

Multi-way Analysis in the Food Industry

Models, Algorithms, and Applications

This monograph was originally written as a Ph. D. thesis (see end of file for original Dutch information printed in the thesis at this page)

MULTI-WAY ANALYSIS IN THE FOOD INDUSTRY

Models, Algorithms & Applications

Rasmus Bro

Chemometrics Group, Food Technology

Department of Dairy and Food Science

Royal Veterinary and Agricultural University

Denmark

Abstract

This thesis describes some of the recent developments in multi-way analysis in the field of chemometrics. Originally, the primary purpose of this work was to test the adequacy of multi-way models in areas related to the food industry. However, during the course of this work, it became obvious that basic research is still called for. Hence, a fair part of the thesis describes methodological developments related to multi-way analysis.

A multi-way calibration model inspired by partial least squares regression is described and applied (N-PLS). Different methods for speeding up algorithms for constrained and unconstrained multi-way models are developed (compression, fast non-negativity constrained least squares regression). Several new constrained least squares regression methods of practical importance are developed (unimodality constrained regression, smoothness constrained regression, the concept of approximate constrained regression). Several models developed in psychometrics that have never been applied to real-world problems are shown to be suitable in different chemical settings. The PARAFAC2 model is suitable for modeling data with factors that shift. This is relevant, for example, for handling retention time shifts in chromatography. The PARATUCK2 model is shown to be a suitable model for many types of data subject to rank-deficiency. A multiplicative model for experimentally designed data is presented which extends the work of Mandel, Gollob, and Hegemann for two-factor experiments to an arbitrary number of factors. A matrix product is introduced which for instance makes it possible to express higher-order PARAFAC models using matrix notation.

Implementations of most algorithms discussed are available in MATLABTM code at <http://newton.foodsci.kvl.dk>. To further facilitate the

understanding of multi-way analysis, this thesis has been written as a sort of tutorial attempting to cover many aspects of multi-way analysis.

The most important aspect of this thesis is not so much the mathematical developments. Rather, the many successful applications in diverse types of problems provide strong evidence of the advantages of multi-way analysis. For instance, the examples of enzymatic activity data and sensory data amply show that multi-way analysis is not solely applicable in spectral analysis – a fact that is still new in chemometrics. In fact, to some degree this thesis shows that the noisier the data, the more will be gained by using a multi-way model as opposed to a traditional two-way multivariate model. With respect to spectral analysis, the application of constrained PARAFAC to fluorescence data obtained directly from sugar manufacturing process samples shows that the uniqueness underlying PARAFAC is not merely useful in simple laboratory-made samples. It can also be used in quite complex situations pertaining to, for instance, process samples.

ACKNOWLEDGMENTS

Most importantly I am grateful to Professor Lars Munck (Royal Veterinary and Agricultural University, Denmark). His enthusiasm and general knowledge is overwhelming and the extent to which he inspires everyone in his vicinity is simply amazing. Without Lars Munck none of my work would have been possible. His many years of industrial and scientific work combined with his critical view of science provides a stimulating environment for the interdisciplinary work in the Chemometrics Group. Specifically he has shown to me the importance of narrowing the gap between technology/industry on one side and science on the other. While industry is typically looking for solutions to real and complicated problems, science is often more interested in generalizing idealized problems of little practical use. Chemometrics and exploratory analysis enables a fruitful exchange of problems, solutions and suggestions between the two different areas.

Secondly, I am most indebted to Professor Age Smilde (University of Amsterdam, The Netherlands) for the kindness and wit he has offered during the past years. Without knowing me he agreed that I could work at his laboratory for two months in 1995. This stay formed the basis for most of my insight into multi-way analysis, and as such he is *the* reason for this thesis. Many e-mails, meetings, beers, and letters from and with Age Smilde have enabled me to grasp, refine and develop my ideas and those of others. While Lars Munck has provided me with an understanding of the phenomenological problems in science and industry and the importance of exploratory analysis, Age Smilde has provided me with the tools that enable me to deal with these problems.

Many other people have contributed significantly to the work presented in this thesis. It is difficult to rank such help, so I have chosen to present these people alphabetically.

Claus Andersson (Royal Veterinary and Agricultural University, Denmark), Sijmen de Jong (Unilever, The Netherlands), Paul Geladi (University of Umeå, Sweden), Richard Harshman (University of Western Ontario, Canada), Peter Henriksen (Royal Veterinary and Agricultural University, Denmark), John Jensen (Danisco Sugar Development Center, Denmark), Henk Kiers (University of Groningen, The Netherlands), Ad

Louwerse (University of Amsterdam, The Netherlands), Harald Martens (The Technical University, Denmark), Magni Martens (Royal Veterinary and Agricultural University, Denmark), Lars Nørgaard (Royal Veterinary and Agricultural University, Denmark), and Nikos Sidiropoulos (University of Virginia) have all been essential for my work during the past years, helping with practical, scientific, technological, and other matters, and making life easier for me.

I thank Professor Lars Munck (Royal Veterinary & Agricultural University, Denmark) for financial support through the Nordic Industrial Foundation Project P93149 and the FØTEK fund.

I thank Claus Andersson, Per Hansen, Hanne Heimdal, Henk Kiers, Magni Martens, Lars Nørgaard, Carsten Ridder, and Age Smilde for data and programs that have been used in this thesis. Finally I sincerely thank Anja Olsen for making the cover of the thesis.

TABLE OF CONTENTS

Abstract	i
Acknowledgments	iii
Table of contents	v
List of figures	xi
List of boxes	xiii
Abbreviations	xiv
Glossary	xv
Mathematical operators and notation	xviii

1. BACKGROUND

1.1 INTRODUCTION	1
1.2 MULTI-WAY ANALYSIS	1
1.3 HOW TO READ THIS THESIS	4

2. MULTI-WAY DATA

2.1 INTRODUCTION	7
2.2 UNFOLDING	10
2.3 RANK OF MULTI-WAY ARRAYS	12

3. MULTI-WAY MODELS

3.1 INTRODUCTION	15
Structure	17
Constraints	18
Uniqueness	18
Sequential and non-sequential models	19
3.2 THE KHATRI-RAO PRODUCT	20

Parallel proportional profiles	20
The Khatri-Rao product	21
3.3 PARAFAC	23
Structural model	23
Uniqueness	25
Related methods	28
3.4 PARAFAC2	33
Structural model	34
Uniqueness	37
3.5 PARATUCK2	37
Structural model	38
Uniqueness	39
Restricted PARATUCK2	40
3.6 TUCKER MODELS	44
Structural model of Tucker3	45
Uniqueness	48
Tucker1 and Tucker2 models	49
Restricted Tucker3 models	50
3.7 MULTILINEAR PARTIAL LEAST SQUARES REGRESSION	51
Structural model	52
Notation for N-PLS models	53
Uniqueness	53
3.8 SUMMARY	54

4. ALGORITHMS

4.1 INTRODUCTION	57
4.2 ALTERNATING LEAST SQUARES	57
4.3 PARAFAC	61
Initializing PARAFAC	62
Using the PARAFAC model on new data	64
Extending the PARAFAC model to higher orders	64
4.4 PARAFAC2	65
Initializing PARAFAC2	67
Using the PARAFAC2 model on new data	67
Extending the PARAFAC2 model to higher orders	68

4.5 PARATUCK2	68
Initializing PARATUCK2	71
Using the PARATUCK2 model on new data	71
Extending the PARATUCK2 model to higher orders	71
4.6 TUCKER MODELS	72
Initializing Tucker3	76
Using the Tucker model on new data	78
Extending the Tucker models to higher orders	78
4.7 MULTILINEAR PARTIAL LEAST SQUARES REGRESSION ...	78
Alternative N-PLS algorithms	83
Using the N-PLS model on new data	84
Extending the PLS model to higher orders	85
4.8 IMPROVING ALTERNATING LEAST SQUARES ALGORITHMS	86
Regularization	87
Compression	88
Line search, extrapolation and relaxation	95
Non-ALS based algorithms	96
4.9 SUMMARY	97

5. VALIDATION

5.1 WHAT IS VALIDATION	99
5.2 PREPROCESSING	101
Centering	102
Scaling	104
Centering data with missing values	106
5.3 WHICH MODEL TO USE	107
Model hierarchy	108
Tucker3 core analysis	110
5.4 NUMBER OF COMPONENTS	110
Rank analysis	111
Split-half analysis	111
Residual analysis	113
Cross-validation	113
Core consistency diagnostic	113
5.5 CHECKING CONVERGENCE	121
5.6 DEGENERACY	122

5.7 ASSESSING UNIQUENESS	124
5.8 INFLUENCE & RESIDUAL ANALYSIS	126
Residuals	127
Model parameters	127
5.9 ASSESSING ROBUSTNESS	128
5.10 FREQUENT PROBLEMS AND QUESTIONS	129
5.11 SUMMARY	132

6. CONSTRAINTS

6.1 INTRODUCTION	135
Definition of constraints	139
Extent of constraints	140
Uniqueness from constraints	140
6.2 CONSTRAINTS	141
Fixed parameters	142
Targets	143
Selectivity	143
Weighted loss function	145
Missing data	146
Non-negativity	148
Inequality	149
Equality	150
Linear constraint	150
Symmetry	151
Monotonicity	151
Unimodality	151
Smoothness	152
Orthogonality	154
Functional constraints	156
Qualitative data	156
6.3 ALTERNATING LEAST SQUARES REVISITED	158
Global formulation	158
Row-wise formulation	159
Column-wise formulation	160
6.4 ALGORITHMS	166

Fixed parameter constrained regression	167
Non-negativity constrained regression	169
Monotone regression	175
Unimodal least squares regression	177
Smoothness constrained regression	181
6.5 SUMMARY	184

7. APPLICATIONS

7.1 INTRODUCTION	185
Exploratory analysis	187
Curve resolution	190
Calibration	191
Analysis of variance	192
7.2 SENSORY ANALYSIS OF BREAD	196
Problem	196
Data	197
Noise reduction	197
Interpretation	199
Prediction	200
Conclusion	203
7.3 COMPARING REGRESSION MODELS (AMINO-N)	204
Problem	204
Data	204
Results	204
Conclusion	206
7.4 RANK-DEFICIENT SPECTRAL FIA DATA	207
Problem	207
Data	207
Structural model	209
Uniqueness of basic FIA model	213
Determining the pure spectra	218
Uniqueness of non-negativity constrained sub-space models ...	221
Improving a model with constraints	222
Second-order calibration	227
Conclusion	227

7.5 EXPLORATORY STUDY OF SUGAR PRODUCTION	230
Problem	230
Data	232
A model of the fluorescence data	235
PARAFAC scores for modeling process parameters and quality .	242
Conclusion	245
7.6 ENZYMATIC ACTIVITY	247
Problem	247
Data	248
Results	249
Conclusion	252
7.7 MODELING CHROMATOGRAPHIC RETENTION TIME SHIFTS	253
Problem	253
Data	253
Results	254
Conclusion	256

8. CONCLUSION

8.1 CONCLUSION	259
8.2 DISCUSSION AND FUTURE WORK	262

APPENDIX

APPENDIX A: MATLAB FILES	265
APPENDIX B: RELEVANT PAPERS BY THE AUTHOR	267

BIBLIOGRAPHY	269
INDEX	285

LIST OF FIGURES

	<i>Page</i>
Figure 1. Graphical representation of three-way array	8
Figure 2. Definition of row, column, tube, and layer	8
Figure 3. Unfolding of three-way array	11
Figure 4. Two-component PARAFAC model	24
Figure 4. Uniqueness of fluorescence excitation-emission model	27
Figure 6. Cross-product array for PARAFAC2	35
Figure 7. The PARATUCK2 model	39
Figure 8. Score plot of rank-deficient fluorescence data	41
Figure 9. Comparing PARAFAC and PARATUCK2 scores	43
Figure 10. Scaling and centering conventions	105
Figure 11. Core consistency – amino acid data	115
Figure 12. Core consistency – bread data	117
Figure 13. Core consistency – sugar data	118
Figure 14. Different approaches for handling missing data	139
Figure 15. Smoothing time series data	154
Figure 16. Smoothing Gaussians	155
Figure 17. Example on unimodal regression	179
Figure 18. Smoothing of noisy data	183
Figure 19. Structure of bread data	197
Figure 20. Score plots – bread data	198
Figure 21. Loading plots – bread data	199
Figure 22. Flow injection system	207
Figure 23. FIA sample data	209
Figure 24. Spectra estimated under equality constraints	216
Figure 25. Pure analyte spectra and time profiles	218
Figure 26. Spectra estimated under non-negativity constraints	222
Figure 27. Spectra subject to non-negativity and equality constraints	223
Figure 28. Using non-negativity, unimodality and equality constraints	226
Figure 29. Fluorescence data from sugar sample	232
Figure 30. Estimated sugar fluorescence emission spectra	236
Figure 31. Comparing estimated emission spectra with pure spectra	237

Figure 32. Scores from PARAFAC fluorescence model	239
Figure 33. Comparing PARAFAC scores with process variables	240
Figure 34. Comparing PARAFAC scores with quality variables	241
Figure 35. Predicting color from fluorescence and process data	244
Figure 36. Structure of experimentally designed enzymatic data	249
Figure 37. Model of enzymatic data	251
Figure 38. Predictions from GEMANOVA and ANOVA	252

LIST OF BOXES

	<i>Page</i>
Box 1. Direct trilinear decomposition versus PARAFAC	31
Box 2. Tucker3 versus PARAFAC and SVD	48
Box 3. A generic ALS algorithm	59
Box 4. Structure of decomposition models	60
Box 5. PARAFAC algorithm	63
Box 6. PARAFAC2 algorithm	66
Box 7. Tucker3 algorithm	74
Box 8. Tri-PLS1 algorithm	82
Box 9. Tri-PLS2 algorithm	83
Box 10. Exact compression	91
Box 11. Non-negativity and weights in compressed spaces	94
Box 12. Effect of centering	103
Box 13. Effect of scaling	106
Box 14. Second-order advantage example	142
Box 15. ALS for row-wise and columns-wise estimation	165
Box 16. NNLS algorithm	170
Box 17. Monotone regression algorithm	176
Box 18. Rationale for using PARAFAC for fluorescence data	189
Box 19. Aspects of GEMANOVA	196
Box 20. Alternative derivation of FIA model	212
Box 21. Avoiding local minima	215
Box 22. Non-negativity for fluorescence data	233

ABBREVIATIONS

ALS	Alternating least squares
ANOVA	Analysis of variance
CANDECOMP	Canonical decomposition
DTD	Direct trilinear decomposition
FIA	Flow injection analysis
FNNLS	Fast non-negativity-constrained least squares regression
GEMANOVA	General multiplicative ANOVA
GRAM	Generalized rank annihilation method
MLR	Multiple linear regression
N-PLS	N-mode or multi-way PLS regression
NIPALS	Nonlinear iterative partial least squares
NIR	Near Infrared
NNLS	Non-negativity constrained least squares regression
PARAFAC	Parallel factor analysis
PCA	Principal component analysis
PLS	Partial least squares regression
PMF2	Positive matrix factorization (two-way)
PMF3	Positive matrix factorization (three-way)
PPO	Polyphenol oxidase
RAFA	Rank annihilation factor analysis
SVD	Singular value decomposition
TLD	Trilinear decomposition
ULSR	Unimodal least squares regression

GLOSSARY

Algebraic structure	Mathematical structure of a model
Component	Factor
Core array	Arises in Tucker models. Equivalent to singular values in SVD, i.e., each element shows the magnitude of the corresponding component and can be used for partitioning variance if components are orthogonal
Dimension	Used here to denote the number of levels in a mode
Dyad	A bilinear factor
Factor	In short, a factor is a rank-one model of an N-way array. E.g., the second score <i>and</i> loading vector of a PCA model is one factor of the PCA model
Feasible solution	A feasible solution is a solution that does not violate any constraints of a model; i.e., no parameters should be negative if non-negativity is required
Fit	Indicates how well the model of the data describes the data. It can be given as the percentage of variation explained or equivalently the sum-of-squares of the errors in the model. Mostly equivalent to the function value of the loss function
Latent variable	Factor
Layer	A submatrix of a three-way array (see Figure 2)
Loading vector	Part of factor referring to a specific (variable-) mode.

If no distinction is made between variables and objects, all parts of a factor referring to a specific mode are called loading vectors

Loss function	The function defining the optimization or goodness criterion of a model. Also called objective function
Mode	A matrix has two modes: the row mode and the column mode, hence the mode is the basic entity building an array. A three-way array thus has three modes
Model	An approximation of a set of data. Here specifically based on a structural model, additional constraints and a loss function
Order	The order of an array is the number of modes; hence a matrix is a second-order array, and a three-way array a third-order array
Profile	Column of a loading or score matrix. Also called loading or score vector
Rank	The minimum number of PARAFAC components necessary to describe an array. For a two-way array this definition reduces to the number of principal components necessary to fit the matrix
Score vector	Part of factor referring to a specific (object) mode
Slab	A layer (submatrix) of a three-way array (Figure 2)
Structural model	The mathematical structure of the model, e.g., the structural model of principal component analysis is bilinear
Triad	A trilinear factor

Tube	In a two-way matrix there are rows and columns. For a three-way array there are correspondingly rows, columns, and tubes as shown in Figure 2
Way	See mode

MATHEMATICAL OPERATORS AND NOTATION

x	Scalar
\mathbf{x}	Vector (column)
\mathbf{X}	Matrix
$\underline{\mathbf{X}}$	Higher-order array
$\underset{\mathbf{x}}{\operatorname{argmin}}(f(\mathbf{x}))$	The argument – \mathbf{x} – that minimizes the value of the function $f(\mathbf{x})$. Note the difference between this and $\min(f(\mathbf{x}))$ which is the minimum function value of $f(\mathbf{x})$.
$\cos(\mathbf{x}, \mathbf{y})$	The cosine of the angle between \mathbf{x} and \mathbf{y}
$\operatorname{cov}(\mathbf{x}, \mathbf{y})$	Covariance of the elements in \mathbf{x} and \mathbf{y}
$\operatorname{diag}(\mathbf{X})$	Vector holding the diagonal of \mathbf{X}
$\max(\mathbf{x})$	The maximum element of \mathbf{x}
$\min(\mathbf{x})$	The minimum element of \mathbf{x}
$\operatorname{rev}(\mathbf{x})$	Reverse of the vector \mathbf{x} , i.e., the vector $[x_1 \ x_2 \ \dots \ x_J]^T$ becomes $[x_J \ \dots \ x_2 \ x_1]^T$
$[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \operatorname{svd}(\mathbf{X}, F)$	Singular value decomposition. The matrix \mathbf{U} will be the first F left singular vectors of \mathbf{X} , and \mathbf{V} the right singular vectors. The diagonal matrix \mathbf{S} holds the first F singular values in its diagonal
$\operatorname{tr}\mathbf{X}$	The trace of \mathbf{X} , i.e., the sum of the diagonal elements of \mathbf{X}
$\operatorname{vec}\mathbf{X}$	The term $\operatorname{vec}\mathbf{X}$ is the vector obtained by stringing out (unfolding) \mathbf{X} column-wise to a column vector (Henderson & Searle 1981). If

$$\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_J],$$

then it holds that

$$\text{vec}\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_J \end{bmatrix}$$

$\mathbf{X} \circ \mathbf{Y}$ The Hadamard or direct product of \mathbf{X} and \mathbf{Y} (Styan 1973). If $\mathbf{M} = \mathbf{X} \circ \mathbf{Y}$, then $m_{ij} = x_{ij}y_{ij}$

$\mathbf{X} \otimes \mathbf{Y}$ The Kronecker tensor product of \mathbf{X} and \mathbf{Y} where \mathbf{X} is of size $I \times J$ is defined

$$\mathbf{X} \otimes \mathbf{Y} = \begin{bmatrix} x_{11}\mathbf{Y} & \cdots & x_{1J}\mathbf{Y} \\ \vdots & & \vdots \\ x_{I1}\mathbf{Y} & \cdots & x_{IJ}\mathbf{Y} \end{bmatrix}$$

$\mathbf{X} | \otimes \mathbf{Y}$ The Khatri-Rao product (page 20). The matrices \mathbf{X} and \mathbf{Y} must have the same number of columns. Then

$$\mathbf{X} | \otimes \mathbf{Y} =$$

$$[\mathbf{x}_1 \otimes \mathbf{y}_1 \quad \mathbf{x}_2 \otimes \mathbf{y}_2 \quad \cdots \quad \mathbf{x}_F \otimes \mathbf{y}_F] =$$

$$\left[\text{vec}(\mathbf{y}_1 \mathbf{x}_1^T) \quad \text{vec}(\mathbf{y}_2 \mathbf{x}_2^T) \quad \cdots \quad \text{vec}(\mathbf{y}_F \mathbf{x}_F^T) \right]$$

\mathbf{X}^+ The Moore-Penrose inverse of \mathbf{X}

$\|\mathbf{X}\|_F^2$ The Frobenius or Euclidian norm of \mathbf{X} , i.e. $\|\mathbf{X}\|_F^2 = \text{tr}(\mathbf{X}^T \mathbf{X})$

CHAPTER 1

BACKGROUND

1.1 INTRODUCTION

The subject of this thesis is multi-way analysis. The problems described mostly stem from the food industry. This is not coincidental as the data analytical problems arising in the food area can be complex. The type of problems range from process analysis, analytical chemistry, sensory analysis, econometrics, logistics etc. The nature of the data arising from these areas can be very different, which tends to complicate the data analysis. The analytical problems are often further complicated by biological and ecological variations. Hence, in dealing with data analysis in the food area it is important to have access to a diverse set of methodologies in order to be able to cope with the problems in a sensible way.

The data analytical techniques covered in this thesis are also applicable in many other areas, as evidenced by many papers of applications in other areas which are emerging in the literature.

1.2 MULTI-WAY ANALYSIS

In standard multivariate data analysis, data are arranged in a two-way structure; a table or a matrix. A typical example is a table in which each row corresponds to a sample and each column to the absorbance at a particular wavelength. The two-way structure explicitly implies that for every sample the absorbance is determined at every wavelength and vice versa. Thus, the data can be indexed by two indices: one defining the sample number and one defining the wavelength number. This arrangement is closely

connected to the techniques subsequently used for analysis of the data (principal component analysis, etc.). However, for a wide variety of data a more appropriate structure would be a three-way table or an array. An example could be a situation where for every sample the fluorescence emission is determined at several wavelengths for several different excitation wavelengths. In this case every data element can be logically indexed by three indices: one identifying the sample number, one the excitation wavelength, and one the emission wavelength. Fluorescence and hyphenated methods like chromatographic data are prime examples of data types that have been successfully exploited using multi-way analysis. Consider also, though, a situation where spectral data are acquired on samples under different chemical or physical circumstances, for example an NIR spectrum measured at several different temperatures (or pH-values, or additive concentrations or other experimental conditions that affect the analytes in different relative proportions) on the same sample. Such data could also be arranged in a three-way structure, indexed by samples, temperature and wavenumber. Clearly, three-way data occur frequently, but are often not recognized as such due to lack of awareness. In the food area the list of multi-way problems is long: sensory analysis (sample \times attribute \times judge), batch data (batch \times time \times variable), time-series analysis (time \times variable \times lag), problems related to analytical chemistry including chromatography (sample \times elution time \times wavelength), spectral data (sample \times emission \times excitation \times decay), storage problems (sample \times variable \times time), etc.

Multi-way analysis is the natural extension of multivariate analysis, when data are arranged in three- or higher way arrays. This in itself provides a justification for multi-way methods, and this thesis will substantiate that multi-way methods provide a logical and advantageous tool in many different situations. The rationales for developing and using multi-way methods are manifold:

- The instrumental development makes it possible to obtain information that more adequately describes the intrinsic multivariate and complex reality. Along with the development on the instrumental side, development on the data analytical side is natural and beneficial. Multi-way

analysis is one such data analytical development.

- Some multi-way model structures are unique. No additional constraints, like orthogonality, are necessary to identify the model. This implicitly means that it is possible to calibrate for analytes in samples of unknown constitution, i.e., estimate the concentration of analytes in a sample where unknown interferences are present. This fact has been known and investigated for quite some time in chemometrics by the use of methods like generalized rank annihilation, direct trilinear decomposition etc. However, from psychometrics and ongoing collaborative research between the area of psychometrics and chemometrics, it is known that the methods used hitherto only hint at the potential of the use of uniqueness for calibration purposes.
- Another aspect of uniqueness is what can be termed computer chromatography. In analogy to ordinary chromatography it is possible in some cases to separate the constituents of a set of samples mathematically, thereby alleviating the use of chromatography and cutting down the consumption of chemicals and time. Curve resolution has been extensively studied in chemometrics, but has seldom taken advantage of the multi-way methodology. Attempts are now in progress trying to merge ideas from these two areas.
- While uniqueness as a concept has long been the driving force for the use of multi-way methods, it is also fruitful to simply view the multi-way models as natural structural bases for certain types of data, e.g., in sensory analysis, spectral analysis, etc. The mere fact that the models are appropriate as a structural basis for the data, implies that using multi-way methods should provide models that are parsimonious, thus robust and interpretable, and hence give better predictions, and better possibilities for exploring the data.

Only in recent years has multi-way data analysis been applied in chemistry. This, despite the fact that most multi-way methods date back to the sixties' and seventies' psychometrics community. In the food industry the hard

science and data of chemistry are combined with data from areas such as process analysis, consumer science, economy, agriculture, etc. Chemistry is one of the underlying keys to an understanding of the relationships between raw products, the manufacturing and the behavior of the consumer. Chemometrics or applied mathematics is the tool for obtaining information in the complex systems.

The work described in this thesis is concerned with three aspects of multi-way analysis. The primary objective is to show successful applications which might give clues to where the methods can be useful. However, as the field of multi-way analysis is still far from mature there is a need for improving the models and algorithms now available. Hence, two other important aspects are the development of new models aimed at handling problems typical of today's scientific work, and better algorithms for the present models. Two secondary aims of this thesis are to provide a sort of tutorial which explains how to use the developed methods, and to make the methods available to a larger audience. This has been accomplished by developing WWW-accessible programs for most of the methods described in the thesis.

It is interesting to develop models and algorithms according to the nature of the data, instead of trying to adjust the data to the nature of the model. In an attempt to be able to state important problems including possibly vague *a priori* knowledge in a concise mathematical frame, much of the work presented here deals with how to develop robust and fast algorithms for expressing common knowledge (e.g. non-negativity of absorbance and concentrations, unimodality of chromatographic profiles) and how to incorporate such restrictions into larger optimization algorithms.

1.3 HOW TO READ THIS THESIS

This thesis can be considered as an introduction or tutorial in advanced multi-way analysis. The reader should be familiar with ordinary two-way multivariate analysis, linear algebra, and basic statistical aspects in order to fully appreciate the thesis. The organization of the thesis is as follows:

Chapter 1: Introduction

Chapter 2: Multi-way data

A description of what characterizes multi-way data as well as a description and definition of some relevant terms used throughout the thesis.

Chapter 3: Multi-way models

This is one of the main chapters since the multi-way models form the basis for all work reported here. Two-way decompositions are often performed using PCA. It may seem that many multi-way decomposition models are described in this chapter, but this is one of the interesting aspects of doing multi-way analysis. There are more possibilities than in traditional two-way analysis. A simple PCA-like decomposition of a data set can take several forms depending on how the decomposition is generalized. Though many models are presented, it is comforting to know that the models PARAFAC, Tucker3, and N-PLS (multilinear PLS) are the ones primarily used, while the rest can be referred to as being more advanced.

Chapter 4: Algorithms

With respect to the application of multi-way methods to real problems, the way in which the models are being fitted is not really interesting. In chemometrics, most models have been explained algorithmically for historical reasons. This, however, is not a fruitful way of explaining a model. First, it leads to identifying the model with the algorithm (e.g. not distinguish between the NIPALS algorithm and the PCA model). Second, it obscures the understanding of the model. Little insight is gained, for example, by knowing that the loading vector of a PCA model is an eigenvector of a cross-product matrix. More insight is gained by realizing that the loading vector defines the latent phenomenon that describes most of the variation in the data. For these reasons the description of the algorithms has been separated from the description of models.

However, algorithms are important. There are few software programs for multi-way analysis available (see Bro et al. 1997), which may make it necessary to implement the algorithms. Another reason why algorithmic aspects are important is the poverty of some multi-way algorithms. While the singular value decomposition or NIPALS algorithm for fitting ordinary two-way PCA models are robust and effective, this is not the case for all

multi-way algorithms. Therefore, numerical aspects have to be considered more carefully.

Chapter 5: Validation

Validation is defined here as ensuring that a suitable model is obtained. This covers many diverse aspects, such as ensuring that the model is correct (e.g. that the least squares estimate is obtained), choosing the appropriate number of components, removing outliers, using proper preprocessing, assessing uniqueness etc. This chapter tries to cover these subjects with focus on the practical use.

Chapter 6: Constraints

Constraints can be used for many reasons. In PCA orthogonality constraints are used simply for identifying the model, while in curve resolution selectivity or non-negativity are used for obtaining unique models that reflect the spectra of pure analytes. In short, constraints can be helpful in obtaining better models.

Chapter 7: Applications

In the last main chapter several applications of most of the models presented in the thesis will be described. Exploratory analysis, curve resolution, calibration, and analysis of variance will be presented with examples from fluorescence spectroscopy, flow injection analysis, sensory analysis, chromatography, and experimental design.

Chapter 8: Conclusion

A conclusion is given to capitalize on the findings of this work as well as pointing out areas that should be considered in future work.

CHAPTER 2

MULTI-WAY DATA

2.1 INTRODUCTION

The definition of multi-way data is simple. Any set of data for which the elements can be arranged as

$$x_{ijk\dots} \quad i=1..I, j=1..J, k=1..K, \dots \quad (1)$$

where the number of indices may vary, is a multi-way array. Only arrays of reals will be considered here. With only one index the array will be a one-way or first-order array – a vector, and with two indices a two-way array – a matrix.

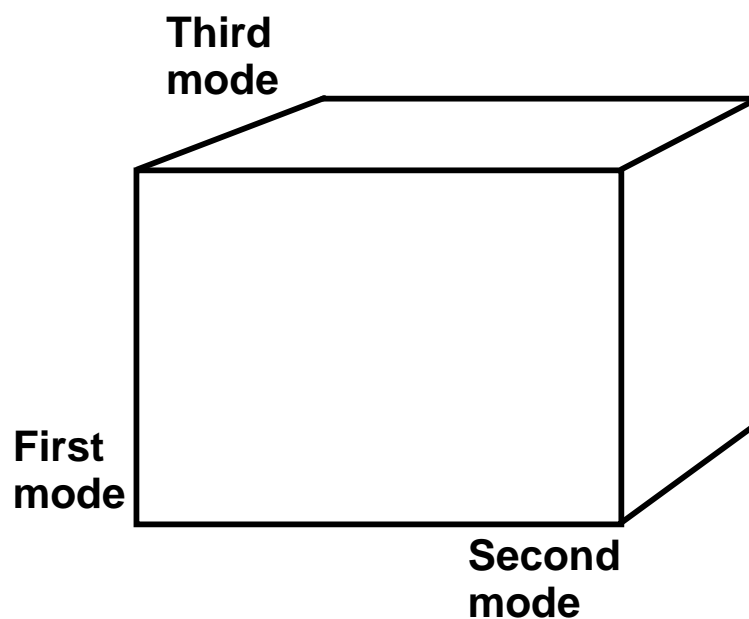


Figure 1. A graphical representation of a three-way data array.

With three indices the data can be geometrically arranged in a box (Figure 1). As for two-way matrices the terms rows and columns are used. Vectors in the third mode will be called tubes (Figure 2). It is also feasible to be able to define submatrices of a three-way array. In Figure 2 a submatrix (gray area) has been obtained by fixing the third mode index. Such a submatrix is usually called a slab, layer, or slice of the array. In this case the slab is called a *frontal* slab as opposed to vertical and horizontal slabs (Kroonenberg 1983, Harshman & Lundy 1984a). In analogy to a matrix each direction in a three-way array is called a way or a mode, and the number of levels in the mode is called the dimension of that mode. In certain contexts a distinction is made between the terms mode and way (Carroll & Arabie 80). The number of ways is the geometrical dimension of the array, while the number of modes are the number of *independent* ways, i.e., a standard variance-covariance matrix is a two-way, one-mode array as the row and column modes are equivalent.

Even though some people advocate for using either tensor (Burdick 1995, Sanchez & Kowalski 1988) or array algebra (Leurgans & Ross 1992) for notation of multi-way data and models, this has not been pursued here. Using such notation is considered both overkill and prohibitive for spreading the use of multi-way analysis to areas of applied research. Instead standard matrix notation will be used with some convenient additions.

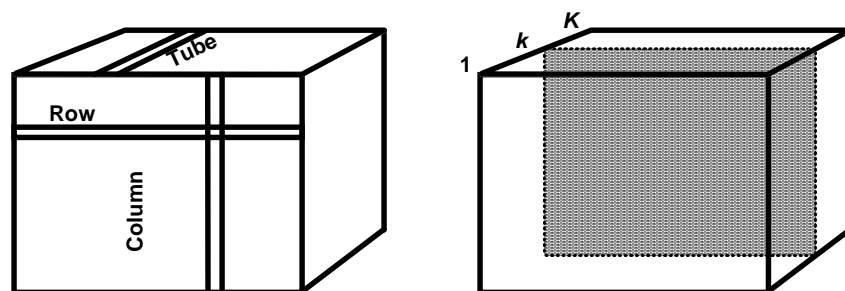


Figure 2. Definition of row, column, tube (left), and the k 'th frontal slab (right).

Scalars are designated using lowercase italics, e.g., x , vectors are generally interpreted as column vectors and designated using bold

lowercase, \mathbf{x} . Matrices are shown in bold uppercase, \mathbf{X} , and all higher-way arrays are shown as bold underlined capitals, $\underline{\mathbf{X}}$. The characters I , J , and K are reserved for indicating the dimension of an array. Mostly, a two-way array – a matrix – will be assumed to be of size $I \times J$, while a three-way array will be assumed to be of size $I \times J \times K$. The lowercase letters corresponding to the dimension will be used to designate specific elements of an array. For example, x_{ij} is the element in the i th row and j th column of the matrix \mathbf{X} . That \mathbf{X} is a matrix follows explicitly from x_{ij} having two indices.

To define sub-arrays two types of notation will be used: A simple intuitive notation will most often be used for brevity, but in some cases a more stringent and flexible method is necessary. Considering first the stringent notation. Given an array, say a three-way array, $\underline{\mathbf{X}}$ of size $I \times J \times K$, any subarray can be denoted by using appropriate indices. The indices are given as a subscript of generic form (i,j,k) where the first number defines the variables of the first mode etc. For signifying a (sub-) set of variables write " $k:m$ " or simply ":" if all elements in the mode are included. For example, the vector obtained from $\underline{\mathbf{X}}$ by fixing the second mode at the fourth variable and the third mode at the tenth variable is designated $\underline{\mathbf{X}}_{(:,4,10)}$. The $J \times K$ matrix obtained by fixing the first mode at the i th variable is called $\underline{\mathbf{X}}_{(i,:)}$.

This notation is flexible, but also tends to get clumsy for larger expressions. Another simpler approach is therefore extensively used when possible. The index will mostly show which mode is considered, i.e., as matrices are normally considered to be of size $I \times J$ and three-way arrays of size $I \times J \times K$, an index i will refer to a row-mode, an index j will refer to column-mode, and an index k to a tube-mode. The j th column of the matrix \mathbf{X} is therefore called \mathbf{x}_j . The matrix \mathbf{X}_k is the k th frontal slab of size $I \times J$ of the three-way array $\underline{\mathbf{X}}$ as shown in Figure 2. The matrix \mathbf{X}_i is likewise defined as a $J \times K$ horizontal slab of $\underline{\mathbf{X}}$ and \mathbf{X}_j is the j th $I \times K$ vertical slab of $\underline{\mathbf{X}}$.

The use of unfolded arrays (see next paragraph) is helpful for expressing models using algebra notation. Mostly the $I \times J \times K$ array is unfolded to an $I \times JK$ matrix (Figure 3), but when this is not the case, or the arrangement of the unfolded array may be dubious a superscript is used for defining the arrangement. For example, an array, $\underline{\mathbf{X}}$, unfolded to an $I \times JK$

matrix will be called $\mathbf{X}^{(I \times JK)}$.

The terms component, factor and latent variable will be used interchangeably for a rank-one model of some sort and vectors of a component referring to one specific mode will be called loading or score vectors depending on whether the mode refers to objects or variables. Loading and score vectors will also occasionally be called profiles.

2.2 UNFOLDING

Unfolding is an important concept in multi-way analysis¹. It is simply a way of rearranging a multi-way array to a matrix, and in that respect not very complicated. In Figure 3 the principle of unfolding is illustrated graphically for a three-way array showing one possible unfolding. Unfolding is accomplished by concatenating matrices for different levels of, e.g., the third mode next to each other. Notice, that the column-dimension of the generated matrix becomes quite large in the mode consisting of two prior modes. This is because the variables of the original modes are combined. There is not one new variable referring to one original variable, but rather a set of variables.

In certain software programs and in certain algorithms, data are rearranged to a matrix for computational reasons. This should be seen more as a practical way of handling the data in the computer, than a way of understanding the data. The profound effect of unfolding occurs when the multi-way structure of the data is ignored, and the data treated as an ordinary two-way data set. As will be shown throughout this thesis, the principle of unfolding can lead to models that are

- less robust
- less interpretable
- less predictive
- nonparsimonious

¹. Note that the term unfolding as used here should not be confused with the meaning of unfolding in psychometrics. Unfolding as used here is also known as reorganization, concatenation or augmentation.

To what extent these claims hold will be substantiated by practical examples. The main conclusion is that these claims generally hold for arrays which can be approximated by multi-way structures and the noisier the data are, the more beneficial it will be to use the multi-way structure. That the data can be approximated by a multi-way structure is somewhat vague.

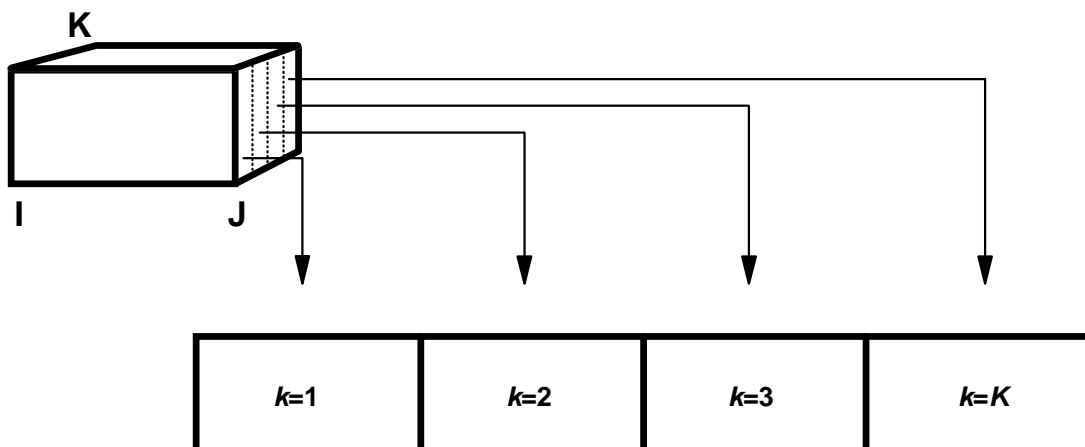


Figure 3. The principle of unfolding. Here the first mode is preserved while the second and third are confounded, i.e., the left-most matrix in the unfolded array is the $I \times J$ matrix equal to the first frontal slab ($k = 1$).

