Z_i and A efficiently.

9.5. CONCLUSION

The final implementation of a data analysis model, or any econometric model for that matter, in computer programs is always of importance. When we use simulations to validate or analyze properties of estimators, we are inclined to use routines in existing statistical program libraries and plug these into our own computer program. Likewise, we are inclined to use these standard routines in our program when we use a regression model to describe real data. However, for the k–dimensional data analysis models in this study, the dimensionality of the data itself is of importance. We direct our attention to the dimensionality of data, not only as data analysts or modelers with an econometric background, but also as programmers.

As a modeler, we see that many expressions (e.g., first– and second–order derivatives) in the five k–dimensional data analysis models have much structure. In addition, the expressions contain Kronecker products, sparse

(0,1)–matrices, and large matrices. As a programmer we realize that these points should be exploited. A first step is to not use existing program libraries and routines that work at a level of pure matrix multiplication. This type of level of programming, which we dub 'programming at face value' is unsatisfactory, both to the programmer and the modeler. The modeler stresses the structure within, and the other points, to each expression. The programmer recognizes the advantages of using a different type of level of programming to address these points. Both modeler and programmer would conclude that if for all five data analysis models real data with three or more dimensions exist, a special calculus is profitable.

We have developed this special calculus. Essentially, it consists of two sets of tools. The first set of tools are the recursive procedures with a minimum use of array–declarations. A procedure which is recursive, permits a variable number of do–loops. Next, this variable number of do–loops permits a choice of any number of modes, k. The minimum use of array–declarations leads to a minimum use of 'at face value' matrix multiplications.

The second set of tools are the three approaches (OP.1, OP.2, and OP.3), with which the (0,1)–matrices in the formulas are handled. We have not yet succeeded in implementing approach OP.1 in all our recursive procedures. For simple formulas such as $C_i(y–Ab)$ we did succeed. For now, it costs too much programming effort to get recursive procedures that are, both, transparent and would include approach OP.1. This is part of our study on computational aspects that deserves more attention in future. We did succeed in implementing approaches OP.2 and OP.3 in all our recursive procedures. The advantage of both approaches is that these require just a slight modification to the procedures of the first set. The recursive procedures stay transparent with minor programming effort. In addition, for OP.2 and OP.3, a minor extra use of array–declarations or function calls, respectively, was needed. Due to both sets of tools, a so–called 'scalar–level calculus' is demonstrated in the computer programs for each of the five data analysis models, using real and artificial data.

This also raises possibilities for research in future, especially on efficient computer manipulation for applications in other fields of research in multidimensional statistical analysis. Examples, amongst others, are

Pereyra and Scherer (1973), De Boor (1979), and Dyksen (1987), who discuss efficient computer manipulation for applications in multidimensional approximation. Percyra and Scherer (1973) propose an algorithm for solving linear systems $(A_1 \otimes ... \otimes A_k)x = y$, with invertible matrix A_i, $i = 1,...,k$, and any vectors x and y of appropriate order. De Boor (1979) slightly modifies their algorithm, by presenting an algorithm that solves $(A_1 \otimes ... \otimes A_k)X = Y$, for invertible matrices A_i, but with matrices X and Y instead of vectors x and y, respectively. Dyksen (1987) also recognizes the fact that a particular matrix factors into the Kronecker product of two or more matrices, is of no value without algorithms for doing efficient computer manipulation of Kronecker products. Dyksen (1987) derives an iterative method to solve discrete elliptical problems of the form $(A_1 \otimes A_2 + A_3 \otimes A_4)X = Y$.

```
    c := Product_ℓ(i+1, k, ℓ);
    for p := 1 to a do
    begin
      for q := 1 to b do
        for r := 1 to c do
          table_C[index+(r-1)*b+q] := index+(q-1)*c+r
    end
  end;
```

Note that by replacing integers $\ell[i]$ with integers $m[i]$ in procedure Create_table_C, we create a vector *table_D*[] that describes D_i.

Pseudo–Pascal source listing for vector *table_B*[] which describes $B_i = I_a \otimes P_{b,c} \otimes I_d$ (where, as an example, $a = m^i$, $b = \ell^i$, $c = m_i$, and $d = \ell_i$ is chosen):

```
  procedure Create_table_B(i: integer) ;
  var
    a, b, c, d, p, q, r, index : integer;
    ℓ, m: intvector;
  begin
    a := Product_ℓ(1, k, m) div m[i];
    b := Product_ℓ(1, k, ℓ) div ℓ[i];
    c := m[i];
    d := ℓ[i];
    for p := 1 to b*c do
      for q := 1 to d do
        for r := 1 to a do
        begin
          index := (r-1)*b*c*d;
          table_B[index+(p-1)*d+q] := (table_P(b, c, p) - 1)*d+index+q
    end
  end;
```

Note that in procedure Create_table_B, we use function *table_P*(*a*, *b*, *index*) stated above. Of course, we could also have inserted the vector *table_P*[] instead, i.e., replace *table_P*(*b*, *c*, *p*) by *table_P*[*p*] .

10. CONCLUSION

The aim of this study was threefold. First, we investigated the possibility of extending several data analysis models that are widely known in data analysis research. The stimulus was that several of these models were extended up to three dimensions of the data in most of the literature on data analysis. There may have been several reasons for this. But whatever other arguments, one argument is easily defeated. This argument is that the description of the models would be tedious, due to awkward notation. But as it turns out in the study, the recently introduced notation by Kapteyn, Neudecker and Wansbeek (1986) makes our task of an extension of several data analysis models manageable. By using additional tools from matrix differential calculus and (0,1)–matrix algebra, we extend five data analysis models to an arbitrary number of dimensions, k.

The second aim of this study then related to the question whether we can describe and analyze (asymptotic) properties of several estimators of the parameters in each model, and whether we can extend several existing numerical solving techniques. One conclusion is that the expressions for the asymptotic distributions (e.g., asymptotic covariances) can be presented in handsome expressions. The form of these expressions is such, that it can be easily seen that they are often enough direct generalizations of results for one–, two–, or three–mode models in earlier literature. But, thanks to the manageability of the expressions, properties of several new estimators also result with minor effort. For instance, the asymptotic properties of a noniterative ULS estimator (consistent, but not asymptotically efficient) in the k–dimensional covariance structure analysis model are derived by assuming a regression context. So, once again, theoretical results to the generalizations are easy to get and to prove. Another conclusion however, is that in practice, estimators and their properties are troublesome to derive by the generalizations of existing numerical solving techniques. Our simulations for several of the k–dimensional models show a numerical instability of solutions.