An easy way to assess initially if this is so is based on the following. For a specific three-way problem, consider a hypothetical two-way matrix consisting of typical data with rows and columns equal to the first and second mode of the three-way array. E.g., if the array is structured as samples \times wavelengths \times elution times, then consider a matrix with samples in the row mode and wavelengths in the column mode. Such a matrix could be adequately modeled by a bilinear model. Consider next a typical matrix of modes one and three (samples \times elution times) as well as mode two and three (wavelengths \times elution times). If all these hypothetical two-way problems are adequately modeled by a bilinear model, then likely, a three-way model will be suitable for modeling the three-way data. Though the problem of deciding which model to use is complicated, this rule of thumb does provide rough means for assessing the appropriateness of multi-way models for a specific problem. For image analysis for example, it is easily concluded that multi-way analysis is not the most suitable

approach if two of the modes are constituted by the coordinates of the picture. Even though the singular value decomposition and methods alike have been used for describing and compressing single pictures other types of analysis are often more useful.

2.3 RANK OF MULTI-WAY ARRAYS

An issue that is quite astonishing at first is the rank of multi-way arrays. Little is known in detail but Kruskal (1977a & 1989), ten Berge et al. (1988), ten Berge (1991) and ten Berge & Kiers (1998) have worked on this issue. A 2×2 matrix has maximal rank two. In other words: Any 2×2 matrix can be expressed as a sum of two rank-one matrices, two principal components for example. A rank-one matrix can be written as the outer product of two vectors (a score and a loading vector). Such a component is called a *dyad*. A triad is the trilinear equivalent to a dyad, namely a trilinear (PARAFAC) component, i.e. an 'outer' product of three vectors. The rank of a three-way array is equal to the minimal number of triads necessary to describe the array. For a $2 \times 2 \times 2$ array the maximal rank is three. This means that there exist $2 \times 2 \times 2$ arrays which cannot be described using only two components. An example can be seen in ten Berge et al. (1988). For a $3 \times 3 \times 3$ array the maximal rank is five (see for example Kruskal 1989). These results may seem surprising, but are due to the special structure of the multilinear model compared to the bilinear.

Furthermore Kruskal has shown that if for example $2 \times 2 \times 2$ arrays are generated randomly from any reasonable distribution the volumes or probabilities of the array being of rank two or three are both positive. This as opposed to two-way matrices where only the full-rank case occurs with positive probability.

The practical implication of these facts is yet to be seen, but the rank of an array might have importance if a multi-way array is to be created in a parsimonious way, yet still with sufficient dimensions to describe the phenomena under investigation. It is already known, that unique decompositions can be obtained even for arrays where the rank exceeds any of the dimensions of the different modes. It has been reported that a ten factor model was uniquely determined from an $8 \times 8 \times 8$ array (Harshman 1970, Kruskal 1976, Harshman & Lundy 1984a). This shows that small arrays

might contain sufficient information for quite complex problems, specifically that the three-way decomposition is capable of withdrawing more information from data than two-way PCA. Unfortunately, there are no explicit rules for determining the maximal rank of arrays in general, except for the two-way case and some simple three-way arrays.

CHAPTER 3

MULTI-WAY MODELS

3.1 INTRODUCTION

In this chapter several old and new multi-way models will be described. It is appropriate first to elaborate a little on what a model is. The term model is not used here in the same sense as in classical statistics.

A model is an approximation of a data set, i.e., the matrix $\hat{\mathbf{X}}$ is a model of the data held in the matrix \mathbf{X} . When a name is given to the model, e.g., a PCA model, this model has the distinct properties defined by the *model specification* intrinsic to PCA. These are the structural basis, the constraints, and the loss function. The PCA model has the distinct *structural basis* or parameterization, that $\hat{\mathbf{X}}$ is bilinear

$$\hat{\mathbf{X}} = \mathbf{AB}^T. \tag{2}$$

The parameters of a model are sometimes estimated under certain *constraints* or restrictions. In this case the following constraints apply: $\mathbf{A}^T\mathbf{A} = \mathbf{D}$ and $\mathbf{B}^T\mathbf{B} = \mathbf{I}$, where \mathbf{D} is a diagonal matrix and \mathbf{I} an identity matrix. Finally an intrinsic part of the model specification is the *loss function* defining the goodness of the approximation as well as serving as the objective function for the *algorithm* used for estimating the parameters of the model. The choice of loss function is normally based on assumptions regarding the residuals of the model. In this thesis only least squares loss functions will be considered. These are optimal for symmetrically distributed homoscedastic noise. For non-homoscedastic or correlated noise the loss

function can be changed accordingly by using a weighted loss function. For noise structures that are very non-symmetrically distributed other loss functions than least squares functions are relevant (see e.g. Sidiropoulos & Bro 1998). The PCA model can thus be defined

MODEL PCA

Given \mathbf{X} ($I \times J$) and the column-dimension F of \mathbf{A} and \mathbf{B} fit the model

$$\hat{\mathbf{X}} = \mathbf{A}\mathbf{B}^T$$

as the solution of

$$\min_{\mathbf{A}, \mathbf{B}} \|\mathbf{X} - \mathbf{A}\mathbf{B}^T\|_F^2 \mid \mathbf{A}^T\mathbf{A} = \mathbf{D}, \mathbf{B}^T\mathbf{B} = \mathbf{I}$$

where \mathbf{D} is a diagonal matrix and \mathbf{I} an identity matrix.

The specific scaling and ordering of \mathbf{A} and \mathbf{B} may vary, but this is not essential here. Using PCA as an example the above can be summarized as

MODEL OF DATA	$\hat{\mathbf{X}}$
STRUCTURAL BASIS	$\hat{\mathbf{X}} = \mathbf{A}\mathbf{B}^T$
CONSTRAINTS	$\mathbf{A}^T\mathbf{A} = \mathbf{D}, \mathbf{B}^T\mathbf{B} = \mathbf{I}$
LOSS FUNCTION	$\ \mathbf{X} - \mathbf{A}\mathbf{B}^T\ _F^2$
MODEL SPECIFICATION	$\min_{\mathbf{A}, \mathbf{B}} \ \mathbf{X} - \mathbf{A}\mathbf{B}^T\ _F^2 \mid \mathbf{A}^T\mathbf{A} = \mathbf{D}, \mathbf{B}^T\mathbf{B} = \mathbf{I}$

Note that, e.g., a PCA model and an unconstrained bilinear model $\hat{\mathbf{X}}$ of the same data will have the same structure and give the exact same model of the data, i.e., the structural basis and the loss function will be identical. Regardless, the models are not identical as the PCA model has additional

constraints on the parameters, that will not be fulfilled by an unconstrained bilinear model in general.

Analyzing data it is important to choose an appropriate structural basis and appropriate constraints. A poor choice of either structure or constraints can grossly impact the results of the analysis. A main issue here is the problem of choosing structure and constraints based on *a priori* knowledge, exploratory analysis of the data, and the goal of the analysis. Several decomposition and calibration methods will be explained in the following. Most of the models will be described using three-way data and models as an example, but it will be shown that the models extend themselves easily to higher orders as well.

STRUCTURE

All model structures discussed in this thesis are conditionally linear; fixing all but one set of parameters yields a model linear in the non-fixed parameters. For some data this multilinearity can be related directly to the process generating the data. Properly preprocessed and well-behaved spectral data is an obvious example of data, where a multilinear model can often be regarded as a good approximate model of the true underlying latent phenomena. The parameters of the model can henceforth be interpreted very directly in terms of these underlying phenomena. For other types of data there is no or little theory with respect to how the data are basically generated. Process or sensory data can exemplify this. Even though multilinear models of such data cannot be directly related to an *a priori* theory of the nature of the data, the models can often be useful due to their approximation properties.

In the first case multilinear decomposition can be seen as curve resolution in a broad sense, while in the latter the decomposition model acts as a feature extraction or compression method, helping to overcome problems of redundancy and noise. This is helpful both from a numerical and an interpretational viewpoint. Note that when it is sometimes mentioned that a structural model is theoretically true, this is a simplified way of saying, that some theory states that the model describes how the data are generated. Mostly such theory is based on a number of assumptions, that are often 'forgotten'. Beer's law stating that the absorbance of an analyte

is directly proportional to the concentration of the analyte only holds for diluted solutions, and even there deviations are expected (Ewing 1985). As such there is no provision for talking about *the* VIS-spectrum of an analyte, as there is no single spectrum of an analyte. It depends on temperature, dilution etc. However, in practical data analysis the interest is not in the everlasting truth incorporating all detailed facets of the problem at hand. For identification, for example, an approximate description of the archetype spectrum of the analyte is sufficient for identifying the analyte. The important thing to remember is that models, be they theoretically based or not, are approximations or maps of the reality. This is what makes them so useful, because it is possible to focus on different aspects of the data without having to include irrelevant features.

CONSTRAINTS

Constraints can be applied for several reasons: for identifying the model, or for ensuring that the model parameters make sense, i.e., conform to a *priori* knowledge. Orthogonality constraints in PCA are applied for identifying the model, while non-negativity constraints are applied because the underlying parameters are known not to be negative.

If the latent variables are assumed to be positive, a decomposition can be made using non-negativity constraints. If this assumption, however, is invalid the resulting latent variables may be misleading as they have been forced to comply with the non-negativity constraint. It is therefore important to have tools for judging if the constraint is likely to be valid. There is no general guideline of how to choose structure and constraints. Individual problems require individual solutions. Constraints are treated in detail in chapter 6.

UNIQUENESS

Uniqueness is an important issue in multi-way analysis. That a structural model is unique means that no additional constraints are necessary to identify the model. A two-way bilinear model is not unique, as there is an infinity of different solutions giving the exact same fit to the data. The rotational freedom of the model means that only after constraining the solution to, e.g., orthogonality as in PCA the model is uniquely defined. For

a unique structural model the parameters cannot be changed without changing the fit of the model. The only nonuniqueness that remains in a unique multilinear model is the trivial scaling and permutations of factors that are allowed corresponding for example to the arbitrariness of whether to normalize either scores or loadings in a two-way PCA model, or to term component two number one. The latter indeterminacy is avoided in PCA by ordering the components according to variance explained and can also be avoided in multilinear models in a similar way. If the fitted model cannot be changed (loadings rotated) then there is only one solution giving minimal loss function value. Assuming that the model is adequate for the data and that the signal-to-noise ratio is reasonable it must be plausible to assume that the parameters of the true underlying phenomena will also provide the best possible fit to the data. Therefore if the model is correctly specified the estimated parameters can be estimates of the true underlying parameters (hence parsimonious and hence interpretable).

SEQUENTIAL AND NON-SEQUENTIAL ALGORITHMS

Another concept of practical importance in multi-way analysis is whether a model can be calculated sequentially or not. If a model can be fitted sequentially it means that the $F-1$ component model is a subset of the F component model. Two-way PCA and PLS models can be fitted sequentially. This property is helpful when several models are being tested, as any higher number of components can be estimated from a solution with a lower number of components. Unfortunately most multi-way models do not have the property that they can be fitted sequentially, the only exceptions being N-PLS and Tucker1. ■

The remainder of the chapter is organized as follows. A matrix product that eases the notation of some models will be introduced. Then four decomposition models will be presented. First, the PARAFAC model is introduced. This is the simplest multi-way model, in that it uses the fewest number of parameters. Then a modification of this model called PARAFAC2 is described. It maintains most of the attractive features of the PARAFAC model, but is less restrictive. The PARATUCK2 model is described next. In its standard form, no applications have yet been seen, but a slightly

restricted version of the PARATUCK2 model is suitable as structural basis for so-called rank-deficient data. The last decomposition model is the Tucker model, which can be divided into the Tucker1, Tucker2, and Tucker3 models. These models are mainly used for exploratory purposes, but have some interesting and attractive features. Besides these four decomposition models, an extension of PLS regression to multi-way data is described.

3.2 THE KHATRI-RAO PRODUCT

It will be feasible in the following to introduce a little known matrix product. The use of Kronecker products makes it possible to express multi-way models like the Tucker3 model in matrix notation. However, models such as PARAFAC, cannot easily be expressed in a likewise simple manner. The Khatri-Rao product makes this possible.

PARALLEL PROPORTIONAL PROFILES

The PARAFAC model is intrinsically related to the principle of *parallel proportional profiles* introduced by Cattell (1944). This principle states that the same set of profiles or loading vectors describing the variation in more than one two-way data set, only in different proportions or with different weights, will lead to a model that is not subject to rotational freedom. In Cattell (1944) it is extensively argued that this principle is the most fundamental property for obtaining meaningful decompositions. For instance, suppose that the matrix \mathbf{X}_1 can be adequately modeled as \mathbf{AB}^T , where the number of columns in \mathbf{A} and \mathbf{B} is, say, two. This model can also be formulated as

$$\mathbf{X}_1 = \mathbf{a}_1 \mathbf{b}_1^T c_{11} + \mathbf{a}_2 \mathbf{b}_2^T c_{12} \quad (3)$$

where c_{11} and c_{12} are both equal to one. Suppose now that another matrix \mathbf{X}_2 can also be described by the same set of scores and loading vectors only in different proportions:

$$\mathbf{X}_2 = \mathbf{a}_1 \mathbf{b}_1^T c_{21} + \mathbf{a}_2 \mathbf{b}_2^T c_{22} \quad (4)$$

where c_{21} and c_{22} are not in general equal to c_{11} and c_{12} . The two models consist of the same (parallel) profiles only in different proportions, and one way of stating the combined model is as

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k \mathbf{B}^T, \quad (5)$$

where \mathbf{D}_k is a diagonal matrix with the element of the k th row of \mathbf{C} with typical element c_{kf} in its diagonal. This in essence is the principle of parallel proportional profiles. Cattell (1944) was the first to show that the presence of parallel proportional profiles would lead to an unambiguous decomposition.

THE KHATRI-RAO PRODUCT

Even though the PARAFAC model can be expressed in various ways as shown in the next paragraph, it seldom leads to a very transparent formulation of the model. In the two-matrix example above the formulation

$$\mathbf{X}_1 = \mathbf{a}_1 \mathbf{b}_1^T c_{11} + \mathbf{a}_2 \mathbf{b}_2^T c_{12},$$

and

$$\mathbf{X}_2 = \mathbf{a}_1 \mathbf{b}_1^T c_{21} + \mathbf{a}_2 \mathbf{b}_2^T c_{22},$$

can also be expressed in terms of the unfolded three-way matrix $\mathbf{X} = [\mathbf{X}_1 \ \mathbf{X}_2]$ as

$$\mathbf{X} = [\mathbf{a}_1 \ \mathbf{a}_2] \begin{bmatrix} \mathbf{b}_1 c_{11} & \mathbf{b}_2 c_{12} \\ \mathbf{b}_1 c_{21} & \mathbf{b}_2 c_{22} \end{bmatrix}^T = [\mathbf{a}_1 \ \mathbf{a}_2] \left[\text{vec}(\mathbf{b}_1 \mathbf{c}_1^T) \ \text{vec}(\mathbf{b}_2 \mathbf{c}_2^T) \right]^T, \quad (6)$$

with $\mathbf{c}_1 = [c_{11} \ c_{21}]^T$ and $\mathbf{c}_2 = [c_{12} \ c_{22}]^T$.

Define the Khatri-Rao product (McDonald 1980, Rao & Mitra 1971) of two matrices with the same number of columns, F , as

$$\mathbf{C} \otimes \mathbf{B} \equiv \left[\text{vec}(\mathbf{b}_1 \mathbf{c}_1^T) \text{vec}(\mathbf{b}_2 \mathbf{c}_2^T) \dots \text{vec}(\mathbf{b}_F \mathbf{c}_F^T) \right] = [\mathbf{c}_1 \otimes \mathbf{b}_1 \ \mathbf{c}_2 \otimes \mathbf{b}_2 \ \dots \ \mathbf{c}_F \otimes \mathbf{b}_F]. \quad (7)$$

The Khatri-Rao product then makes it possible to specify, e.g., the PARAFAC model as

$$\mathbf{X} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T. \quad (8)$$

In the following it will be evident that the Khatri-Rao product makes model specification easier and more transparent, especially for higher-order PARAFAC models. E.g., a four-way PARAFAC model can simply be written

$$\mathbf{X}^{(I \times J \times K \times L)} = \mathbf{A}(\mathbf{D} \otimes \mathbf{C} \otimes \mathbf{B})^T, \quad (9)$$

where \mathbf{D} is the fourth mode loading matrix. The Khatri-Rao product has several nice properties, but only a few important ones will be mentioned here. The Khatri-Rao product has the associative property that

$$(\mathbf{D} \otimes \mathbf{C}) \otimes \mathbf{B} = \mathbf{D} \otimes (\mathbf{C} \otimes \mathbf{B}). \quad (10)$$

Also like regular matrix multiplication, the distributive property is retained

$$(\mathbf{B} + \mathbf{C}) \otimes \mathbf{D} = \mathbf{B} \otimes \mathbf{D} + \mathbf{C} \otimes \mathbf{D}. \quad (11)$$

Finally, it will be relevant to know the following property of the Khatri-Rao product

$$(\mathbf{A} \otimes \mathbf{B})^T (\mathbf{A} \otimes \mathbf{B}) = (\mathbf{A}^T \mathbf{A}) \circ (\mathbf{B}^T \mathbf{B}), \quad (12)$$

where the operator, \circ , is the Hadamard product. The proofs for the above relations are easy to give and are left for the reader.

3.3 PARAFAC

PARAFAC (PARAllel FACtor analysis) is a decomposition method, which can be compared to bilinear PCA, or rather it is *one* generalization of bilinear PCA, while the Tucker3 decomposition (page 44) is another generalization of PCA to higher orders (Harshman & Berenbaum 1981, Burdick 1995). The PARAFAC model was independently proposed by Harshman (1970) and by Carroll & Chang (1970) who named the model CANDECOMP (CANonical DECOMPosition). In case of a three-way analysis a decomposition of the data is made into triads or trilinear components. Instead of one score vector and one loading vector as in bilinear PCA, each component consists of one score vector and two loading vectors. It is common three-way practice not to distinguish between scores and loadings as these are treated equally numerically. The important difference between PCA and PARAFAC is that in PARAFAC there is no need for requiring orthogonality to identify the model.

STRUCTURAL MODEL

The structural model behind two-way principal component analysis is a bilinear model

$$\hat{x}_{ij} = \sum_{f=1}^F a_{if} b_{jf} . \quad (13)$$

Likewise a PARAFAC model of a three-way array is given by three loading matrices, **A**, **B**, and **C** with typical elements a_{if} , b_{jf} , and c_{kf} . The PARAFAC model is defined by the structural model

$$\hat{x}_{ijk} = \sum_{f=1}^F a_{if} b_{jf} c_{kf} . \quad (14)$$

Using the Kronecker product the structural model can also be written

$$\hat{\mathbf{X}}^{(I \times JK)} = \sum_{f=1}^F \mathbf{a}_f (\mathbf{c}_f^T \otimes \mathbf{b}_f^T), \quad (15)$$

where $\hat{\mathbf{X}}^{(I \times JK)}$ is the three-way model unfolded to an $I \times JK$ matrix. This corresponds quite closely to specifying the PCA model as

$$\hat{\mathbf{X}} = \sum_{f=1}^F \mathbf{a}_f \mathbf{b}_f^T.$$

Here \mathbf{a}_f , \mathbf{b}_f , and \mathbf{c}_f are the f th columns of the loading matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} respectively. In matrix notation the PARAFAC model is normally written

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k \mathbf{B}^T + \mathbf{E}_k, \quad k = 1, \dots, K \quad (16)$$

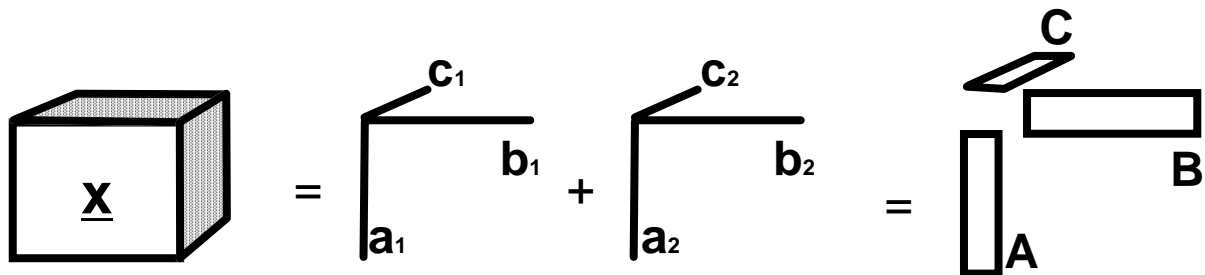


Figure 4. A two-component three-way PARAFAC model of the three-way array \mathbf{X} (residuals omitted for brevity). The vector and matrix products are equivalent to ordinary outer products, i.e., the first component given by vectors \mathbf{a}_1 , \mathbf{b}_1 , and \mathbf{c}_1 , gives a rank-one part of the model of the same size as \mathbf{X} , each element being a triple product $a_{i1}b_{j1}c_{k1}$.

where \mathbf{D}_k is a diagonal matrix holding the k th row of \mathbf{C} in its diagonal and \mathbf{E}_k is a matrix of residuals. The matrix expression (16) of the PARAFAC model seems to indicate that there is a difference in how the different modes are treated in the model. This is not the case. The PARAFAC model is symmetric in the sense that if the array is reordered so that, e.g., the first

mode becomes the third and vice versa, then the PARAFAC models of the two data sets will be identical, except that the first and third mode loadings have been interchanged. The expression $\mathbf{A}\mathbf{D}_k\mathbf{B}^T$ does shed some light on how the third mode loadings can be interpreted, specifically on how the PARAFAC model is related to the principle of parallel proportional profiles (page 20). Let $\mathbf{S}_k = \mathbf{A}\mathbf{D}_k$, then $\mathbf{X}_k = \mathbf{S}_k\mathbf{B}^T + \mathbf{E}_k$, i.e., for each variable k a bilinear model is obtained of \mathbf{X}_k and for different k s the models will be identical except that the individual components are weighted differently through the matrix \mathbf{D}_k . In Figure 4 a two-component model of a three-way array \mathbf{X} is illustrated graphically.

Instead of the matrix notation in equation 16 another simplifying matrix formulation is possible by the introduction of the Khatri-Rao product (page 20). Using this, the PARAFAC model can be formulated in terms of the unfolded array as

$$\mathbf{X}^{(I \times JK)} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}^{(I \times JK)}, \quad (17)$$

omitting residuals for brevity. In this way the complete model can be expressed in a simple matrix notation much in line with stating a bilinear model as

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T + \mathbf{E}. \quad (18)$$

The Khatri-Rao product also eases comparison of the PARAFAC model with the matrix formulation of the Tucker3 model (equation 40) as well as enabling higher-order PARAFAC models to be formulated using matrix notation.

UNIQUENESS

An obvious advantage of the PARAFAC model is the uniqueness of the solution. In bilinear models there is the well-known problem of rotational freedom. The loadings in a spectral bilinear decomposition reflect the pure spectra of the analytes measured, but it is not possible without external

information to actually find the pure spectra because of the rotation problem. This fact has prompted a lot of different methods for obtaining more interpretable models than PCA and models alike (Scarminio & Kubista 1993, Sarabia et al. 1993, Faber et al. 1994), or for rotating the PCA solution to more appropriate solutions. Most of these methods, however, are more or less arbitrary or have ill-defined properties. This is not the case with PARAFAC. In most circumstances the model is uniquely identified from the structure, and hence no postprocessing is necessary as the model is *the* best model in the least squares sense. Further, if the data are approximately trilinear, the true underlying phenomena will be found if the right number of components is used and the signal-to-noise ratio is appropriate (Harshman 1972, Kruskal 1976 & 1977). This important fact is what originally initiated R. A. Harshman to develop the method based on the idea of parallel proportional profiles of Cattell (1944). There are several examples where the PARAFAC model coincides with a physical model, e.g., fluorescence excitation-emission, chromatography with spectral detection, and spatiotemporal analysis of multichannel evoked potentials (Field & Graupe 1991). In Figure 4 an example is given of the practical meaning of uniqueness for modeling fluorescence data. The uniqueness properties of the PARAFAC model are sometimes stated as the model having *unique axes*. As opposed to a bilinear model where the subspace spanned by the data can be uniquely determined, the PARAFAC model not only determines the subspace but also the position of the axes defining the subspace. Hence the name unique axes.

Harshman (1972) and Leurgans et al. (1993) among others have shown, that unique solutions can be expected if the loading vectors are linear independent in two of the modes, and furthermore in the third mode the less restrictive condition is that no two loading vectors are linear dependent. Kruskal (1977a & 1989) gives even less restricted conditions for uniqueness. He uses the k -rank of the loading matrices, which is a term introduced by Harshman & Lundy (1984b). If any combination of k_A columns of \mathbf{A} have full column-rank, and this does not hold for k_A+1 , then the k -rank of \mathbf{A} is k_A . The k -rank is thus related, but not equal, to the rank of the matrix, as the k -rank can never exceed the rank. Kruskal proves that if

$$k_A + k_B + k_C \geq 2F + 2, \quad (19)$$

then the PARAFAC solution is unique. Here k_A is the k -rank of \mathbf{A} , k_B is the k -rank of \mathbf{B} , k_C is the k -rank of \mathbf{C} and F is the number of PARAFAC components sought. None of the above conditions are strong enough to cover all situations where uniqueness can be expected, but they do give sufficient conditions for uniqueness. Note that disregarding the above rule, a one-component solution is always unique. This even holds for a two-way decomposition.

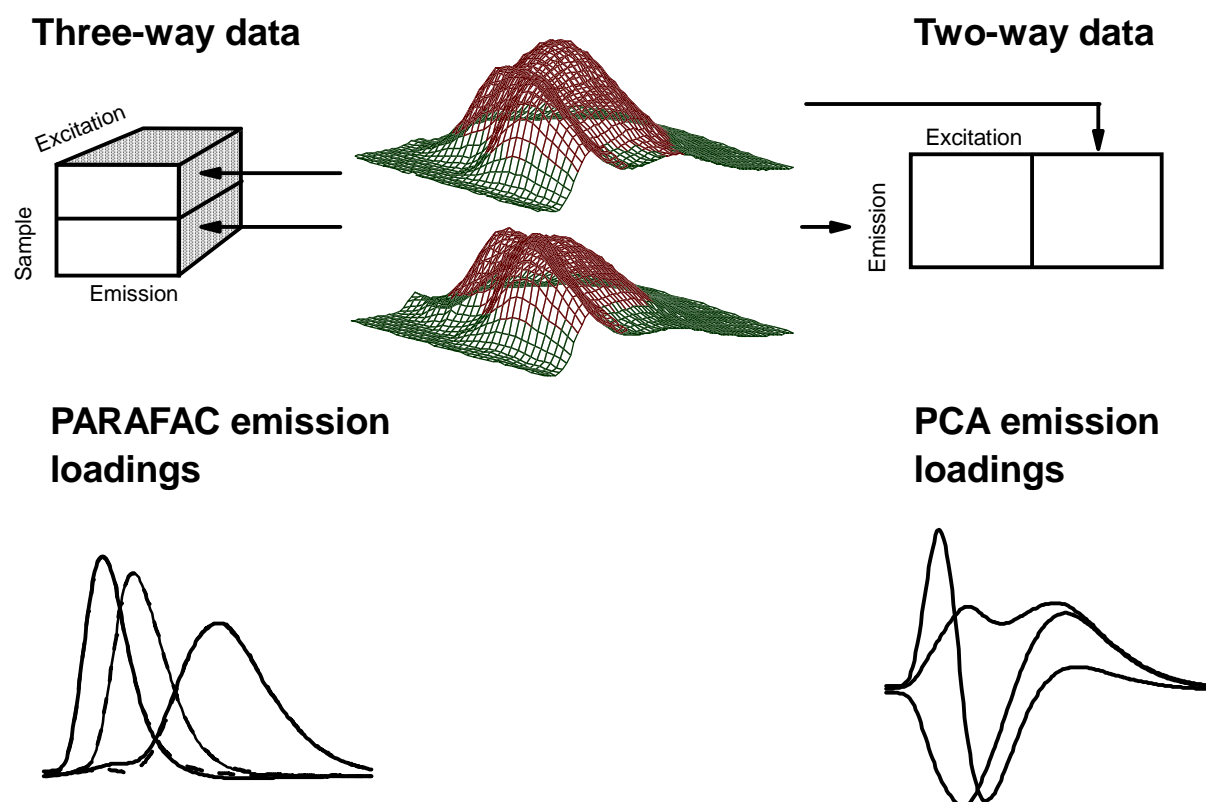


Figure 5. Two samples with different amounts of tryptophan, tyrosine, and phenylalanine measured fluorometrically (excitation-emission) giving two landscape/matrices of data shown in the top middle. The data can be arranged and decomposed as a three-way array (left) or as an unfolded two-way array (right). The pure emission spectra are shown by dashed lines together with the PARAFAC emission loading vectors in the bottom left corner. In the bottom right corner the corresponding orthogonal PCA emission loadings are shown.

The mathematical meaning of uniqueness is that the estimated PARAFAC component matrices cannot be rotated without a loss of fit. Consider a two-way F -component bilinear model

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T + \mathbf{E}.$$

This model can be rotated by any non-singular $F \times F$ matrix, \mathbf{P} , as

$$\mathbf{A}\mathbf{B}^T = \mathbf{A}\mathbf{P}\mathbf{P}^{-1}\mathbf{B}^T$$

i.e., using as scores $\mathbf{A}\mathbf{P}$ and as loadings $\mathbf{B}(\mathbf{P}^{-1})^T$ a model is obtained with exactly the same fit to the data. This is often termed the rotational ambiguity of PCA. In PARAFAC no such rotational indeterminacy exists. The model

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{B}^T$$

if rotatable, could equally well be expressed as

$$\mathbf{A}\mathbf{D}_k\mathbf{B}^T = \mathbf{A}\mathbf{T}\mathbf{T}^{-1}\mathbf{D}_k\mathbf{S}\mathbf{S}^{-1}\mathbf{B}^T. \quad (20)$$

This expression implies that instead of \mathbf{A} , \mathbf{B} , and \mathbf{D}_k the loadings $\mathbf{A}\mathbf{T}$, $\mathbf{B}(\mathbf{S}^{-1})^T$ and $\mathbf{T}^{-1}\mathbf{D}_k\mathbf{S}$ may as well be used. However, for the PARAFAC model to hold the third mode loading matrices $\mathbf{T}^{-1}\mathbf{D}_k\mathbf{S}$ must be diagonal. Because of this requirement only very special \mathbf{T} 's (and \mathbf{S} 's) can be valid, namely those that preserve the diagonality (Harshman 1972). In practice this means that \mathbf{T}/\mathbf{S} (and hence $\mathbf{T}^{-1}/\mathbf{S}^{-1}$) have to be permutation or scaling matrices. Therefore the only indeterminacy in the least squares solution is the order of the components and the scaling of the loading vectors. This indeterminacy is trivial and not a matter of great concern.

RELATED METHODS

The methods mentioned below are all based on the trilinear model underlying the three-way PARAFAC model. However, these methods are

interesting either because they are applied in other types of analyses, or because they are estimated differently than PARAFAC.

CANDECOMP: The trilinear – or multilinear in general – model behind PARAFAC has been reinvented several times. Most notably Carroll & Chang (1970) developed the identical model CANDECOMP at the same time that Harshman developed PARAFAC. The CANDECOMP model was not inspired by the basic principles for interpretable models laid down by Cattell (1944), but rather CANDECOMP was developed as a natural extension in the field of multidimensional scaling. As an intrinsic part of INDSCAL (individual differences scaling) it has been widely used in psychometrics and related areas for a long time. As such CANDECOMP and PARAFAC are identical and thus in this context the theory of CANDECOMP does not add much to the use of PARAFAC.

PMF3: Paatero (1994) has developed an interesting algorithm for fitting two-way bilinear models under non-negativity constraints using a weighted loss function. The model and algorithm has been coined positive matrix factorization or PMF. Later the method was extended to the three-way case (Paatero 1997), thus being identical to the PARAFAC model under non-negativity constraints and with a weighted loss function. To distinguish the two- and three-way case the methods are called PMF2 and PMF3 respectively. The interesting aspect of the PMF3 algorithm is that it is not based on alternating least squares (page 57), but rather seeks to update all parameters simultaneously using a Gauss-Newton approach (page 96). The algorithm is usually faster than the standard PARAFAC algorithms especially for difficult problems (Hopke et al. 1998). The primary problem with PMF3 is that the memory requirement of the algorithm increases fast with the size of the problem, in practice preventing the algorithm from being applicable to large data sets. This problem could potentially be circumvented by using a compression of the data set prior to fitting the model, as described in chapter 4. Another obstacle in the usefulness of the PMF3 algorithm is that it does not generalize to other models or orders of data. P. Paatero is currently developing an algorithm called the multilinear engine which will be more general and flexible than any algorithm described in this

thesis. This newer algorithm simply takes as input a structural equation and a set of constraints, and from this information decomposes the array. Thus instead of being restricted to PARAFAC, Tucker3 or any other predefined structural model, the structure can be determined freely according to the problem at hand.

PRINCIPAL COMPONENT FACTOR ANALYSIS: Appellof & Davidson re-invented the PARAFAC model and algorithm in 1981 inspired by the CANDECOMP algorithm. In a relatively short paper they present a thorough description of the model, its usefulness for fluorometric data, and algorithmic shortcuts for an ALS procedure identical to the original procedure of Harshman (1970). In the paper no name was given to the method, but in a series of papers (Russell & Gouterman 1988a & b, Russell et al. 1988) the model and algorithm was called principal component factor analysis.

RAFA, GRAM, DTD: Another algorithm for fitting the trilinear model that has received much attention in chemometrics has several acronyms:

RAFA	- rank annihilation factor analysis,
GRAFA	- generalized rank annihilation factor analysis,
GRAM	- generalized rank annihilation method,
TLD	- trilinear decomposition,
DTD	- direct trilinear decomposition,
NBRA	- nonbilinear rank annihilation.

Several other methods exist that are essentially identical but not given any names. RAFA is the mother of these methods. GRAFA is identical to GRAM. TLD is identical to DTD and the broad term TLD should generally be avoided as PARAFAC is also a trilinear decomposition.

Ho and coworkers (Ho et al. 1978, Ho et al. 1980 & 1981) developed an algorithm called rank annihilation factor analysis – RAFA – for estimating the concentration of an analyte in an unknown matrix solely using the unknown sample and a pure standard. This amazing property has later been

coined the *second-order advantage*, as this property is obtained by using the second-order or two-way structure of the individual sample instead of merely unfolding the matrix to a long vector or first-order structure. The second-order advantage is identical to the uniqueness of the trilinear structure, and the structural model underlying the GRAM, RAFA, and DTD algorithm is the same as that of the PARAFAC model. The idea behind RAFA was based on reducing the rank of the calibration sample by subtracting the contribution from the analyte of interest, that is, if the signal from the analyte of interest is subtracted from the sample data, then the rank of this matrix will decrease by one as the contribution of the analyte of interest to the rank is one in case of ordinary bilinear rank-one data like chromatographic or fluorescence data. This was intuitively appealing but the method itself was somewhat unsatisfactory and slow.

DIRECT TRILINEAR DECOMPOSITION VERSUS PARAFAC
BOX 1
DTD

Only two (pseudo-) samples

Unknown objective

Fast algorithm

No constraints

Only three-way

PARAFAC

Any number of samples

Least squares objective

Slow

Constraints possible

N-way

Lorber (1984 & 1985) found that the algorithm could be automated by realizing that the sought reduction in rank could be expressed as a generalized eigenvalue problem. Sanchez & Kowalski (1986) generalized the method into the generalized rank annihilation method – GRAM – in which several components could be present/absent in both calibration and standard sample. Wilson et al. (1989) and Wang et al. (1993) extended GRAM to setups where the contribution of a single analyte to the signal does not correspond to a rank-one signal as presupposed in the trilinear

model underlying GRAM. This method is called nonbilinear rank annihilation – NBRA – and is numerically similar to GRAM, but special attention is needed for correctly estimating the concentrations.

A problem in GRAM arises because eigenvectors and values involved in fitting the model can be complex for different reasons. This is undesirable as the pure spectra, profiles etc. can henceforth not be estimated. Li et al. (1992), Li and Gemperline (1993), Booksh et al. (1994), and Faber et al. (1994) discuss different approaches to remedy this mathematical artifact of the method by similarity transformations.

The GRAM method is based on the trilinear model and fits the model using a (generalized) eigenvalue problem. The method is restricted by one mode having maximally dimension two (i.e., two samples). However, estimating the parameters of the model using two samples, will give the spectral and chromatographic loadings in case of a chromatography-spectroscopy analysis. As these parameters are fixed for new samples as well, the concentrations of analytes for new samples can be found by simple regression on the fixed parameters. Used in this way the method is called DTD (Sanchez & Kowalski 1990). This extension is normally based on defining a set of two synthetic samples based on linear combinations of the original samples. Sanchez and Kowalski advocate for using a Tucker1 model for that purpose, while Sands & Young (80) propose a somewhat different scheme for an algorithm that can be considered equivalent to DTD.

As the DTD and PARAFAC model are structurally identical, then what are the differences? The PARAFAC model is a least-squares model whereas the DTD model has no well-defined optimization criterion (see Box 1). For noise-free data it gives the correct solution but for noisy data there is no provision for the quality of the model. The advantage of DTD is the speed of the algorithm and for precise trilinear data it often gives good results. The advantage of the PARAFAC model is the possibility to modify the algorithm according to external knowledge, such as using weighted regression when uncertainties are available, or using non-negativity when negative parameters are known not to conform to reality (see for example Box 14 on page 142). Also advantageous for the PARAFAC model is the easiness with which it can be extended to higher-order data. Mitchell &

Burdick (1993), Leurgans et al. (1993), and Sands & Young (1980) all investigate GRAM-like methods and compare them with PARAFAC for their use in curve resolution. Tu & Burdick (1992) also describe the differences between some of these methods. They find GRAM inferior to PARAFAC but suggest using GRAM for initialization of the PARAFAC algorithm. Sanchez & Kowalski (1990) suggest the same (see however page 62). Kiers and Smilde (1995) compared different versions of GRAM and PARAFAC. They prove under which conditions these methods can be expected to be able to predict the concentration of analytes in an unknown sample possibly with unknown interferences using only one standard.

The basic principle of GRAM/DTD has been invented several times. As early as 1972 Schönemann developed a similar algorithm, essentially and interestingly based on the same idea of Cattell (1944) through Meredith (1964) that Harshman generalized to the PARAFAC model and algorithm. Schönemann's GRAM-like method has later been refined in a more stable and sensible way by, e.g., de Leeuw and Pruzansky (1978).

3.4 PARAFAC2

In some cases a data set – ideally trilinear – does not conform to the PARAFAC model. The reason can be sampling problems or physical artifacts. Another problem occurs when the slabs of the array are not of the same row (or column) dimension. An example could be that in a longitudinal analysis certain subjects did not go along all the way (some persons died before the analysis ended, some batches were stopped due to breakdown etc.). It turns out that both of these problems can, in certain cases, be remedied with the use of the PARAFAC2 model.

Consider a chromatographic data set obtained with spectral detection of some sort, very generally structured as wavelength \times elution time \times run/sample. In many situations such a data set will be well suited for a PARAFAC model. Fitting a PARAFAC model will give the parameters **A** (\mathbf{a}_f being the estimated spectrum of analyte f), **B** (\mathbf{b}_f being the profile or chromatogram of analyte f), and **C** (c_{kf} being the relative concentration of analyte f in sample k). If there are shifts in retention time from run to run the PARAFAC model can give misleading results since in the PARAFAC model

it is assumed that the profile of each analyte be the same in every run. Instead of forcing the data into the PARAFAC model it is possible to define a model, that does not require the chromatographic profiles to be identical from run to run, but allows for some sort of deviation. The PARAFAC2 model is one way of doing so.

STRUCTURAL MODEL

Consider the PARAFAC model

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k \mathbf{B}^T, \quad k = 1, \dots, K \quad (21)$$

disregarding noise. Instead of modeling the \mathbf{X}_k matrices directly consider a model of an array consisting of the matrices $\mathbf{X}_k \mathbf{X}_k^T$. Assuming first that the PARAFAC model is valid, then for the cross-product array it holds that

$$\begin{aligned} \mathbf{X}_k \mathbf{X}_k^T &= \\ (\mathbf{A} \mathbf{D}_k \mathbf{B}^T)(\mathbf{A} \mathbf{D}_k \mathbf{B}^T)^T &= \\ \mathbf{A} \mathbf{D}_k \mathbf{B}^T \mathbf{B} \mathbf{D}_k \mathbf{A}^T &= \\ \mathbf{A} \mathbf{D}_k \mathbf{H} \mathbf{D}_k \mathbf{A}^T, \quad k = 1, \dots, K & \end{aligned} \quad (22)$$

where

$$\mathbf{H} = \mathbf{B}^T \mathbf{B}. \quad (23)$$

Define \mathbf{Y} as

$$\mathbf{Y} = [\mathbf{Y}_1 \ \mathbf{Y}_2 \ \dots \ \mathbf{Y}_K] \quad (24)$$

where

$$\mathbf{Y}_k = \mathbf{X}_k \mathbf{X}_k^T. \quad (25)$$

The relation between $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ is shown graphically in Figure 6.

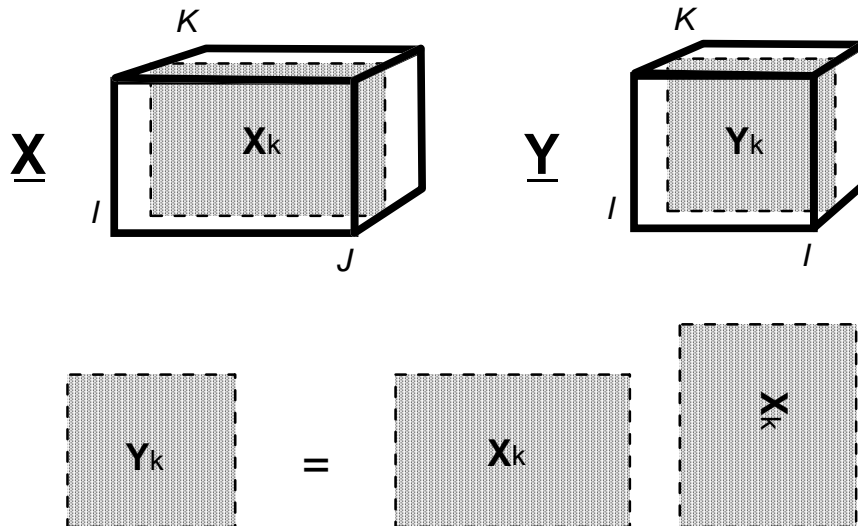


Figure 6. The cross-product array $\underline{\mathbf{Y}}$ is obtained from the raw data $\underline{\mathbf{X}}$ such that each frontal slice in $\underline{\mathbf{Y}}$ is the cross-product of the corresponding slice in $\underline{\mathbf{X}}$.

Rearrange $\mathbf{Y}^{(I \times I \times K)}$ as an $K \times II$ array called $\mathbf{Y}^{(K \times II)}$. Then it can be shown that it is possible to state the model as

$$\mathbf{Y}^{(K \times II)} = (\mathbf{C}^T \otimes \mathbf{C}^T)^T \text{diag}(\text{vec} \mathbf{H}) (\mathbf{A} \otimes \mathbf{A})^T. \quad (26)$$

This model is called the PARAFAC2 model. Observe, that since only \mathbf{Y}_k and not \mathbf{X}_k is modeled, it is possible to have \mathbf{X}_k matrices of different column dimensions.

Only \mathbf{H} , which is the cross-product matrix of \mathbf{B} , is estimated, not \mathbf{B} itself. Thus, in this model the profiles of \mathbf{B} are not required to be identical at different runs, only the cross-product of \mathbf{B} must be the same. In the chromatographic example the implication is that \mathbf{A} (spectrum estimates) and \mathbf{C} (concentration estimates) should be the same irrespective of elution time, but the chromatographic profiles need not be the same in every experiment. Only the covariance or crossproduct matrix of the profile loading matrix should be the same to conform to the PARAFAC2 model.

This is naturally less stringent than requiring that the profiles be identical as in the PARAFAC model, but it is difficult to see directly what types of deviations from the trilinear model are then allowed. Kiers et al. (1998) have shown that requiring

$$\mathbf{H} = \mathbf{B}^T \mathbf{B} \quad (27)$$

is equivalent to assuming the model

$$\hat{\mathbf{X}}_k = \mathbf{A} \mathbf{D}_k (\mathbf{B}_k)^T = \mathbf{A} \mathbf{D}_k (\mathbf{P}_k \mathbf{B})^T, \quad k = 1, \dots, K \quad (28)$$

where \mathbf{A} and \mathbf{D}_k are defined as usual, \mathbf{B} is an $F \times F$ matrix, and \mathbf{P}_k is a columnwise orthonormal matrix of size $J \times F$ ($J \geq F$). It is easily verified that if \mathbf{B}_k , the profiles for the k th mode, can be expressed as

$$\mathbf{B}_k = \mathbf{P}_k \mathbf{B}, \quad k = 1, \dots, K \quad (29)$$

then it holds that

$$\mathbf{B}_k^T \mathbf{B}_k =$$

$$(\mathbf{P}_k \mathbf{B})^T (\mathbf{P}_k \mathbf{B}) =$$

$$\mathbf{B}^T \mathbf{B} = \mathbf{H}, \quad k = 1, \dots, K \quad (30)$$

since

$$\mathbf{P}_k^T \mathbf{P}_k = \mathbf{I}, \quad k = 1, \dots, K \quad (31)$$

From this it follows, that if the data at hand do not conform to the PARAFAC model and if the deviation from the trilinear model can be expressed as in equation 28, then the data can be modeled by the PARAFAC2 model. For chromatographic data the problem of shifts in retention time can be

modeled by the PARAFAC2 model if the data are measured such that a baseline is present both before and after the appearance of the peaks. Also, overlapping peaks must, at least approximately, be shifted similarly. However, even though the PARAFAC2 model cannot model the kind of deviation present in the data perfectly, using PARAFAC2 instead of merely PARAFAC can still be helpful. Typically if the data are too complex for the PARAFAC model, using the less restrictive PARAFAC2 will at least partly remedy the problem, yet still enable a unique approximate model of the data.

UNIQUENESS

The PARAFAC2 model is unique in certain cases but the uniqueness of the PARAFAC2 model has been much less studied than the uniqueness of the PARAFAC model. Harshman & Lundy (1996) and ten Berge & Kiers (1996) give certain results on when uniqueness can be expected. For a rank two problem ten Berge & Kiers show that an $I \times I \times 3$ array is sufficient for obtaining uniqueness if non-negativity of (only) **C** is imposed and the first and second mode loading matrices have full column-rank. If no non-negativity is imposed the third mode has to be at least of dimension four or five. In all cases certain mild assumptions of the parameter matrices are to be fulfilled. The results so far indicate that uniqueness can be obtained as in the PARAFAC model but requires somewhat more levels of variation (see also Kiers et al. 1998).

3.5 PARATUCK2

PARATUCK2 is a generalization of the PARAFAC model, that adds some of the flexibility of Tucker's three-mode models (page 44) while retaining some of PARAFAC's uniqueness properties. The name PARATUCK2 indicates its similarity to both the PARAFAC and the Tucker2 model (page 49). The PARATUCK2 model is well suited for a certain class of multi-way problems that involve interactions between factors. This can occur, for example, in kinetic experiments where end products are related to the precursor, or in setups involving a pH gradient where both the acidic and basic forms relate to the same analyte. The PARATUCK2 model (Harshman & Lundy 1994) has been suggested in the psychometric literature, but

no applications have yet been published, partly perhaps because no algorithm has been published.

To give an understanding of the PARATUCK2 method and the usefulness of its provision for factor interactions, it may be helpful to read the application discussed on page 207. It nicely illustrates the appropriateness of the PARATUCK2 model in rank-deficient problems.

STRUCTURAL MODEL

The general PARATUCK2 model is defined as

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k^{\mathbf{A}}\mathbf{H}\mathbf{D}_k^{\mathbf{B}}\mathbf{B}^{\mathbf{T}} + \mathbf{E}_k \quad k = 1, \dots, K \quad (32)$$

\mathbf{X}_k is an $I \times J$ matrix, \mathbf{A} is an $I \times R$ matrix of loadings. The left \mathbf{D}_k matrix, $\mathbf{D}_k^{\mathbf{A}}$, is an $R \times R$ diagonal matrix, \mathbf{H} is an $R \times S$ matrix. The right \mathbf{D}_k matrix $\mathbf{D}_k^{\mathbf{B}}$ is an $S \times S$ diagonal matrix, \mathbf{B} is a $J \times S$ matrix and \mathbf{E}_k is a residual matrix. A matrix $\mathbf{C}^{\mathbf{A}}$ is defined that in its k th row contains the diagonal of $\mathbf{D}_k^{\mathbf{A}}$. A matrix $\mathbf{C}^{\mathbf{B}}$ is defined similarly. Due to the introduction of \mathbf{H} , the number of factors in the \mathbf{A} and \mathbf{B} mode need not be the same, and this is what gives the model its Tucker likeness. The term $\mathbf{D}_k^{\mathbf{A}}\mathbf{H}\mathbf{D}_k^{\mathbf{B}}$ can be compared to the core of a Tucker2 decomposition (van der Kloot & Kroonenberg 1985, Ross & Leurgans 1995), but due to the restricted structure of the core compared to Tucker2 uniqueness is retained (Harshman & Lundy 1996).

In Figure 7 the mathematical model is depicted graphically. The Tucker2 model is equal to the model shown in Figure 7a, but in PARATUCK2 the so-called extended core array, \mathbf{G} , is specifically modeled as shown in Figure 7b. The structure of each frontal layer of \mathbf{G} is a function of \mathbf{H} , $\mathbf{C}^{\mathbf{A}}$, and $\mathbf{C}^{\mathbf{B}}$, i.e., $\mathbf{G}_k = \mathbf{D}_k^{\mathbf{A}}\mathbf{H}\mathbf{D}_k^{\mathbf{B}}$.

As for the PARAFAC and the PARAFAC2 model it is possible to state the model using the Khatri-Rao product as

$$\mathbf{X}^{(K \times I \times J)} = ((\mathbf{C}^{\mathbf{B}})^{\mathbf{T}} \otimes (\mathbf{C}^{\mathbf{A}})^{\mathbf{T}})^{\mathbf{T}} \text{diag}(\text{vech}\mathbf{H})(\mathbf{B} \otimes \mathbf{A})^{\mathbf{T}} + \mathbf{E}_k \quad (33)$$

Though less appealing than the compact PARAFAC expression, the use of the Khatri-Rao product does lead to a matrix-formulation of the complete model, and also illustrates that the **A** and **B** loadings are interacting with the elements of the matrix $((\mathbf{C}^B)^T \otimes (\mathbf{C}^A)^T)^T \text{diag}(\text{vec}\mathbf{H})$ giving the magnitude of these interactions.

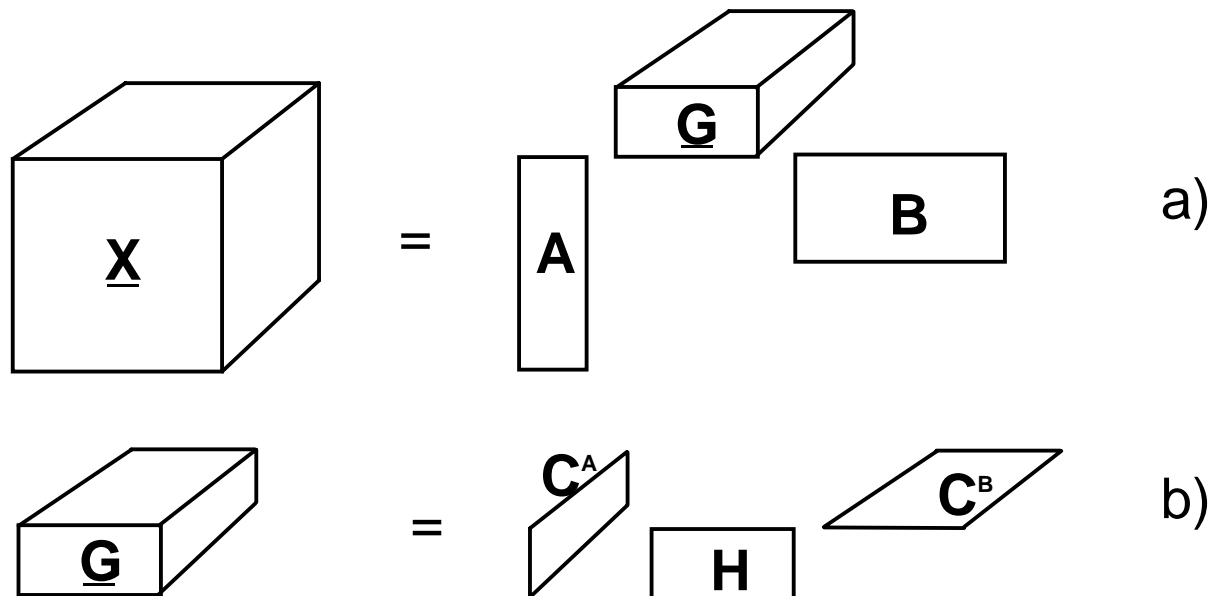


Figure 7. Graphical representation of a PARATUCK2 model. In a) the complete model is shown. The structure of **G** is shown in b).

UNIQUENESS

Only Harshman & Lundy (1996) have worked on proving the uniqueness properties of the PARATUCK2 model. They give experimental evidence of uniqueness being obtainable for models with different 'left' and 'right' dimensions ($R \neq S$), but treat only in detail the situation where R equals S . For this situation they prove that any model will be unique if all loading matrices including **H** are of full rank and there is proper variation in the matrices \mathbf{C}^A and \mathbf{C}^B . Proper variation in this case means, that there are sufficiently many levels of the third mode varying independently in a sense described in more detail in Harshman & Lundy (1996). Due to the way they prove uniqueness it is required to have at least 5 levels in the third mode for a two-factor model, 15 for a three-factor model, and 35 for a four-factor model. They further restrict **H** to having no zero elements. The requirements may seem harsh, and indeed several of them can be relaxed

substantially, however, at the cost of more complicated proofs (which already are quite complicated). In essence, what is indicated in Harshman & Lundy (1996) is, that the PARATUCK2 model will mostly be unique if the loading matrices are of full rank and the dimensions of the array not too small.

RESTRICTED PARATUCK2

In this section a restricted PARATUCK2 model will be described that is well suited for modeling rank-deficient data. Rank-deficiency arises in a number of situations (Amrhein et al. 1996) when there is linear dependency between components. Kinetic data, for example, are often subject to rank-deficiency as the formation of some products, hence the concentrations of these, depend on the amount of their precursors. Consider a two-way model

$$\mathbf{X} = \mathbf{A}\mathbf{H}\mathbf{B}^T + \mathbf{E} \quad (34)$$

where \mathbf{A} ($I \times F$) and \mathbf{B} ($J \times F$) are loading matrices, \mathbf{H} ($F \times F$) is an interaction matrix and \mathbf{E} a residual matrix. For rank reduced SVD it holds that \mathbf{H} is a diagonal matrix but it is possible to introduce non-diagonal elements thereby allowing for interaction between loading vectors. This does not provide increased modeling power in the two-way case, but extending the model to multi-way data it does make a difference. For simplicity the two-way case is used as an example first.

Let the first mode be a sample mode and \mathbf{A} be estimates of concentrations, and let the second mode be a spectral mode and \mathbf{B} estimates of spectra. It may happen that there are three analytes in the data, but due to their interrelation the amount of the first two analytes is identical. The rank of the matrix would thus be two even though three spectral phenomena are present. To cope with this, it is possible to specify a model as in equation 34 but where \mathbf{A} is now of size $I \times 2$, \mathbf{B} of size $J \times 3$ and \mathbf{H} of size 2×3 . In this case setting

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (35)$$

would give a perfect model as this implies that the concentration of the first and second spectral phenomena are identical (\mathbf{a}_1). Writing out the corresponding model it would read

$$\hat{\mathbf{X}} = \mathbf{AHB}^T = \mathbf{a}_1\mathbf{b}_1^T + \mathbf{a}_1\mathbf{b}_2^T + \mathbf{a}_2\mathbf{b}_3^T \quad (36)$$

showing that there are two linear independent variations in the sample mode, but three in the spectral mode. Often the interaction pattern as defined by \mathbf{H} is not known but subject to optimization.

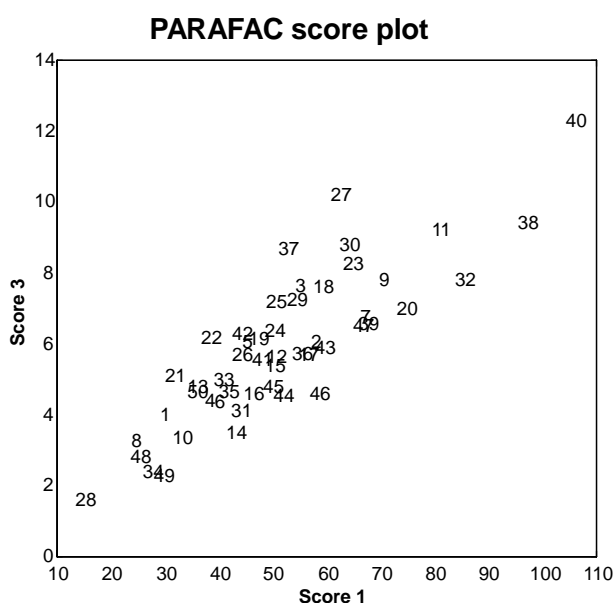


Figure 8. Score three versus one from a PARAFAC model of the fluorescence data.

The two-way model described here is not identified unless additional constraints are introduced. Consider, however, a situation as above but where the data are three-way. The last mode may for example be a chromatographic mode. Instead of the model in equation 34 the model can then be expressed

$$\mathbf{X}^{(I \times JK)} = \mathbf{AH}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E} \quad (37)$$

where the matrix $\mathbf{C} \otimes \mathbf{B}$ indicates that both \mathbf{B} (spectra) and \mathbf{C} (chromatographic profiles) are of rank three. This model can equivalently be expressed as

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k \mathbf{B}^T + \mathbf{E}, \quad (38)$$

where \mathbf{D}_k is a diagonal matrix with the k th row of \mathbf{C} in its diagonal. Comparing with equation 32 the similarity with the PARATUCK2 model is evident. The model can thus be considered a restricted PARATUCK2 model where \mathbf{C}^A or equivalently \mathbf{C}^B is fixed as a matrix of ones. An alternative way of interpreting the model is as a restricted PARAFAC model where the first mode loading matrix is restricted to the form $\mathbf{A}\mathbf{H}$. On page 207 an example on the use of such a restricted PARATUCK2 is given. Another example will be shortly described here to illustrate one situation in which the restricted PARATUCK2 model comes in handy.

Fifty samples of cow milk from cows of different breeds and from different regions were collected². Each sample was measured spectrofluorometrically from 200 to 520 nm excitation (step length 20 nm) and 300 to 650 nm emission (step length 2 nm) on an Aminco-Bowman Series 2 fluorometer at 40°C after preservation. Several multi-way models were investigated. A three-component PARAFAC model seems in many respects to be suitable. Rank analysis as well as loading plots from a PARAFAC and a Tucker3 model clearly pointed to the presence of three distinct spectral phenomena. However, score plots in the sample mode reveal that two components are almost collinear (Figure 8). The cause of this may be that the two underlying phenomena are simply correlated by nature. In any case, the model may not be stable as the PARAFAC model requires that no two loading vectors are collinear in order to be identified. As the second and third mode loadings in this case need to be of rank three, another model than the PARAFAC model will have to be used. Using Tucker3 with model dimensions two, three, and three is one possibility, but

². The application is shown here with courtesy of P. W. Hansen, Foss Electric A/S, DK. The purpose of collecting the samples were to make prediction models for fat, protein, lactose, and total solids, but is not relevant here.

this will only solve the rank problem by introducing rotational indeterminacy. Considering the problem, it is apparent that the restricted PARATUCK2 model is close to what is sought. By fitting the model

$$\mathbf{X} = \mathbf{A}\mathbf{H}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E} \quad (39)$$

with \mathbf{A} consisting of two columns and \mathbf{B} and \mathbf{C} of three columns and defining \mathbf{H} as in equation 35, a model is obtained where each phenomenon given by $\mathbf{c}_f \otimes \mathbf{b}_f$ is represented in a sample by the variation of two scores and the interaction of these defined by \mathbf{H} . In Figure 9 the three-component PARAFAC model and the two/three-component PARATUCK2 model is shown. The loadings of the PARATUCK2 model are similar to the PARAFAC scores. Thus, when rank-deficiency occurs the PARATUCK2 model is a viable alternative to the overly flexible Tucker3 model.

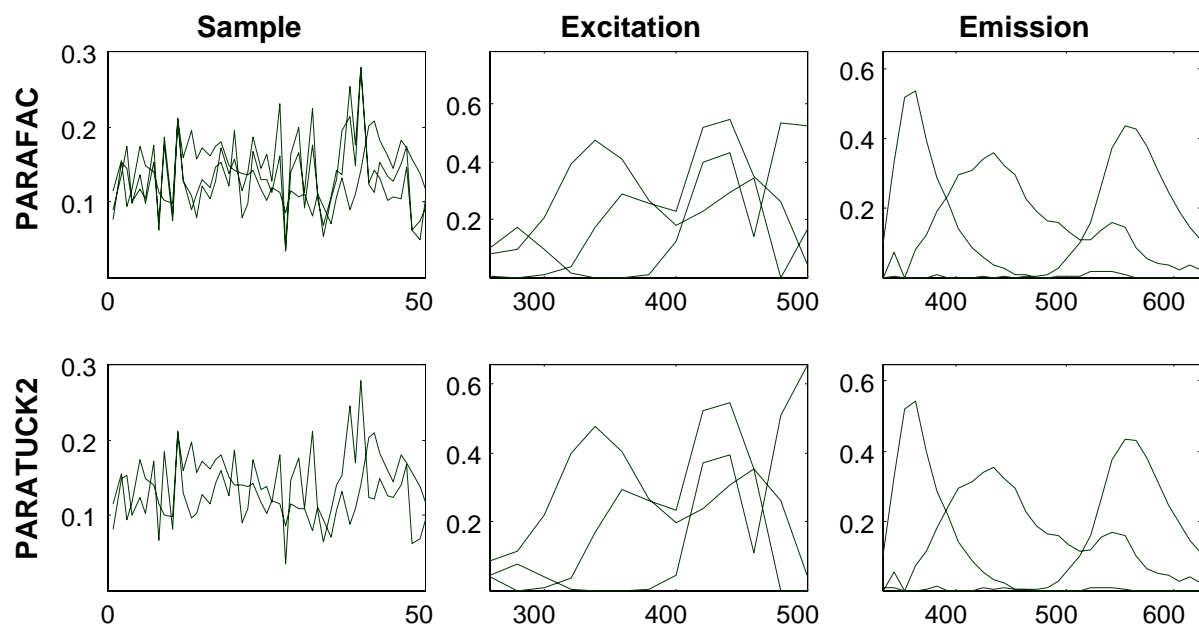


Figure 9. Comparing parameters from a PARAFAC (top figures) and a PARATUCK2 (bottom figures) model. Notice the correlations in the first mode PARAFAC scores and the similarity of the two models.

The main reason for the development of the PARATUCK2 model is that some types of three-way data are too complex to be modeled by the

trilinear PARAFAC model. A natural thing to do is then to use the more flexible three-mode factor analysis models Tucker3 or Tucker2 (to be described next). However, in most cases these models are unnecessarily flexible giving fitted models that are difficult to interpret. The way to overcome the interpretational difficulties of the Tucker models has been either to use postrotation steps (page 48) or to use constrained versions of the model (page 50). In many cases, however, it is possible to simply use the restricted PARATUCK2 model instead. The introduction of the interaction matrix, \mathbf{H} , instead of a three-way core array as in the Tucker3 model seems to be more in line with many problems characterized as rank-deficient (Amrhein et al. 1996).

The PARAFAC and PARATUCK2 models can be considered constrained versions of the Tucker3 model, meaning that any PARAFAC and PARATUCK2 model can also be defined as a restricted Tucker3 model. But in the sense that many problems are well handled by, e.g., the PARAFAC model, it is sensible to describe and implement the PARAFAC model independently from Tucker3. This is beneficial from a numerical point of view as more efficient algorithms can be devised. It is also beneficial from a cognitive point of view as the characteristics of the PARAFAC model does not have to be understood and developed in the light of a more complex model. It is the claim here, that the structure of the restricted PARATUCK2 model has application in many areas and hence deserves to be treated in its own rights more than as one special case of the Tucker3 model.

3.6 TUCKER MODELS

During the sixties L. Tucker developed a series of three-way models, which are now known as the Tucker1, Tucker2, and Tucker3 models (Kroonenberg & de Leeuw 1980). The models are also collectively called three-mode principal component analysis or originally three-mode factor analysis though it would by most people be conceived as a component model (Tucker 1963 & 1966). Several successful applications have been demonstrated in quite different areas such as chromatography (de Ligny et al. 1984), environmental analysis (Gemperline et al. 1992) and person perception analysis (van der Kloot & Kroonenberg 1985). The most

important model is the Tucker3 model, inasmuch as Tucker2 can be seen as a specific case of the Tucker3 model and the Tucker1 model simply corresponds to unfolding the array to a two-way matrix followed by a PCA. Hence focus will be on the Tucker3 model in the following.

STRUCTURAL MODEL OF TUCKER3

An $I \times J \times K$ array $\underline{\mathbf{X}}$ is given and a Tucker3 model of rank D in the first mode, E in the second mode, and F in the third mode is sought. Written in matrix notation letting \mathbf{X} be the $I \times JK$ unfolded array the Tucker3 model reads

$$\mathbf{X} = \mathbf{A}\mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}. \quad (40)$$

The matrix \mathbf{G} is the $(D \times E \times F)$ core array $\underline{\mathbf{G}}$ arranged as a $D \times EF$ matrix. The elements of the core define how individual loading vectors in the different modes interact. In equation 40 \mathbf{A} has size $I \times D$, \mathbf{B} has size $J \times E$, and \mathbf{C} has size $K \times F$ and these matrices hold the loadings in the first, second, and third mode respectively. Here only Tucker3 models with orthogonal \mathbf{A} , \mathbf{B} , and \mathbf{C} are considered, though this restriction is not mandatory.

To shed a little light on the Tucker3 model consider another formulation of the structural model

$$\hat{x}_{ijk} = \sum_{d=1}^D \sum_{e=1}^E \sum_{f=1}^F a_{id} b_{je} c_{kf} g_{def}. \quad (41)$$

The SVD formulation of a two-way PCA model would normally read

$$\hat{x}_{ij} = \sum_{f=1}^F a_{if} b_{jf} g_{ff}, \quad (42)$$

or alternatively

$$\hat{\mathbf{X}} = \mathbf{AGB}^T. \quad (43)$$

The indices of the singular values, g_{ff} , indicate that the matrix \mathbf{G} is a diagonal matrix, which again means that only the f th column of \mathbf{A} is interacting with the f th column of \mathbf{B} , thus the PCA model can be written as a sum of F separate outer products

$$\mathbf{X} = \mathbf{a}_1 g_{11} \mathbf{b}_1^T + \mathbf{a}_2 g_{22} \mathbf{b}_2^T + \dots + \mathbf{a}_F g_{FF} \mathbf{b}_F^T + \mathbf{E}. \quad (44)$$

Consider a situation where not only the diagonal, but all elements of \mathbf{G} can be nonzero. Now for every column of \mathbf{A} a product with each column of \mathbf{B} will be present. The corresponding model still has the matrix formulation given in equation 43, but written out it is

$$\mathbf{X} = \mathbf{a}_1 g_{11} \mathbf{b}_1^T + \mathbf{a}_1 g_{12} \mathbf{b}_2^T + \dots + \mathbf{a}_1 g_{1F} \mathbf{b}_F^T + \dots + \mathbf{a}_2 g_{21} \mathbf{b}_1^T + \dots + \mathbf{a}_2 g_{2F} \mathbf{b}_F^T + \dots + \mathbf{a}_F g_{FF} \mathbf{b}_F^T + \mathbf{E}, \quad (45)$$

in all F^2 products. For PCA/SVD the increased complexity would not provide any more modeling power. The model in equation 45 can always be rotated into the model in equation 44. Hence, there is no reason for using the more complex model as the simpler fits equally well and is easier to interpret. For three-way and higher-way models, however, the two approaches do not coincide with respect to fit. The diagonal approach generalizes to PARAFAC, where only loading vectors with the same column number interact. This corresponds to having a core array \mathbf{G} with zeros except in the superdiagonal ($g_{111}, g_{222}, \dots, g_{FFF}$). If nonzero off-superdiagonal elements are allowed the model will be the Tucker3 model, and opposite to the two-way situation the increased number of core-elements leads to a model that fits better except in extreme cases. An interesting aspect of the Tucker3 model, is that the number of components need not be the same in the different modes (cf. equation 41). Unlike the PARAFAC model, the column dimensions of the loading matrices can hence be accommodated individually in each mode.

The Tucker3 core of a Tucker3 model with orthogonal loading matrices

can be seen as a regression of $\underline{\mathbf{X}}$ onto a set of truncated basis matrices, or as the coordinates of $\underline{\mathbf{X}}$ in the truncated space defined by the loading matrices. For orthogonal loading matrices this transformation can be written as

$$\mathbf{G} = \mathbf{A}^T \mathbf{X} (\mathbf{C} \otimes \mathbf{B}). \quad (46)$$

The use of the Kronecker product in equation 46 is merely a convenient way of expressing the three transformations simultaneously in matrix notation. What really happens is that $\underline{\mathbf{X}}$ is regressed on \mathbf{A} in the first mode, $\mathbf{A}^T \mathbf{X}^{(I \times JK)}$. The product $\mathbf{A}^T \mathbf{X}^{(I \times JK)}$ is then regressed on \mathbf{B} in the second mode, and finally regressed on \mathbf{C} in the third mode.

Besides giving the magnitudes of interactions, the core can be considered an approximation of $\underline{\mathbf{X}}$. It approximates the *variation* of the original array by expressing it in terms of the truncated basis matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} . An approximation of the original data can thus be obtained by transforming \mathbf{G} back into the original space as in equation 40.

To recapitulate the SVD version of two-way PCA can be formulated as

$$\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix}^T \quad (47)$$

for a two-component model. This formulation corresponds to the PARAFAC generalization of PCA. It may also be formulated as

$$\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 \end{bmatrix} \begin{bmatrix} g_{11} & g_{21} \\ g_{12} & g_{22} \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix}^T, \quad (48)$$

which generalizes to the Tucker3 model (Box 2). The Tucker3 model is best seen as a way to generalize PCA to higher orders, i.e., its usefulness rests in its capability to compress variation, extract features, explore data, generate parsimonious models etc. There are few examples on the use of Tucker3 models for data that can be assumed to be generated by a

process according to the Tucker3 model. This as opposed to the PARAFAC, the PARAFAC2, and the restricted PARATUCK2 model, which coincide with several physical models.

TUCKER3 VERSUS PARAFAC AND SVD

BOX 2

Tucker3

PARAFAC

SVD

$$\hat{x}_{ijk} = \sum_{d=1}^D \sum_{e=1}^E \sum_{f=1}^F a_{id} b_{je} c_{kf} g_{def} \quad \hat{x}_{ijk} = \sum_{f=1}^F a_{if} b_{jf} c_{kf} g_{fff} \quad \hat{x}_{ij} = \sum_{f=1}^F a_{if} b_{jf} g_{ff}$$

UNIQUENESS

The Tucker3 model has rotational freedom, and is hence not structurally unique as the PARAFAC model. This can be seen by replacing the model

$$\mathbf{X} = \mathbf{AG}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}, \quad (49)$$

with a model where, e.g., the first mode loadings (or one or more other modes) have been rotated by a non-singular quadratic matrix \mathbf{S} . Then by counter-rotating \mathbf{G} with the inverse of this matrix the model

$$\mathbf{X} = \mathbf{ASS}^{-1}\mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T = \mathbf{AG}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E} \quad (50)$$

is obtained. As can be seen this model is completely equivalent to the original model, hence rotation is possible. Even imposing orthogonality does not provide an identified solution, since rotations by any orthogonal matrix of any of the loading matrices will provide new orthogonal loading matrices.

Actually it has been shown that the structural model is so redundant that several parameters, mostly more than half of the elements, in the core, \mathbf{G} , can be forced to zero without changing the fit of the model (Kiers et al.

1997). This clearly shows that the Tucker models are unnecessarily complex and explains why they give 'ambiguous' results. As in two-way analysis the rotational freedom has prompted the need for rotations of solutions in order to increase interpretability (Brouwer & Kroonenberg 1991, Kiers 1992, Henrion 1993, Kiers 1998a, Murakami et al. 1998, ten Berge & Kiers 1998 and Henrion & Andersson 1997).

TUCKER1 AND TUCKER2 MODELS

The Tucker1 and Tucker2 models can be seen as extreme cases of the Tucker3 model. The Tucker3 model will equal the Tucker2 model if one index, say the first, is running over all is , i.e.,

$$\hat{x}_{ijk} = \sum_{d=1}^I \sum_{e=1}^E \sum_{f=1}^K a_{id} b_{je} c_{kf} g_{def} \quad (51)$$

then a_{id} can be absorbed in g_{def} yielding

$$h_{ief} = \sum_{d=1}^I a_{id} g_{def} \quad (52)$$

and by using the core with elements for h_{ief} the model can be written

$$\hat{x}_{ijk} = \sum_{e=1}^E \sum_{f=1}^F b_{je} c_{kf} h_{ief} \quad (53)$$

or equivalently

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{H}^{(I \times EF)} (\mathbf{C} \otimes \mathbf{B})^T . \quad (54)$$

In this model the core array, \mathbf{H} , has the same first-mode dimension as \mathbf{X} and is also sometimes referred to as the extended core array, meaning that the core array is extended as compared to the Tucker3 core (cf. Figure 7).

When two instead of one index is running over all possible combinations

the model becomes

$$\hat{x}_{ijk} = \sum_{d=1}^D \sum_{e=1}^J \sum_{f=1}^K a_{id} b_{je} c_{kf} g_{def} \quad (55)$$

which is the Tucker1 model. By absorbing b_{je} and c_{kf} in g_{def} as above it follows that the model can be expressed

$$\hat{x}_{ijk} = \sum_{d=1}^D a_{id} h_{jkd} \quad (56)$$

which is equivalent to

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{AH}^{(D \times JK)} \quad (57)$$

The Tucker1 model is the well-known unfolding technique, where the data array is rearranged to a matrix which is then subjected to ordinary two-way PCA. The Tucker2 is a model of intermediate complexity as compared with the Tucker1 and Tucker3 model.

RESTRICTED TUCKER3 MODELS

Primarily due to non-uniqueness, the Tucker models have not received much attention in spectral analysis yet. They have primarily been used for exploratory analysis (Henrion et al. 1992, Gemperline et al. 1992, Henrion et al. 1995). However, the Tucker3 model forms the basis for what is termed constrained or restricted Tucker models initially proposed by Kiers (1992). In restricted Tucker models chemical (or other) knowledge is used to restrict especially the core elements, forcing individual elements to attain specific values. Kiers & Smilde (1997) showed that in this way it is possible in some situations to define models that uniquely estimate sought properties like spectra, concentrations etc. The use of restricted Tucker models has been promoted in chemometrics by Smilde et al. (1994a & b). It has primarily been proposed as a method for solving problems where the

second-order signal from the analyte of interest has a medium rank (thus not full rank) as opposed to idealized fluorescence spectroscopy or chromatography where each analyte is supposed to contribute with a rank-one signal. The restricted Tucker model can be seen as a structural model accommodated to a specific chemical problem.

3.7 MULTILINEAR PARTIAL LEAST SQUARES REGRESSION

There are several approaches to calibration when using multi-way methods. The standard multivariate two-way approach in chemometrics is using principal component regression (PCR) or partial least squares regression (PLS), where the array of independent variables is decomposed into a set of scores. The dependent variable(s) is then regressed on these scores instead of the original variables. The same approach is of course also possible in three-way analysis. The analog to PCR is to use, for example, PARAFAC or Tucker3 to decompose the array and then subsequently regress the dependent variables on the scores.

In chemometrics PLS regression is a widely used approach for obtaining multivariate regression models. The theory and advantages of PLS regression will not be described in detail here, but may be found in the literature (Martens & Næs 1989, de Jong & Phatak 1997). The main difference between PCR and PLS is that in PLS the independent data are modeled such that variation specifically relevant for predicting the dependent variables is emphasized. Rather than decomposing the independent data into a least squares bilinear model, a model of the dependent and a model of the independent data is obtained such that the score vectors from these models have pairwise maximal covariance. That is, components are found in \mathbf{X} and \mathbf{Y} simultaneously and such that the scores in the \mathbf{X} and \mathbf{Y} spaces have maximal covariance. Since the covariance is the product of the correlation between the scores and the variance of each score, these three measures are collectively maximized. Maximizing the variation of the score vectors ensures that the model is real and not due to small random variation. Maximizing the correlation (the linear relationship) ensures that it is possible to predict the \mathbf{Y} score from the \mathbf{X} score thus optimizing the predictive ability of the model.

In 1989 Ståhle extended the PLS regression model to three-way data by extending the two-way algorithm in a straightforward manner. The optimality of the proposed algorithm, however, was not substantiated. Later Bro developed a general multi-way PLS (N-PLS) regression model which was shown to be optimal according to the underlying theory of PLS (Bro 1996). Smilde (1997a) and de Jong (1998) further elaborated on the properties of N-PLS. Additionally, de Jong showed that the three-way version of the PLS1 regression method developed by Bro was numerically equivalent to the method earlier suggested by Ståhle, and Smilde (1997b) also devised a new approach to making PLS-like multi-way calibration models based on the theory of principal covariate regression (de Jong & Kiers 1992).

STRUCTURAL MODEL

In the three-way version of PLS the three-way array of independent variables is decomposed into a trilinear model similar to the PARAFAC model, only for N-PLS, the model is not fitted in a least squares sense but seeks in accordance with the philosophy of PLS to describe the covariance of the dependent and the independent variables. This is achieved by simultaneously fitting a multilinear model of the dependent variables, a multilinear model of the independent variables, and a regression model relating the two decomposition models.