Luckily enough, this is not true in general. We have derived satisfactory results for real data that are analyzed and interpreted in the k–dimensional PCA model. In a similar way, we were able to deliver reliable estimation results for real and simulated data in the k–mode CSA model.

The third aim of the study was to investigate the possibility of developing matching program routines. This aim appears naturally, when one strives for efficient computer programs. By efficiency is meant the efficient use of computer memory and the efficient use of the number of operations (i.e., multiplications and additions). To gain this efficiency, we should not use routines from existing statistical program libraries, and simply plug these into our own computer program. These routines work at a level of pure matrix multiplication, which is unsatisfactory, both to the programmer and the modeler. The modeler stresses the structure within each formula, the occurrence of large matrices, and the occurrence of sparse (0,1)–matrices. The programmer recognizes the advantages of using a different type of programming to address these points. This type of programming uses elementwise multiplication of matrices and iterative updating of the results by recursive procedures. In this way, both the issue of structure and size of formulas is addressed. The issue of appearance of (0,1)–matrices in formulas is addressed by using either one of two approaches for recording the operative work of (0,1)–matrices. One approach is to use a vector of integers that describes the shuffling a (0,1)–matrix performs to a vector or matrix. This leads to a declaration of a vector array of integers. The other approach is to use a function that computes elements of the same vector array of integers, over and over again, but without any declaration of a vector whatsoever. Both approaches keep the recursive procedures transparent.

SAMENVATTING

Meerdimensionale matrices komen voor in de psychometrie, in de data–analyse in het algemeen, en in de econometrie in engere zin. Diverse modellen voor analyse van meerdimensionale data, zoals weergegeven in de meerdimensionale matrices, zijn terug te vinden in de literatuur van bovenstaande vakgebieden. Maar de modellen beperken zich tot op heden voornamelijk tot een beschrijving van data met een maximum van drie dimensies. De generalisatie van de modellen tot k dimensies beperkt zich dan ook vaak tot een korte beschrijving van de gevolgen van deze generalisatie voor diverse formules voor drie dimensies. Er zijn diverse mogelijke redenen aan te voeren voor de beperking van een generalisatie tot drie dimensies. Een reden kan zijn dat de generalisatie naar k dimensies niet nodig is, bijvoorbeeld omdat geen data voorhanden zijn met meer dan drie dimensies. Een andere reden kan zijn dat men verkeert in de veronderstelling dat de generalisatie een trivialiteit betreft. Of het kan zijn dat men denkt dat de sommatienotatie juist te bewerkelijk is om de modellen voor meer dan drie dimensies te beschrijven. De laatste reden is echter te omzeilen door gebruik te maken van notatie die recentelijk is ingevoerd door Kapteyn, Neudecker en Wansbeek (1986), voor een k–dimensionale principale–componenten–analyse (PCA) model.

In de onderhavige studie worden een aantal bekende en minder bekende drie–dimensionale data–analyse–modellen gegeneraliseerd naar k dimensies met behulp van de notatie van Kapteyn, Neudecker en Wansbeek (1986).

In de studie worden specifieke lineair–algebraïsche hulpmiddelen gebruikt voor de behandeling van de meerdimensionale modellen. De hulpmiddelen komen voort uit de matrix–differentiatie–calculus en (0,1)–matrix–algebra. In hoofdstukken 1, 2 en 3 wordt ingegaan op de diverse matrix–algebra–regels. Met name hoofdstuk 3 is gewijd aan verscheidene eigenschappen van diverse (0,1)–matrices, te weten de permutatie matrix P, de commutatie matrix C_i (en D_i) en de blok–mutatie matrix B_i.

In hoofdstuk 4 wordt het k–dimensionale PCA model besproken, grotendeels gebaseerd op Kapteyn, Neudecker en Wansbeek (1986). Het model wordt gebruikt

voor de analyse van vier–dimensionale data uit een advertentie–onderzoek. De schattingsresultaten van twee schattingsmethoden, te weten een kleinste–kwadraten aanpak volgens Kapteyn, Neudecker en Wansbeek (1986) en een kanonieke aanpak volgens Lastovicka (1981), worden vergeleken. Er wordt een interpretatie van de schattingen van de latente componenten volgens beide schattingsmethoden gegeven. Verder wordt beschreven hoe de stochastiek in dit model wordt geïntroduceerd. We beschrijven twee aanpakken om stochastiek te introduceren. We prefereren de aanpak waarbij alle dimensies symmetrisch worden beschouwd boven de aanpak waarbij één dimensie asymmetrisch wordt beschouwd. We beschrijven een meest–aannemelijke schatter waarvan de asymptotische verdeling wordt gegeven.

Hoofdstuk 5 is grotendeels gebaseerd op Wansbeek en Verhees (1986). Dit hoofdstuk behandelt het k–dimensionale faktor–analyse–model als generalisatie van het drie–dimensionale model van Bentler en Lee (1978, 1979). De eerste–orde condities voor meest–aannemelijkheid schatting, gewogen–kleinste–kwadraten schatting, en ongewogen–kleinste–kwadraten schatting worden gespecificeerd. Twee iteratieve oplossingsmethoden voor het vinden van de schatters worden behandeld, en zijn terug te voeren tot methoden die gebruik maken van de informatie–matrix. De informatie–matrix blijkt ook voor het k–dimensionale faktor–analyse–model tot een elegante struktuur gebracht te kunnen worden. De asymptotische eigenschappen van de schatters zijn ook in een elegante uitdrukking te geven, mede dankzij de (0,1)–matrix–algebra en de bovengenoemde notatie. Programmeer–aspecten en simulaties van het model worden kort belicht. De programmeer–aspecten en simulaties tonen aan dat het k–dimensionale model op papier in een elegante vorm is te specificeren, maar dat een feitelijke applikatie bewerkelijker is. De programmeer–aspecten zijn van dien aard dat een apart hoofdstuk (9) is ingeruimd voor een discussie van onze aanpak in computer–programma's voor dit model en de andere modellen. We komen hier later op terug.