Assume that both the dependent and the independent data are three-way. Let $\underline{\mathbf{X}}$ be the $I \times J \times K$ array of independent data and \mathbf{X} the $I \times JK$ unfolded array. Let $\underline{\mathbf{Y}}$ be the $I \times L \times M$ array of dependent data and \mathbf{Y} the $I \times LM$ unfolded array. The N-PLS model decomposes $\underline{\mathbf{X}}$ as

$$\mathbf{X} = \mathbf{T}(\mathbf{W}^K \otimes \mathbf{W}^J)^T + \mathbf{E}_x, \quad (58)$$

i.e., a trilinear model similar to the PARAFAC model, and $\underline{\mathbf{Y}}$ as

$$\mathbf{Y} = \mathbf{U}(\mathbf{Q}^M \otimes \mathbf{Q}^L)^T + \mathbf{E}_y. \quad (59)$$

Note that the notation has changed here compared to the prior models. The score vectors of $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ are called \mathbf{t} and \mathbf{u} respectively and the weight vectors \mathbf{w} and \mathbf{q} , to conform with standard PLS notation. A superscript J or K , L or M respectively is used to specify which mode the vectors refer to. The decomposition models above are estimated component-wise under the following restrictions: Find a set of vectors \mathbf{w}^J , \mathbf{w}^K , \mathbf{q}^L , and \mathbf{q}^M such that the least squares score vectors \mathbf{t} and \mathbf{u} have maximal covariance. Through these models of the data sets the prediction model between $\underline{\mathbf{Y}}$ and $\underline{\mathbf{X}}$ is found using a regression model for the so-called inner relation

$$\mathbf{U} = \mathbf{T}\mathbf{B} + \mathbf{E}_U. \quad (60)$$

When the dependent variables of a new sample are to be predicted the following holds. From the model of $\underline{\mathbf{X}}$ (equation 58) \mathbf{T} can be determined. Through equation 60 the scores in the $\underline{\mathbf{Y}}$ -space can be predicted and through the model of $\underline{\mathbf{Y}}$ (equation 59) the prediction of $\underline{\mathbf{Y}}$ is obtained. There are several delicate points in actually implementing an algorithm for this model. These will be described in detail in the next chapter.

Note that the decomposition model of $\underline{\mathbf{X}}$, the decomposition model of $\underline{\mathbf{Y}}$ and the regression model (60) relating these two models, *together* constitute the N-PLS regression model.

NOTATION FOR N-PLS MODELS

The multilinear PLS models are called N-PLS models in general. If an Arabic is used after PLS it defines the order of the dependent data. A single dependent variable gives a PLS1 model, a two-way matrix of dependent variables give PLS2 etc. To specify the order of the independent data a prefix can be used. For a two-way array the model is called bi-PLS, for a three-way array tri-PLS etc.

UNIQUENESS

The N-PLS model is unique in the sense that it consists of successively

estimated one-component models. As noted page 27 a one-component multilinear model is always unique. However, the uniqueness in this case will seldom infer that real underlying phenomena like pure-analyte spectra can be recovered, because the model assumptions do not reflect any fundamental or theoretical model.

3.8 SUMMARY

In this chapter several multi-way models have been discussed. The trilinear PARAFAC model has the attractive feature of giving unique solutions under mild conditions which has important implications for many modeling problems in the harder sciences as well as exploratory problems. In describing the PARAFAC model a matrix product called the Khatri-Rao product was described. It makes it possible to state the unique axis models using ordinary matrix algebra. The models become more transparent, and especially for higher order models this is advantageous.

The Tucker3 model is more flexible than the PARAFAC model, hence also more difficult to interpret. It is, however, still simpler than the alternative approach, unfolding, in that it uses much fewer parameters. In the Tucker3 model a new concept is introduced, namely interaction of factors. In the Tucker3 model, e.g., the first loading vector in the first mode, the third in the second mode and the fourth in the third mode constitute *one* component, the magnitude being determined by the size of the core element (1,3,4) in a manner similar to a singular value in SVD.

Two advanced decomposition models have also been discussed, the PARAFAC2 and the PARATUCK2 model. The PARAFAC2 model distinguishes itself from the ordinary PARAFAC model by allowing for certain deviations from exact trilinearity as compared to the PARAFAC model and yet still retaining the uniqueness properties. This gives the possibility to model, e.g., chromatographic data with retention time shifts. The PARATUCK2 model is a model of intermediate complexity compared to the PARAFAC and the Tucker models. It allows for interactions between factors but in a more restricted way than in the Tucker models. Due to this restrictedness PARATUCK2 provides unique models under certain conditions. The unconstrained PARATUCK2 model has not yet been applied in the literature, but a restricted version of the model, has some

very direct connections to rank-deficient problems where the relationship between constituents in, e.g., a kinetic system causes problems in other modeling approaches.

Finally the extension of PLS to multi-way data has been discussed. Unlike the other models, N-PLS is a two-block model. A set of independent and a set of dependent data are decomposed simultaneously generating a calibration model.

CHAPTER 4

ALGORITHMS

4.1 INTRODUCTION

In this chapter it will be shown how to fit the models that have been described. It is not strictly essential to know the algorithms to use the models. However, it does add a deeper understanding of the methodology and also enables one to accommodate the algorithms to a specific situation if necessary. Most algorithms are based on Alternating Least Squares (ALS) and it thus seems reasonable to first describe the principle behind ALS.

4.2 ALTERNATING LEAST SQUARES

The principle of ALS is old (Yates 1933), and consists of simply dividing the parameters into several sets. Each set of parameters is estimated in a least squares sense conditionally on the remaining parameters. The estimation of parameters is repeated iteratively until no change is observed in the parameter values or in the fit of the model to the data. The reason for dividing the parameters into groups is to make it possible to use simpler algorithms for estimating the parameters. Consider a bilinear model:

$$\hat{\mathbf{X}} = \mathbf{AB}^T. \tag{61}$$

To estimate \mathbf{A} and \mathbf{B} simultaneously using the least squares loss function

$$\min_{\mathbf{A}, \mathbf{B}} \|\mathbf{X} - \mathbf{A}\mathbf{B}^T\|_F^2 \quad (62)$$

is a rather difficult nonlinear problem whereas estimating \mathbf{B} given \mathbf{A} is simply

$$\mathbf{B} = \mathbf{X}^T(\mathbf{A}^+)^T. \quad (63)$$

This will not solve the overall problem, but it will improve any current estimate of \mathbf{B} . Afterwards a new and better estimate of \mathbf{A} given \mathbf{B} can be determined as

$$\mathbf{A} = \mathbf{X}(\mathbf{B}^+)^T \quad (64)$$

and a better estimate of \mathbf{B} can then be obtained etc. This is the basic idea of alternating least squares algorithms. Redefine the global problem (equation 62) into subproblems that are easy to solve. Then iteratively solve these problems until convergence. As all estimates of parameters are least squares estimates such an algorithm may only improve the fit or keep it the same if converged. Therefore the accompanying loss function value will be strictly monotonic decreasing. Since the problem is a bounded-cost problem (the loss function can not be less than zero) convergence follows. This property is very attractive and one of the reasons for the widespread use of ALS (see e.g. de Leeuw et al. 1976).

In many cases the problem may have several local minima, which means that convergence to the global optimum can seldom be guaranteed, but is dependent on data, model, and algorithm. While some ALS-algorithms like NIPALS (Martens & Næs 1989) for fitting a PCA model, or most algorithms for fitting the Tucker3 and N-PLS model (Kroonenberg 1983), are fast and stable, most algorithms for fitting, for example, the PARAFAC model *can* occasionally be problematic for certain types of data (Harshman & Lundy 1984b). Simple repetitions of the analysis can reveal if global convergence has not been achieved, as convergence to the same *local* optimum several consecutive times is unlikely if the analysis is started from different initial parameter sets (see page 121).

The benefit of ALS algorithms is the simplicity of the involved sub-steps as compared to an algorithm working simultaneously on the entire problem, the fact that ALS algorithms are guaranteed to converge and the many variations possible from drawing from the general theory of least squares regression. The drawback in difficult cases can be a rather slow convergence.

Given an array $\underline{\mathbf{X}}$ consider the general model

$$\underline{\mathbf{X}} = f(\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots) + \mathbf{E}. \quad (65)$$

To estimate the parameters \mathbf{A} , \mathbf{B} , \mathbf{C} , etc. an ALS algorithm can be formulated as shown in box 3.

A GENERIC ALS ALGORITHM

BOX 3

1. Initialize the parameters
2. \mathbf{A} is the solution to $\underset{\mathbf{A}}{\operatorname{argmin}} \|\underline{\mathbf{X}} - f(\mathbf{A}, \mathbf{B}, \dots, \mathbf{C})\|_F^2$
3. \mathbf{B} is the solution to $\underset{\mathbf{B}}{\operatorname{argmin}} \|\underline{\mathbf{X}} - f(\mathbf{A}, \mathbf{B}, \dots, \mathbf{C})\|_F^2$
4. Estimate following sets of parameters similarly
5. Return to step 2 until convergence

The first step of initializing the parameters will be treated under the description of each specific algorithm. The last step involves determining whether convergence has been achieved. Usually the iterative algorithm is stopped when either the parameters or the fit of the model does not change much. Notice that in the algorithm outlined above the parameters have been divided into sets that are arranged in matrices. However, this is merely an example. The general idea is to divide the parameters into as few sets as possible in order to avoid the appearance of local minima and slow convergence. Hence, the parameters should be divided into few smaller subsets, but such that each arising subproblem is easily solved. Therefore, if dividing into smaller parameter sets, like solving for individual

columns of a loading matrix, is more practical this is of course also possible.

An issue, which will not be treated in detail here, is the order in which the parameter sets should be estimated. It is most feasible to update the particular set of parameters which will bring about the largest decrease in the loss function value relative to the time it takes to estimate the parameters. In practice, the most common approach is to update the sets of parameters sequentially without considering the relative efficiency of each update.

MODELS

BOX 4

PARAFAC

$$\mathbf{X}^{(I \times J \times K)} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T$$

$$x_{ijk} = \sum_{f=1}^F a_{if} b_{jf} c_{kf}$$

PARAFAC2

$$\mathbf{X}^{(K \times I \times I)} = (\mathbf{C}^T \otimes \mathbf{C}^T)^T \text{diag}(\text{vec} \mathbf{H})(\mathbf{A} \otimes \mathbf{A})^T$$

$$x_{ijk} = \sum_{e=1}^E \sum_{f=1}^F a_{ie} a_{jf} h_{ef} c_{ke} c_{kf}$$

PARATUCK2

$$\mathbf{X}^{(K \times I \times J)} = ((\mathbf{C}^B)^T \otimes (\mathbf{C}^C)^T)^T \text{diag}(\text{vec} \mathbf{H})(\mathbf{B} \otimes \mathbf{A})^T$$

$$x_{ijk} = \sum_{r=1}^R \sum_{s=1}^S a_{ir} b_{js} h_{rs} c_{kr}^A c_{ks}^B$$

TUCKER3

$$\mathbf{X}^{(I \times J \times K)} = \mathbf{A} \mathbf{G} (\mathbf{C}^T \otimes \mathbf{B}^T)$$

$$x_{ijk} = \sum_{d=1}^D \sum_{e=1}^E \sum_{f=1}^F a_{id} b_{je} c_{kf} g_{def}$$

PARAFAC2 is shown in the indirect fitting mode, i.e., \mathbf{X}_k is a cross-product matrix obtained from the raw data

In the following algorithms for estimating the parameters of PARAFAC, PARAFAC2, PARATUCK2, Tucker3, and N-PLS models will be given. For clarity the models are given in Box 4 above. After discussing the basics of

the algorithms several approaches for speeding up the slower ones will be given.

4.3 PARAFAC

The three-way PARAFAC model is defined as

$$\hat{\mathbf{X}} = \mathbf{A}(\mathbf{C}\mathbf{B}|\otimes|\mathbf{B})^T, \quad (66)$$

and the corresponding loss function is

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \|\mathbf{X} - \mathbf{A}(\mathbf{C}\mathbf{B}|\otimes|\mathbf{B})^T\|_F^2. \quad (67)$$

To estimate \mathbf{A} conditionally on \mathbf{B} and \mathbf{C} formulate the optimization problem as

$$\|\mathbf{X} - \mathbf{AZ}^T\|_F^2 \quad (68)$$

where \mathbf{X} is $\underline{\mathbf{X}}$ unfolded to an $I \times JK$ array and $\mathbf{Z} = (\mathbf{C}\mathbf{B}|\otimes|\mathbf{B})$. It then follows that the least squares estimate of \mathbf{A} can be found as

$$\mathbf{A} = \mathbf{X}(\mathbf{Z}^T)^+ = \mathbf{XZ}(\mathbf{Z}^T\mathbf{Z})^{-1}. \quad (69)$$

From the symmetry of the problem it follows that \mathbf{B} and \mathbf{C} can be updated in similar ways, but already Harshman (1970) and Carroll & Chang (1970) noted that a simpler updating scheme is possible due to the special structure of the problem. It is possible to calculate \mathbf{XZ} and $\mathbf{Z}^T\mathbf{Z}$ directly from \mathbf{B} , \mathbf{C} , and \mathbf{X} . It can be shown that

$$\mathbf{XZ} = \mathbf{X}_1\mathbf{B}\mathbf{D}_1 + \mathbf{X}_2\mathbf{B}\mathbf{D}_2 + \dots + \mathbf{X}_K\mathbf{B}\mathbf{D}_K, \quad (70)$$

and

$$\mathbf{Z}^T\mathbf{Z} = (\mathbf{B}^T\mathbf{B}) \circ (\mathbf{C}^T\mathbf{C}), \quad (71)$$

where the operator, \circ , signifies the Hadamard (element-wise) product (Styan 1973). For the three-way model different but similar formulas can be used for each mode (see, e.g., Kiers & Krijnen 1991), but for extending the algorithm to models of an arbitrary number of modes it is sensible to only use the formulae given here and then reshape the data array correspondingly. Thus the updating scheme for a three-way PARAFAC model is based on repeatedly updating **A**, **B**, and **C** as shown in Box 5.

In step 4 some savings can be achieved in case an unweighted loss function is used. For each loading an orthogonal basis can be obtained from a simple Gram-Schmidt orthogonalization. These orthogonalized loadings are called **U**, **V**, and **Z** for the first, second, and third mode respectively. Since these bases span the space of the model exactly the projection onto these leaves the model intact. The projected data reads

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{U}\mathbf{U}^T \mathbf{X}^{(I \times JK)} (\mathbf{Z}\mathbf{Z}^T \otimes \mathbf{V}\mathbf{V}^T) = \mathbf{A}(\mathbf{C} | \otimes | \mathbf{B})^T, \quad (72)$$

and the regression of the data on these bases can be expressed

$$\mathbf{Y}^{(F \times FEF)} = \mathbf{U}^T \mathbf{X}^{(I \times JK)} (\mathbf{Z} \otimes \mathbf{V}), \quad (73)$$

which is a small array having the same sum-of-squares of elements as the model. This array corresponds to the core array of a Tucker3 model (page 45). Instead of calculating the fit by the residuals one may regress the data onto the orthogonalized interim loading matrices and simply calculate the sum-of-squares from the thus obtained small array. The loss function value will equal the sum-of-squares of \mathbf{X} (which need only be calculated once) subtracted the sum-of-squares of the elements of the core array.

INITIALIZING PARAFAC

Good starting values for the ALS algorithm can help speeding up the algorithm and help in assuring that the global minimum is found. Several possible kinds of initializations have been proposed. Harshman & Lundy (1984a) advocate for using random starting values and starting the algorithm from several different starting points. If the same solution is reached several times there is little chance that a local minimum is reached

due to an unfortunate initial guess. In (Sands & Young 80, Burdick et al. 90, Sanchez & Kowalski 90, Li & Gemperline 93) it is proposed to use starting values based on GRAM or similar methods.

PARAFAC ALS ALGORITHM

BOX 5

Initialize **B** and **C**

$$1. \mathbf{Z} = (\mathbf{C} | \otimes | \mathbf{B})$$

$$\mathbf{A} = \mathbf{X}^{(I \times J \times K)} \mathbf{Z} (\mathbf{Z}^T \mathbf{Z})^+$$

$$2. \mathbf{Z} = (\mathbf{C} | \otimes | \mathbf{A})$$

$$\mathbf{B} = \mathbf{X}^{(J \times I \times K)} \mathbf{Z} (\mathbf{Z}^T \mathbf{Z})^+$$

$$3. \mathbf{Z} = (\mathbf{B} | \otimes | \mathbf{A})$$

$$\mathbf{C} = \mathbf{X}^{(K \times I \times J)} \mathbf{Z} (\mathbf{Z}^T \mathbf{Z})^+$$

4. Go to step 1 until relative change in fit is small

With respect to speed, however, there is often little advantage of using these initialization methods, unless the data conform very well to the model. They will often get the algorithm in the right direction, but typically many steps can still be required to find the actual solution (see e.g. Kiers 1998b). Rather, the advantage is if the ALS algorithm tends to get stuck in local minima, a good initialization might help overcoming this problem. However, it may also happen that a GRAM initialization leads to a local minimum, in which case it is of little help. Other practical problem with the eigenvalue-based methods are how to extend them to higher orders as well as using them for constrained models. These problems has not yet been addressed. The most sensible conclusion to draw from this disagreement in the literature is that one is best off using both random and eigenvalue-based initialization. If both methods agree there is no problem. If not, one has to consider what causes the disagreement (see also the discussion on page

121 and 124).

USING THE PARAFAC MODEL ON NEW DATA

Usually when using an existing model on new data, one is interested in estimating the scores of one or several new samples. Assuming that the first mode refers to samples the problem of estimating the scores of a sample \mathbf{X} ($J \times K$) is

$$\underset{\mathbf{a}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{B} \operatorname{diag}(\mathbf{a}) \mathbf{C}^T\|_F^2 = \underset{\mathbf{a}}{\operatorname{argmin}} \|\operatorname{vec} \mathbf{X} - (\mathbf{C} \otimes \mathbf{B}) \mathbf{a}\|_F^2. \quad (74)$$

Note that \mathbf{B} and \mathbf{C} are given from the prior obtained model. The solution to this problem is simply a least squares problem. Define a matrix $\mathbf{Z} = \mathbf{C} \otimes \mathbf{B}$. Then \mathbf{a} is given

$$\mathbf{a} = \mathbf{Z}^+ \operatorname{vec} \mathbf{X}. \quad (75)$$

EXTENDING THE PARAFAC MODEL TO HIGHER ORDERS

The PARAFAC model is easy to extend to higher orders. Consider an F -component four-way PARAFAC model of an $I \times J \times K \times M$ array $\underline{\mathbf{X}}$ given by the four loading matrices, \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} . The corresponding model is

$$\mathbf{X}^{(I \times JKM)} = \mathbf{A}(\mathbf{D} \otimes \mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}. \quad (76)$$

Note the simplicity of extending the model to higher orders with the use of the Khatri-Rao product. To update the first mode loadings \mathbf{A} define the matrix \mathbf{Z} ($JKM \times F$) as

$$\mathbf{Z} = (\mathbf{D} \otimes \mathbf{C} \otimes \mathbf{B}). \quad (77)$$

Then the conditional estimate of \mathbf{A} can be calculated as

$$\mathbf{A} = \mathbf{XZ}(\mathbf{Z}^T \mathbf{Z})^{-1}. \quad (78)$$

As for the three-way model computationally efficient formulas can be used

instead of the direct approach above as already noted by Carroll & Chang (1970). Define $\mathbf{G} = \mathbf{XZ}$. This product can be calculated efficiently using summation signs as

$$g_{if} = \sum_{j=1}^J \sum_{k=1}^K \sum_{m=1}^M x_{ijkm} b_{jf} c_{kf} d_{mf} \quad (79)$$

though if implemented in MATLABTM such an expression would need to be compiled or rewritten in a vectorial way to function efficiently. The matrix $\mathbf{Z}^T \mathbf{Z}$ is easily obtained as

$$\mathbf{Z}^T \mathbf{Z} = (\mathbf{B}^T \mathbf{B}) \circ (\mathbf{C}^T \mathbf{C}) \circ (\mathbf{D}^T \mathbf{D}). \quad (80)$$

4.4 PARAFAC2

Even though the PARAFAC2 model resembles the PARAFAC model it is not possible to fit the PARAFAC2 model with the PARAFAC algorithm unless the second mode loadings are constrained to be orthogonal. R. A. Harshman proposed the PARAFAC2 model in 1972, but it was not until 1993 that an algorithm for fitting the model was published by H. A. L. Kiers. This algorithm is quite complicated due to the special characteristics of the model, and later Kiers et al. (1998) proposed another simpler algorithm described below.

As noted on page 36 the direct fitting PARAFAC2 loss function can be stated

$$\min_{\mathbf{A}, \mathbf{C}, \mathbf{B}, \mathbf{P}_k} \sum_{k=1}^K \|\mathbf{X}_k - \mathbf{A} \mathbf{D}_k (\mathbf{P}_k \mathbf{B})^T\|_F^2, \quad (81)$$

where \mathbf{D}_k is a diagonal matrix holding the k th row of \mathbf{C} . An ALS algorithm for this model is easier to implement than the algorithm for the cross-product matrices. It has the additional advantage of being simpler to constrain to, e.g., non-negativity and also handles missing data more easily. The algorithm for fitting the above model is called a *direct fitting*

approach according to Kruskal (1978), as opposed to an algorithm working on cross-product matrices which is termed an indirect fitting approach (cf. SVD and eigenvalue based approaches for fitting a PCA model). Observe that

$$\min_{\mathbf{A}, \mathbf{C}, \mathbf{B}, \mathbf{P}_k} \sum_{k=1}^K \|\mathbf{X}_k - \mathbf{A}\mathbf{D}_k(\mathbf{P}_k\mathbf{B})^T\|_F^2, \quad (82)$$

is equivalent to

$$\min_{\mathbf{A}, \mathbf{C}, \mathbf{B}, \mathbf{P}_k} \sum_{k=1}^K \|\mathbf{X}_k\mathbf{P}_k - \mathbf{A}\mathbf{D}_k\mathbf{B}^T\|_F^2, \quad (83)$$

PARAFAC2 ALS ALGORITHM

BOX 6

Initialize \mathbf{A} , \mathbf{B} and \mathbf{C}

1. For every k , $k = 1, \dots, K$

$$\mathbf{Q}_k = \mathbf{X}_k^T \mathbf{A} \mathbf{D}_k \mathbf{B}^T$$

$$\mathbf{P}_k = \mathbf{Q}_k (\mathbf{Q}_k^T \mathbf{Q}_k)^{-.5}$$

2. For every k , $k = 1, \dots, K$

$$\mathbf{Y}_k = \mathbf{X}_k \mathbf{P}_k$$

3. Determine \mathbf{A} , \mathbf{B} , and \mathbf{C} from one iteration of a PARAFAC-ALS on \mathbf{Y} .

4. Go to step 1 until relative change in fit is small

which can be seen to be an ordinary PARAFAC model of an array with frontal slabs $\mathbf{X}_k\mathbf{P}_k$. The updates for \mathbf{A} , \mathbf{B} , and \mathbf{C} (\mathbf{D}_k) can therefore be found from an ordinary PARAFAC algorithm on the above array. Note, that the

slabs $\mathbf{X}_k \mathbf{P}_k$ are of size $I \times F$ which is generally smaller than the original slabs ($I \times J$ in case all slabs have the same dimensionality).

The matrices \mathbf{P}_k can be estimated from

$$\underset{\mathbf{P}_k}{\operatorname{argmin}} \|\mathbf{X}_k - \mathbf{M}_k \mathbf{P}_k^T\|_F^2, \quad (84)$$

subject to $\mathbf{P}_k^T \mathbf{P}_k = \mathbf{I}$, the matrix \mathbf{I} being the identity matrix. Above

$$\mathbf{M}_k = \mathbf{A} \mathbf{D}_k \mathbf{B}^T. \quad (85)$$

The solution is described on page 154 and is³

$$\mathbf{P}_k = \mathbf{X}_k^T \mathbf{M}_k (\mathbf{M}_k^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{M}_k)^{-1/2}. \quad (86)$$

The complete algorithm follows directly and is shown in Box 6.

INITIALIZING PARAFAC2

Kiers (1993) suggests starting the PARAFAC2 algorithm by setting \mathbf{C} to ones and \mathbf{A} to the first singular vectors of $\sum \mathbf{X}_k \mathbf{X}_k^T$. For indirect fitting $\mathbf{B}^T \mathbf{B}$ can then be determined from \mathbf{A} and \mathbf{C} . In line with the other multi-way algorithms it is suggested here to use random values instead or additionally in order to be able to verify convergence and assess uniqueness.

USING THE PARAFAC2 MODEL ON NEW DATA

Predicting the scores of a new sample is somewhat complicated and depends on which modes play which part in the model. If the third mode is the object mode the following loss function applies. The data of the new sample is called \mathbf{X} ($I \times J$)

$$\underset{\mathbf{c}, \mathbf{P}}{\min} \|\mathbf{X} - \mathbf{A} \operatorname{diag}(\mathbf{c})(\mathbf{P}\mathbf{B})^T\|_F^2. \quad (87)$$

³. $(\mathbf{S})^{-1/2}$ is normally calculated using SVD. If \mathbf{S} does not have full rank, a truncated SVD has to be used. This can happen if elements in \mathbf{D}_k are zero, e.g., due to non-negativity constraints.

The uniqueness of the solution follows from the special structure of \mathbf{P} (orthogonal). An algorithm for solving the problem is identical to the PARAFAC2 algorithm, keeping \mathbf{A} and \mathbf{B} fixed, i.e., not updating these.

EXTENDING THE PARAFAC2 MODEL TO HIGHER ORDERS

A four-way array $\underline{\mathbf{X}}$ of size $I \times J \times K \times M$ can be modeled by a similar approach as for the three-way case. However, the analyst must choose which mode is the 'problematic' mode (mode two above), where only the cross-product of the profiles are modeled, *and* also over which mode these cross-products are to be estimated (mode three above).

Recall, that for the three-way case each two-way slab, \mathbf{X}_k , is post-multiplied by an orthogonal matrix \mathbf{P}_k . The array with two-way slabs $\mathbf{X}_k \mathbf{P}_k$ can be modeled by an ordinary three-way PARAFAC model. For the four-way case each three-way array, $\underline{\mathbf{X}}_m$ is postmultiplied by an orthogonal matrix \mathbf{P}_m . The four-way array with three-way (unfolded) slabs $\mathbf{X}_m \mathbf{P}_m$ can then be modeled by a four-way PARAFAC model. Hence the principle remains the same even though the order of the array changes.

4.5 PARATUCK2

The loss function defining the three-way PARATUCK2 model is

$$\min_{\mathbf{A}, \mathbf{D}_k^{\mathbf{A}}, \mathbf{H}, \mathbf{D}_k^{\mathbf{B}}, \mathbf{B}^{\mathbf{T}}}_{k=1} \sum_{k=1}^K \|\mathbf{X}_k - \mathbf{A} \mathbf{D}_k^{\mathbf{A}} \mathbf{H} \mathbf{D}_k^{\mathbf{B}} \mathbf{B}^{\mathbf{T}}\|_{\mathbf{F}}^2 \quad k = 1, \dots, K \quad (88)$$

and an ALS algorithm for fitting the model will be shown below. The algorithm looks a little complicated, but it follows directly from the structural model.

To update \mathbf{A} conditionally on the remaining parameters observe the following. For one level of k the model of \mathbf{X}_k is

$$\mathbf{X}_k = \mathbf{A} \mathbf{F}_k + \mathbf{E}_k, \quad k = 1, \dots, K \quad (89)$$

where

$$\mathbf{F}_k = \mathbf{D}_k^{\mathbf{A}} \mathbf{H} \mathbf{D}_k^{\mathbf{B}} \mathbf{B}^{\mathbf{T}}. \quad k = 1, \dots, K \quad (90)$$

This follows directly from the definition of the PARATUCK2 model (equation 88). As \mathbf{A} remains fixed over all ks ($k=1\dots K$) the simultaneous least squares solution over all ks must be given by

$$[\mathbf{X}_1 \ \mathbf{X}_2 \ \dots \ \mathbf{X}_K] = \mathbf{A}[\mathbf{F}_1 \ \mathbf{F}_2 \ \dots \ \mathbf{F}_K] + [\mathbf{E}_1 \ \mathbf{E}_2 \ \dots \ \mathbf{E}_K]. \quad (91)$$

From this equation follows that the estimate of \mathbf{A} is

$$\mathbf{A} = \mathbf{X}(\mathbf{F}^+)^{\mathbf{T}}, \quad (92)$$

where $\mathbf{F} = [\mathbf{F}_1 \ \mathbf{F}_2 \ \dots \ \mathbf{F}_K]$.

The estimation of the k th row of $\mathbf{C}^{\mathbf{A}}$ is based on the following. The PARATUCK2 model is

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k^{\mathbf{A}} \mathbf{H} \mathbf{D}_k^{\mathbf{B}} \mathbf{B}^{\mathbf{T}} + \mathbf{X} \mathbf{E}_k. \quad (93)$$

Substituting

$$\mathbf{F}_k = \mathbf{B} \mathbf{D}_k^{\mathbf{B}} \mathbf{H}^{\mathbf{T}}, \quad (94)$$

the model is

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k^{\mathbf{A}} \mathbf{F}_k^{\mathbf{T}} + \mathbf{X} \mathbf{E}_k. \quad (95)$$

Since $\mathbf{D}_k^{\mathbf{A}}$ is a diagonal matrix the model can also be expressed

$$\text{vec} \mathbf{X}_k = (\mathbf{F}_k \otimes \mathbf{I})^{\mathbf{T}} (\mathbf{C}_{(k,:)}^{\mathbf{A}})^{\mathbf{T}} + \text{vec} \mathbf{E}_k \quad (96)$$

where $\mathbf{C}_{(k,:)}^{\mathbf{A}}$ is the k th row of $\mathbf{C}^{\mathbf{A}}$. From this the least squares solution follows

as

$$\mathbf{C}_{(k,:)}^A = (\mathbf{Z}^+ \text{vec} \mathbf{X}_k)^T, \quad (97)$$

where

$$\mathbf{Z} = (\mathbf{F}_k \otimes \mathbf{A}). \quad (98)$$

To estimate the interaction matrix \mathbf{H} do the following. For one level in the third mode, k , the PARATUCK2 model is

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k^A \mathbf{H} \mathbf{D}_k^B \mathbf{B}^T \Rightarrow$$

$$\text{vec} \mathbf{X}_k = \text{vec}(\mathbf{A} \mathbf{D}_k^A \mathbf{H} \mathbf{D}_k^B \mathbf{B}^T) \Rightarrow$$

$$\text{vec} \mathbf{X}_k = (\mathbf{B} \mathbf{D}_k^B \otimes \mathbf{A} \mathbf{D}_k^A) \text{vec} \mathbf{H} \Rightarrow$$

$$\begin{bmatrix} \text{vec} \mathbf{X}_1 \\ \text{vec} \mathbf{X}_2 \\ \vdots \\ \text{vec} \mathbf{X}_K \end{bmatrix} = \begin{bmatrix} (\mathbf{B} \mathbf{D}_1^B \otimes \mathbf{A} \mathbf{D}_1^A) \\ (\mathbf{B} \mathbf{D}_2^B \otimes \mathbf{A} \mathbf{D}_2^A) \\ \vdots \\ (\mathbf{B} \mathbf{D}_K^B \otimes \mathbf{A} \mathbf{D}_K^A) \end{bmatrix} \text{vec} \mathbf{H}. \quad (99)$$

Thus \mathbf{H} can be found as

$$\text{vec} \mathbf{H} = \mathbf{Z}^+ \text{vec} \mathbf{X}, \quad (100)$$

where

$$\mathbf{Z} = \begin{bmatrix} (\mathbf{BD}_1^{\mathbf{B}} \otimes \mathbf{AD}_1^{\mathbf{A}}) \\ (\mathbf{BD}_2^{\mathbf{B}} \otimes \mathbf{AD}_2^{\mathbf{A}}) \\ \vdots \\ (\mathbf{BD}_K^{\mathbf{B}} \otimes \mathbf{AD}_K^{\mathbf{A}}) \end{bmatrix}. \quad (101)$$

The problem of estimating $\mathbf{C}^{\mathbf{B}}$ and \mathbf{B} is symmetrical to the description of the estimation of $\mathbf{C}^{\mathbf{A}}$ and \mathbf{A} with all arrays and matrices arranged accordingly.

INITIALIZING PARATUCK2

Unless prior estimates of the parameters are available some other scheme for initializing the PARATUCK2 model must be used. Using random values, it is possible to redo the analysis and check for nonuniqueness and convergence by observing if the same model (fit) is obtained every time. Alternatively loading vectors obtained from doing an unfolded PCA (Tucker1) analysis on each mode can be used for initialization. The third mode loadings $\mathbf{C}^{\mathbf{A}}$ and $\mathbf{C}^{\mathbf{B}}$ could be set to the first R and S PCA loading vectors respectively. The matrix \mathbf{H} can be set to the identity if R equals S and random numbers otherwise.

USING THE PARATUCK2 MODEL ON NEW DATA

If the first mode is the sample mode the prediction of the scores of a new sample \mathbf{X} ($J \times K$) is defined as the minimum of the loss function

$$\min_{\mathbf{a}} \|(\text{vec}\mathbf{X})^{\mathbf{T}} - \mathbf{a} \left[\mathbf{D}_1^{\mathbf{A}} \mathbf{H} \mathbf{D}_1^{\mathbf{B}} \mathbf{B}^{\mathbf{T}} \quad \mathbf{D}_2^{\mathbf{A}} \mathbf{H} \mathbf{D}_2^{\mathbf{B}} \mathbf{B}^{\mathbf{T}} \quad \dots \quad \mathbf{D}_K^{\mathbf{A}} \mathbf{H} \mathbf{D}_K^{\mathbf{B}} \mathbf{B}^{\mathbf{T}} \right]\|_F^2 \quad (102)$$

where \mathbf{a} is a $1 \times F$ row vector. The solution follows immediately from equation 92.

EXTENDING THE PARATUCK2 MODEL TO HIGHER ORDERS

The PARATUCK2 model extends itself easily to higher orders. As the

model is not symmetric it has to be decided in which way the model should be extended. Consider for example a situation where the first mode would really be an unfolded mode consisting originally of two modes. If these two modes for example are fluorescence excitation and emission, then it immediately follows that instead of estimating the first mode loadings unfolded, it could be more appropriately modeled bilinearly. A similar argument could hold for the second mode. For the third mode a similar setup could probably also be envisioned, though in the basic PARATUCK2 model this seems less fruitful.

4.6 TUCKER MODELS

The loss function for the Tucker3 model is

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{G}} \|\mathbf{X} - \mathbf{A}\mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T\|_F^2 \quad (103)$$

subject to \mathbf{A} , \mathbf{B} , and \mathbf{C} are orthogonal. The column dimensions of \mathbf{A} , \mathbf{B} , and \mathbf{C} , are D , E , and F respectively, and \mathbf{G} is the $D \times E \times F$ core array. The orthonormality constraints can be relaxed if for example having non-negative parameters is desired. In the following only the orthonormal model will be considered. The Tucker3 model is a quadrilinear model as the model consists of four sets of parameters each set being conditionally linear.

Originally the algorithms proposed to fit the Tucker models were not least squares algorithms, but later Kroonenberg & de Leeuw (1980) devised such algorithms based on alternating least squares. The primary algorithm is called TUCKALS3 (or TUCKALS2 for the Tucker2 model) for fitting the model in a least squares sense with orthonormal loading vectors. Röhmel et al. (1983) and ten Berge et al. (1987) gave additional results for fitting the Tucker models where the requirement of orthogonal loading matrices were relaxed. Note, that relaxing these constraints are inactive in the sense that the fit does not improve.

The core array \mathbf{G} can be found conditional on \mathbf{A} , \mathbf{B} , and \mathbf{C} by a simple regression of \mathbf{X} onto \mathbf{A} , \mathbf{B} , and \mathbf{C} . In matrix notation this reads

$$\mathbf{G}^{(D \times EF)} = \mathbf{A}^T \mathbf{X}^{(I \times JK)} (\mathbf{B} \otimes \mathbf{C}) , \quad (104)$$

for orthogonal loading matrices. If the model is perfect, then $\underline{\mathbf{G}}$ contains the exact same information as $\underline{\mathbf{X}}$ merely expressed using different 'coordinates'. From the definition of $\underline{\mathbf{G}}$ it follows that the Tucker3 model of $\underline{\mathbf{X}}$ can be stated

$$\mathbf{AG}(\mathbf{C}^T \otimes \mathbf{B}^T) = \mathbf{AA}^T \mathbf{X} (\mathbf{C} \otimes \mathbf{B}) (\mathbf{C}^T \otimes \mathbf{B}^T) = \mathbf{AA}^T \mathbf{X} (\mathbf{CC}^T \otimes \mathbf{BB}^T) \quad (105)$$

For \mathbf{B} and \mathbf{C} fixed it follows that finding the optimal \mathbf{A} is equal to minimizing the norm of $(\mathbf{X} - \mathbf{AA}^T \mathbf{M})$, where $\mathbf{M} \equiv \mathbf{X}(\mathbf{CC}^T \otimes \mathbf{BB}^T)$. Let $\mathbf{P}_A = \mathbf{AA}^+$ be the orthogonal projector onto the column-space of \mathbf{A} and $\mathbf{Q}_A = \mathbf{I} - \mathbf{AA}^+$ the anti-projector, i.e., the projector onto the null-space of \mathbf{A} . If \mathbf{A} has orthonormal columns then $\mathbf{A}^+ = \mathbf{A}^T$, hence $\mathbf{AA}^T = \mathbf{P}_A$ is a projector. The same applies to \mathbf{BB}^T and \mathbf{CC}^T , and also to \mathbf{HH}^T , where $\mathbf{H} \equiv \mathbf{C} \otimes \mathbf{B}$. The minimization problem can then be expressed

$$\min_{\mathbf{A}} \|\mathbf{X} - \mathbf{P}_A \mathbf{X} \mathbf{P}_H\|_F^2 = \quad (106)$$

$$\min_{\mathbf{A}} \|\mathbf{X} \mathbf{Q}_H + \mathbf{X} \mathbf{P}_H - \mathbf{P}_A \mathbf{X} \mathbf{P}_H\|_F^2 .$$

As \mathbf{Q}_H and \mathbf{P}_H span orthogonal spaces the above can be expressed

$$\min_{\mathbf{A}} (\|\mathbf{X} \mathbf{Q}_H\|_F^2 + \|\mathbf{X} \mathbf{P}_H - \mathbf{P}_A \mathbf{X} \mathbf{P}_H\|_F^2) . \quad (107)$$

As $\|\mathbf{X} \mathbf{Q}_H\|_F^2$ is constant this is equivalent to

$$\min_{\mathbf{A}} \|\mathbf{X} \mathbf{P}_H - \mathbf{P}_A \mathbf{X} \mathbf{P}_H\|_F^2 = \quad (108)$$

$$\min_{\mathbf{A}} \|\mathbf{Q}_A \mathbf{X} \mathbf{P}_H\|_F^2 =$$

$$\max_{\mathbf{A}} \|\mathbf{P}_A \mathbf{X} \mathbf{P}_H\|_F^2 . \quad (109)$$

The latter maximum is attained when $\mathbf{M} = \mathbf{X}\mathbf{P}_H$ is projected onto its own 'principal sub-space', i.e, when the column space of \mathbf{A} equals the space spanned by the first D left singular vectors of \mathbf{M} . These equal the space spanned by the first D eigenvectors of $\mathbf{M}\mathbf{M}^T$. Since $\mathbf{M}\mathbf{M}^T = \mathbf{X}\mathbf{P}_H\mathbf{P}_H^T\mathbf{X}^T = \mathbf{X}\mathbf{P}_H\mathbf{X}^T = \mathbf{X}\mathbf{H}\mathbf{H}^T\mathbf{X}^T$, using the property of symmetry and idempotency of orthogonal projection operators, the sought matrix can be obtained from the matrix $\mathbf{X}\mathbf{H} = \mathbf{X}(\mathbf{C} \otimes \mathbf{B})$ by extracting the left singular vectors⁴. From this a generic Tucker3 algorithm can be derived as described in Box 7 (see also Kroonenberg & de Leeuw 1980). The expression $[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}, D)$ means that a truncated SVD of \mathbf{X} is performed retaining the first D components. The matrix \mathbf{U} contains the first D left singular vectors, \mathbf{V} the first D right singular vectors, and \mathbf{S} is a diagonal matrix holding the first D singular values in its diagonal.

TUCKER3 ALS ALGORITHM

BOX 7

Initialize \mathbf{B} and \mathbf{C}

1. $[\mathbf{A}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}), D)$
2. $[\mathbf{B}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(J \times IK)}(\mathbf{C} \otimes \mathbf{A}), E)$
3. $[\mathbf{C}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(K \times IJ)}(\mathbf{B} \otimes \mathbf{A}), F)$
4. Go to step 1 until relative change in fit is small
5. $\mathbf{G} = \mathbf{A}^T \mathbf{X}(\mathbf{C} \otimes \mathbf{B})$

The algorithms for fitting the Tucker3 model are usually among the fastest of the multi-way algorithms. However, for large problems the fitting procedure will require increasing computational efforts. In an attempt to

⁴. The above derivation originally stems from Andersson & Bro (1998) but was kindly suggested modified as here by S. de Jong.

improve the speed several different implementations of the Tucker3 algorithm have been explored in Andersson & Bro (1998). The interest was specifically aimed at developing an algorithm in the MATLABTM environment that is suitable for large data arrays. Nine different implementations were developed and tested on real and simulated data.

The size of the arrays considered were such that the computer has physical memory to hold the array and intermediate working arrays. If the array size exceeds what the physical computer memory can hold other problems arise and other algorithms may be better; e.g., the one proposed by Alsberg & Kvalheim (1994a & b). Their algorithm does not compute an exact least squares solution, but rather approximate the solution by finding suitable truncated bases for the different modes. An efficient algorithm based on loss-less compression for the case of one very large mode has also been proposed (Kiers et al. 1992).

There are several important steps in implementing the Tucker3 algorithm for large problems: avoiding the use of Kronecker products and unnecessarily large working matrices and using a fast method for estimating a truncated orthonormal basis for a matrix. In the following the update of \mathbf{A} will be used as an example.

It is common to express the Tucker3 model and algorithm using Kronecker products (equation 103). While intuitively appealing for providing simple matrix expressions for array models, Kronecker products should not be used in actual implementation as it leads to large intermediate arrays and excessively many elementary operations. Instead the arrays should be rearranged continuously as exemplified below. This is justified by the fact that rearranging a matrix is fast as it only requires changes in indices, not real computations. The Kronecker multiplication $\mathbf{X}(\mathbf{C} \otimes \mathbf{B})$ can be written in matrix notation:

$$\mathbf{W}^{(E \times IK)} = \mathbf{B}^T \mathbf{X}^{(J \times IK)}$$

$$\mathbf{V}^{(F \times IE)} = \mathbf{C}^T \mathbf{W}^{(K \times IE)}$$

$$\mathbf{X}(\mathbf{C} \otimes \mathbf{B}) = \mathbf{V}^{(I \times EF)}. \tag{110}$$

Though more complicated to look at, this way of computing the product is faster than directly using the Kronecker products especially for large arrays.

The essential part of the Tucker3 algorithm is the estimation of orthonormal bases, \mathbf{A} , \mathbf{B} , and \mathbf{C} . Using $\mathbf{X}(\mathbf{C}\mathbf{C}^T \otimes \mathbf{B}\mathbf{B}^T)$, the size of the matrix from which \mathbf{A} is estimated is $I \times JK$. Using $\mathbf{X}(\mathbf{C} \otimes \mathbf{B})$ the size is only $I \times EF$. In addition, the computation of $\mathbf{X}(\mathbf{C} \otimes \mathbf{B})$ is faster than the computation of $\mathbf{X}(\mathbf{C}\mathbf{C}^T \otimes \mathbf{B}\mathbf{B}^T)$. In Andersson & Bro (1998) several procedures have been tested for determining \mathbf{A} given the matrix $\mathbf{X}(\mathbf{C} \otimes \mathbf{B})$

- SVD
- Bauer-Rutishauser (Rutishauser 1969, Bauer 1957)
- Gram-Schmidt orthogonalization (Longley 1984)
- QR orthogonalization
- NIPALS

For details on the implementations please refer to Andersson & Bro (1998). For large arrays it is found that NIPALS is the most efficient algorithm in terms of speed. This as opposed to Kroonenberg et al. (1989) who suggested and compared implementations of the Gram-Schmidt orthogonalization and Bauer-Rutishauser.

INITIALIZING TUCKER3

A straightforward initialization method often suggested is to use the singular vectors from an SVD in each unfolded mode, i.e.:

$$[\mathbf{B}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(J \times I \times K)}, E) \quad (111)$$

$$[\mathbf{C}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(K \times I \times J)}, F). \quad (112)$$

Here the expression $[\mathbf{B}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(J \times I \times K)}, E)$ means that \mathbf{B} is set equal to the first E left singular vectors of an SVD of the matrix $\mathbf{X}^{(J \times I \times K)}$. The algorithm is then started by estimating \mathbf{A} given these initial values of \mathbf{B} and \mathbf{C} . A slight change is suggested here.

$$[\mathbf{B}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X}^{(J \times I \times K)}, E) \quad (113)$$

$$\mathbf{R}^{(E \times I \times K)} = \mathbf{B}^T \mathbf{X}^{(J \times I \times K)}$$

$$[\mathbf{C}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{R}^{(K \times I \times E)}, F). \quad (114)$$

When estimating \mathbf{A} this is done from the matrix $\mathbf{C}^T \mathbf{R}$ rearranged appropriately. In this way, the initial \mathbf{A} , \mathbf{B} and \mathbf{C} are likely to be closer to solution than using the former approach. This can be explained as follows.

Both initialization methods use the same initialization of \mathbf{B} . Let \mathbf{C}^{opt} be the loading matrix obtained from equation 114 and \mathbf{C}^{svd} the one obtained from equation 112. Considering a model where \mathbf{B} is given as above \mathbf{C}^{opt} is the solution to

$$\min_{\mathbf{C}} \|\mathbf{X} - \mathbf{X}(\mathbf{C}\mathbf{C}^T \otimes \mathbf{B}\mathbf{B}^T)\|_F^2. \quad (115)$$

hence \mathbf{C}^{opt} will be the one that lowers the loss function of this model maximally given \mathbf{B} , while this can then not be the case for \mathbf{C}^{svd} unless it is identical to \mathbf{C}^{opt} which will only happen for noise-free data. The matrix \mathbf{C}^{svd} would be the optimal \mathbf{C} if \mathbf{B} was not known. The reason for this is that to find the optimal \mathbf{C} given \mathbf{B} only the part of \mathbf{X} which is within the column space of \mathbf{B} has to be considered. This is exactly what is obtained using the above approach. Thus the projected data matrix

$$\mathbf{X}(\mathbf{C}\mathbf{C}^T \otimes \mathbf{B}\mathbf{B}^T) \quad (116)$$

from which \mathbf{A} is subsequently estimated preserves the maximal amount of the original variation in \mathbf{X} when \mathbf{C} is equal to \mathbf{C}^{opt} . Therefore the model obtained from the least squares solution \mathbf{A} to

$$\min_{\mathbf{A}} \|\mathbf{X} - \mathbf{A}\mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T\|_F^2 \quad (117)$$

is likely to have a lower fit when using \mathbf{C}^{opt} than when using \mathbf{C}^{svd} . This can also be stated in the following way: both initialization methods (estimating \mathbf{A} , \mathbf{B} , and \mathbf{C}) give an interim solution to the Tucker3 model. Using unfolding and SVD this solution is found only under the restrictions that \mathbf{B} and \mathbf{C} are optimal rank-reduced approximations of the respective variable spaces, and

that \mathbf{A} is the least squares solution to the Tucker3 model given \mathbf{B} and \mathbf{C} . Using the suggested approach the additional constraint is imposed that \mathbf{C} is optimal not only with respect to the variable space in the third mode but also with respect to the known part of the Tucker3 model (\mathbf{B}).

That this leads to a better initial setting is not guaranteed but most often it is the case. In addition and importantly the matrices from which the initial estimates are obtained are smaller hence faster to compute. The gain in speed will be higher the higher the order of the array is and the larger the dimensions of the array are.

USING THE TUCKER MODEL ON NEW DATA

Predicting the scores of a new sample, \mathbf{X} , of size $J \times K$ amounts to minimizing

$$\min_{\mathbf{a}} \|(\text{vec}\mathbf{X})^T - \mathbf{a}^T \mathbf{G}(\mathbf{C}^T \otimes \mathbf{B}^T)\|_F^2 \quad (118)$$

This is a least squares problem and easily solved as such.

EXTENDING THE TUCKER3 MODEL TO HIGHER ORDERS

As for PARAFAC the extension to higher orders is quite simple due to the symmetry of the model. For a four-way model the loss function is

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{G}} \|\mathbf{X} - \mathbf{A}\mathbf{G}(\mathbf{D}^T \otimes \mathbf{C}^T \otimes \mathbf{B}^T)\|_F^2 \quad (119)$$

and an algorithm for updating, e.g., \mathbf{A} is simply based on finding the best-fitting D -dimensional sub-space of the matrix $\mathbf{H} = \mathbf{X}(\mathbf{D} \otimes \mathbf{C} \otimes \mathbf{B})$ analogously to the three-way model. General N -way algorithms for the Tucker3 model have been described by Röhmel et al. (1983) and Kapteyn et al. (1986).

4.7 MULTILINEAR PARTIAL LEAST SQUARES REGRESSION

The loss function of PLS is difficult to state globally. Rather a component-wise loss function can be derived. To extend PLS to three-way arrays,

consider first the two-way PLS model. The ordinary two-way PLS1 algorithm (bi-PLS1) can be described as consisting of two steps. For each component a rank-one model is built of both \mathbf{X} and \mathbf{y} . Then these models are subtracted from \mathbf{X} and \mathbf{y} , and a new set of components is found from the residuals. The calculation of components, is the essential part of the algorithm, and can be regarded as a problem of determining a weight vector, \mathbf{w} , to maximize a certain function.

To calculate a component in two-way PLS, a one-component model of \mathbf{X} is sought of the form

$$\hat{x}_{ij} = t_i w_j \quad (120)$$

where the t s are scores and the w s weights. For a given weight vector, \mathbf{w} , the least squares solution of determining \mathbf{t} is simply obtained as

$$\mathbf{t} = \mathbf{X}\mathbf{w} \quad (121)$$

or using summation signs

$$t_i = \sum_{j=1}^J x_{ij} w_j . \quad (122)$$

The objective function for finding the first component is given

$$\max_{\mathbf{w}} \left[\text{cov}(\mathbf{t}, \mathbf{y}) \mid t_i = \sum_{j=1}^J x_{ij} w_j \wedge \|\mathbf{w}\|_F^2 = 1 \right] . \quad (123)$$

This expression tells to find a vector \mathbf{w} , that yields a score vector, \mathbf{t} , with maximal covariance with \mathbf{y} . The covariance between \mathbf{t} and \mathbf{y} can also be expressed using summations

$$\max_{\mathbf{w}} \left[\sum_{i=1}^I t_i y_i \mid t_i = \sum_{j=1}^J x_{ij} w_j \wedge \|\mathbf{w}\|_F^2 = 1 \right]. \quad (124)$$

This equation is not strictly correct, since there is no correction for degrees of freedom, but as this correction is constant for a given component, it will not affect the maximization. Also equation 124 does not express the covariance, if \mathbf{X} has not been centered. The expressions for covariance and the least squares solution of finding the t s can be combined to

$$\max_{\mathbf{w}} \left[\sum_{i=1}^I \sum_{j=1}^J y_i x_{ij} w_j \mid \|\mathbf{w}\|_F^2 = 1 \right] \quad (125)$$

Since \mathbf{X} and \mathbf{y} are known beforehand the summation over i can be done before actually estimating \mathbf{w} . This summation will yield a vector of size $J \times 1$, that is called \mathbf{z} leading to

$$\max_{\mathbf{w}} \left[\sum_{j=1}^J z_j w_j \mid \|\mathbf{w}\|_F^2 = 1 \right]. \quad (126)$$

Since \mathbf{w} is restricted to be of length one, the maximum value of the above expression is reached, when \mathbf{w} is a unit vector in same direction as \mathbf{z} . Therefore, the solution of finding \mathbf{w} yields

$$\mathbf{w} = \frac{\mathbf{z}}{\|\mathbf{z}\|_F} = \frac{\mathbf{X}^T \mathbf{y}}{\|\mathbf{X}^T \mathbf{y}\|_F}. \quad (127)$$

This in essence defines the bilinear PLS model. For further details on the PLS model the reader is referred to the literature (e.g. Martens & Næs 1989, de Jong & Phatak 1997).

For a three-way array of independent variables the goal of the algorithm

is to make a decomposition of the array $\underline{\mathbf{X}}$ into triads. A triad consists of one score vector (\mathbf{t}) and two weight vectors; one in the second mode called \mathbf{w}^J and one in the third mode called \mathbf{w}^K . The model of $\underline{\mathbf{X}}$ is given

$$\hat{x}_{ijk} = t_i w_j^J w_k^K, \quad i=1, \dots, I; j=1, \dots, J; k=1, \dots, K \quad (128)$$

i.e., a trilinear model just like the three-way PARAFAC model. By the same reasoning as for two-way PLS, the tri-PLS model can be expressed as the problem of finding suitable unit-length weight vectors \mathbf{w}^J and \mathbf{w}^K . Through the model of equation 128 the least squares scores are given as

$$t_i = \sum_{j=1}^J \sum_{k=1}^K x_{ijk} w_j^J w_k^K, \quad i = 1, \dots, I \quad (129)$$

The problem is to find a set of weight vectors \mathbf{w}^J and \mathbf{w}^K that produces a score vector with maximal covariance with \mathbf{y} . Both \mathbf{w}^J and \mathbf{w}^K are, as before, of length one, but these constraints will be left out of the expressions for simplicity. The optimization criterion can be expressed

$$\begin{aligned} \max_{\mathbf{w}^J, \mathbf{w}^K} \left[\sum_{i=1}^I t_i y_i \mid t_i = \sum_{j=1}^J \sum_{k=1}^K x_{ijk} w_j^J w_k^K \right] = \\ \max_{\mathbf{w}^J, \mathbf{w}^K} \left[\sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K y_i x_{ijk} w_j^J w_k^K \right] = \\ \max_{\mathbf{w}^J, \mathbf{w}^K} \left[\sum_{j=1}^J \sum_{k=1}^K z_{jk} w_j^J w_k^K \right] \end{aligned} \quad (130)$$

where \mathbf{Z} is now a matrix instead of a vector. To maximize this expression, formulate it in terms of matrices as

$$\max_{\mathbf{w}^J, \mathbf{w}^K} [(\mathbf{w}^J)^T \mathbf{Z} \mathbf{w}^K] \Rightarrow$$

$$(\mathbf{w}^J, \mathbf{s}, \mathbf{w}^K) = \text{SVD}(\mathbf{Z}, 1) \quad (131)$$

where $\text{SVD}(\mathbf{Z}, 1)$ means the first component of a singular value decomposition of \mathbf{Z} . The problem of finding \mathbf{w}^J and \mathbf{w}^K is simply accomplished by calculating this set of vectors. This follows directly from the properties of SVD (see for instance Malinowski 1991, paragraph 3.3.2).

tri-PLS1 ALGORITHM

BOX 8

center \mathbf{X} and \mathbf{y} and let \mathbf{y}_0 equal \mathbf{y}
 $f=1$

1. Calculate \mathbf{Z}
2. Determine \mathbf{w}^J and \mathbf{w}^K by SVD
3. Calculate \mathbf{t} . $\mathbf{T} = [\mathbf{t}_1 \ \mathbf{t}_2 \ \dots \ \mathbf{t}_f]$
4. $\mathbf{b} = (\mathbf{T}^T \mathbf{T})^{-1} \mathbf{T}^T \mathbf{y}_0$
5. Each sample, \mathbf{X}_i is replaced with
 $\mathbf{X}_i - \mathbf{t}_i \mathbf{w}^J (\mathbf{w}^K)^T$ and $\mathbf{y} = \mathbf{y}_0 - \mathbf{T} \mathbf{b}$
6. $f = f + 1$. Continue from 1 until proper description of \mathbf{y}_0

index f omitted on \mathbf{w}^J , \mathbf{w}^K , \mathbf{t} , and \mathbf{b} for brevity

The complete trilinear PLS1 algorithm follows from the above and the two-way PLS algorithm (Box 8⁵).

As the score vectors from different components are not orthogonal, the calculation of the regression coefficients in step 4 has to be performed

⁵. Note that in step four of the algorithm for N-PLS1 in Bro (96) the residual \mathbf{y} is used instead of \mathbf{y}_0 in the regression step. This however, will lead to the exact same model and is also the algorithm obtained by using the general N-PLS m model for one dependent variable.

taking all calculated score vectors into account.

The algorithm outlined above corresponds to the bilinear PLS1, in that there is only one dependent variable, \mathbf{y} . If several dependent variables are present, it is possible to model each dependent variable separately by this algorithm, but it is also possible to calibrate for all analytes simultaneously as in the PLS2 algorithm. The algorithm for several dependent variables, which is iterative, is shown in Box 9. It follows directly from tri-PLS1 and bi-PLS2.

Tri-PLS2 ALGORITHM

BOX 9

center \mathbf{X} and \mathbf{Y}

let \mathbf{u} equal a column in \mathbf{Y} .

$f=1$

1. Calculate the matrix \mathbf{Z} using \mathbf{X} and \mathbf{u}
2. Determine \mathbf{w}^J and \mathbf{w}^K by SVD
3. Calculate \mathbf{t}
4. $\mathbf{q} = \mathbf{Y}^T \mathbf{t} / |\mathbf{Y}^T \mathbf{t}|$
5. $\mathbf{u} = \mathbf{Y} \mathbf{q}$
7. If convergence continue, else step 1.
8. $\mathbf{b} = (\mathbf{T}^T \mathbf{T})^{-1} \mathbf{T}^T \mathbf{u}$
9. $\mathbf{X}_i = \mathbf{X}_i - t_i \mathbf{w}^J (\mathbf{w}^K)^T$ and $\mathbf{Y} = \mathbf{Y} - \mathbf{T} \mathbf{b} \mathbf{q}^T$
10. $f = f + 1$. Continue from 1 until proper description of \mathbf{Y}

index f omitted on \mathbf{w}^J , \mathbf{w}^K , \mathbf{t} , \mathbf{u} , \mathbf{q} , and \mathbf{b} for brevity

ALTERNATIVE N-PLS ALGORITHMS

It is possible to calculate residuals in \mathbf{X} by an extra set of loading vectors (\mathbf{p}) just as in ordinary PLS. To calculate these, step 1 and 2 in the algorithm are repeated, but now \mathbf{t} takes the place of \mathbf{y} , and \mathbf{p}^J and \mathbf{p}^K take the place of \mathbf{w}^J and \mathbf{w}^K . Thus, a trilinear model is calculated conditionally on \mathbf{t} . However, this will not yield orthogonal scores. Since that is the primary purpose of introducing extra loadings in the two-way PLS model, it is

omitted here. In this respect the multilinear model is quite similar to the bi-PLS version originally proposed by Martens (see Martens & Næs 1989), as also elaborated on by Smilde (1997a). It is also possible to orthogonalize the independent variables prior to estimating the components as suggested by Ståhle (1989). However, in correspondence with the equality of the orthogonal and non-orthogonal scores two-way PLS model, this does not make a difference with respect to the predictions as shown by de Jong (1998). It does, however, lead to a more complicated, unfolded, model of $\underline{\mathbf{X}}$. In fact de Jong (98) has shown that the deflation of $\underline{\mathbf{X}}$ is immaterial as long as \mathbf{y} is deflated properly. Hence the three-way version of N-PLS and Ståhles PLS will give identical predictions. Not deflating $\underline{\mathbf{X}}$ will speed up the algorithm, but will also mean that no component-wise residuals in the $\underline{\mathbf{X}}$ -space are available for diagnostics.

An alternative modification suggested by de Jong (p.c.) would be to replace the models of $\underline{\mathbf{X}}$ and of $\underline{\mathbf{Y}}$ with overall least squares trilinear models given the scores. That is, for a given number of scores the loadings are determined from a PARAFAC model of the data with the scores fixed. This has not been pursued here, but would lead to several interesting properties both with respect to the models and the algorithm. First of all the models in equations (58) and (59) would become least squares models given the scores, which is an attractive feature. Whether this would lead to more predictive models is not certain. It would, however, lead to a more comparable model of $\underline{\mathbf{X}}$ in terms of bilinear PLS, as the bilinear model of $\underline{\mathbf{X}}$ obtained in traditional bi-PLS is a least squares model given the scores. Algorithmically, such an approach would be more complicated as well as slower.

USING THE N-PLS MODEL ON NEW DATA

Depending on what kind of information is needed different possibilities exist for using an N-PLS model on new data. If only the prediction of \mathbf{y} is wanted it is possible to obtain a model that directly relate $\underline{\mathbf{X}}$ to \mathbf{y} . This has been shown in detail by Smilde (1997a) and de Jong (1998) for the PLS1 model. The first score vector \mathbf{t}_1 is found as the least squares solution

$$\mathbf{t}_1 = \mathbf{X}\mathbf{w}_1, \quad (132)$$

where \mathbf{X} is the array $\underline{\mathbf{X}}$ unfolded to an $I \times JK$ matrix and

$$\mathbf{w}_1 = \mathbf{w}_1^K \otimes \mathbf{w}_1^J. \quad (133)$$

The second score vector can be found from projecting the residual $\mathbf{X} - \mathbf{t}_1 \mathbf{w}_1^T$ onto $\mathbf{w}_2 (= \mathbf{w}_2^K \otimes \mathbf{w}_2^J)$, i.e.,

$$\begin{aligned} \mathbf{t}_2 &= (\mathbf{X} - \mathbf{t}_1 \mathbf{w}_1^T) \mathbf{w}_2 = \\ &(\mathbf{X} - \mathbf{X} \mathbf{w}_1 \mathbf{w}_1^T) \mathbf{w}_2 = \\ &\mathbf{X}(\mathbf{I} - \mathbf{w}_1 \mathbf{w}_1^T) \mathbf{w}_2, \end{aligned} \quad (134)$$

etc. This derivation leads to a general formula for a matrix

$$\mathbf{R} = \left[\mathbf{w}_1 (\mathbf{I} - \mathbf{w}_1 \mathbf{w}_1^T) \mathbf{w}_2 \dots \left(\prod_{f=1}^{F-1} (\mathbf{I} - \mathbf{w}_f \mathbf{w}_f^T) \right) \mathbf{w}_F \right], \quad (135)$$

for which it holds that

$$\mathbf{T} = \mathbf{X} \mathbf{R}. \quad (136)$$

From the relation

$$\hat{\mathbf{y}} = \mathbf{T} \mathbf{b} \quad (137)$$

it follows that

$$\mathbf{b}_{\text{pls}} = \mathbf{R} \mathbf{b} \quad (138)$$

will be the regression coefficients that directly computes $\hat{\mathbf{y}} = \mathbf{T} \mathbf{b}$ from \mathbf{X} .

EXTENDING THE TRI-PLS MODEL TO HIGHER ORDERS

It is possible to extend the algorithms shown above to any desired order.

This will be shown here for quadri-linear PLS1. A vector holding the dependent variable, \mathbf{y} , and a four-way array of independent variables, \mathbf{X} ($I \times J \times K \times M$) is given. As shown in the three-way algorithm, a score vector, \mathbf{t} , is sought with maximum covariance with \mathbf{y} . This is obtained by calculating a $J \times K \times M$ array, \mathbf{Z} , where the jk th element is given by the dot-product $\mathbf{y}^T \mathbf{X}_{(:,j,k,m)}$. To find the first component of a quadri-linear PLS solution, seek the one-component decomposition of this array, that explains most of the variance. This solution can be found using PARAFAC (which is fast when only one component is sought). The decomposition of the array by PARAFAC immediately yields three weight vectors, \mathbf{w}^J , \mathbf{w}^K and \mathbf{w}^M , which, normalized, then again determines the score vector. For a five-way table the principle is analogous, but there are now four weight vectors to estimate for each component. These can be found by a four-way one-component PARAFAC model. To calculate one-component PARAFAC models, an algorithm specifically aimed at this purpose has been designed by Bro (1996) which updates two modes simultaneously.

It is also possible to extend the outlined PLS algorithm to dependent variables of higher orders. This can be done in a similar way as extending the \mathbf{X} model to higher orders. This is easily seen by realizing, that bilinear PLS2 is symmetrical in the way weights and scores are found for \mathbf{X} and \mathbf{Y} . Likewise, a trilinear decomposition of \mathbf{Y} can be found iteratively by making a matrix corresponding to \mathbf{Z} , by properly multiplying \mathbf{Y} and \mathbf{t} , and then decomposing it by SVD into two loading vectors, which then implicitly define the score vector \mathbf{u} . For applications of higher-order PLS models see Nilsson et al. (1997) and <http://newton.foodsci.kvl.dk/rasmus/npls.htm>.

4.8 IMPROVING ALTERNATING LEAST SQUARES ALGORITHMS

The following descriptions mainly deals with how to improve the speed and stability of ALS algorithms for fitting the PARAFAC model, as this is *the* model being most difficult to estimate. However, algorithms for fitting the PARAFAC2, PARATUCK2, or the restricted Tucker3 model can also benefit from the proposals.

Alternating least squares is an attractive approach because it ensures an improvement of the solution in every iteration. A major drawback of

some ALS algorithms is the time required to fit the models, especially when the number of variables is high. Most ALS algorithms have a poor convergence rate. Several hundred or thousands of iterations are sometimes necessary before convergence is achieved. With a data array of size 50 x 50 x 50 the parameters of a PARAFAC model might take hours to calculate on a moderate computer (depending on implementation and convergence criterion). This is problematic when recalculation of the model is necessary, which is often the case, e.g., during outlier detection. To make PARAFAC a workable method it is therefore of utmost importance to develop faster algorithms. Using more computer power could of course solve the problem but there is an annoying tendency of the data sets always to be a little larger, than what is optimal for the current computer power.

First it will be described how ALS algorithms can sometimes be speeded up by using regularized regression. This is mainly feasible in situations where the involved regression problems are ill-posed, since this is what regularization specifically deals with. Secondly a new approach to compression is developed, which seems highly beneficial for many types of multi-way problems, and which leads to faster computations in general. Next a simple improvement of the iterative search is described, which is simply a line search or extrapolation step. Finally, alternatives to ALS are considered. On page 154 another way of speeding up difficult problems is described.

REGULARIZATION

It has been reported in several instances, especially in connection to the PARAFAC model, that extremely slow convergence can occur temporarily in ALS algorithms. This has been termed swamps (page 124). These swamps have been explained by the fact that the least squares problems solved during the iterative procedures can be ill-posed. This means that the individual regression problems being solved will be quite uncertain in the sense that the uncertainty of the estimated parameters is large. As is known from the treatment of ill-posed problems in many areas some sort of regularization can be fruitful for lowering this variance (at the cost of biased estimates). Rayens & Mitchell (1997) investigated the effect of using

ridge regression for fitting the PARAFAC model. By the use of ridge regression they were able to speed up ill-posed problems considerably. However, they experienced in some cases, that the solution was slightly biased due to the use of ridge regression. An important observation is noted in Hopke et al. (1998), namely that ALS algorithms can also be slow in cases with no ill-conditioning. Hence regularization can only be hoped to partly remedy the slowness of ALS.

COMPRESSION

In this paragraph an approach is developed for compressing a multi-way array prior to fitting a multilinear model with the purpose of speeding up the iterative procedure (Bro & Andersson 1998). A method is developed for a rich variety of structural models with optional constraints on the factors. It is based on three key aspects: A fast implementation of the Tucker3 algorithm, which serves as compression method, the optimality theorem of the CANDELINC model, which ensures that the compressed array preserves the original variation maximally, and a set of guidelines for how to incorporate optional constraints.

A way to increase the speed of ALS algorithms is to compress the data array initially and then subsequently fit the model to the compressed data. This is natural since a multi-way model is *per se* a compression of the original data into fewer parameters, implying that the systematic variation in the data is expressible in less than the original number of data points. Hence, the model to be fitted should also be possible to fit to a condensed representation of the systematic variation in the data. After estimating the parameters of the model in the compressed space, these can then be transformed to the original space, and hopefully provide a good approximate solution to the solution that would be found if fitting the model in the original space. To ensure this, the model is re-fitted from the original data using the first model as initial estimate. This way the least squares solution to the problem is found, and if the compressed model is valid, only few iterations are necessary when fitting the model to the raw data.

In the sequel the model used to compress the data will be referred to as the *compression model* and the model operating on the compressed array the *analytical model*.

Alsberg & Kvalheim (1994a & b) have described in a series of papers a method for compressing highdimensional arrays. Kiers & Harshman (1997) have shown that their approach is identical to the CANDELINC (CANonical DEcomposition with LINear Constraints) approach. In CANDELINC the original array is compressed by expressing the array in the coordinates of a lowdimensional sub-space defined by a set of bases in each mode. Only orthogonal bases are allowed but any non-orthogonal basis can be orthogonalized prior to compression without any loss of information (Carroll et al. 1980). The Alsberg & Kvalheim approach was developed specifically for fitting Tucker3 models, while the CANDELINC approach is valid for fitting any multi-way model. Furthermore as stressed by Kiers & Harshman there is no need for special algorithms in the CANDELINC approach. Simply compress the array, use any existing multi-way algorithm on the compressed array, and decompress the result by postmultiplying the solution with the bases. This, however, only holds for unconstrained models with a nonweighted least squares optimization criterion as will be shown. The only important constraint that does not require any special attention is orthogonality. If orthogonal loadings are found in, e.g., a PARAFAC model of the compressed array, then the backtransformed solution will also be orthogonal.

Here, the Tucker3 model is suggested for finding the compression bases as the Tucker3 algorithm is fast and has, in a least squares sense, the property of providing optimal bases. Alsberg & Kvalheim suggest different bases in their work. *If* the size of the array is so large that fitting the Tucker3 model is in practice impossible due to the computer capacity then these suggested bases are sensible, but if the computer capacity is sufficient it is not sensible to use other bases than those given by the Tucker3 algorithm.

In the following, three-way arrays will be used as an example but the developed theory is directly applicable for arrays of any order. Also the ALS procedure for fitting the PARAFAC model will be used throughout though compression is equally applicable for other models *and* algorithms.

An $I \times J \times K$ array \mathbf{X} is given. Suppose that the pseudo-rank in each of the three modes is D , E , and F respectively. This is the rank of a sub-space of the particular mode spanning the systematic variation or the number of

linearly independent phenomena when noise is not present (Tomišić & Simeon 1993). Define \mathbf{U} of size $I \times D$ as an orthogonal basis for the systematic variation in the first mode, where D is determined by the analyst. Let an orthogonal matrix \mathbf{V} of size $J \times E$ define the variable space in the second mode and an orthogonal matrix \mathbf{Z} of size $K \times F$ define the variable space in the third mode. A G -component PARAFAC model is sought for the $I \times J \times K$ array $\underline{\mathbf{X}}$. This model is defined through \mathbf{A} ($I \times G$), \mathbf{B} ($J \times G$), and \mathbf{C} ($K \times G$) as

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \|\underline{\mathbf{X}} - \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T\|_F^2. \quad (139)$$

As \mathbf{A} is describing the systematic variation in the first mode of $\underline{\mathbf{X}}$ it must hold approximately that a matrix exists such that

$$\mathbf{A} = \mathbf{U}\Delta, \quad (140)$$

as \mathbf{U} is a basis for the systematic variation. Similar relations hold for the second and third mode:

$$\mathbf{B} = \mathbf{V}\Theta \quad (141)$$

and

$$\mathbf{C} = \mathbf{Z}\Psi. \quad (142)$$

This is the same as saying that the PARAFAC model is linearly constrained to the sub-spaces \mathbf{U} , \mathbf{V} , and \mathbf{Z} . The CANDELINC model was developed for fitting multi-way models under such linear constraints (Carroll et al. 1980). The theory of the CANDELINC model states that if a PARAFAC model of $\underline{\mathbf{X}}$ given by \mathbf{A} , \mathbf{B} , and \mathbf{C} is sought, subject to the above constraints, then it is only necessary to estimate the (much) smaller matrices Δ , Θ , and Ψ . More importantly these matrices can be found by fitting a PARAFAC model to an array $\underline{\mathbf{Y}}$ of size $D \times E \times F$ found by expressing the projection of $\underline{\mathbf{X}}$ with respect to the orthonormal bases \mathbf{U} , \mathbf{V} , and \mathbf{Z} . The projected array is

defined as

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{U}\mathbf{U}^T \mathbf{X}^{(I \times JK)} (\mathbf{Z}\mathbf{Z}^T \otimes \mathbf{V}\mathbf{V}^T), \quad (143)$$

and thus the compressed array \mathbf{Y} is defined

$$\mathbf{Y}^{(D \times EF)} = \mathbf{U}^T \mathbf{X}^{(I \times JK)} (\mathbf{Z} \otimes \mathbf{V}). \quad (144)$$

Fitting a G -component PARAFAC model to \mathbf{Y} will give the loading matrices $\mathbf{\Delta}$ ($D \times G$), $\mathbf{\Theta}$ ($E \times G$), and $\mathbf{\Psi}$ ($F \times G$), and through the relations of equations (140 - 142) the associated loading matrices in the original spaces can be calculated.

EXACT COMPRESSION

BOX 10

Algorithms for fitting the PARAFAC model and the Tucker3 model in situations where only one mode is high-dimensional have been given by Kiers & Krijnen (1991) and Kiers et al. (1992). These methods are exact and implicitly based on the fact that the rank in the high-dimensional mode is limited by the dimensions of the remaining modes. If the product of the two smallest dimensions (say I and J) is smaller than the dimension in the third mode, then it can be shown that the numerical rank of the third mode is bounded by the former mentioned product. In the present approach this means that in situations with one high-dimensional mode, one can simply compress with a basis of dimension IJ in the third mode. This corresponds to computing a Tucker1 model of the data and will provide a compressed array that exactly preserves the variation of the original array.

If the span of \mathbf{U} , \mathbf{V} , and \mathbf{Z} covers the systematic variation of \mathbf{X} , then the model fitted to \mathbf{Y} (equation 144) will give the sought solution. In Carroll et al. (1980) this is shown for any model that can be regarded as a Tucker3

model or a restricted version of a Tucker3 model. The PARAFAC, PARATUCK2, and PARAFAC2 models can all be regarded as restricted versions of Tucker3 and can hence be found by fitting the models to the compressed array without loss of information under the constraints of equations 140 - 142. Also, the Tucker3 model itself may conveniently be fitted and explored from a compressed array. The crucial point in compressing is to find good truncated bases for the respective modes. If these are appropriate, it is expectable that the analytical model found from the compressed data will be almost equal to the model fitted to the raw data. One possibility for finding these bases would be to use the singular vectors from a singular value decomposition (Tucker1) of the array properly unfolded for each mode, i.e.

Tucker1 based compression matrices

$$\begin{aligned} [\mathbf{U}, \mathbf{S}, \mathbf{T}] &= \text{svd}(\mathbf{X}^{(I \times JK)}, D) \\ [\mathbf{Z}, \mathbf{S}, \mathbf{T}] &= \text{svd}(\mathbf{X}^{(J \times IK)}, E) \\ [\mathbf{V}, \mathbf{S}, \mathbf{T}] &= \text{svd}(\mathbf{X}^{(K \times IJ)}, F). \end{aligned} \tag{145}$$

Compressing with such bases is a natural approach and has also been used for speeding up the PARAFAC-ALS algorithm (Appellof & Davidson 1981). A better way, though, to define optimal bases is to say that the projected array as defined in equation 143 should give a least squares fit to $\underline{\mathbf{X}}$. As the compressed array $\underline{\mathbf{Y}}$ is equivalent to the projected array, this ensures that the compressed array preserves the variation of $\underline{\mathbf{X}}$ maximally. The definition of the projection in equation 143 is equivalent to the definition of the Tucker3 model. It therefore immediately follows that orthogonal loading matrices of a $D \times E \times F$ Tucker3 model will provide optimal bases for calculating the compressed array. The compressed is therefore equivalent to the core of the Tucker3 model

Tucker3 based compression matrices

$$\min_{\mathbf{U}, \mathbf{V}, \mathbf{Z}, \mathbf{Y}} \|\mathbf{X} - \mathbf{UY}(\mathbf{Z}^T \otimes \mathbf{V}^T)\|_F^2 \tag{146}$$

Realizing this, it then follows that a fast Tucker3 model is the key to a successful compression method (see also Box 10 and 11). After obtaining the array \underline{Y} any suitable model can be found as described in Kiers & Harshman (1997) and Carroll et al. (1980), and exemplified above for the PARAFAC model.

It is important that most systematic variation be incorporated into the compressed array. This is especially true when the subsequent model to be fitted is constrained in some sense. It is of little concern whether the compressed array is of size $7 \times 7 \times 7$ or $11 \times 11 \times 11$ with respect to the speed of the algorithm, but it may have a significant influence on the quality of the model if not all systematic variation is retained in the $7 \times 7 \times 7$ array. Real data seldom conform exactly to a mathematical model, which means that some systematic variation in the residuals must be expected. Different systematic variation will be excluded in different models and if, e.g., a three-component PARAFAC model is sought it will not always suffice to compress the array using a $3 \times 3 \times 3$ Tucker3 model. A rule of thumb is to use at least one or two more components for compression than what is to be used in the analytical model if an unconstrained model is sought. If the analytical model is constrained, subtle differences in the data may be crucial for obtaining the right model and more components are mostly necessary in the compression model. Using, for example five extra components compared with the number of components in the analytical model, would probably ensure a valid model in most cases⁶.

When fitting the analytical model in the compressed space it may be verified that the compression model has captured all relevant variation by monitoring that the loading parameters of the analytical model of the compressed array with the highest row number are close to zero. This will hold if a sufficient number of components are used in the compression model because the latter components of the Tucker3 compression model will then only describe noise irrelevant for the analytical model.

⁶. The results presented in Kiers (1998b) do indicate that using only the same number of factors in the Tucker3 model as in the subsequent analytical model will work satisfactory in cases with very little model error

NON-NEGATIVITY & WEIGHTS IN COMPRESSED SPACES**BOX 11**

The characteristics of scaling and centering multi-way arrays are described on page 101. These rules also applies to models in compressed spaces. If the noise is homoscedastic, it is possible to use a weighted loss function in order to incorporate the specific uncertainties of the data elements in the compression model. Alternatively, known uncertainties may be ignored when fitting the Tucker3 compression model. A compressed array \underline{Y} is obtained and the uncertainties of the elements of this array can be found by compressing the array of uncertainties using the same bases as for compressing \underline{X} . A new array of uncertainties of the elements of the core array is then available, which can be used directly in the subsequent analysis of the compressed array.

If the resulting loading matrices of the full PARAFAC model are required to be non-negative this poses some problems, as the bounded least squares problem of the uncompressed problem turns into a more general and complicated inequality constrained least squares problem in the compressed space. Currently no method seems able to handle this special situation efficiently but the problem is being worked on (page 149).

In Bro & Andersson (1998) the compression approach is used to speed up the fitting of the PARAFAC model. The Tucker3 compression model outperforms the Tucker1 compression model with respect to:

- Speed
- Number of floating point operations
- Closeness of solution to the true solution

For the Tucker3 based compression method it is found that for spectral data a compression model with the same or one or two more components than the analytical model is sufficient. The model fitted to \underline{Y} is nearly identical to the model fitted directly to \underline{X} .