Hoofdstuk 6 is gewijd aan het k–dimensionale covariantie–struktuur–analyse (CSA) model als generalisatie van het twee–dimensionale CSA model van Swain (1980). Het hoofdstuk is gebaseerd op Verhees en Wansbeek (1988). Drie van de vier schattingstechnieken zijn onder een noemer te brengen, hetgeen een algemene expressie oplevert voor de schatters, die te vertalen is naar de kleinste–kwadraten–oplossing van een enigszins aangepast regressie–model. De vierde schatter is een non–iteratieve (consistente) gewogen–kleinste–kwadraten

schatter die aanzienlijk minder bewerkelijk is dan de overige drie schatters. De werking van de ontwikkelde computer-programmatuur en de uitkomsten volgens het model worden getoond aan de hand van de schattingsresultaten voor drie datasets en diverse simulaties.

Hoofdstuk 7 introduceert een aantal onderwerpen die verband houden met het regressie-model voor analyse van k-dimensionale data met interdependentie. Het model is geïntroduceerd door Wansbeek en Buyze (1981) voor de mogelijke analyse van inkomenswaardering data. Ook voor dit model worden eerste- en tweede-orde condities afgeleid, alsmede enkele oplossingsmethoden om meest-aannemelijke schatters en gewogen-kleinste-kwadraten schatters te vinden. Onder de aanname dat de ontwikkelde programmatuur voor dit model korrekt is, tonen de simulaties dat er meerdere optima aanwezig kunnen zijn en dat het pad van convergentie zeer gevoelig is voor de keuze van startwaarden.

Hoofdstuk 8 behandelt een k-dimensionaal Poisson-regressie-model voor analyse van meerdimensionale kruisklassifikatie-tabellen. Dit model verschilt van klassieke loglineaire modellen vanwege de Kronecker-produkt-struktuur die wordt opgelegd en de interpretatie van de parameters. Er wordt een meest-aannemelijke schatter en een iteratieve gewogen-kleinste-kwadraten schatter afgeleid. Voor beide schatters wordt de asymptotische verdeling besproken. Verder blijken de eerste-orde en tweede-orde condities in het PCA model veel gelijkenis te tonen met de eerste-orde en tweede-orde condities in het Poisson-regressie-model.

Voor allevijf data-analyse-modellen geldt dat in de praktijk, diverse matrices in de expressies voor de eerste-orde- en tweede-orde-condities grote dimensies kunnen aannemen. Ook is het duidelijk dat er veel struktuur aanwezig is binnen de eerste-orde en tweede-orde condities in de vijf data-analyse-modellen. Die struktuur wordt onder andere bewerkstelligd door de aanwezigheid van Kronecker-producten. Ook de aanwezigheid van vele, schaarsgevulde, (0,1)-matrices is van belang voor de struktuur van de condities.

De grootte van matrices, de aanwezigheid van vele en diverse (0,1)-matrices in de vijf data-analyse-modellen, en de wens om een willekeurige keuze van het aantal dimensies k toe te laten voor empirisch onderzoek van meerdimensionale data, leidt tot de ontwikkeling van procedures

die niet standaard voorkomen in programma bibliotheken. Een calculus op scalair niveau in de procedures laat een willekeurige keuze toe van de grootte van matrices. Rekursiviteit van de procedures laat een willekeurige keuze toe van het aantal dimensies k. De operaties van (0,1)-matrices worden beschreven door functies of geheeltallige vectoren, die op een eenvoudige wijze in de procedures zijn op te nemen. De eenvoudige wijze van opname van deze functies of vectoren, garandeert dat de procedures inzichtelijk en van redelijke eenvoud blijven. Dit is iets wat niet bereikt kan worden met de derde manier van beschrijving van de operatie van (0,1)-matrices. Deze derde manier bestaat uit het toevoegen van nieuwe – en permuteren van de bestaande – 'do-loops' in de procedures. In hoofdstuk 9 wordt deze specifieke programmatuur voor bovenstaande k-dimensionale data-analyse-modellen besproken.

Het onderzoek levert de volgende drie conclusies, die zijn samengevat in het laatste hoofdstuk (10). Ten eerste, de drie-dimensionale data-analyse-modellen zijn op eenvoudige wijze te generaliseren naar k dimensies. Met behulp van de notatie van Kapteyn, Neudecker en Wansbeek (1986), matrix-differentiatie-calculus, en (0,1)-matrices, is getoond dat inzichtelijke expressies voor eerste-orde en tweede-orde condities zijn af te leiden.

Ten tweede, dankzij de eenvoud van deze condities, zijn de asymptotische eigenschappen van de diverse schatters in de diverse modellen op eenvoudige wijze af te leiden. Op papier hebben de asymptotische covariantie-matrices van de schatters een eenvoudige struktuur. Ook blijkt de beschrijving van numerieke-oplossing-technieken die gebruik maken van de condities, op papier, geen problemen op te leveren. De schatters zouden dus, in theorie, op eenvoudige wijze te bepalen moeten zijn met koppeling van procedures in bestaande computer-programma-bibliotheken.

Onze derde conclusie geeft echter aan dat deze laatste uitspraak enigzins dient te worden afgezwakt voor modellen waar grote matrices voorkomen. De derde conclusie luidt dan ook als volgt. Als voor de vijf data-analyse-modellen drie- of meer-dimensionale data gebruikt worden, waarbij dimensie-lengten grote waarden aannemen, dan is programmatuur met een scalaire-niveau-calculus-benadering te verkiezen boven programmatuur met 'rechttoe-rechtaan' declaratie en multiplicatie van matrices.

References

Aaker, D.A. and D.E. Bruzzone (1981), Viewer perceptions of prime-time television advertising. *Journal of Advertising Research*, 21, 15–27.

Aigner, D.J., Hsiao, C., Kapteyn, A. and T.J. Wansbeek (1984), Latent variables models in Econometrics. In: Griliches, Z. and M.D. Intriligator (Eds.), *Handbook of econometrics, Vol. II.*

Aitchison, J. and S.D. Silvey (1958), Maximum likelihood estimation of parameters subject to restraints. *Annals of Mathematics*, 29, 813–823.

Amemiya, T., Fuller, W.A. and S.G. Pantula (1987), The asymptotic distributions of some estimators for a factor analysis model. *Journal of Multivariate Analysis*, 2, 51–64.