Using the Tucker3-based compression is generally 5 to 80 times

cheaper than fitting the model from the raw data in terms of flops as compared to only a 3 to 40 times cheaper with respect to speed⁷. The general observation is that the model fitted in the compressed space has to be very close to the true model to be effective. Therefore if the data are very noisy and there is a significant amount of model error it is important to use a sufficient number of components for the compression model. It also means that it is important to incorporate any constraints directly in the fitting of the analytical model in the compressed space.

LINE SEARCH, EXTRAPOLATION AND RELAXATION

Another method for speeding up the ALS algorithm is to use the 'temporal' information in the iterations. The simple idea is to perform a predefined number of cycles of ALS-iterations and then these estimates of the loadings are used to predict new estimates element-wise (Appellof & Davidson 1981, Harshman 1970, Ross & Leurgans 1995). There are two good reasons for using the temporal information in the iterations of, e.g., the PARAFAC-ALS algorithm. (i) It is only in the first few iterations that major changes occur in the estimates of the elements of the loadings. For the main fraction of iterations only minor modifications of the factors occur. (ii) The changes in each element of the factors are most often systematic and quite linear over short ranges of iterations.

To make it profitable to extrapolate it is necessary, that the time required to extrapolate is less, than the time required to perform a corresponding number of iterations. This to some extent limits the applicability of the method, because very ingenious extrapolations tend to be too slow. Several implementations have been tried ending up with a simple algorithm by Claus A. Andersson. At the i th iteration the estimated loadings, e.g, \mathbf{A} are saved as $\mathbf{A1}$. After the $(i+1)$ th iteration a linear regression is performed for each element to predict the value of the elements a certain number of iterations ahead. As only two values of each element are used in the regression, the prediction can simply be written

$$\mathbf{A}_{\text{new}} = \mathbf{A1} + (\mathbf{A} - \mathbf{A1})d, \quad (147)$$

⁷. The size of the used data sets were $5 \times 51 \times 201$ and $268 \times 371 \times 7$.

where d is the number of iterations to predict ahead. Letting

$$d = it^{1/3} \quad (148)$$

where it is the number of iterations has proven useful empirically. When applying the extrapolation, it is feasible not to extrapolate during the first, say five, iterations, because the variations in the elements are unstable in the beginning. If some modes are constrained, extrapolation has to wait longer for the iterations to be stable. Furthermore if the extrapolations fail to improve the fit persistently (more than four times) the number d is lowered from $it^{1/n}$ to $it^{1/n+1}$

An issue that is quite obvious yet little explored is to consider the parameter estimate changes with iterations as a multivariate time-series. Exploring the data as such could possibly lead to improved algorithms for using the temporal information.

NON-ALS BASED ALGORITHMS

There is little doubt, that ALS as a technique has some intrinsic problems, which must be circumvented somehow if multi-way analysis should gain widespread use. Several authors have explored and investigated the use of other algorithms (e.g. Hayashi 1982). Most notably P. Paatero developed his PMF3 algorithm for fitting the three-way PARAFAC model. The algorithm PMF3 uses a Gauss-Newton approach for iteratively minimizing the PARAFAC loss function simultaneously over all modes.

Consider an interim estimate of the PARAFAC model:

$$\mathbf{X} = \mathbf{A}_0(\mathbf{C}_0 \otimes \mathbf{B}_0)^T + \mathbf{E}_0, \quad (149)$$

where \mathbf{E}_0 is the interim residual. The difference between the true least squares solution (\mathbf{A} , \mathbf{B} , and \mathbf{C}) and the interim solution is given

$$\mathbf{A} = \mathbf{A}_0 + \Delta\mathbf{A}, \quad \mathbf{B} = \mathbf{B}_0 + \Delta\mathbf{B}, \quad \mathbf{C} = \mathbf{C}_0 + \Delta\mathbf{C}, \quad (150)$$

hence the sought solution is the one that minimizes

$$\|\mathbf{X} - (\mathbf{A}_0 + \Delta\mathbf{A})[(\mathbf{C}_0 + \Delta\mathbf{C}) \otimes (\mathbf{B}_0 + \Delta\mathbf{B})]^T\|_F^2, \quad (151)$$

which is only a function of $\Delta\mathbf{A}$, $\Delta\mathbf{B}$, and $\Delta\mathbf{C}$. Vectorizing the corresponding loss function the normal equation of this (underdetermined) system can be solved with respect to $\Delta\mathbf{A}$, $\Delta\mathbf{B}$, and $\Delta\mathbf{C}$ by regularized (ridge) regression. Weighted loss functions and some constraints can be incorporated as discussed at length in Paatero (1997).

This approach supposedly yields a more robust and fast algorithm than using ALS (Paatero 1997, Hopke et al. 1998). The price to be paid, however, is that the algorithm requires a more skilled user, that the memory requirements prevent the algorithm from being used on large (spectral) data, and that the algorithm does not generalize to N-way models. The problem with memory requirements can be circumvented using compression (page 88), but the problem of extending the algorithm is less easily solved. In order to obtain a more general algorithm, P. Paatero has suggested the *multilinear engine* based on a conjugate gradient technique. Little experience has yet been gained, but it does seem to circumvent both the problems with ALS (slow convergence) and PMF3 (generality and memory requirements).

4. 9 SUMMARY

In this chapter algorithms for the models described in chapter 3 have been given. Further it has been shown how to use a current model on new data as well as how to extend the algorithms to higher order models. Finally several useful techniques for speeding up slower ALS algorithms have been given.

CHAPTER 5

VALIDATION

5.1 WHAT IS VALIDATION?

Validation can be seen as the part of the analysis where it is investigated if valid conclusions can be drawn from the model: Does the model actually *generalize* the data in a parsimonious way, i.e., express the main variation in the data as simple as possible? Has the model been influenced by certain samples or variables in such a way that specific parts rather than the data as such are described? Has the algorithm converged?

Computational validation is concerned with whether the estimated parameters are indeed the parameters, that optimize the loss function defining the model. For a data analyst this aspect should ideally not be of concern, but unfortunately many multi-way models are not simple to fit. *Statistical validation* is for example related to appropriateness of distributional assumptions and how well the properties of the data fit the assumptions of the model. *Explanatory validation* is concerned with how well the model reflects the real phenomena under investigation, i.e., the appropriateness of the model.

R. A. Harshman (1984) gives a thorough description of how to validate a model. It is reasonable to sum up the essence of his paper, but then focus on the problems more specifically related to multi-way modeling. He divides validation into four levels: Zero-fit diagnostics, one-fit diagnostics, many-fit (single data set) diagnostics, and many-fit (multiple data sets) diagnostics.

Zero-fit diagnostics: are those related to the data *before* any model has

been fit. Zero-fit diagnostics includes check for outliers and reliability of the data through replicates. It also includes initial judging of which model to use, perhaps guided by initial unfolding models.

One-fit diagnostics: are those used after fitting one specific model and useful for validating this particular model. Not so much assessing by comparison to other competing models but more the consistency of the model itself. The diagnostics can be divided into those relating to the parameters and to the residuals. *Parameters:* Investigation of a model includes monitoring correlations of loadings to detect model mis-specifications, constant factors which could perhaps be eliminated by proper preprocessing or be signs of only random variation in a mode. Checking for signs of non-linearities, interpretability, convergence etc. *Residuals:* Checking for heteroscedasticity, outliers, or systematic variation. Comparing goodness-of-fit with estimated data reliability etc.

Many-fit (single data set) diagnostics: are useful for comparing either models fitted from different starting points, or different alternative models. Diagnostics at this stage can again be divided into those related to parameters and those related to residuals. *Parameters:* Study parameters across successive iterations, different solutions or different dimensionalities to investigate which seem sensible, interpretable, robust, converged etc. *Residuals:* Similar to one-fit related diagnostics, but here the focus is on whether the algorithms have converged, if one model is significantly better than another etc.

Many-fit (multiple data sets) diagnostics: These diagnostics mainly refer to when several data sets are available for the same problem, or when the data set is so large that it can be split into several sets for validation. In that case the theory of resampling is applicable, making possible bootstrapping, cross-validation, split-half analyses, randomization tests etc. ■

This general description of practical validation will now be substantiated by dwelling on more specific problems:

-
- Preprocessing of data
 - Selecting a model
 - Determining the number of components
 - Validating whether the least squares model is found
 - Dealing with degeneracy
 - Assessing uniqueness properties
 - Outlier detection
 - Assessing robustness

As for ordinary multivariate analysis these issues are all strongly connected. This tends to complicate the validation. In-depth knowledge of the problem at hand is what makes the difference in most situations. Subjects not specifically related to multi-way analysis (transformations, general residual analysis etc.) will not be treated in detail as information on these subjects can be found in any textbook on standard multivariate analysis.

5.2 PREPROCESSING

Preprocessing of higher-order arrays is more complicated than in the two-way case, though understandable in light of the multilinear variation presumed to be an acceptable model of the data. Centering serves the same purpose as in two-way analysis, namely to remove constant terms in the data, that may otherwise at best need an extra component, at worst make modeling impossible.

All models described here are implicitly based on that the data are ratio-scale (interval-scale with a natural origin), i.e., that there is a natural zero which really does correspond to zero (no presence means no signal) and that the measurements are otherwise proportional such that doubling the amount of a phenomenon implies that its corresponding contribution to the signal is doubled. If data are not approximately ratio-scale then centering the data is also mandatory.

Centering is performed to make the data compatible with the structural model. Scaling on the other hand is a way of making the data compatible with the least squares loss function normally used. Scaling does not change the structural model of the data, but only the weight paid to errors of specific elements in the estimation. Scaling is dramatically simpler than

using a weighted loss function (page 145), and is therefore to be preferred to this, if approximate homoscedastic data can be obtained by scaling. Centering and scaling will be described using three-way arrays in the following.

CENTERING

Centering, e.g., the first mode of an array can be done by unfolding the array to an $I \times JK$ matrix, and then center this matrix as in ordinary two-way analysis:

$$X_{ijk}^{\text{cent}} = X_{ijk} - \frac{\sum_{i=1}^I X_{ijk}}{I} . \quad (152)$$

This is often referred to as single-centering. The centering shown above is also called centering *across* the first mode, which is the terminology suggested by ten Berge (1989). The centering can of course be applied to any of the modes, depending on the problem. If centering is to be performed across more than one mode, it has to be done by first centering one mode, and then center the outcome of this centering. If two centerings are performed in this way, it is often referred to as double-centering. Triple-centering means centering across all three modes one at a time. In Kruskal (1983), Harshman & Lundy (1984b), and ten Berge (1989) the effect of both scaling and centering on the multilinear behavior of the data is described. It turns out that centering one mode at a time, is the only appropriate way of centering, with respect to the assumptions of the PARAFAC or any other multilinear model. Centering one mode at a time essentially removes any constant levels in that particular mode. Centering for example matrices instead of columns will destroy the multilinear behavior of the data, because more constant levels are introduced than eliminated. The same holds for other kinds of centering. For instance, if it is known that the true model consists of one PARAFAC term (a trilinear component) and an overall level, it may seem feasible to fit a PARAFAC model to the original data subtracted the grand level. However, even though the mathematical structure might theoretically be true, the subtraction of the grand level

introduces some artifacts in the data, not easily described by the PARAFAC model. In this case even though the grand level has been subtracted *two components* are still necessary to describe the data.

THE EFFECT OF CENTERING

BOX 12

Consider a synthetic data set $\underline{X} = \mathbf{a}(\mathbf{c} \otimes \mathbf{b})^T + 10$ where $\mathbf{a} = [1 \ 2 \ 3 \ 4]^T$ and \mathbf{b} and \mathbf{c} are identical to \mathbf{a} . No noise is added.

Consider the following alternative PARAFAC models using

- i) \underline{X} and one component
- ii) \underline{X} and two components
- iii) \underline{X} minus the overall average of \underline{X} and one component
- iv) \underline{X} centered across the first mode and one component
- v) \underline{X} and two components, fixing one component to a constant

The fit values of these five models are given below

- i) 99.54%
- ii) 100.00%
- iii) 71.65%
- iv) 100.00%
- v) 100.00% (constant estimated to 10.000000)

Subtracting the grand level prior to modeling does not lead to a simpler model. Either the generally applicable single-centering must be used (iv) or the grand level specifically estimated (v).

This shows that the preprocessing has not achieved its goal of simplifying the subsequent model. If on the other the data are centered across one mode the data can be modeled by a one-component model. Another possibility is to fit a two-component model but constraining one component to constant loadings in each mode, thus reflecting the grand level. This provides a model with a unique estimate of the grand level (see Box 12).

SCALING

If the uncertainties of the individual data elements are known it can be feasible to use these in the decomposition. If the uncertainty of a given variable remains almost the same over all other modes, it will suffice to scale the array accordingly. After scaling, an unconstrained model is fitted to the scaled array. If the uncertainties vary also within specific variables or if an iteratively re-weighted approach is desired for robustness, then the model must be fitted using a weighted loss function.

Scaling in multi-way analysis has to be done, taking the trilinear model into account. It is not, as for centering, appropriate to scale the unfolded array column-wise, but rather whole slabs or submatrices of the array should be scaled. If variable j of the second mode is to be scaled (compared to the rest of the variables in the second mode), it is necessary to scale by the same scalar all columns where variable j occurs. This means that whole matrices instead of columns have to be scaled. For a four-way array, three-way arrays would have to be scaled. Mathematically scaling within the first mode can be described

$$X_{ijk}^{\text{scal}} = \frac{X_{ijk}}{s_i} \quad (153)$$

where setting

$$s_i = \sqrt{\left(\sum_{j=1}^J \sum_{k=1}^K X_{ijk}^2 \right)} \quad (154)$$

will scale to unit squared variation. The scaling shown above is referred to as scaling *within* the first mode. When scaling within several modes is desired, the situation is a bit complicated because scaling one mode affects the scale of the other modes. If scaling to norm one is desired within several modes, this has to be done iteratively, until convergence (ten Berge 1989).

Another complicating issue, is the interdependence of centering and

scaling. Scaling within one mode disturbs prior centering across the same mode, but not across other modes. Centering across one mode disturbs scaling within all modes (Harshman & Lundy 1984b). Hence only centering across arbitrary modes or scaling within one mode is straightforward, and furthermore not all combinations of iterative scaling and centering will converge. In practice, though, it need not influence the outcome much if an iterative approach is not used. Scaling to a sum-of-squares of one is arbitrary anyway and it may be just as reasonable to just scale within the modes of interest once, thereby having at least mostly equalized any huge differences in scale. Centering can then be performed after scaling and thereby it is assured that the modes to be centered are indeed centered.

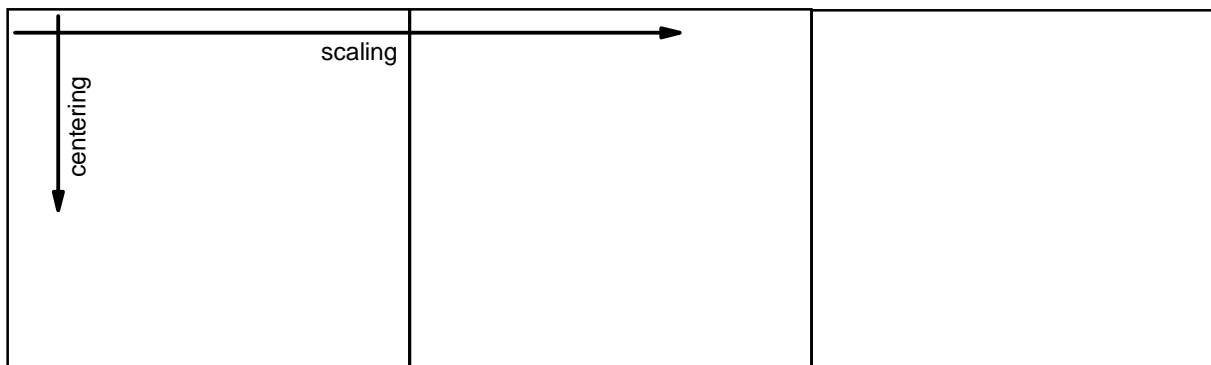


Figure 10. Three-way unfolded array. Centering must be done across the columns of this matrix, i.e., each column is subtracted by the same scalar. Scaling has to be done on the rows, that is, all elements of a row are divided by the same scalar.

The appropriate centering and scaling procedures can most easily be summarized in a figure where the array is shown unfolded to a matrix (Figure 10). Note, that this figure only shows the directions in which to center and scale; not the order in which such operations have to be performed. Centering must be done across the columns of this matrix, while scaling should be done within the rows of this matrix. The common approach of scaling the columns of a data-matrix is not appropriate for the above unfolded data. The consequence of such a scaling is that more components are necessary than if suitable scaling is used, and that the

resulting model will be more difficult to interpret (Box 13).

THE EFFECT OF SCALING

BOX 13

Consider a synthetic data set $\underline{\mathbf{X}} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T$ where \mathbf{A} is a 4×2 matrix of random numbers (\mathbf{B} and \mathbf{C} defined likewise). Consider the following alternative two-component PARAFAC models:

- i) Using $\underline{\mathbf{X}}$
- ii) Using $\underline{\mathbf{X}}$ centered and scaled as a two-way matrix, $\mathbf{X}^{(I \times JK)}$
- iii) Using $\underline{\mathbf{X}}$ scaled within, e.g., mode two.

The fit values of these three models are given below

- i) 100.00%
- ii) 98.80%
- iii) 100.00%

A two-component model is appropriate, and should be even after scaling (iii). However, using ordinary two-way scaling methods (ii), destroys the multilinear structure of the data and deteriorates the model.

CENTERING DATA WITH MISSING VALUES

When data are missing preprocessing such as centering turns into modeling, as it cannot be performed prior to fitting the model. Rather a model has to be specified specifically to handle simultaneously the structural model and the centering sought. This is similar to the situation for two-way models like PCA. If only few data points are missing a pragmatic approach can be adopted using an approximate centering instead of a least squares (i.e. by disregarding missing values), but this is not always acceptable. As scaling generally depends on centering the same applies to scaling. In the following only centering will be described for simplicity. Suppose a model is sought centered across the first mode. The corresponding global model is

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{1}\bar{\mathbf{x}}^T + \mathbf{M}^{(I \times JK)} \quad (155)$$

$\mathbf{1}$ being an $I \times 1$ vector of ones, and $\bar{\mathbf{x}}^T$ being the $1 \times JK$ vector holding the mean-values as in ordinary two-way analysis. The matrix \mathbf{M} is the model, which in case of, e.g., three-way PARAFAC would be $\mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T$.

The beauty of preprocessing is, that merely by first centering the data and then subsequently fitting the model to the centered data the least squares solution to the total problem is obtained (Kruskal 1984). However, this does not hold for data with missing values. Instead the model of the whole problem has to be fitted simultaneously. There are several ways of doing this. One approach is to replace initially the missing elements with suitable values. Using these estimates of the missing values, the data array is now complete and a standard model can be fitted. First the data are preprocessed and then the model parameters are estimated. Then new estimates of the missing values are calculated from the model of the data as well as the mean values. The whole procedure is repeated using these values until convergence. In practice, each submodel is not iterated to convergence, but rather a few or only one iteration would be performed before recalculating the missing values and re-preprocessing. This way of dealing with missing values is identical to the one used in the PARAFAC program of Harshman & Lundy (1994).

5.3 WHICH MODEL TO USE

Depending on the purpose of the analysis several different models may be useful. Choosing which model is better is therefore part of the overall validation procedure. For calibration problems different models can be compared in terms of how well they *predict* the dependent variables. Sometimes *a priori* knowledge of the structure of the data is available (e.g., that fluorescence data can be approximated by a trilinear model), but often this is not the case. If no external information is available for which comparison of different models can be based, other approaches have to be used. In the following a discussion is given of how to assess the structure of data based on the mathematical properties of the data and model. No exact rules will be given, but rather some guidelines that may be helpful for the data analyst. The most basic rule, which will be substantiated

in the application chapter (seven), is that multi-way data are often best modeled by a multi-way model.

MODEL HIERARCHY

It is important to have a feeling for the hierarchy of the different possible models. Kiers (1991) shows that PARAFAC can be considered a constrained version of Tucker3, and Tucker3 a constrained version of two-way unfold PCA (Tucker1). Any data set that can be modeled adequately by a PARAFAC model can thus also be modeled by Tucker3 or Tucker1, but PARAFAC uses fewer parameters. A Tucker1 model always *fits* data better than a Tucker3 model, which again will *fit* better than a PARAFAC model, all except for extreme cases where the models may fit equally well. If a PARAFAC model is adequate, Tucker3 and Tucker1 models will tend to use the excess parameters to model noise or model the systematic variation in a redundant way (see page 196 & 204). Therefore it is generally preferable to use the simplest adequate model. This principle of using the simplest possible model is old, in fact dating back as long as to the fourteenth century (Occam's razor), and is now also known as the law or principle of parsimony (Seasholz & Kowalski 1993). The Tucker1 model can be considered the most complex and flexible model, as it uses most parameters, while PARAFAC is the most simple and restricted model.

It is apparent that the reason for using multi-way methods is not to obtain better fit, but rather more adequate, robust and interpretable models. This is equivalent to the difference between using multiple linear regression and partial least squares regression (PLS) for multivariate calibration. Multiple linear regression is known to give the best fit to the dependent variable of the calibration data, but in most cases PLS has better predictive power. Partial least squares regression can be seen as a constrained version of multiple linear regression, where the constraints on the regression vector help the model focusing on the systematic part of the data. In the same way multi-way models are less sensitive to noise and further give loadings that can be directly related to the different modes of the multi-way array.

That unfold-PCA/Tucker1 can give complicated models can be illustrated with an example. For an F -component Tucker1 solution to an I

$\times J \times K$ array unfolded to an $I \times JK$ matrix, the model consists of $F(I+JK)$ parameters (scores and loading elements). A corresponding Tucker3 model with equal number of components in each mode would consist of $F(I+J+K)+F^3$, and PARAFAC $F(I+J+K)$ parameters. For example, consider a $98 \times 371 \times 4$ array modeled by a 5 component solution (page 204). A Tucker1 model of the 98×1484 unfolded array consists of 7910 parameters, a Tucker3 model of 2490 and a PARAFAC model of 2365 parameters. Clearly, the Tucker1 model will be more difficult to interpret than the multi-way models in general. It is interesting to note that the Tucker3 model will often contain only slightly more parameters than the PARAFAC model, hence the Tucker3 model is a good alternative to unfolding when PARAFAC is not appropriate.

How then, do these observations help in choosing the most appropriate model? The following holds:

$$\text{SSE}(\text{Tucker1}) < \text{SSE}(\text{Tucker3}) < \text{SSE}(\text{PARAFAC}). \quad (156)$$

$\text{SSE}(\text{model})$ is the sum-of-squares of errors for the given model. In order for this ranking to hold, it is assumed that the same number of components is used in all three models as well as the same number of components in each mode of the Tucker3 model. Only the most important decomposition models are shown here for simplicity. If for example the PARATUCK2 model is used with either its left or right third mode loadings eliminated (fixed to ones, see page 40) it would be positioned between the Tucker3 and the PARAFAC model. Suppose that for a given data set a PARAFAC model is valid. From the above discussion and the inequality above it follows, that most likely SSE of a Tucker1, Tucker3, and a PARAFAC model of equivalent dimension will be similar though highest for the PARAFAC model. This is so, because they will all be capable of modeling the systematic variation. Even though the Tucker1 and the Tucker3 model have additional modeling power, this will only describe the (smaller) random variation. Hence the models will be almost equally good with respect to SSE. Another extreme occurs in a situation where no multi-way structure is appropriate but the unfold-PCA model is. Then $\text{SSE}(\text{Tucker1})$ will be significantly smaller than $\text{SSE}(\text{Tucker3})$ and $\text{SSE}(\text{PARAFAC})$. Much in line

with how a scree-plot is used, the appropriate complexity of the model may be determined by scrutinizing the fit of different models this way (Kiers 1991). If possible using instead of fit the sum-of-squares of errors in modeling new samples can give a clearer picture. For the example described on page 204 it is evident that all decomposition models (Tucker1, Tucker3, PARAFAC) are equally good at modeling the data, so there is little sense in using Tucker1 or Tucker3.

TUCKER3 CORE ANALYSIS

An intriguing and exploratory way of finding the appropriate structural model is to investigate obtainable simplicity from a Tucker3 core array. As the Tucker3 model has rotational freedom it is possible to rotate the core by counter-rotating the loading matrices accordingly without changing the fit of the model. If the core can be rotated approximately to a superdiagonal, then it is clear evidence that a PARAFAC model is appropriate. It can be sought either to rotate the core to a hypothesized structure like super-diagonality or simply to a simple structure in some well-defined way. Yet another possibility is to seek a transformation of the core array with as many zeros as possible, either by rotation preserving the fit, or by simply constraining elements to be zero. There will not be any examples of this here, because for all applications treated simpler models like PARAFAC and PARATUCK2 are adequate. For more information on these subjects the reader is referred to the references noted on page 49 and 50.

5.4 NUMBER OF COMPONENTS

It is difficult to decide the best rank, i.e., the column-dimensions of the loading matrices, of most multi-way models. With experience a feeling for which results are good and which results are bad is gained. This can be important for making good models. The use of experience and intuition can also be more systematically used. Often certain things are known about the underlying phenomena in the data. Spectra of certain analytes might be known, the shape of chromatographic profiles might be known or the non-negativity of certain phenomena might be known. These kinds of hard facts can be informative when comparing different models. In Ross & Leurgans (1995) and Durell et al. (1990) some examples on how to use residuals and

external knowledge to choose the appropriate number of components are shown. In the following different tools for determining the appropriate number of components are given.

RANK ANALYSIS

For a two-way matrix the row- and column ranks are identical for mathematical reasons. For three- and higher-order arrays this is not so. The rank in a specific mode shows how many linear independent linear variations are present in the given mode. To determine the rank in the first mode of a three-way array unfold the $I \times J \times K$ array to an $I \times JK$ matrix as

$$\mathbf{X}^{(I \times JK)} = [\mathbf{X}_1 \mathbf{X}_2 \dots \mathbf{X}_K]. \quad (157)$$

The rank of this matrix will reveal the number of linear independent phenomena in the first mode. In practice the numerical rank of the matrix is not of interest. Numerically a data matrix of real measurements will most often have full rank, because the data are noisy. Instead an estimate of what is sometimes called the pseudo-rank is sought. This is the minimum rank of a basis spanning the systematic variation or equivalently the rank in case no noise was present. Many authors have worked on how to estimate the pseudo-rank of a matrix. Common approaches are the use of cross-validation (Wold 1978, Eastment & Krzanowski 1982) or Malinowski's indicator function (Malinowski 1991). For other methods see Piggot & Sharman (1986), Gemperline & Salt (1989) or Fay et al. (1991).

SPLIT-HALF ANALYSIS

Harshman & Lundy (1984a) and Harshman & de Sarbo (1984) advocate for using split-half analysis for determining the rank of unique models. The split-half analysis is a type of jack-knife analysis where different subsets of the data are analyzed independently. Due to the uniqueness of, e.g., the PARAFAC model, the same result – same loadings – will be obtained in the nonsplitted modes from models of any suitable subset of the data, if the correct number of components is chosen. If too many or too few components are chosen the model parameters will differ if the model is fitted to different data sets. Even though the model may be unique, the model

parameters will be dependent on the specific sampling as the amount of underlying phenomena present in the data set determines which linear combination of the intrinsic set of profiles and the noise will give a unique solution for the specific model at hand. To judge if two models are equal the indeterminacy in multilinear models has to be respected: The order and scale of components may change if not fixed algorithmically. If a model is stable in a split-half sense it is a clear indication that the model is real; that it captures essential variation, that not only pertains to the specific samples. If, on the other hand, some components are not stable in a split-half sense, it indicates that they may not be real, hence the model is not valid. It may also happen, though, that the phenomenon reflected in the non-stable component is simply only present in specific subsets. Therefore non-stability in a split-half analysis is not always as conclusive as stability.

When performing a split-half experiment it has to be decided which mode to split. Splitting should be performed in a mode with a sufficient number of *independent* variables/samples. With a highdimensional spectral mode, an obvious idea would be to use this spectral mode for splitting, but the collinearity of the variables in this mode would impede sound results. If the spectra behave additively the two data sets would in practice be identical, hence split-half analysis would not be possible.

In order to avoid that an unlucky splitting of the samples causes some phenomena to be absent in certain groups, the following approach is often adopted. The samples are divided into two groups: A and B. If the samples are presumed to have some kind of correlation in time the sets are constructed contiguously, i.e., A consists of the first half of the samples and B of the last. Accidentally it may happen that one of these sets does not contain information on all latent phenomena. To assure or at least increase the possibility, that the sets to be analyzed cover the complete variation two more sets are generated, C and D. The set C is made from the first half of A and B and the set D consists of the last half the of samples in A and B. These four sets are pairwise independent. A model is fitted to each of the data sets, and if the solution replicates over set A and B or over set C and D, correctness of the solution is empirically verified.

The split-half approach may also sometimes be used for verifying if non-(tri)-linearities are present. If spectral data are modeled and there are

indications of non-linearities in certain wavelength areas this may be verified by making separate models of different wavelength areas. If no non-linearities are present, the same scores should be obtained in the different areas, possibly using sub-models of different dimensions if some phenomena are only present in certain areas. If the same scores are not obtained, it could indicate dissimilar interrelations in different areas, hence nonlinearities.

RESIDUAL ANALYSIS

As in bilinear models, the characteristics of a model can be judged by the residuals of the model: If systematic variation is left in the residuals, it is an indication that more components can be extracted. If a plot of the residual sum of squares versus the number of components (a scree-plot) sharply flattens out for a certain number of components, this is an indication of the true number of components. If the residual variance is larger than the known experimental error, it is indicative of more systematic variation in the data. To calculate variance-like estimators Durell et al. (1990) suggest using the number of parameters. Such degrees of freedom might be used for exploratory purposes, but they are not to be taken as statistically correct numbers of degrees of freedom. Such are currently not available.

CROSS-VALIDATION

Cross-validation and methods alike are commonly used in chemometrics for determining the model dimensionality. This is also possible in multi-way modeling. In chemometrics it has become common to do cross-validation in two-way modeling by leaving out samples. It is not in general profitable to use this approach for multi-way decompositions. However, work is in progress to develop a more sound cross-validation procedure for multi-way models (A. K. Smilde, p.c.) based on the work of Eastment & Krzanowski (1982).

CORE CONSISTENCY DIAGNOSTIC

A new approach called the *core consistency diagnostic* is suggested for determining the appropriate number of components for multi-way models. It applies especially to the PARAFAC model, but also any other model, that

can be considered a restricted Tucker3 model. First the principle behind the method will be given and it will then be shown that it can indeed be an effective way of judging model complexity.

Consider a three-way PARAFAC model. Normally the structural model is stated

$$\mathbf{X} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}, \quad (158)$$

but it may equivalently be stated as a restricted Tucker3 model

$$\mathbf{X} = \mathbf{A}\mathbf{T}^{(F \times FF)}(\mathbf{C} \otimes \mathbf{B})^T + \mathbf{E}, \quad (159)$$

where the core array \mathbf{T} is a binary array with zeros in all places except for the superdiagonal which contains only ones. After having fitted the PARAFAC model (\mathbf{A} , \mathbf{B} , and \mathbf{C}), verification that the trilinear structure is appropriate can be obtained by calculating the least-squares Tucker3 core given \mathbf{A} , \mathbf{B} , and \mathbf{C} , i.e.

$$\underset{\mathbf{G}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{A}\mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T\|_F^2. \quad (160)$$

Note that this model is based on the PARAFAC loading vectors as also described in Lundy et al. (1989). As these are not orthogonal the algorithm explained on page 72 can not be used here. Instead a regression model can be constructed to solve the problem based on the loss function

$$\underset{\mathbf{G}}{\min} \|\operatorname{vec}\mathbf{X} - (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})\operatorname{vec}\mathbf{G}\|_F^2. \quad (161)$$

If the PARAFAC model *is* valid then \mathbf{G} should resemble \mathbf{T} . If the data can not approximately be described by a trilinear model or too many components are used then the core, \mathbf{G} , will differ from \mathbf{T} . To explain this, assume that an F -component trilinear model is an adequate model for the systematic part of the data. An additional component (or rather the $F+1$ -component model as a whole) will not only describe the systematic variation but also

the random part which is distributed more evenly in the variable space.

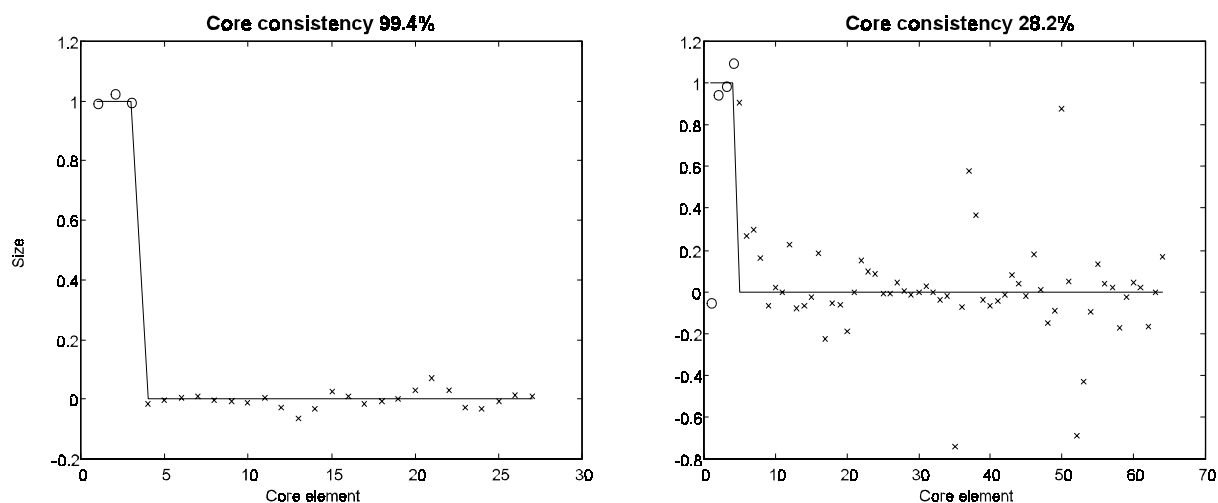


Figure 11. Core consistency plot of a three-component PARAFAC model (left) and a four-component model (right) of five samples of fluorescence excitation-emission (a $5 \times 201 \times 61$ array). Each sample contains different amounts of tryptophan, tyrosine, and phenylalanine, and should theoretically be modeled by a three-component model. The data are due to C. A. Andersson and have also been described in Box 14 (page 142) using two of the five samples. In the plots the circles are the superdiagonal elements of $\underline{\mathbf{G}}$ hence these should ideally be one. The crosses are the off-superdiagonal elements which should ideally be zero. The line segment is made from the elements of $\underline{\mathbf{T}}$ and is hence the target that $\underline{\mathbf{G}}$ should resemble.

Hence the extra component, even though it is forced to be trilinear, will be descriptive not only of trilinear variation, but also variation distributed all over the array. This follows because, if an extra component could be found that is not descriptive of evenly (or rather non-trilinear) distributed variation, then naturally a component could be found descriptive of trilinear variation. In that case an extra component *would* be appropriate. Hence, per definition, a least squares model with an extra component will not only describe trilinear variation. Therefore the core array of equation 160 will provide a better-fitting model by deviating from $\underline{\mathbf{T}}$ (see Bro & Kiers 98).

A simple way to assess if the model structure is reasonable is therefore to monitor the distribution of superdiagonal and off-superdiagonal elements of \mathbf{G} . If the superdiagonal elements are all close to one, and the off-superdiagonal elements are close to zero the model is not overfitting. If, on the other hand, this is not the case then either too many components have been extracted, the model is mis-specified, or gross outliers disturb the model. It is possible to calculate the superidentity of \mathbf{G} to obtain a single parameter for the model quality. The superidentity or *core consistency* is defined here as

$$\text{Core Consistency} = 100 \left(\frac{1 - \sum_{d=1}^F \sum_{e=1}^F \sum_{f=1}^F (g_{def} - t_{def})^2}{\sum_{d=1}^F \sum_{e=1}^F \sum_{f=1}^F t_{def}^2} \right) \quad (162)$$

i.e. percentage of the variation in \mathbf{G} consistent the variation in \mathbf{T} . It is called the *core consistency* as it reflects how well the Tucker3 core fits to the assumptions of the model. The difference between using superdiagonality and superidentity is generally small. The superidentity, though, is directly reflecting the sought consistency.

The core consistency diagnostic may at first seem less strong than other approaches for determining dimensionality, but it is actually extremely powerful. This will be exemplified here by showing the results for some of the PARAFAC models discussed in this thesis (for more examples see Bro & Kiers 98). In the following figures the distribution of the core elements are shown in so-called *core consistency plots* for models of the same complexity as used in the actual applications and models using one more component. Note the following important points

- For *all* models the appropriate complexity was determined by other means than core consistency.
- The data vary from simple laboratory data with almost perfect trilinear

structure (amino acids) over more complicated though probably quite trilinear data (sugar) to very noisy data with no *a priori* knowledge of reasonable structure (bread).

In Figure 11 the core consistency plot is shown for two competing models of a simple fluorescence data set. It is easy to see that the four-component model is strongly overfitting the data, as several off-superdiagonal elements of \mathbf{G} are larger or of similar size as the superdiagonal elements. In this case there is thus no doubt that the three-component solution is preferable to the four-component solution.

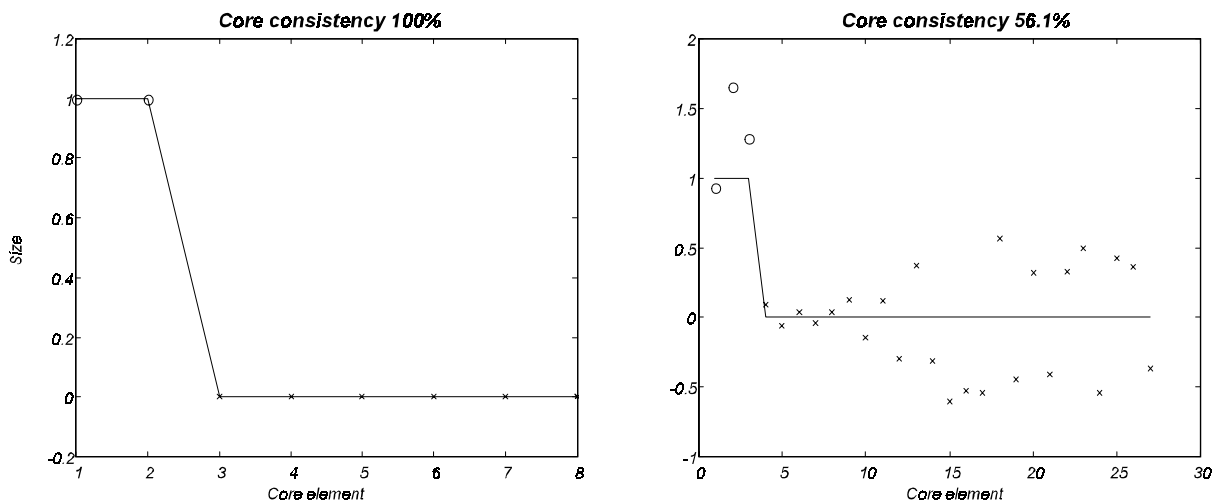


Figure 12. Core consistency plots of a two- and a three-component PARAFAC model of the bread data described on page 196.