Andersen, E.B. (1980), *Discrete statistical models with social science applications.* Amsterdam : North-Holland.

Anderson, T.W. (1958), *An introduction to multivariate statistical analysis.* New York: Wiley and Sons.

Anderson, T.W. (1973), Asymptotically efficient estimation of covariance matrices with linear structure. *The Annals of Statistics*, 1, 135–141.

Anderson, T.W. and H. Rubin (1956), Statistical inference in factor analysis. In: Neyman, J. (Ed.), *Proceedings of the Third Berkely Symposium on Mathematical Statistics and Probability, Vol. V.* Berkely: University of California Press.

Archer, C.O. and R.I. Jennrich (1973), Standard errors for rotated factor loadings. *Psychometrika*, 38, 581–591.

d'Aubigny, G. and E. Polit (1989), Some optimality properties of the generalization of the Tucker method to the analysis of n-way tables with specified metrics. In: Coppi, R. and S. Bolasco (Eds.), *Multiway data analysis.* Amsterdam: North-Holland.

Balestra, P. (1976), *La dérivation matricielle.* Collection de l'Ime, no.12. Paris: Sirey.

Bard, J. (1974), *Nonlinear parameter estimation.* New York: Academic Press.

Bartholomew, D.J. (1985), Foundations of factor analysis: some practical implications. *British Journal of Mathematical and Statistical Psychology*, 38, 1–10.

Bartholomew, D.J. and R.P. McDonald (1986), Foundations of factor analysis: a further comment. *British Journal of Mathematical and Statistical Psychology*, 39, 228–229.

Beale, E.M.L. and R.J.A. Little (1975), Missing values in multivariate analysis. *Journal of the Royal Statistical Society, Series B*, 129–146.

Bekker, P. (1986a), A note on the identification of restricted factor loading matrices. *Psychometrika*, 51, 607–611.

Bekker, P. (1986b), Identification in restricted factor models, and the evaluation of rank conditions. Department of Economics, Groningen University.

Bekker, P. and J. de Leeuw (1987), The rank of reduced dispersion matrices. *Psychometrika*, 52, 125–135.

Bekker, P. and A. Merckens (1988), Computerized evaluation of rank conditions: identification of simultaneous equation models with measurement error.

Bentler, P.M. (1976), Multistructure statistical model applied to factor analysis. *Multivariate Behavorial Research*, 11, 3–25.

Bentler, P.M. (1982), Confirmatory factor analysis via noniterative estimation: a fast, inexpensive method. *Journal of Marketing Research*, 19, 417–424.

Bentler, P.M. (1983), Some contributions to efficient statistics in structural models: specification and estimation of moment structures. *Psychometrika*, 48, 493–517.

Bentler, P.M. and D.G. Bonett (1980), Significance tests and goodness of fit in the analysis of covariance structures. *Psychological Bulletin*, 88, 588–606.

Bentler, P.M. and T.K. Dijkstra (1985), Efficient estimation via linearization in structural models. In: Krishnaiah, P.R. (Ed.), *Multivariate analysis Vol. VI*, 9–42. New York: Academic Press.

Bentler, P.M. and S.–Y. Lee (1978a), Statistical aspects of a three mode factor analysis model. *Psychometrika*, 43, 343–352.

Bentler, P.M. and S.–Y. Lee (1978b), Matrix derivatives with chain rule and rules for simple, Hadamard and Kronecker products. *Journal of Mathematical Psychology*, 7, 255–262.

Bentler, P.M. and S.–Y. Lee (1979), A statistical development of three–mode factor analysis. *British Journal of Mathematical and Statistical Psychology*, 32, 87–104.

Bentler, P.M., Poon, W.–Y. and S.–Y. Lee (1988), Generalized multimode latent variable models: implementation by standard programs. *Computational Statistics and Data Analysis*, 6, 107–118.

Bentler, P.M. and J.S. Tanaka (1983), Problems with EM algorithms for ML factor analysis. *Psychometrika*, 48, 247–251.

Bentler, P.M. and D.G. Weeks (1980), Linear structural equations with latent variables. *Psychometrika*, 45, 289–308.

Bentler, P.M. and D.G. Weeks (1982), Multivariate analysis with latent variables. In: Krishnaiah, P.R. and L.N. Kanal (Eds.), *Handbook of Statistics*, *Vol.* 2, 747–771.

Bieber, S.L. and W. Meredith (1986), Transformation to achieve a longitudinally stationary factor pattern matrix. *Psychometrika*, 51, 537–547.

Bishop, Y.M.M., Fienberg, S.E. and P.W. Holland (1975), *Discrete multivariate analysis. Theory and practice.* Cambridge (Mass.): M.I.T. Press.

Bloxom, B. (1968), A note on invariance in three–mode factor analysis. *Psychometrika*, 33, 347–350.

Bock, R.D. and R.E. Bargmann (1966), Analysis of covariance structures. *Psychometrika*, 31, 507–534.

Bonett, D.G., Bentler, P.M. and J.A. Woodward (1989), Asymptotic results for a conditional Poisson log–linear model. *Statistic and Probability Letters*, 7, 267–270.

Brien, C.J., James, A.T. and W.N. Venables (1988), An analysis of correlation matrices: variables cross–classified by two factors. *Biometrika*, 75, 469–476.

Brown, C.H. (1983), Asymptotic comparison of missing data procedures for estimating factor loadings. *Psychometrika*, 48, 269–291.

Browne, M.W. (1974), Generalized least squares estimators in the analysis of covariance structures. *South African Statistical Journal*, 8, 1–24.

Browne, M.W. (1982), Covariance structures. In: Hawkins, D.M. (Ed.), *Topics in applied multivariate analysis.* London: Cambridge University Press.

Browne, M.W. (1984a), The decomposition of multitrait–multimethod matrices. *The British Journal of Mathematical and Statistical Psychology*, 37, 1–21.

Browne, M.W. (1984b), Asymptotically distribution–free methods for the analysis of covariance structures. *The British Journal of Mathematical and Statistical Psychology*, 37, 62–83.

Browne, M.W. (1987), Robustness of statistical inference in factor analysis and related models. *Biometrika*, 72, 375–384.

Browne, M.W. (1989), Relationships between an additive model and a multiplicative model for multitrait–multimethod matrices. In: Coppi, R. and S. Bolasco (Eds.), *Multiway data analysis*. Amsterdam: North–Holland.

Cameron, A.C. and P.K. Trivedi (1986), Econometric models based on count data: comparisons and applications of some estimators and tests. *Journal of Applied Econometrics*, 1, 29–53.

Campbell, D.T. and D.W. Fiske (1959), Convergent and discriminant validation by the multitrait–multimethod matrix. *Psychological Bulletin*, 56, 81–105.

Campbell, D.T. and E.J. O'Connell (1967), Method factors in multitrait–multimethod matrices: multiplicative rather than additive ? *Multivariate Behavioral Research*, 2, 409–426.

Carroll, J.D. and J.J. Chang (1970), Analysis of individual differences in multidimensional scaling via an N–way generalization of the Eckart–Young decomposition. *Psychometrika*, 35, 283–319.

Carroll, J.D., Pruzansky, S. and J.B. Kruskal (1980), CANDELINC: A general approach to multidimensional analysis of many–way arrays with linear constraints on parameters. *Psychometrika*, 45, 3–23.

Clarke, M.R.B. (1970), A rapidly convergent method for maximum likelihood factor analysis. *British Journal of Mathematical and Statistical Psychology*, 23, 43–52.

Clarkson, D.B. (1979), Estimating the standard errors of the rotated factor loadings by jackknifing. *Psychometrika*, 44, 297–313.

Cramer, J.S. (1986), *Econometric applications of maximum likelihood methods*. New York: Cambridge University Press.

Cudeck, R. (1988), Multiplicative models and MTMM matrices. *Journal of Educational Statistics*, 13, 131–147.

Cudeck, R. and M.W. Browne (1983), Cross–validation of covariance structures. *Multivariate Behavorial Research*, 18, 147–167.

Dahlquist, G., Björk, Å. and N. Anderson (1974), *Numerical methods*. New Jersey: Prentice Hall.

Dahm, P.F. and W.A. Fuller (1986), Generalized least squares estimation of the functional multivariate linear errors–in–variables model. *Journal of Multivariate Analysis*, 19, 132–141.

De Boor, C. (1979), Efficient computer manipulation of tensor products. *ACM Transactions on Mathematical Software*, 5, 173–182.

De Leeuw, J., Keller, W.J. and T.J. Wansbeek (1983), *Interfaces between econometrics and psychometrics*; issue of the *Annals of Applied Econometrics*, supplement to the *Journal of Econometrics*.

De Leeuw, J., Heiser, W., Meulman, J. and F. Critchley (1986), *Multidimensional data analysis*. Leiden: DSWO Press.

Dempster, A.P., Laird, N.M. and D.B. Rubin (1977), Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39, 1–38.

d'Esopo, D.A. (1959), A convex programming procedure. *Naval Research Logistic Quarterly*, 11, 33–42.

Dijkstra, T.K. (1987), Some properties of scale invariant covariance structures. Department of Economics, Groningen University.

Don, F.J.H. (1982), Restrictions on variables. *Journal of Econometrics*, 18, 363–393.

Don, F.J.H. (1985), The use of generalized inverses in restricted maximum likelihood. *Linear Algebra and its Applications*, 70, 225–240.

Don, F.J.H and A.P. van der Plas (1981), A note on the characteristic polynomial of the commutation matrix. *Linear Algebra and its Applications*, 37, 135–142.

Dwyer, P.S. (1967), Some applications of matrix derivatives in multivariate analysis. *Journal of the American Statistical Association*, 62, 607–625.

Dyksen, W.R. (1987), Tensor product generalized ADI methods for separable elliptic problems. *SIAM Journal on Numerical Analysis*, 24, 59–76.

Ehrenberg, A.S.C. (1974), Repetitive advertising and the consumer. *Journal of Advertising Research*, 14, 25–33.

Etezadi-Amoli, J. and R.P. McDonald (1983), A second generation nonlinear factor analysis. *Psychometrika*, 48, 315–342.

Fienberg, S.E. (1970a), An iterative procedure for estimation in contingency tables. *Annals of the Mathematical Statistician*, 41, 907–917.

Fienberg, S.E. (1970b), Quasi-independence and maximum likelihood estimation in incomplete contingency tables. *Journal of the American Statistical Association*, 65, 1610–1616.

Fienberg, S.E. (1972), The analysis of incomplete multiway contingency tables. *Biometrics*, 28, 177–202.

Fienberg, S.E. and M.M. Meyer (1983), Loglinear models and categorical data analysis with psychometric and econometric applications. In: De Leeuw, J., Keller, W.J. and T.J. Wansbeek (Eds.), *Interfaces between Econometrics and Psychometrics*; issue of the *Annals of Applied Econometrics*, supplement to the *Journal of Econometrics*.

Finkbeiner, C. (1979), Estimation for the multiple factor model when data are missing. *Psychometrika*, 44, 409–420.

Flamer, S. (1983), Assessment of the multitrait–multimethod matrix validity of Likert scales via confirmatory factor analysis. *Multivariate Behavorial Research*, 18, 275–308.

Fletcher, R. and M.J.D. Powell (1964), A rapidly convergent descent method for minimization. *The Computer Journal*, 2, 163–168.

Fletcher, R. and C.M. Reeves (1964), Function minimization by conjugate gradients. *The Computer Journal*, 7, 149–153.

Frome, E.L. (1983), The analysis of rates using Poisson regression models. *Biometrics*, 39, 665–674.

Geweke, J.F. and K.J. Singleton (1980), Interpreting the likelihood ratio statistic in factor models when sample size is small. *Journal of the American Statistical Association*, 75, 133–137.

Geweke, J.F. and K.J. Singleton (1981), Maximum likelihood "confirmatory" factor analysis of economic time series. *International Economic Review*, 22 37–54.

Goldfeld, S.M. and R.E. Quandt (1972), *Nonlinear methods in econometrics*. Amsterdam: North–Holland.

Goodman, L.A. (1970), The multivariate analysis of qualitative data. Interactions among multiple classifications. *Journal of the American Statistical Association*, 65, 226–255.

Goodman, L.A. (1978), *Analyzing qualitative/categorical data*. Amsterdam: Addison–Wesley.

Goodman, L.A. (1979a), Multiplicative models for the analysis of occupational mobility tables, and other kinds of cross–classification tables. *American Journal of Sociology*, 84, 804–819.