In Figure 12 the core consistency plot of a two-component model of the bread data is perfect. The core elements on the superdiagonal (to the left in the figure) are close to one and all off-diagonal elements are zero. Further the superidentity is 100%. Hence there is no doubt that the two-component model is suitable from this point of view. For a three-component model the picture changes. This model is clearly inferior to the two-component model. Note that in this case the data consist of different assessors' judgement of different breads with respect to different attributes.

These data are noisy and there is no theory stating that the structure of the data should be trilinear. A two-component model seems appropriate, but the somewhat intermediate core consistency of the three-component solution indicates that some additional systematic variation is present there though. Further analysis is necessary in this case to elucidate the appropriateness of the three-component model.

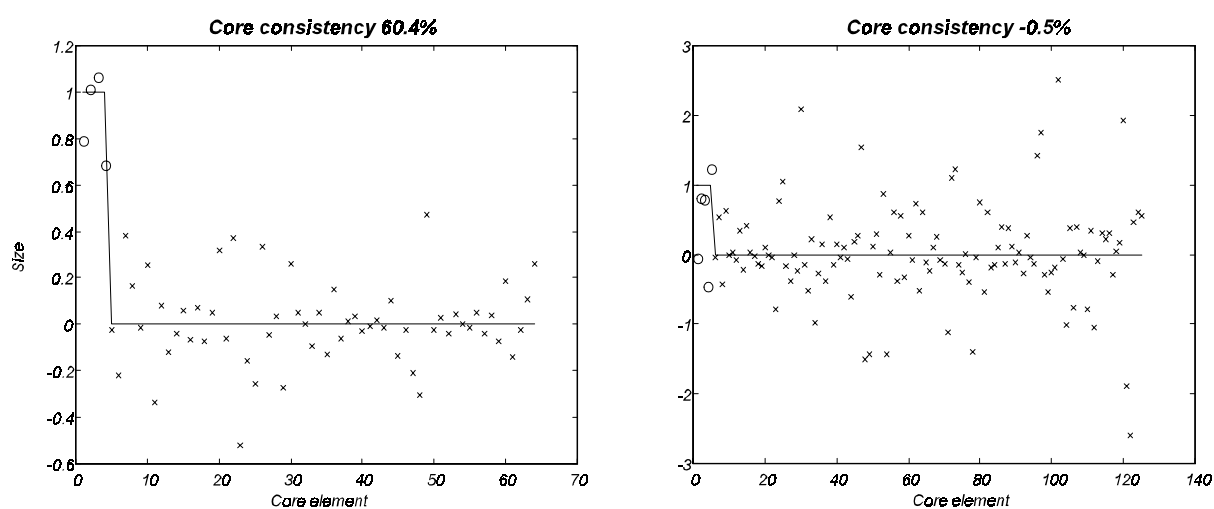


Figure 13. Core consistency plots of a four- and a five-component PARAFAC model of the sugar data described on page 230.

In Figure 13 the core consistency plots are not as clear-cut as for the other examples. Even though the five-component PARAFAC solution is inappropriate as judged from the core consistency plot, the four-component model is not perfect either. However, as the diagonality is high and as discussed on page 230, the data *are* difficult to model adequately, the four-component model may still be the best choice. Further, the deviation from perfect superidentity of the four-component solution simply points to the problems also described in detail in chapter seven.

If the core consistency diagnostic is to be used for other types of models, e.g., the restricted PARATUCK2 model (page 40) the principle is basically the same. For a given model the loading matrices of the model are used for calculating a Tucker3 core. To be able to compare with a

Tucker3 core, the corresponding restricted Tucker3 core of the model must be used, just like the superidentity array $\underline{\mathbf{I}}$ is used for the PARAFAC model. For a restricted PARATUCK2 model this array can be identified by restating the model as a restricted Tucker3 model. Suppose the restricted PARATUCK2 model has first mode model dimension R and second and third mode model dimension S . Then

$$\mathbf{X} = \mathbf{A}\mathbf{H}(\mathbf{C} \otimes \mathbf{B})^T \Rightarrow$$

$$\mathbf{X} = \mathbf{A}\mathbf{T}^{(R \times SS)}(\mathbf{C} \otimes \mathbf{B})^T, \quad (163)$$

where the core array $\underline{\mathbf{T}}$ has a specific structure. The element

$$t_{rss} \equiv h_{rs}, \quad (164)$$

and all other elements are zero. It is easily verified that the structural model thus posed is identical to the restricted PARATUCK2 model. It is of no concern if the interaction matrix is fixed or estimated. In both cases $\underline{\mathbf{G}}$ is compared to the expected core $\underline{\mathbf{T}}$.

When some loading matrices in a PARAFAC model do not have full column rank ten Berge & Kiers (p.c.) has pointed out that the core consistency diagnostic does not work because the rank of the problem defining the core is deficient. However, this problem can be easily circumvented. Assume that the first mode loadings, \mathbf{A} , of a PARAFAC model has dimensions 2×4 , i.e., there are only two levels in the first mode but four components. Such a situation occurs frequently in second-order calibration. Let \mathbf{A}_1 be two columns of \mathbf{A} that are not collinear and let \mathbf{A}_2 be the remaining columns. Define a 2×4 matrix \mathbf{H} as

$$\mathbf{H} = [\mathbf{I} \ \mathbf{A}_1^+ \ \mathbf{A}_2] \quad (165)$$

where \mathbf{I} is the two by two identity matrix. It then holds that the PARAFAC model

$$\mathbf{X}^{(I \times JK)} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T = \mathbf{A}_1 \mathbf{H}(\mathbf{C} \otimes \mathbf{B})^T, \quad (166)$$

i.e., the PARAFAC model can be posed as a restricted PARATUCK2 model. Since all loading matrices of this restricted PARATUCK2 model are of full rank the model can thus be tested as an ordinary restricted PARATUCK2 model. In practice, a QR decomposition can be used for rearranging the model.

The core consistency diagnostic may also be used for judging other models. It is not necessary that the model to test is unique, but it is essential that the model used to test the parameters against is less restricted than the model being judged. It is not feasible, for example, to test a two-way PCA model (\mathbf{TP}^T) against the 'core' of a bilinear model where the core is not restricted to be diagonal (\mathbf{AGB}^T). The posed problem here is to estimate

$$\underset{\mathbf{G}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{AGB}^T\|_F^2, \quad (167)$$

where \mathbf{A} and \mathbf{B} are set equal to \mathbf{T} and \mathbf{P} from the PCA model. The calculated core \mathbf{G} will be equal to the identity matrix. This is so because the model \mathbf{AGB}^T is mathematically equivalent to the model \mathbf{TP}^T . Having \mathbf{G} equal to the identity matrix is implicitly given even in the PCA model and as the model \mathbf{AGB}^T does not offer any increased modeling power no other setting of \mathbf{G} can give a better-fitting model.

The core consistency diagnostic often gives a clear-cut answer to whether a model is appropriate or not. It does not, however, tell if the model is *the* correct model. For a data set that can be modeled by, say, a three-component PARAFAC model, one will find that a one- and two-component PARAFAC model is also valid. The core consistency will consequently show that all these models are valid in the sense that they do not overfit. By assuming that noise is not trilinear however, it follows that the valid model with the highest number of components must be the one to choose. Also, though, it must be considered that another model structure or a model using other constraints or preprocessing may be more appropriate.

If a data set is modeled by, e.g., PARAFAC models of increasing number of components, the superidentity will typically decrease monotonically with the number of components. After the maximal number of

appropriate components the superidentity will decrease, though, much more dramatically, and often more clearly than if using a scree-plot or similar.

The core consistency diagnostic helps in choosing the proper model complexity. Further, no a priori assumptions regarding residuals are required, since it is the deterministic and systematic rather than the probabilistic part of the data that is being used for assessing the model. The results shown here for data of quite different nature indicate that it has a versatile applicability and it is suggested that it is used to supplement other methods for determining dimensionality.

5.5 CHECKING CONVERGENCE

When an algorithm has converged according to the convergence criterion used, it is important to know if the model can be considered the least squares model. For models like N-PLS and the Tucker models problems seldom occur. For PARAFAC, PARAFAC2, PARATUCK2, and similar models problems are sometimes encountered.

Two problems may arise: i) The algorithm stopped prematurely, ii) the algorithm converged but not to the least squares solution. If the algorithm has not converged to a minimum the cause is that the stopping criterion used in the algorithm is not sufficiently stringent. For, e.g., the PARAFAC algorithm mentioned in appendix A, the default convergence criterion is 10^{-6} meaning that if one iteration only improves the loss function value by less than 10^{-6} relative to the current loss function value, the algorithm is considered to have converged. This choice of stopping criterion is suitable in most cases. However, if the model parameters are very difficult to estimate a lower criterion may have to be used. This can be the case if the parameters are very correlated or if too many components are used. In such case the algorithm will stop at different loss function values if estimated several times from different initial parameter settings. Using a more stringent convergence criterion will remedy this problem.

In some situations local minima may exist. If this is the case the algorithm may converge to one of these instead of the global minimum. If the model parameters are estimated several times from different starting points the algorithm may converge to different minima. Unlike the above

situation, however, the number of local minima is usually finite and therefore several successive fitted models will yield only a finite number of different solutions. The difficulty when local minima arise is how to determine whether the global minimum has been found at all. The solution with the lowest loss function value is the best candidate. But, it is impossible to ascertain if a solution exist yielding a model with lower associated loss function value.

Mostly when a problem is well-posed model fitting is quite straightforward. Local minima problems are usually encountered when too many components are being extracted or the problem is ill-defined in some way. Imposing additional constraints or otherwise changing the model specification slightly may make the problem of local minima vanish. An example on this is given on page 215. Here the initial use of approximate orthogonality (page 154) seems to help minimizing the occurrence of the problem.

To assess if the global minimum has been attained it is thus essential to start the fitting procedure from several different sets of initial values, because otherwise the fitted models will be identical leaving no measures for assessing convergence.

5.6 DEGENERACY

Degenerate solutions are sometimes encountered when fitting models with oblique factors. Degeneracy is a situation where the algorithm has difficulties in correctly fitting the model for various reasons. The fitted models are hence often unstable and unreliable.

A typical sign of a degeneracy is that two of the components become almost identical but with opposite sign or contribution to the model. The contribution of the components almost cancels each other. In, e.g., PARAFAC each rank-one component can be expressed as the vectorized rank-one array obtained as

$$\mathbf{z}_f = \mathbf{c}_f \otimes \mathbf{b}_f \otimes \mathbf{a}_f \quad (168)$$

of size $IJK \times 1$. For the degenerate model it would hold that for some f and g

$$\mathbf{z}_f \approx -\mathbf{z}_g \Rightarrow$$

$$\cos(\mathbf{z}_f, \mathbf{z}_g) = \frac{\mathbf{z}_f^T \mathbf{z}_g}{\|\mathbf{z}_f\| \|\mathbf{z}_g\|} \approx -1. \quad (169)$$

This means that the loading vectors in component f and component g will be almost equal in shape, but negatively correlated.

An indication of degenerate solutions can thus be obtained by monitoring the correlation between all pairs of components. For three-way PARAFAC the measure in equation 169 can be calculated as

$$TC_{fg} = \cos(\mathbf{z}_f, \mathbf{z}_g) = \quad (170)$$

$$\cos(\mathbf{a}_f, \mathbf{a}_g) \cos(\mathbf{b}_f, \mathbf{b}_g) \cos(\mathbf{c}_f, \mathbf{c}_g) =$$

$$\frac{\mathbf{a}_f^T \mathbf{a}_g}{\|\mathbf{a}_f\| \|\mathbf{a}_g\|} \frac{\mathbf{b}_f^T \mathbf{b}_g}{\|\mathbf{b}_f\| \|\mathbf{b}_g\|} \frac{\mathbf{c}_f^T \mathbf{c}_g}{\|\mathbf{c}_f\| \|\mathbf{c}_g\|}$$

f and g indicating the f th and g th component. This measure is called the congruence coefficient (Tucker 1951)⁸. It is easy to show that the symmetric matrix \mathbf{TC} containing all possible congruence coefficients can be computed as

$$\mathbf{TC} = (\mathbf{A}^T \mathbf{A}) \circ (\mathbf{B}^T \mathbf{B}) \circ (\mathbf{C}^T \mathbf{C}), \quad (171)$$

for the three-way PARAFAC model if all loading vectors are scaled to length one.

For a degeneracy, the more iterations are performed, the closer the correlation between the two components will come to minus one. However, the correlation will theoretically never reach minus one (Harshman & Lundy 1984b, Kruskal 1984). Degeneracy occurs when the model is simply

⁸. Mitchell & Burdick (1993 & 1994) refer to TC as the uncorrected correlation coefficient.

inappropriate, for example, when data are not appropriately modeled by a trilinear model. In (Kruskal et al. 1989) some of these situations are referred to as *two-factor degeneracies*. When two factors are interrelated a Tucker3 model is appropriate and fitting PARAFAC models can then yield degenerate models that can be shown to converge completely only after infinitely many iterations, while the norm of the correlated loading vectors diverge. In such a case the Tucker (or restricted versions) or unfold models might be better (Kruskal et al. 1989, Kiers 1992, Smilde et al. 1994a & b). Another way of circumventing degenerate solutions is by applying orthogonality or non-negativity constraints on the model as this effectively prohibits negative correlations. It does not however remove the essential problem causing the degeneracy, nor will it always remove the problem of prolonged estimation. Degeneracies can also be caused by poor preprocessing. In such a case degenerate solutions are observed even for a low number of components, even when other information indicates that further systematic information is present. Extracting too many components can also give degeneracy-like problem. This will be easily recognizable, by the fact that models with fewer components yield nondegenerate solutions.

A degeneracy-like situation sometimes occurs where the correlation is just an interim algorithmic problem, that will disappear after a number of iterations. Such a situation has been referred to as a *swamp*, because the algorithm gets stuck only improving the loss function slowly for a long period. Some have reported that one way of overcoming swamps is by the use of regularization. Specifically ridge regression has been proposed by Rayens & Mitchell (1997) while Kiers (1998b) has advocated for a similar approach though as part of a more complex setting. Mitchell & Burdick (1993 & 1994) investigated swamps and found it profitable to do several runs of a few iterations, and only use those runs that are not subject to degeneracy. This is helpful for avoiding swamps that are merely caused by an unlucky initialization. This problem can also be solved using initial approximate orthogonality (page 154).

5.7 ASSESSING UNIQUENESS

As already described split-half analysis can help in checking whether a found solution generalizes over several samplings. While uniqueness only

pertains to a specific model of a specific data set, split-half stability is an even stronger criterion. Hence if a model is stable in a split-half sense, one may safely assume that the models involved are also unique, while the opposite is not necessarily true (see page 235). Unfortunately, in some cases there are not sufficiently many samples, and it is therefore not possible to reliably fit models from subsets of the original set of samples.

For PARAFAC uniqueness is almost always present. The practical problems in judging fitted PARAFAC models, thus mainly pertains to checking for convergence and if the uniqueness of the model can be interpreted in line with the underlying phenomena generating the data, e.g., by split-half analysis.

For some models, however, the uniqueness properties are not as well known. This holds especially for models like restricted PARATUCK2 and restricted Tucker3. Assessing uniqueness of such models can be divided into two parts: assessing if the structural model is unique in itself, and if this is not the case, assessing if the parameters can be determined uniquely by adding suitable constraints.

The uniqueness properties of the PARAFAC model are very attractive, regardless of whether one is interested in curve resolution or not. The fact that there is only one best solution simplifies interpretation of the model. Hence, for a given structural model it is interesting to know *a priori* if the model can be expected to be unique. Stringent mathematical proofs can sometimes be derived. E.g., for the PARAFAC2 and the PARATUCK2 model uniqueness has been proven under certain more or less mild conditions (Harshman & Lundy 1996, ten Berge & Kiers 1996). Also importantly, Kiers & Smilde (1998) have proposed different general means for assessing the uniqueness properties of hypothesized (restricted Tucker3) models.

Paatero (1998) has suggested that one may use the Jacobian of the loss function to investigate potential uniqueness of a specific solution. The rationale for this is that if there exists a direction in the multivariate space of all parameters where the derivative is zero, then one may change the parameters in that direction without affecting the fit. Hence, the solution is not unique. Thus, the uniqueness properties of a model given by specifically estimated parameters may be investigated by assessing the rank of the

Jacobian matrix. If this is of full rank⁹ the model is unique. In practice several numerical problems may complicate the assessments, but the simplicity and intuitive appeal definitely calls for further investigation of this approach. One may also envision this approach used for general inferences regarding the uniqueness of specific structural models under certain conditions regarding the rank or k -rank of the loading matrices.

Another way to assess uniqueness is by studying empirically if several fitted models of the data yield the same parameters. If this is not the case non-uniqueness has been verified (see Figure 24). If parameters of several re-fitted models are identical uniqueness cannot be rejected. However, it does not constitute a proof of uniqueness. Models of real data have local minima and non-smooth error surfaces which may make most estimates end up in the same solution although the real solution is by no means unique. Alternatively, uniqueness can be at least better assessed by decomposing synthetic data with the same characteristics as the hypothesized model. As the data are noise-free no problems arising from the noise part will occur, and non-uniqueness is therefore more likely to show up.

The above discussion *only* pertains to uniqueness arising from the structural model. For models that are known not to be unique, it is interesting to investigate if uniqueness can be obtained by the use of constraints. This area, however, is scarcely described in the chemometrics literature. While, it is advised, in curve resolution, to use additional appropriate constraints to narrow the set of possible solutions, only selectivity constraints have been shown to be able to provide definite uniqueness under certain conditions. As discussed elsewhere, non-negativity may also provide uniqueness, since non-negativity, if active, yields zero-parameters, which is equivalent to enforcing selectivity.

5.8 INFLUENCE & RESIDUAL ANALYSIS

Leverages and residuals can be used for influence and residual analysis. Such analyses are performed more or less as in standard multivariate analysis, and will therefore not be described in much detail here.

⁹. If scaling indeterminacies have not been fixed algorithmically each indeterminacy give rise to exactly loss of one rank in the Jacobian.

RESIDUALS

Residuals are easily calculated by subtracting the model from the data. These residuals can be used for calculating variance-like estimates (Smilde & Doornbos 1992) or they can be plotted in various ways to detect trends and systematic variation (Ross & Leurgans 1995). One difficulty arises in residual analysis, because degrees of freedom are not available. This is implicitly related to the discussion of rank on page 12. Consider for example a $2 \times 2 \times 2$ array. Such an array will contain 8 elements, hence 8 degrees of freedom. Each PARAFAC component of this array will consist of 6 parameters (2 in each mode). Even though a two-component model will comprise 12 parameters this is not always enough to describe all the variation in the data, as some $2 \times 2 \times 2$ arrays are of rank three. Thus simple counting of parameters does not lead to any good approximation of degrees of freedom. In fact, the problem of finding explicit rules for the maximal rank of arrays, directly points to the difficulty of defining degrees of freedom. Also, the fact that different randomly made arrays can have different ranks indicate, that degrees of freedom do not exist *a priori*, but has to be determined from the specific data. This is a very surprising situation.

MODEL PARAMETERS

As for other decomposition methods plots of the parameters are often helpful in elucidating the interrelations in terms of correlated or clustered variables/objects or outliers. In accordance with the theory of biplots (Gabriel 1971), e.g., triplots, also called joint plots, for a three-way PARAFAC model can be generated. For models that incorporate interactions between factors (most notably Tucker3) such plots should be interpreted with great care, taking in to considerations the magnitudes of interactions as given by the core array.

One may also use Mahalanobis distances ('leverages') for describing the relative importance of different objects or variables according to the model. For a given mode with loading matrix \mathbf{A} the leverages are calculated as

$$\mathbf{v} = \text{diag}(\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T), \quad (172)$$

The leverage for the i th sample or variable, v_i , is the i th element of \mathbf{v} and has a value between zero and one (Cook & Weisberg 1982). A high value indicates an influential sample or variable, while a low value indicates the opposite. Samples or variables with high leverages and low in case of a variable mode must be investigated to verify if they are inappropriate for the model (outliers) or are indeed influential and acceptable. If a new sample is fit to an existing model, the sample leverage can be calculated using the new scores for that sample as in ordinary regression analysis. The leverage is then no longer restricted to be below one.

5.9 ASSESSING ROBUSTNESS

When a model has been chosen it is essential to verify that it is 'real'. One way of validating that the chosen model is robust and hence not a product of chance is to verify that minor changes in the modeling procedure will not affect the conclusions drawn from the model. This can be verified in a number of ways:

PROBABILISTIC VARIATION: Models may be fitted under slightly different distributional assumptions using loss functions that are based on: least squares, weighted least squares, least absolute deviations etc.

SAMPLING VARIATION: It may be verified that the model does not essentially change when using different samples using cross-validation, bootstrapping, etc.

VARIABILITY: The usefulness of a model may be confirmed from how replicates are modeled. These should be close to each other in the model space, and the residual variation should be comparable to the variation between replicates.

EXTERNAL VALIDITY: The extrapolation properties may be tested by verifying that the model also works on new samples that were not part of the model building step.

ALGORITHMIC VALIDITY: Changing, e.g., convergence criterion should not

change the fitted model.

MEASUREMENT LEVEL AND ORDER: The model should not be sensitive to slight changes in basic assumptions of the measurements level and order characteristics (page 156).

MODEL VALIDITY: The model should reflect what is already known about the problem. Conclusions should not change by small changes in the structural model or constraints.

5.10 FREQUENT PROBLEMS AND QUESTIONS

When actually trying to do multi-way analysis in practice many questions arise; several related to basic problems of understanding the intrinsic two-way nature of many programs and the multi-way nature of the data. Here a selection of common problems is discussed.

What is multi-way PCA?

It is unfortunate and confusing, that the term multi-way PCA is used in chemometrics for unfolding and doing ordinary two-way PCA (hence Tucker1) modeling. This is unfortunate since multi-way PCA is easily confused with multi-mode PCA which is the term accepted for the Tucker models in general. It also seems to confuse model structure with data structure. In multi-way PCA (unfolding PCA) the multi-way array is rearranged to a two-way array and analyzed as such; hence, there is no usage of the multi-way structure in the modeling step. Instead of using the term multi-way PCA (or PLS for that matter) unfold-PCA or similar should be used.

Is multi-way analysis only useful for spectral analysis?

No! Though multi-way models seem appropriate for modeling spectral data, the benefits of using multi-way analysis for non-spectral data can actually be more significant. This is similar to two-way analysis. The structure imposed by using multi-way models as opposed to unfolding and doing two-way analysis can make an immense difference for noisy data.

Consider a simple hypothetical example using spectroscopy. For ten

samples the fluorescence emission spectrum (200 wavelengths) is measured at five excitation wavelengths. Assume that the samples only contain tyrosine in different amounts. The array is thus $10 \times 200 \times 5$, and the unfolded array 10×1000 .

When modeling the unfolded data by a one-component bilinear model the following assumption is implicitly made: The latent phenomenon (as given by the emission spectrum of tyrosine) at a certain instance (a specific excitation) is *completely* independent of the phenomenon (emission spectrum) at another instance (excitation). Clearly this is seldom correct; most often the loadings or phenomena will be similar at different occasions.

In multi-way analysis the opposite assumption holds: The phenomenon (emission spectrum) describing the variation at one instance (excitation) is exactly the same as the phenomenon describing it at another instance, though of a different magnitude. This may often also be incorrect, but mostly closer to the truth than the assumption underlying unfolding techniques. Furthermore, to the degree that a latent phenomenon cannot be described in such a simple manner, using extra components can remedy the approximation error simpler than using *one* unfolding component.

The above can be compared with how common multivariate regression techniques work. In multivariate regression a regression vector that indirectly gives a description of the dependent variable is sought. In multiple linear regression this is done by finding the regression vector that maximizes the fit. In PCR and PLS and methods alike the interest is also in maximizing the fit, but under the constraint that the regression coefficients are related (through the model of the independent variables). The PCR/PLS approach is advantageous even in cases where the bilinear model of \mathbf{X} is not correct in terms of an underlying physical model, hence not only for, e.g., spectral analysis. In the same sense, e.g., unfold-PCA maximizes the fit of the model to the data in a bilinear sense. The multi-way model does the same, but under the constraint that the loading vectors on different occasions are related.

Can an unfolded array be scaled like in two-way analysis prior to multi-way modeling?

Suppose an $I \times J \times K$ array is arranged as an $I \times JK$ matrix in the computer. If the variables are measured on different scales it may be tempting to just scale or center and scale this matrix as in two-way analysis. This, however, will destroy the structure of the data, if the data do conform to the structural model. This has been explained in more detail on page 104. The important message is, that two-way preprocessing tools should not be adopted for multi-way arrays. They are not appropriate.

How many components can be extracted from a data set of only two samples?

Generally many. If a $2 \times J \times J$ three-way data set is arranged as a matrix of size $2 \times JJ$ only two bilinear components can be extracted. If, alternatively, the data are arranged as, e.g., a $J \times 2J$ matrix then J components can be extracted. However, when treated as a three-way array the number of components that can be extracted is even higher. The number of meaningful components that can be extracted of course depends on the nature of the data, but theoretically the largest integer less than $3J/2$ components can be extracted (Kruskal 1989), hence more than when the array is unfolded.

A column in the loading matrix only consists of zeros

Sometimes when non-negativity constraints are used it may happen that a column in a loading matrix (intermediately) has only zeros. The unfortunate consequence is that the interim model essentially loses rank and the fitting of the model becomes impossible. It may happen that the true solution is the one with zeros in one column, in which case the model should be fitted using one component less. Most often, though, the problem is simply a numerical one which is easily remedied. Suppose \mathbf{A}^{old} is the prior estimate of \mathbf{A} and \mathbf{A}^{curr} is the current. Assuming that \mathbf{A}^{old} is unproblematic and \mathbf{A}^{curr} has a column of zeros, a better estimate of \mathbf{A} is simply found as e.g.,

$$\mathbf{A}^{\text{new}} = \frac{1}{2}(\mathbf{A}^{\text{old}} + \mathbf{A}^{\text{curr}}). \quad (173)$$

This estimate of \mathbf{A} will not be the conditional least-squares estimate of \mathbf{A} , but it will give an estimate that improves the model without lowering the

rank.

The model does not converge to the same solution every time

If the fit of several fitted models of the same data differ it is sensible to assume that the model has simply stopped prematurely. For some models the algorithm converges extremely fast, while for others many iterations are required for fitting the model. If the fit of repeated fittings are identical but the model parameters differ, then it may simply be concluded that the model is not identified (unique). Incorporating certain constraints or using fewer components may help avoiding that. If the model consistently converges to different solutions, local minima exist, in which case a slight change of the model (changing constraints, structure, loss function etc.) may be helpful in avoiding this.

The loadings shift in sign from estimation to estimation

All multilinear models have an intrinsic indeterminacy with respect to scaling and permutation. That is, component one may be exchanged with component two, or the scale or sign of a loading vector may change if another loading vector of the same component is changed accordingly. This holds even for PCA, but it is standard procedure to adopt a scaling and permutation convention in PCA, so that the first component has the largest variance and, e.g., the loading vector is scaled to norm one. Similar conventions can be adopted for other models. Care has to be taken, however, in how these conventions are implemented if the model is fitted, e.g., under equality constraints between components.

The model looks poor though it describes almost all variation

Often, in spectral applications, no centering is performed in order to make the results more interpretable and because centering is not necessary. However, if typical spectral data has not been centered any moderately appropriate model will explain most of the variation due to the high common level of the variables.

5.11 SUMMARY

In this chapter practical aspects of multi-way modeling have been

discussed under the general framework of validation. There are many levels of validation and the most important one is to validate the model with respect to how well it explains, generalizes, or predicts the real-world problem it is aimed at describing. This aspect has not been discussed much here, partly because it is problem-dependent and mainly because it would only be a repetition of theory from standard multivariate analysis. Instead the focus has been on data-analytical aspects specific to multi-way modeling.

Choosing a model and model complexity has been discussed including a description of a new tool called the core consistency diagnostic for investigating adequacy of a posed model. Preprocessing of multi-way arrays has been explained emphasizing that care has to be taken not to destroy any structure in the data.

Especially for the PARAFAC, the PARAFAC2, and the PARATUCK2 model it is important to be able to assess convergence, degeneracy, and uniqueness. Several diagnostics for this purpose have been explained, the most important ones being to fit the data several times from different starting points and fitting different subsets of the samples. Finally influence and residual analysis and overall model validity has been discussed.

CHAPTER 6

CONSTRAINTS

6.1 INTRODUCTION

Constraining a model can sometimes be helpful. For example resolution of spectra may be wanted. To ensure that the estimated spectra make sense it may be reasonable to estimate the spectra under non-negativity constraints as most spectral parameters are known to be non-negative. Constraints can for example help to

- Obtain parameters that do not contradict with *a priori* knowledge
Ex.: Require chromatographic profiles to have but one peak
- Obtain unique solution where otherwise a non-unique model would be obtained
Ex.: Use selective channels in data to obtain uniqueness
- Test hypotheses
Ex.: Investigate if tryptophane is present in sample
- Avoiding degeneracy and numerical problems
Ex.: Enabling a PARAFAC model of data otherwise inappropriate for the model
- Speed up algorithms
Ex.: Use truncated bases to reexpress problem by a smaller problem
- Enable quantitative analysis of qualitative data
Ex.: Incorporate sex and job type in a model for predicting income

Some researchers argue that constraining, e.g., the PARAFAC model is superfluous, as the structural model in itself should be unique. However, there are several good reasons for wanting to use constraints. Not all models are unique like the PARAFAC model. And even though the model is unique, the model may not provide a completely satisfactory description of the data. Rayleigh scatter in fluorescence spectroscopy is but one instance where slight model inadequacy can cause the fitted model to be misleading (page 230). Constraints can be helpful in preventing that. In other situations numerical problems or intrinsic ill-conditioning can make a model problematic to fit. At a more general level constraints may be applied simply because they are known to be valid. This can give better estimates of model parameters and of the data (page 207).

A constrained model will fit the data poorer than an unconstrained model, but if the constrained model is more interpretable and realistic this may justify the decrease in fit. Applying constraints should be done carefully considering the appropriateness beforehand, considering why the unconstrained model is unsatisfactory, and critically evaluating the effect afterwards. In some cases there is confidence in that the constraint is appropriate. If spectral data are analyzed negative parameters are often unthinkable. In such a case the model should be fitted under non-negativity constraints. In other cases there is uncertainty in the degree of appropriateness of the constraint. For example, in the flow injection analysis example (page 207) it is used that all analytes have the same elution profile. If, however, the analytes differ markedly chemically, this may not be fulfilled exactly. In such a case it can be appropriate to apply the constraint softly, i.e., letting a solution fulfilling the constraint be more probable than a solution that does not. This can be accomplished using a penalty approach, in which a penalty is added to the loss function if the solution differs from the constraint. By adjusting the size of the penalty the degree of fulfillment of the constraint can be controlled.

Some researchers use an intuitive approach for imposing constraints. Using non-negativity constraints as an example a common approach is to use an unconstrained estimation procedure and then subsequently set negative values to zero. Naturally this will lead to a feasible solution, i.e., a solution that is strictly non-negative. This approach can not be

recommended for several reasons. First of all, an estimate obtained from such an approach will have no well-defined optimality property. This can make it difficult to distinguish between problems pertaining to the algorithm, the model, and the data. That is, if the fitted model is unsatisfactory, it becomes more difficult to assess the cause of the problem, because an additional source of error has been introduced, namely the properties of the constraining algorithm.

The possible problems arising from using such ad hoc methods can be demonstrated easily by an example. Consider a matrix \mathbf{Z} of independent variables and a matrix \mathbf{y} of dependent variables. Let

$$\mathbf{Z} = \begin{bmatrix} 73 & 71 & 52 \\ 87 & 74 & 46 \\ 72 & 2 & 7 \\ 80 & 89 & 71 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 49 \\ 67 \\ 68 \\ 20 \end{bmatrix}. \quad (174)$$

The regression vector, \mathbf{a} , of the least squares problem

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Za}\|_F^2 \quad (175)$$

can be found as $\mathbf{a} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} = [1.123 \ 0.917 \ -2.068]^T$. Setting the negative element to zero the estimated solution under non-negativity constraint is $\mathbf{a} = [1.123 \ 0.917 \ 0]^T$. The root mean squared error corresponding to this regression vector equals 103. If the non-negativity constrained regression vector is properly estimated the result is $\mathbf{a} = [0.650 \ 0 \ 0]^T$, yielding a root mean squared error of 20. Thus, the approximate regression vector is not even close to an optimal or good solution. Another serious problem with this approximate approach is that when included in a multi-way algorithm it can cause the algorithm to diverge, i.e., successive iterations yield intermediate models that describe the data poorer and poorer.

Another example showing the importance of using as efficient and direct algorithm as possible comes from the restricted PARATUCK2 model shown

on page 42. The data array contains very many missing values (36.4%). Additionally the restricted PARATUCK2 model is not structurally unique as two of the loading vectors in the spectral modes have a rotational ambiguity. Only, when applying non-negativity the model is unique. The important aspect here is that this model is quite difficult to fit. For handling missing values two different approaches was used. One is to simply ignore the missing values, hence only optimizing over the present elements. This is achieved by using weighted regression as explained page 146. This approach directly optimizes the loss function over all non-missing elements. Another approach is based on iteratively imputing missing elements from the interim model of the data. As there are then no missing data an ordinary algorithm can be used for fitting the model. This approach can be shown to have identical optimum as the former, but it does not optimize the wanted objective function directly. This situation is to some extent comparable to using poorly defined algorithms for constrained problems. Even though the algorithm does not directly optimize the loss function the hope is that the final solution will be close to the true least squares solution. Here, the imputation method, even though it is actually very well-defined as compared to many ad hoc constrained regression algorithms, serves a similar role. Normally, there is no difference between using the two different approaches for handling missing data. In this case though, using imputation does not give the least squares solution even when initiated by a very good approximate solution. In Figure 14 the excitation loading vectors obtained by the two algorithms are shown. Evidently there are large discrepancies. The model corresponding to the left plot explains 99.546% of the variation in the data while the one to the right explains 99.535%¹⁰. Hence the two models are almost equally good as judged from the fit even though the parameters are very different. This case is rather extreme because of the large amount of missing data and the rotational indeterminacy of the structural model, but it is exactly in the difficult cases, that it is important to be able to discern between problems pertaining to the algorithm, problems pertaining to the model and problems pertaining to the data. Using a well-defined and sound algorithm will help in that.

¹⁰.For both models the fitting procedure was based on fitting several randomly started algorithms and picking the best.

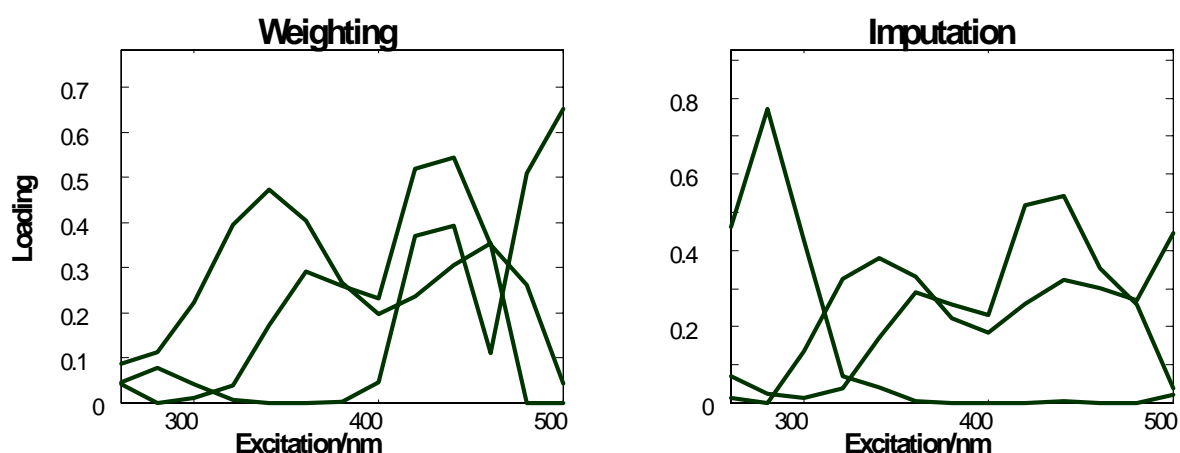


Figure 14. Resulting emission loading vectors obtained using either weighting (left) or imputation (right) for handling missing data when modeling fluorescence data with many missing elements.

One of the main reasons for using ALS algorithms is that they are guaranteed to converge monotonically. When incorporating new sub steps in an ALS algorithm it is therefore important to make sure that these sub steps also give conditional least squares estimates to retain the convergence properties.

DEFINITION OF CONSTRAINTS

A constrained model is characterized by the model specification contains restrictions on some parameters. This chapter, however, will also contain some additional subjects, such as situations where the loss function is not simply the minimum sum-of-squares of the residuals of the structural model. Specifically, fitting a model with a weighted loss function or fitting a model from incomplete data (missing values) will also be treated here.

The unconstrained three-way PARAFAC model is implicitly given by the objective function

$$\min_{\mathbf{a}_i, \mathbf{b}_j, \mathbf{c}_k} \left\| \mathbf{x}_{ijk} - \sum_{f=1}^F \mathbf{a}_{if} \mathbf{b}_{jf} \mathbf{c}_{kf} \right\|_F^2 \quad (176)$$

whereas a model defined by the objective function

$$\min_{\mathbf{a}, \mathbf{b}, \mathbf{c}} \left\| \mathbf{x}_{ijk} - \sum_{f=1}^F \mathbf{a}_{if} \mathbf{b}_{jf} \mathbf{c}_{kf} \right\|_F^2 \quad (177)$$

subject to $\mathbf{b}_{jf} \geq 0$

is a constrained model as the elements of \mathbf{B} are required to be non-negative.

EXTENT OF CONSTRAINTS

Constraints can be applied as either approximate or exact constraints. Applying, e.g., non-negativity as an exact constraint means that all parameters should be non-negative, while applying non-negativity as an approximate constraint, means that negativity of parameters is allowed but non-negativity is preferred. Most constraints described in the sequel are exact constraints, but can be easily modified to become approximate constraints. This can be helpful when it is uncertain if the constraint is appropriate or if it is known only to be approximately true. Requiring that a spectral loading equals the spectrum of an analyte obtained from a pure solution may be too strict as the spectrum can change slightly in the sample matrix, so applying the constraint as an approximate constraint will be more adequate.

UNIQUENESS FROM CONSTRAINTS

One of the main issues in using constraints is whether uniqueness can be obtained from an otherwise non-identified model by using constraints. It is important to emphasize again that there are many other reasons for using constraints, e.g.:

- Simply using the right structure can be considered a constraint that helps filtering away noise as exemplified in the sensory example page 196.
- The unimodality and equality constraints in the FIA example page 207

are not necessary for obtaining uniqueness of the concentration estimates. However, since they are appropriate applying them results in better estimates of concentrations and spectra. Applying constraints in this case also helps avoiding problems with local minima and swamps.

- The main purpose of applying the unimodality constraint in the sugar example page 230 is to obtain better visual appearance for easier identification of the chemical analytes. The models fitted without unimodality are unique, but they are not stable in a split-half sense, i.e., they are partly determined by irrelevant variation.
- Orthogonality constraints are often used in multi-way models for avoiding numerical problems or to be able to partition variance more easily. As discussed on page 154 and in the FIA application page 207 it is also helpful for making some ALS algorithms faster and less susceptible to local minima.