Goodman, L.A. (1979b), Simple models for the analysis of association in cross–classifications having ordered categories. *Journal of the American Statistical Association*, 74, 537–552.

Goodman, L.A. (1981), Association models and canonical correlation in the analysis of cross–classifications having ordered categories. *Journal of the American Statistical Association*, 76, 320–334.

Graham, A. (1981), *Kronecker products and matrix calculus with applications*. New York: Wiley.

Greenacre, M. (1988), Correspondence analysis of multivariate categorical data by weighted least squares. *Biometrika*, 75, 457–467.

Grizzle, J.E. and O.D. Williams (1972), Log linear models and tests of independence for contingency tables. *Biometrics*, 28, 137–156.

Haberman, S.J. (1974a), *The analysis of frequency data.* Chicago: University of Chicago Press.

Haberman, S.J. (1974b), Log–linear models for frequency tables with ordered classifications. *Biometrics*, 30, 589–600.

Hägglund, G. (1982), Factor analysis by instrumental variables methods. *Psychometrika*, 47, 209–222.

Harman, H.H. (1976), *Modern factor analysis.* Chicago: University of Chicago Press.

Harman, H.H. and W.H. Jones (1966), Factor analysis by minimizing residuals (minres). *Psychometrika*, 31, 351–368.

Harris, C.W. (1963), Canonical factor models for the description of change. In: Harris, C.W. (Ed.), *Problems in measuring change.* Maddison (Wisc.): University of Wisconsin Press.

Harshman, R.A. and M.E. Lundy (1984), The Parafac model for three–way factor analysis and multidimensional scaling. In: Law, H.G., Snyder, C.W., Hattie, J.A. and R.P. McDonald (Eds.), *Research methods for multimode data analysis.* New York: Preager.

Hartwig, R.E. and S.B. Morris (1975), The universal flip matrix and the generalized faro–shuffle. *Pacific Journal of Mathematics*, 50, 445–455.

Hausman, J., Hall, B.H. and Z. Griliches (1984), Econometric models for count data with an application to the patents – R&D relationship. *Econometrica*, 52, 909–938.

Henderson, H.V. and S.R. Searle (1981), The Vec–permutation matrix, the Vec operator and Kronecker products: a review. *Linear and Multilinear Algebra*, 9, 271–288.

Hilton, T.L., Beaton, A.E. and C.P. Bower (1971), *Stability and instability in academic growth: a compilation of longitudinal data.* Report, USOE, Research no. 0–0140. Princeton, N.J.: Educational Testing Service.

Holmquist, B. (1985), The direct product permuting matrices. *Linear and Multilinear Algebra*, 17, 117–141.

Hung, C.–H. and T.L. Markham (1975), The Moore–Penrose inverse of a partitioned matrix. *Linear Algebra and its Applications*, 11, 73–86.

Ihara, M and Y. Kano (1986), A new estimator of the uniqueness in factor analysis. *Psychometrika*, 51, 563–566.

Jennrich, R.I. (1986), A Gauss–Newton algorithm for exploratory factor analysis. *Psychometrika*, 51, 277–284.

Jennrich, R.I. and S.M. Robinson (1969), A Newton–Raphson algorithm for maximum likelihood factor analysis. *Psychometrika*, 34, 111–123.

Jöreskog, K.G. (1967), Some contributions to maximum likelihood factor analysis. *Psychometrika*, 32, 443–482.

Jöreskog, K.G. (1969), A general approach to confirmatory maximum likelihood factor analysis. *Psychometrika*, 34, 183–202.

Jöreskog, K.G. (1971), Simultaneous factor analysis in several populations. *Psychometrika*, 36, 409–426.

Jöreskog, K.G. (1973), Analysis of covariance structures. In: P.R. Krishnaiah (Ed.), *Multivariate analysis, Vol. III*. New York: Academic Press.

Jöreskog, K.G. and A.S. Goldberger (1972), Factor analysis by generalized least squares. *Psychometrika*, 37, 243–260.

Jöreskog, K.G. and D. Sörbom (1977), Statistical models and methods for analysis of longitudinal data. In: Aigner, D.J. and A.S. Goldberger (Eds.), *Latent variables in socio–economic models*. Amsterdam: North–Holland.

Jöreskog, K.G. and D. Sörbom (1981), Lisrel V, analysis of linear relationships by maximum likelihood and least squares methods. Uppsala: Department of Statistics, University of Uppsala.

Kaiser, H.F. (1958), The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23, 187–200.

Kapteyn, A. (1977), *A theory of preference formation*. PhD–thesis, Leiden University.

Kapteyn, A., Neudecker, H. and T.J. Wansbeek (1986), An approach to n–mode components analysis. *Psychometrika*, 51, 269–275.

Kapteyn, A. and T.J. Wansbeek (1982), Identification of the parameters in the factor analysis model. Unpublished CBS report.

Keller, W.J. and T.J. Wansbeek (1983), Multivariate methods for quantitative and qualitative data. In: De Leeuw, J., Keller, W.J. and T.J. Wansbeek (Eds.), *Interfaces between econometrics and psychometrics*; issue of the *Annals of Applied Econometrics*, supplement to the *Journal of Econometrics*.

Kroonenberg, P.M. (1983), *Three–mode principal components analysis*. Leiden: DSWO Press.

Kroonenberg, P.M. and J. de Leeuw (1980), Principal component analysis of three–mode data by means of alternating least squares algorithms. *Psychometrika*, 45, 69–97.

Krugman, H.E. (1972), Why three exposures may be enough. *Journal of Advertising Research*, 12, 11–14.

Krugman, H.E. (1977), Memory without recall, exposure without perception. *Journal of Advertising Research*, 17, 7–12.

Kruskal, J.B. (1977), Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, 18, 95–138.

Kruskal, J.B. (1988), Rank and geometry of three way arrays, and uniqueness theorems. Paper presented at Multiway '88, international meeting on the analysis of multiway data matrices, Rome.

Lastovicka, J.L. (1981), The extension of component analysis to four–mode matrices. *Psychometrika*, 46, 47–57.

Law, H.G., Snyder, C.W., Hattie, J.A. and R.P. McDonald (1984), *Research methods for multimode data analysis*. New York: Preager.

Lawley, D.N. and A.E. Maxwell (1971), *Factor analysis as a statistical method*. London: Butterworth.

Leavitt, C. (1970), A multidimensional set of rating scales for television commercials. *Journal of Applied Psychology*, 54, 427–429.

Lee, S.–Y. (1977), *Some algorithms for covariance structure analysis*. Unpublished doctoral dissertation, University of California, Los Angeles.

Lee, S.–Y. (1980), Estimation of covariance structure models with parameters subject to functional restraints. *Psychometrika*, 45, 309–324.

Lee, S.–Y. (1985), Analysis of covariance and correlation structures. *Computational Statistics and Data Analysis*, 2, 279–295.

Lee, S.–Y. and P.M. Bentler (1980), Some asymptotic properties of constrained generalized least squares estimation in covariance structure models. *South African Statistical Journal*, 14, 121–136.

Lee, S.–Y. and R.I. Jennrich (1979), A study of algorithms for covariance structure analysis with specific comparisons using factor analysis. *Psychometrika*, 44, 99–113.

Lee. S.–Y. and K.–L. Tsui (1982), Covariance structure analysis in several populations. *Psychometrika*, 47, 297–308.

Levin, J. (1965), Three–mode factor analysis. *Psychological Bulletin*, 64, 442–452.

MacRae, E.C. (1974), Matrix derivatives with an application to an adaptive linear decision problem. *The Annals of Statistics*, 2, 337–346.

Magnus, J.R. (1985), On differentiating eigenvalues and eigenvectors. *Econometric Theory*, 1, 179–191.

Magnus, J.R. and H. Neudecker (1979), The commutation matrix: some properties and applications. *The Annals of Statistics*, 7, 381–394.

Magnus, J.R. and H. Neudecker (1980), The elimination matrix: some lemmas and applications. *SIAM Journal on Algebraic and Discrete Methods*, 1, 422–449.

Magnus, J.R. and H. Neudecker (1988), *Matrix differential calculus.* Chichester: Wiley.

Marquardt, D.W. (1963), An algorithm for least squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11, 431–441.

McCullagh, P. (1986), The conditional distribution of goodness–of–fit statistics for discrete data. *Journal of the American Statistical Association*, 81, 104–107.

McCullagh, P. and J.A. Nelder (1983), *Generalized linear models.* New York: Chapman and Hall.

McCulloch, C.E. (1982), Symmetric matrix derivatives with applications. *Journal of the American Statistical Association*, 77, 679–682.

McDonald, R.P. (1979), The simultaneous estimation of factor loadings and scores. *British Journal of Mathematical and Statistical Psychology*, 32, 212–228.

McDonald, R.P. (1982), A note on the investigation of local and global identifiability. *Psychometrika*, 47, 101–103.

McDonald, R.P. (1984), The invariant factors model for multimode data. In: Law, H.G., Snyder, C.W., Hattie, J.A. and R.P. McDonald (Eds.), *Research methods for multimode data analysis.* New York: Preager.

McDonald, R.P. (1986), Describing the elephant: structure and function in multivariate data. *Psychometrika*, 51, 513–534.

Merckens, A., Verhees, J. and T.J. Wansbeek (1988), Generating an interface between the computer programming code and the vec–algebra for multimode data analysis; the dual approach of parameter–converging optimization techniques. Department of Economics, Groningen University.

Mooijaart, A. (1985), Factor analysis for non–normal variables. *Psychometrika*, 50, 323–342.

O'Muircheartaigh, C.A.O. and C. Payne (1977), *The analysis of survey data, Vol. 2: model fitting.* Chichester: Wiley.

Mulaik, S.A. (1972), *The foundations of factor analysis.* New York: McGraw–Hill.

Mulaik, S.A. (1986), Factor analysis and Psychometrika: major developments. *Psychometrika*, 51, 23–33.

Murakami, T. (1983), Quasi three–mode principal component analysis – a method for assessing the factor change. *Behaviormetrika*, 14, 27–48.

Murray, W. (1972), *Numerical methods for unconstrained optimization.* New York: Academic Press.

Nelder, J.A. and R.W.M. Wedderburn (1972), Generalized linear models. *Applications in Statistics*, 24, 259–261.

Neudecker, H. (1969), Some theorems on matrix differentiation with special reference to Kronecker matrix products. *Journal of the American Statistical Association*, 64, 953–963.

Neudecker, H. (1977), On some (0,1) matrices. University of Amsterdam, AE–report 11/77.

Neudecker, H. (1981a), On the matrix formulation of Kaiser's varimax criterion. University of Amsterdam, AE–note N4/81.

Neudecker, H. (1981b), Two germane matrix derivatives: properties and relationships. University of Amsterdam, AE–report 23/81.

Neudecker, H. and T.J. Wansbeek (1983), Some results on commutation matrices, with statistical applications. *The Canadian Journal of Statistics*, 11, 221–231.

Nevels, K. (1986), A direct solution for pairwise rotations in Kaiser's varimax method. *Psychometrika*, 51, 327–329.

Nunnally, J.C. (1978), *Psychometric theory*. New York: McGraw–Hill.

Okamoto, M. and M. Ihara (1983), A new algorithm for the least squares solution in factor analysis. *Psychometrika*, 48, 597–605.

Oldenburger, R. (1940), Higher dimensional determinants. *American Mathematics Monthly*, 47, 25–33.

Oldenburger, R. (1941), Factorability of general symmetric matrices. *Compositio Mathematica*, 7, 223–228.

Olkin, I. (1973), Testing and estimation for structures which are circularly symmetric in blocks. In: Kabe, D.G. and R.P. Gupta (Eds.), *Multivariate statistical inference*. Amsterdam: North–Holland.

Olkin, I. (1985), Estimating a Cholesky decomposition. *Linear Algebra and its Applications*, 67, 201–205.

Pereyra, V. and G. Scherer (1973), Efficient computer manipulation of tensor products with applications to multidimensional approximation. *Mathematics of Computation*, 27, 595–605.

Pollock, D.S.G. (1985a), Matrix derivatives: a conflict of definitions. Paper, Queen Mary College, University of London.

Pollock, D.S.G. (1985b), Tensor products and matrix differential calculus. *Linear Algebra and its Applications*, 67, 169–193.

Rao, C.R. (1973), *Linear statistical inference and its applications*. New York: Wiley.

Rindskopf, D. (1983), Parameterizing inequality constraints on unique variances in linear structural models. *Psychometrika*, 48, 73–83.

Rindskopf, D. (1984), Using phantom and imaginary latent variables to parameterize constraints in linear structural models. *Psychometrika*, 49, 37–47.

Rubin, D.B. and T.H. Szatrowski (1982), Finding maximum likelihood estimates of patterned covariance matrices by the EM algorithm. *Biometrika*, 69, 657–660.

Rubin, D.B. and D.T. Thayer (1982), EM algorithms for ML factor analysis. *Psychometrika*, 47, 69–76.

Rubin, D.B. and D.T. Thayer (1983), More on EM for ML factor analysis. *Psychometrika*, 48, 253–257.

Sands, R. and F.W. Young (1980), Component models for three–way data: an alternating least squares algorithm with optimal scaling features. *Psychometrika*, 45, 39–67.

Schlinger, M.J. (1979), A profile of responses to commercials. *Journal of Advertising Research*, 19, 37–46.

Searle, S.R. (1982), *Matrix algebra useful for statistics*. New York: Wiley.

Shapiro, A. (1982), Rank–reducibility of a symmetric matrix and sampling theory of minimum trace factor analysis. *Psychometrika*, 47, 187–199.

Shapiro, A. (1983), Asymptotic distribution theory in the analysis of covariance structures. *South African Statistical Journal*, 17, 33–81.

Shapiro, A. (1986), Asymptotic theory of overparameterized structural models. *Journal of the American Statistical Association*, 81, 142–149.

Shapiro, A. and M.W. Browne (1983), On the investigation of local identifiability: a counterexample. *Psychometrika*, 48, 303–304.

Sherin, R.J. (1966), A matrix formulation of Kaiser's varimax criterion. *Psychometrika*, 31, 535–538.

O'Sullivan, F., Yandell, B.S. and W.J. Raynor (1986), Automatic smoothing of regression functions in generalized linear models. *Journal of the American Statistical Association*, 81, 96–103.

Swain, A.J. (1975), A class of factor analysis estimation procedures with common asymptotic sampling properties. *Psychometrika*, 40, 315–335.

Swain, A.J. (1980), A direct product model for covariance structure. Unpublished paper. Department of Primary industries, Brisbane, Australia.

Swaminathan, H. (1984), Factor analysis of longitudinal data. In: Law, H.G., Snyder, C.W., Hattie, J.A. and R.P. McDonald (Eds.), *Research methods for multimode data analysis*. New York: Preager.

Ten Berge, J.M.F. (1984), A joint treatment of varimax rotation and the problem of diagonalizing symmetric matrices simultaneously in the least-squares sense. *Psychometrika*, 49, 347–358.

Ten Berge, J.M.F., de Leeuw, J. and P.M. Kroonenberg (1987), Some additional results on principal components analysis of three–mode data by means of alternating least squares. *Psychometrika*, 52, 183–191.

Tenenhaus, M. (1988), Generalized canonical analysis and principal component analysis of three–way data. Paper presented at Multiway '88, international meeting on the analysis of multiway data matrices, Rome.

Tracy, D.S. and P.S. Dwyer (1969), Multivariate maxima and minima with matrix derivatives. *Journal of the American Statistical Association*, 64, 1576–1594.

Tracy, D.S. and K.G. Jinadasa (1987), Matrix derivatives of patterned matrices and statistical applications. In: Puri, M.L., Vilaplana, J.P. and W. Wertz (Eds.), *New perspectives in theoretical and applied statistics*. New York: Wiley.

Tracy, D.S. and R.P. Singh (1972), A new matrix product and its applications in partitioned matrix differentiation. *Statistica Neerlandica*, 26, 143–157.

Tracy, D.S. and R.P. Singh (1975), Some applications of matrix differentiation in the general analysis of covariance structures. *Sankhyā: The Indian Journal of Statistics, Series A*, 37, 269–280.

Tucker, L.R. (1963), Implications of factor analysis of three–way matrices for measurement of change. In: Harris, C.W. (Ed.), *Problems in measuring change*. Madison: University of Wisconsin Press.

Tucker, L.R. (1966), Some mathematical notes on three–mode factor analysis. *Psychometrika*, 31, 279–311.

Verhees, J. (1985), Four–mode principal components analysis: evaluation of solutions from two alternative algorithms. Department of Economics, Groningen University.

Verhees, J. and T.J. Wansbeek (1988), The factorial covariance structure model. Department of Economics, Groningen University.

Vetter, W.J. (1975), Vector structures and solutions of linear matrix equations. *Linear Algebra and its Applications*, 10, 181–188.

Wansbeek, T.J. (1980), *Quantitative effects in panel data modeling*. PhD–thesis, Leiden University.

Wansbeek, T.J. (1988), Vec operator. In: Kotz S., Johnson, N.L. and C.B. Read (Eds.) *Encyclopedia of Statistical Sciences, Vol. 9*. New York: Wiley.

Wansbeek, T.J. and J. Buyze (1981), *Multidimensional matrices and multiplicative interdependence in regression analysis.* Den Haag: CBS (preliminary) report.

Wansbeek, T.J. and J. Verhees (1986), The algebra of multimode factor analysis. To appear in: *The Second Special Issue on Linear Algebra and Statistics (SILAX) of Linear Algebra and its Applications.*

Wansbeek, T.J. and J. Verhees (1988), Analysis of poisson distributed k–way data. Department of Economics, Groningen University.

Wansbeek, T.J. and J. Verhees (1989), Models for multidimensional matrices in econometrics and psychometrics. In: Coppi, R. and S. Bolasco (Eds.), *Multiway data analysis.* Amsterdam: North–Holland.

Wells, W.D., Leavitt, C. and M. McConville (1971), A reaction profile for TV commercials. *Journal of Advertising Research*, 11, 11–17.

Wiens, D.P. (1985), On some pattern–reduction matrices which appear in statistics. *Linear Algebra and its Applications*, 67, 233–258.

Zegers, F.E. and J.M.F. ten Berge (1983), A fast and simple computational method of minimum residual factor analysis. *Multivariate Behavorial Research*, 18, 331–340.

INDEX OF TABLES AND FIGURES

AUTHOR INDEX

J

K

T