Some constraints can be effective for obtaining uniqueness but most can only help reducing the possible feasible set of parameters, thereby giving a smaller span of possible solutions. For structural models uniqueness properties can sometimes be verified, e.g., that the PARAFAC model is unique under certain specified conditions. For constraints however, this is not always so. Non-negativity can infer uniqueness automatically (ten Berge & Kiers 1996), or implicitly by enforcing appropriate zero-parameters so that rotations are not possible. There can be large differences in the degree of uniqueness that can be expected from different constraints. Constraining loading vectors to be unimodal is not likely to lead to uniqueness, while constraining individual loading vectors to be symmetric over a mode is (Manne 1995).

6.2 CONSTRAINTS

Many types of constraints can be imagined. It is appropriate to consider first the different types of constraints and their typical areas of applications. In a subsequent section specific algorithms for some of the constrained problems will be developed.

SECOND-ORDER ADVANTAGE**BOX 14****Spectral calibration with unknown interferents**

In PCA the structural model is $\mathbf{X} = \mathbf{TP}^T$. The scores in \mathbf{T} can be interpreted as pseudo-concentrations, i.e., concentrations of *latent* variables defined by the loadings vectors. In PARAFAC the model is $\mathbf{X} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T$. Here \mathbf{B} and \mathbf{C} are estimates of pure-analyte spectra, i.e., \mathbf{A} contains estimates of relative concentrations of real analytes. Theoretically there is therefore no need for a regression step as in PCR/PLS as the scores directly reflect the concentrations (up to a scaling factor).

Example:

Calibration set: One sample with *one* analyte (2.67 μM Trp)

Test set: One sample with three analytes (Trp, Tyr, Phe)

The data for each sample is a 201×61 array of fluorescence emission-excitation of the sample. The data were analyzed using GRAM and PARAFAC and also using PARAFAC with non-negativity constraints and fixing the estimated concentrations of interferents to zero in the standard sample.

Reference	GRAM	PARAFAC	PARAFAC constrained
0.879 μM	0.920 (4.6)	0.920 (4.6)	0.882 (0.3)

The number in parentheses is the relative error in percent. It is seen that all three approaches work well. Unconstrained PARAFAC and GRAM perform similarly, and the constrained PARAFAC better. From this example it seems that incorporating the *a priori* knowledge is helpful in getting better results.

FIXED PARAMETERS

Fitting a model subject to certain parameters are fixed is a quite simple problem, which can be useful for different purposes.

If a structural model is sought that is not represented by any known model it may be possible to obtain the model using a known model as a basis by fixing certain parameters. For an illustrative example of little practical use one may obtain a PARAFAC model from a Tucker3 model by fixing the elements of the core to a superidentity array, i.e., an array where there are only ones on the superdiagonal (see page 40 and 192 for other examples).

If a constant baseline is present or if a lower-order effect is to be estimated in an ANOVA model this corresponds to setting certain loading vectors to ones (page 194). If a second-order calibration problem is to be solved using pure standards, then it is known that the concentration of interferences is zero in the standard, which may be incorporated into the model (Box 14). An algorithm for solving the problem of having fixed parameters is shown on page 167.

TARGETS

If certain spectra are known or supposed to be present in the data it may be assured that these spectra are also present in the model by fixing some of the spectral loadings to equal the preknown spectra. This can be done similarly to fixing parameters only in this situation it may be feasible to use an approximate implementation of the constraint in order to allow for possible deviations in the spectra caused for example by matrix effects (see page 167).

SELECTIVITY

In traditional curve resolution the problem is to uniquely decompose a data set into estimated pure spectra and profiles (chromatography). Often, the data used for curve resolution are two-way data which are modeled by a bilinear model. As the bilinear model has rotational freedom it is necessary to use additional restrictions to obtain a unique solution. It is well-known (Manne 1995) that if certain variables are selective, i.e., contain information from one analyte only, or if some variables are completely unaffected by one analyte, then unique or partly unique resolution can be obtained. Selectivity is *the* constraint for obtaining unique decompositions in two-way analysis. It naturally follows that knowledge of selective variables in multi-

way decompositions will also be beneficial for circumventing possible non-uniqueness. There is a multitude of methods for determining the presence and degree of selectivity (see references page 190). What primarily is added here is a way to express and incorporate any sort of selectivity precisely into the overall model formulation. Hopefully such an approach could lead to more robust models.

Suppose that a multilinear model is sought for a data set, be it two-way or higher-way. For a given mode a set of parameters, \mathbf{A} , is sought. From other sources knowledge is available on the selectivity pattern of this parameter set. Imagine \mathbf{A} being an estimate of the chromatographic profiles of analytes in a sample, each column being an estimate of a certain analyte's time profile and each row corresponding to a specific time. If it is known that only one analyte is present at the first, say two points in time, then this corresponds to requiring that in the two first rows of \mathbf{A} all elements should be zero apart from in one column. The form of \mathbf{A} will thus be

$$\mathbf{A} = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & 0 & & 0 \\ a_{31} & a_{32} & & a_{3F} \\ \vdots & & \ddots & \\ a_{I1} & a_{I2} & & a_{IF} \end{bmatrix}, \quad (178)$$

where all nonzero elements are subject to optimization. The problem of estimating \mathbf{A} under the constraints of these selectivity requirements comes down to the problem of having fixed parameters.

There is an important distinction between the way selectivity is typically used in two-way analysis and the approach outlined here. For the above problem in a two-way setting the spectrum of the first analyte would be estimated using only the first two points in time, i.e., the variables where selectivity is present. This way less robust results are obtained as the whole data set is not used for estimating the spectrum. The benefit, however, is that uniqueness is guaranteed in situations where the overall

model would not always be unique.

When it is not known beforehand which parameters are almost zero, a penalized approach can be used where all parameters are estimated subject to zero being the most attractive value (ridge regression). After fitting the model visual inspection can reveal which parameters are zero and if a unique solution has been obtained. If uniqueness has not been obtained the model is re-fitted with a higher penalty for deviating from zero until uniqueness is obtained. Finally a model is fitted without a penalty but with the zero-parameters now forced to zero. The patterns of zeros in this model will ensure that the model is unique. If the discrepancy between this model and the original non-unique model is small the unique model may be the preferable choice.

WEIGHTED LOSS FUNCTION

Suppose the uncertainty of individual data elements are known. It is then possible to take this into account by fitting the model, \mathbf{M} , with a weighted loss function

$$\min_{\mathbf{M}} \|\mathbf{W} \circ (\mathbf{X} - \mathbf{M})\|_F^2, \quad (179)$$

where \mathbf{W} holds the uncertainty of individual data elements and ' \circ ' is the Hadamard (element-wise) product. Weighted regression gives the possibility to incorporate knowledge of the uncertainty of the data elements in the decomposition, and also the possibility to do iteratively reweighted regression for obtaining more robust models. Weighting also provides a way of handling missing data as missing elements can be simply set to weight zero when fitting the model. There are other ways of treating incomplete data to be described.

For improving the robustness of estimated parameters iteratively reweighted regression may be used (Phillips & Edward 1983, Griep et al. 1995). The basic principle is that the interim model is fitted and from this the interim model residuals of each data element can be calculated. These interim residuals are then used for defining a set of weights used in the subsequent iterative cycle using a weighted regression approach. The

basic rationale for this approach is that elements with high uncertainty and outliers will be downweighted as they will have high residuals. However, it is well-known from regression analysis that high-leverage points, i.e. outliers, will often have small residuals as they tend to steer the model towards them, while low- or medium-leverage points, i.e., good data points, can have quite high residuals for the same reason (Cook & Weisberg 1982). The residuals themselves are hence not always good estimates of the appropriateness of the data point with respect to the model. In regression analysis the leverage-corrected or studentized residuals are proposed as a better measure of the sought feature. Similar means for correcting residuals of decomposition models are currently not available.

Regression incorporating individual weights is easily performed using weighted regression. However, when other constraints are imposed than weighting, it may happen that ordinary weighted regression becomes difficult. An example is unimodality as discussed in Bro & Sidiropoulos (1998). Kiers (1997), however, has shown how to modify any unweighted ordinary least squares algorithm to a weighted algorithm. The basic principle is to iteratively fit the ordinary least squares algorithm to a transformed version of the data. This transformation is based on the actual data as well as on the current model-estimate and the weights. In each step, ordinary least squares fitting to the transformed data improves the weighted least squares fit of the actual data. For details on this approach the reader is referred to Kiers (1997).

MISSING DATA

Missing values should be treated with care in any model. Simply setting the values to zero is sometimes suggested, but this is a *very* dangerous approach. The missing elements may just as well be set to 1237 or some other value. There is nothing special about zero. Another approach is to impute the missing elements from an ANOVA model or something similar. While better than simply setting the elements to zero, this is still not a good approach. In two-way PCA and any-way PLS fitted through NIPALS-like algorithms the approach normally advocated for in chemometrics (Martens & Næs 1989) is to simply skip the missing elements in the appropriate inner products of the algorithm. This approach has been shown to work well for

a small amount of randomly missing data, but also to be problematic in some cases (Nelson et al. 1996).

A better way, though, to handle missing data follows from the idea that the model is fitted by optimizing the loss function *only* considering non-missing data. This is a more sensible way of handling randomly missing data. The loss function for any model of incomplete data can thus be stated

$$\min_{\mathbf{M}} \|\mathbf{W} \circ (\mathbf{X} - \mathbf{M})\|_F^2, \quad (180)$$

where \mathbf{X} is a matrix containing the data and \mathbf{M} the model (both unfolded). The structure (and constraints) of \mathbf{M} are given by the specific model being fitted. The matrix \mathbf{W} contains weights that are either one if corresponding to an existing element or zero if corresponding to a missing element. If weighted regression is desired \mathbf{W} is changed accordingly keeping the zero elements at zero. The natural way to fit the model with missing data is thus by a weighted regression approach covered in the preceding section¹¹.

Another approach for handling incomplete data is to impute the missing data iteratively during the estimation of the model parameters. The missing data are initially replaced with either sensible or random elements. A standard algorithm is used for estimating the model parameters using all data. After each iteration the model of \mathbf{X} is calculated, and the missing elements are replaced with the model estimates. The iterations and replacements are continued until no changes occur in the estimates of the missing elements and the overall convergence criterion is fulfilled. It is easy to see, that when the algorithm has converged, the elements replacing the missing elements will have zero residual.

How then, do these two approaches compare? Kiers (1997) has shown that the two approaches give identical results, which can also be realized by considering data imputation more closely. As residuals corresponding

¹¹. Note that, though the standard approach in two-way PCA/PLS is also based on a sort of weighted regression, this does not lead to an optimal model according to equation 180, because the components are estimated successively. Actually, algorithms for PCA and PLS based on NIPALS do not produce orthogonal scores and loadings when missing values are present.

to missing elements will be zero they do not influence the parameters of the model, which is the same as saying they should have zero weight in the loss function. Algorithmically, however, there are some differences. Consider two competing algorithms for fitting a model of data with missing elements; one where the parameters are updated using weighted least squares regression with zero weights for missing elements and one where ordinary least squares regression and data imputation is used. Using direct weighted least squares regression instead of ordinary least squares regression is computationally more costly per iteration, and will therefore slow down the algorithm. Using iterative data imputation on the other hand often requires more iterations due to the data imputation (typically 30-100% more iterations). It is difficult to say which method is preferable as this is dependent on implementation, size of the data, and size of the computer. Data imputation has the advantage of being easy to implement, also for problems which are otherwise difficult to estimate under a weighted loss function.

NON-NEGATIVITY

Fitting models subject to non-negativity constraints is of practical importance in chemistry as well as many other disciplines (Durell et al. 1990, Paatero 1997, Bro 1997, ten Berge et al. 1993). Most physical properties like concentrations, absorptivities, etc. are non-negative. In Lawton & Sylvestre (1971) it was conjectured that using non-negativity will significantly reduce the feasible space of the parameters to be estimated. Thereby the estimated components may be determined up to a small rotational ambiguity. Another, stronger property, of using non-negativity, however, is that it leads to an increased number of zero elements which may prevent rotations and therefore in some situations provides uniqueness. Non-negativity is an inequality constraint but of a simple kind, namely a bounded problem. Such a problem can be efficiently solved by an active set algorithm (Gill et al. 1981). For ALS, however, the standard algorithms tend to be slow. On page 169 a tailor-made algorithm of Bro & de Jong (1997) is given, that cuts computation time considerably.

INEQUALITY CONSTRAINTS

When compression (page 88) is used some constraints become more complicated. Consider a three-way PARAFAC model where non-negativity is imposed on the first mode loadings \mathbf{A} . When fitting a PARAFAC model to the compressed array, \mathbf{Y} , it is the backtransformed parameters that should be non-negative. If the matrix \mathbf{U} defines this transformation (equation 140) and $\mathbf{\Delta}$ is the first mode loadings estimated from the compressed array then the non-negativity constraint can be written

$$\mathbf{A} = \mathbf{U}\mathbf{\Delta} \geq \mathbf{0}. \quad (181)$$

This is an inequality constraint on $\mathbf{\Delta}$, which can be solved in a variety of ways. Inequality constraints are a quite general type of constraints, that can also be useful, e.g., for imposing smoothness. Standard algorithms for handling inequality constrained regression are not appropriate for the problem here, however, due to the number of constraints, as these algorithms only handle problems where the matrix \mathbf{U} has full row rank.

Work is currently in progress finding an appropriate algorithm for this specialized purpose. Algorithms used are based on direct active set algorithms (Gill et al. 1981) and quadratic programming. A complicating issue in solving the problem is that some constraints, i.e., rows of \mathbf{U} , may be redundant and thereby unnecessarily complicate the computations. As \mathbf{U} is generally known beforehand it seems reasonable to remove such constraints before fitting the model. Finding these constraints is a relatively complicated task, however, that requires a lot of computations.

It may seem at first that an algorithm for this inequality constrained problem will be too complicated to be beneficial. However, though the algorithm is complicated, it will not during the main fraction of iterations be computationally complex. If the active set is correct only an equality constrained least squares problem has to be solved and checked. Also, it has been shown in several instances (Bro & Andersson 1998, Kiers 1998b) that it can make a significant difference in terms of computational cost if the model fitted to the compressed array is not very close to the true solution. This especially holds for constrained problems. Hence the closeness of the

solution obtained from the compressed data is important, and it is therefore important to have algorithms for imposing non-negativity in compressed spaces.

EQUALITY CONSTRAINTS

When working with, for example, closed data it may be helpful to incorporate the closure directly in the model. If the loading matrix **A** contains concentrations that are subject to closure, then it holds that each row should sum to one (or some other constant value). Such a constraint may be incorporated specifically by the use of an equality constraint. Equality constraints are defined similarly to inequality constraints as the problem of estimating a matrix (or vector) **A** subject to

$$\mathbf{PA} = \mathbf{Q}, \quad (182)$$

or

$$\mathbf{PA}^T = \mathbf{Q}. \quad (183)$$

Several different implementations of equality constrained regression can be seen in Lawson & Hanson (1974). For a practical example see the FIA application page 207.

LINEAR CONSTRAINTS

Linear constraints means that the estimated loadings are required to be linearly related to some space, that is, **A** must obey

$$\mathbf{A} = \mathbf{PG}, \quad (184)$$

where **A** is the loading matrix to be estimated and **P** is a predefined matrix. Such a linear constraint can be helpful if a model is sought where the solution is to be within the variable space of, e.g., an experimental design. It may also be helpful in compressing arrays prior to estimating a model (page 169). As noted earlier linear constraints are solved by the use of the CANDELINC model (Carroll et al. 1980).

SYMMETRY

Manne (1995) has pointed out the important fact, that many constraints do not guarantee uniqueness. One relatively soft constraint that is, however, likely to bring about uniqueness is symmetry. If, for example, chromatographic profiles are known to be symmetric around their maximum mode, then requiring symmetry might be enough to ensure uniqueness. No special-made algorithm has yet been devised for symmetry. If unimodality is required additionally to symmetry the problem may be solved by an exhaustive search for the best mode location using either monotone regression, quadratic programming (Luenberger 1973), or dynamic programming (Sidiropoulos & Bro 1998). If no unimodality is required one may simply do an exhaustive search for the mode of symmetry using equality constraints. This will not be pursued here.

MONOTONICITY

The amount of substrate in a kinetic experiment will typically decrease monotonically in time. If a model is fitted where, e.g., a specific loading vector is an estimate of the amount of substrate, it is possible to constrain the parameters to obey the monotonicity. An algorithm for estimating such a problem is given on page 175.

UNIMODALITY

In curve resolution, e.g., chromatography, it is quite common to work with data types where the underlying phenomena generating the data can be assumed to be unimodal. The development of least squares unimodal regression is important as currently either ad hoc or very restrictive methods are used for enforcing unimodality in curve resolution (Karjalainen & Karjalainen 1991, Gemperline 1986, Knorr et al. 1981, Frans et al. 1985). One approach often used in iterative algorithms is to simply change elements corresponding to local maxima on an estimated curve so that the local maxima disappear. Clearly such a method does not have any least squares (or other well-defined) property. The restrictive methods typically enforce the profiles to be Gaussians, but there is seldom provision for assuming that, e.g., chromatographic peaks are even approximately Gaussian. Least squares estimation under unimodality constraints seems

to be more appropriate than the overly restricted Gaussian approach and more well-defined and well-behaved than simply changing parameters without considering the accompanying changes in the loss function.

On page 177 an algorithm for fitting least squares models under unimodality constraints is presented¹². In Bro & Sidiropoulos (1998) it is discussed how to extend the algorithm for unimodal least squares regression to oligomodality using dynamic programming, but this will not be pursued here.

SMOOTHNESS

Incorporating smoothness as a constraint is tempting in many situations. Here a method will be described for obtaining smoothness in case the deviations from smoothness are primarily caused by random noise.

A common approach for obtaining smoothness of, e.g., a time-series signal is to use a local smoother like a zeroth-order Savitzky-Golay, a median, or a moving average filter. Alternatively, wavelets or Fourier transforms have been employed for removing low-frequency variation. All these methods, however, suffer from a major problem. They do not optimize any well-defined criterion. Even though it is possible to envision a set-up where smoothness is imposed in an ALS algorithm, this is likely to lead to an ill-defined algorithm yielding poor results. Additionally, most of these methods are based on local smoothing and do not always produce results that look smooth.

Consider the least squares estimation of a vector, α , possibly a column of a loading matrix. An estimate of α is sought under the constraint of being smooth. A reasonable way to define a smooth estimate, \mathbf{a} , is that the change in the first derivative from point to point is little. The discrete estimate of the second derivative for an element a_j of \mathbf{a} is

$$(a_{j-1} - a_j) - (a_j - a_{j+1}), \quad (185)$$

¹². Frisen (1986) and Geng & Shi (1990) have also developed a similar algorithm for a related problem. Note, however, that the algorithm suggested in Frisen (1986) for unimodal regression with fixed mode location is erroneous and the overall algorithms suggested in both papers are based on exhaustive searches, that are much more time-consuming than the one given here.

assuming here that the elements of \mathbf{a} are equidistantly spaced. Consider the $J-2$ vector of the discrete second derivatives of each element of the J vector \mathbf{a} excluding the end points

$$\mathbf{e} = \begin{bmatrix} (a_1 - a_2) - (a_2 - a_3) \\ (a_2 - a_3) - (a_3 - a_4) \\ \vdots \\ (a_{J-2} - a_{J-1}) - (a_{J-1} - a_J) \end{bmatrix} = \begin{bmatrix} a_1 - 2a_2 + a_3 \\ a_2 - 2a_3 + a_4 \\ \vdots \\ a_{J-2} - 2a_{J-1} + a_J \end{bmatrix}. \quad (186)$$

The vector \mathbf{e} can be found as \mathbf{Pa} where \mathbf{P} ($J-2 \times J$) is the second difference operator defined as below

$$\mathbf{e} = \begin{bmatrix} 1 & -2 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & & 0 & 0 & 0 \\ \vdots & & & \ddots & & & & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 & 1 \end{bmatrix} \mathbf{a} = \mathbf{Pa}. \quad (187)$$

An efficient way to impose smoothness is to require the norm of \mathbf{e} to be below a certain value. Thus, estimating \mathbf{a} subject to $\|\mathbf{e}\| < \rho$ is a reasonable and well-defined way of obtaining smoothness. It is also possible to pay different weights to different parts of the data by weighting the elements of \mathbf{e} differentially. The use of the norm of \mathbf{e} for obtaining smoothness has been described in several different applications e.g. in Hastie & Tibshirani (1990)¹³.

¹³. The work reported here on smoothness forms the basis for a collaboration with N. Sidiropoulos on developing an algorithm for smooth unimodal regression.

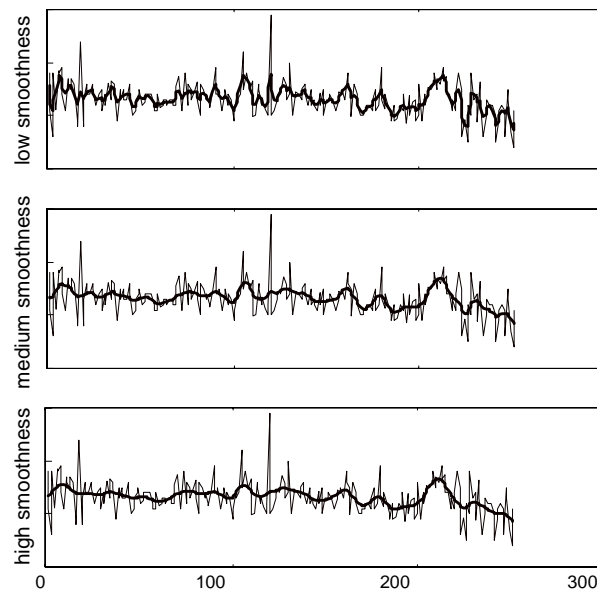


Figure 15. Arbitrary noisy time series and smoothed version (thick lines) shown for increasingly imposed smoothness.

The penalty approach for obtaining smoothness can be used in many situations. Consider a noisy time series (Figure 15). Before using such a time series in an analysis it may be fruitful to smooth it in order to eliminate the high frequency variation.

In Figure 15 a time series has been estimated subject to increasing degrees of smoothness. As evidenced by the figures using the smoothness constraint accomplishes exactly what is perceived as increasing smoothness. Another example is shown in Figure 16 for a simulated series consisting of a Gaussian with noise added. Again, imposing smoothness using the second differences operator works exactly as expected.

An algorithm for smooth regression is given on page 181.

ORTHOGONALITY

When curve resolution or visual appearance is not an issue a decomposition may be stabilized by constraining the loading vectors in certain modes to be orthogonal. This will also enable variance partitioning of the otherwise correlated components (Kiers & Krijnen 1991, Heiser & Kroonenberg 1997) as well as help avoiding problems with degeneracy (Lundy et al. 1989). The problem of estimating an orthogonal least squares loading matrix can be

expressed as finding

$$\min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 \quad (188)$$

subject to $\mathbf{A}^T\mathbf{A} = \mathbf{I}$.

The solution to this problem has been devised by Cliff (1966) and is simply

$$\mathbf{A} = \mathbf{Y}^T\mathbf{Z}(\mathbf{Z}^T\mathbf{Y}\mathbf{Y}^T\mathbf{Z})^{-1/2}. \quad (189)$$

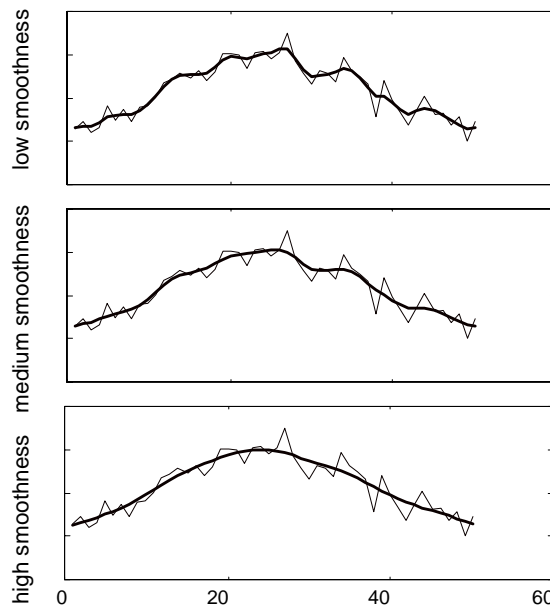


Figure 16. Smoothing of a noisy Gaussian. Smoothed estimate shown with thick lines for increasing smoothness.

A model fitted under orthogonality constraints will differ from an unconstrained model. However, if the unconstrained model is unique, then so is the orthogonality constrained model. Uniqueness, though, does not imply that true underlying phenomena in a curve resolution sense are found. This is only the case if the posed model including the constraints is a valid description of the data. This will seldom be the case if orthogonality is imposed.

The fact that orthogonality constrained models are faster to estimate in

some cases, is interesting and can be used for speeding up slow algorithms as well as help avoiding problems with local minima and swamps. This can be done by imposing approximate orthogonality in one mode such that orthogonality is only imposed significantly during the first iterations and not at all during later iterations. This can be achieved by using a penalty function for deviations from the orthogonal solution (see Box 21 page 215).

FUNCTIONAL CONSTRAINTS

Consider a situation where, e.g., the loading vectors of one mode should ideally follow an exponential decay. In such a case this can be enforced by estimating each loading vector subject to being an exponential. Restricting loading vectors to be of a specific functional shape should be done with care. Unlike for example non-negativity and unimodality such estimation of the parameters subject to functional constraints is based on just a few degrees of freedom. Therefore, the results must be interpreted with caution, since they may be misleading by the beauty of the solution regardless of appropriateness.

QUALITATIVE DATA

It is common for all methods discussed thus far, that they are metric, i.e. they only work properly for ratio- or interval-scale data. When, e.g., qualitative data in some sense are used the methods will not be optimal. To cope with this, the strategy of alternating between least squares and optimal scaling is useful. Any least squares method can be accommodated to any type of qualitative data by the use of optimal scaling. The principle is simple and elegant. The least squares metric model is fitted to the raw data, and then the model of the data is transformed to conform to the characteristics of the data type. Subsequently the transformed model is used as input to the least squares model, and this procedure is repeated until convergence.

The optimal scaling depends on the nature of data measured. There are three characteristics needed for defining the nature of the data (Stevens 1946). The measurement *level*, the measurement *process*, and the measurement *conditionality*. The measurement level describes the order and 'discreteness' of the measurements and can be either nominal, ordinal,

or numerical. The measurement process describes the nature of the process generating the data and can be either discrete or continuous. The measurement conditionality describes how different measurements relate to each other.

Nominal data are data where only discrete values can be obtained and where there is no intrinsic order of the different levels. An example could be nationality. If the measurement process is discrete the data are nominal-discrete which means that the process generating the process is discrete. Nationality is nominal-discrete as a person is either Dutch or German. There is not a continuous underlying process generating the nationality. On the other hand suppose that the terms red, blond, and black hair are used as variable settings. This will give rise to nominal data as there is no order of hair color, but the underlying process generating the hair color can be regarded as continuous and hence hair color is nominal-continuous.

For ordinal data there is an additional order among the different levels of a variable, an example being income classes. Again the data can be generated by either a discrete or a continuous process.

Numerical data, are the data often used or at least perceived in chemometrics. Such data may be either interval or ratio scale depending on whether or not a natural zero exists. Such data may also be generated by either a discrete or a continuous process.

Finally the conditionality of the data has to be defined. The conditionality tells which data elements are related and which are not. For example, though income class and color may both be considered to be ordinal data, there is no intrinsic order *between* the two types of data; hence the ordinality only refers to different income classes and different colors separately.

Optimal scaling is an appealing and apparently sound technique. Any model appropriate for continuous data can be used for qualitative or mixed data by the use of optimal scaling. A large number of publications including the Gifi book (1990) illustrate optimal scaling for a variety of problems (Takane et al. 1977, Sands & Young 1980, Young 1981). In chemometrics the experience with optimal scaling is limited. Some researchers have expressed some concern with respect to how well optimal scaling works in practice. Harshman & Lundy (1984a) conjecture that in practice the

difference between the results obtained from optimal scaled data and the original data will be insignificant, while H. A. L. Kiers (1989, page 51-53) and the author of this thesis have experienced that optimal scaling can result in severe overfitting, because the data are modified continuously to follow a pre-specified model. What remains, however, is that more experience with optimal scaling in chemometrics is needed.

6.3 ALTERNATING LEAST SQUARES REVISITED

Most algorithms for fitting multi-way models are based on ordinary least squares regression or at least the algorithms can be constructed based on regression. The basic problem to solve in most ALS algorithms is that of minimizing

$$\|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 \quad (190)$$

over \mathbf{A} . The matrix \mathbf{A} can be a loading matrix or a core array. Sometimes this regression problem can be solved directly, but for certain constraints it is fruitful first to modify it. It will be explained in the following how to estimate \mathbf{A} either directly, row-wise or column-wise.

GLOBAL FORMULATION

In different sub steps of most ALS algorithms (chapter 4) the fundamental problem being solved can be expressed

Problem **GLOBAL**: Given \mathbf{Y} ($I \times M$) and \mathbf{Z} ($I \times J$) find a matrix \mathbf{A} that minimizes

$$\min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2. \quad (191)$$

Unconstrained least squares solution:

$$\mathbf{A} = \mathbf{Y}^T(\mathbf{Z}^+)^T$$

The matrices \mathbf{Y} and \mathbf{Z} depend on the problem being solved and \mathbf{A} is the set

of parameters to be estimated (loading matrix or core array). For PARAFAC solving for the first mode loadings \mathbf{Y} would equal $\mathbf{X}^{(JK \times I)}$ and \mathbf{Z} would equal $\mathbf{C} \otimes \mathbf{B}$. For Tucker3 \mathbf{Z} would equal $\mathbf{C} \otimes \mathbf{B}$. Note that the above multivariate regression problem can also be expressed as a multiple linear regression problem. Let

$$\mathbf{t} = \text{vec} \mathbf{Y}, \quad (192)$$

of size $IM \times 1$ and

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Z} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Z} & & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{Z} \end{bmatrix} \quad (193)$$

of size $IM \times JM$. Then the above problem **GLOBAL** can be equivalently expressed

$$\min_{\mathbf{a}} \|\mathbf{t} - \mathbf{Qa}\|_F^2 \quad (194)$$

where

$$\mathbf{a} = \text{vec}(\mathbf{A}^T). \quad (195)$$

Expressing the problem as a multiple regression problem can be helpful for incorporating certain constraints.

ROW-WISE FORMULATION

It is important for the following discussion to note, that \mathbf{A} may also be estimated column- or row-wise. To solve problem **GLOBAL** row-wise consider the formulation

$$\begin{aligned} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 &= \\ \left\| \begin{bmatrix} \mathbf{y}_1 - \mathbf{Z}\mathbf{A}_{(1,:)}^T & \mathbf{y}_2 - \mathbf{Z}\mathbf{A}_{(2,:)}^T & \cdots & \mathbf{y}_J - \mathbf{Z}\mathbf{A}_{(J,:)}^T \end{bmatrix} \right\|_F^2 &= \\ \|\mathbf{y}_1 - \mathbf{Z}\mathbf{A}_{(1,:)}^T\|_F^2 + \|\mathbf{y}_2 - \mathbf{Z}\mathbf{A}_{(2,:)}^T\|_F^2 + \cdots + \|\mathbf{y}_J - \mathbf{Z}\mathbf{A}_{(J,:)}^T\|_F^2 . \end{aligned} \quad (196)$$

Hence the overall problem can be solved by solving for each row of \mathbf{A} separately unless some constraints are imposed over rows. Letting the column vector \mathbf{a} be a transposed row of \mathbf{A} and \mathbf{y} the corresponding column of \mathbf{Y} , it therefore holds that to find the optimal \mathbf{a} conditionally on the remainder of \mathbf{A} it is only necessary to consider the problem

Problem ROWWISE: Given \mathbf{y} ($l \times 1$) and \mathbf{Z} ($l \times J$) find a vector \mathbf{a} that minimizes

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Z}\mathbf{a}\|_F^2 \quad (197)$$

Unconstrained least squares solution:

$$\mathbf{a} = \mathbf{Z}^+ \mathbf{y} \quad (198)$$

The vector \mathbf{a} is the i th transposed row of \mathbf{A} and \mathbf{y} is the i th column of \mathbf{Y} of problem **GLOBAL**. Cycling over all i s will provide the overall solution to problem **GLOBAL**.

COLUMN-WISE FORMULATION

Instead of estimating \mathbf{A} row-wise it may be feasible to consider a column-wise update instead. Thus consider

$$\|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 = \|\mathbf{Y} - (\mathbf{z}_1\mathbf{a}_1^T + \mathbf{z}_2\mathbf{a}_2^T + \cdots + \mathbf{z}_M\mathbf{a}_M^T)\|_F^2 . \quad (199)$$

Unlike for the row-wise estimation no separation of the loss function is possible into contributions from each column. If an algorithm is sought for column-wise estimation then the estimation has to be done conditionally on the remaining columns of \mathbf{A} . Define a new matrix \mathbf{T} as

$$\mathbf{T} = \mathbf{Y} - \mathbf{Z}_{-f}\mathbf{A}_{-f}^T, \quad (200)$$

where \mathbf{Z}_{-f} is the matrix obtained by ignoring the f th column of \mathbf{Z} , and \mathbf{A}_{-f} being defined likewise by ignoring the f th column of \mathbf{A} . Then

$$\|\mathbf{Y} - (\mathbf{Z}_{-f}\mathbf{A}_{-f}^T + \mathbf{z}_f\mathbf{a}_f^T)\|_F^2 = \|\mathbf{T} - \mathbf{z}_f\mathbf{a}_f^T\|_F^2 \quad (201)$$

and hence the problem of estimating the f th column of \mathbf{A} can be expressed

Problem COLUMNWISE: Given \mathbf{T} ($l \times M$) = $\mathbf{Y} - \mathbf{Z}_{-f}\mathbf{A}_{-f}^T$, and \mathbf{z}_f ($l \times 1$) find a vector \mathbf{a}_f that minimizes

$$\min_{\mathbf{a}_f} \|\mathbf{T} - \mathbf{z}_f\mathbf{a}_f^T\|_F^2 \quad (202)$$

Unconstrained least squares solution:

$$\mathbf{a}_f = \mathbf{T}^T\mathbf{z}_f(\mathbf{z}_f^T\mathbf{z}_f)^{-1} \quad (203)$$

In the sequel problem **COLUMNWISE** will be shortly stated as

$$\min_{\mathbf{a}} \|\mathbf{T} - \mathbf{z}\mathbf{a}^T\|_F^2. \quad (204)$$

Note an important difference between problem **COLUMNWISE** and the former problems. Updating the columns of \mathbf{A} one at a time will *not* provide the overall least squares solution to problem **GLOBAL** because the parameters of each column depend on the setting of the remaining parameters in \mathbf{A} .

As for the global and row-wise problems the columnwise problem can

also be expressed as a multiple regression problem as

$$\min_{\mathbf{a}} \|\mathbf{q} - \mathbf{R}\mathbf{a}\|_F^2 \quad (205)$$

where

$$\mathbf{q} = \text{vec}\mathbf{T} \quad (206)$$

and

$$\mathbf{R} = \begin{bmatrix} \mathbf{z} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{z} & & \mathbf{0} \\ \vdots & & \ddots & \\ \mathbf{0} & \mathbf{0} & & \mathbf{z} \end{bmatrix}. \quad (207)$$

A very important aspect of the column-wise problem is that the contribution of each parameter to the loss function is independent of the remaining parameters. This means that the optimal value of each parameter can be determined separately. The implication of this will be elaborated on after the following lemma.

Lemma 1: Consider the optimization problem

$$\min_{\mathbf{a}} \|\mathbf{T} - \mathbf{z}\mathbf{a}\|_F^2 \quad (208)$$

subject to \mathbf{a} is constrained (e.g., non-negative or unimodal). The solution to this problem is

$$\min_{\mathbf{a}} \|\boldsymbol{\alpha} - \mathbf{a}\|_F^2 \quad (209)$$

subject to the same constraints. Above α is the solution to the unconstrained problem, i.e., $\alpha = \frac{\mathbf{T}^T \mathbf{z}}{\mathbf{z}^T \mathbf{z}}$

Proof

Let

$$\alpha = \frac{\mathbf{T}^T \mathbf{z}}{\mathbf{z}^T \mathbf{z}} \quad (210)$$

and

$$\mathbf{E} = \mathbf{T} - \mathbf{z}\alpha^T.$$

Then

$$\min_{\mathbf{a}} \|\mathbf{T} - \mathbf{z}\mathbf{a}^T\|_F^2 =$$

$$\min_{\mathbf{a}} \|\mathbf{z}\alpha^T + \mathbf{E} - \mathbf{z}\mathbf{a}^T\|_F^2,$$

yielding

$$\min_{\mathbf{a}} \left[\text{tr}(\mathbf{E}^T \mathbf{E}) + 2\text{tr}(\mathbf{E}^T \mathbf{z}(\alpha - \mathbf{a})^T) + \text{tr}((\alpha - \mathbf{a})\mathbf{z}^T \mathbf{z}(\alpha - \mathbf{a})^T) \right]. \quad (211)$$

Since $\text{tr}(\mathbf{E}^T \mathbf{E})$ is constant, $\mathbf{E}^T \mathbf{z}$ is a vector of zeros, and $\mathbf{z}^T \mathbf{z}$ is constant it follows that

$$\text{argmin}_{\mathbf{a}} \|\mathbf{T} - \mathbf{z}\mathbf{a}^T\|_F^2 =$$

$$\text{argmin}_{\mathbf{a}} \|\alpha - \mathbf{a}\|_F^2. \quad (212)$$

subject to the given constraints. ■

The above lemma provides a dramatic simplification of many otherwise complicated constrained problems (see Bro & Sidiropoulos 1998). Consider, for example, a non-negativity constraint. Normally active set algorithms (page 169) or an iterative penalized algorithm is needed, but for the column-wise algorithm it is possible to simply estimate the unconstrained solution and then set any negative values to zero. It follows directly from Lemma 1 that this will be the least squares solution to the problem¹⁴.

The drawback of column-wise estimation is that the overall algorithm tends to get slower and less robust than when using problem **ROWWISE** or **GLOBAL**. This is due to the fact that for the column-wise estimation the parameters are estimated conditionally on *all* remaining parameters, while for the row-wise estimation the parameters are estimated conditionally only on the parameters in other modes (given in the matrix **Z**). As finding the optimal setting of each row is independent of remaining rows in the same mode, the optimal setting of a given row does not depend on the remaining rows, i.e., the solution to problem **ROWWISE**. Hence

$$\operatorname{argmin}_{\mathbf{A}_{(i,:)}} \|\mathbf{y}_i - \mathbf{Z}\mathbf{A}_{(i,:)}^T\|_F^2 = \left[\mathbf{A}_{(i,:)} \mid \min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 \right] \quad (213)$$

while for the solution to problem **COLUMNWISE** it holds that

$$\operatorname{argmin}_{\mathbf{a}_f} \|\mathbf{T} - \mathbf{z}_f\mathbf{a}_f^T\|_F^2 \neq \left[\mathbf{a}_f \mid \min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2 \right]. \quad (214)$$

¹⁴. Heiser & Kroonenberg (1997) also suggested the use of a column-wise update in the PARAFAC algorithm identical to the approach described here. They call it a triadic update to signify that if the three-way PARAFAC model is updated column-wise by first estimating the first loading vector in every mode, then the second etc., then the algorithm updates triads.

ALS ALGORITHMS IN PRACTICE**BOX 15**

Consider a simple algorithm for fitting an F -component bilinear model to an $I \times J$ matrix \mathbf{X} , the model being $\mathbf{X} = \mathbf{A}\mathbf{B}^T$. First initialize \mathbf{B} .

1. $\mathbf{A} = \mathbf{X}(\mathbf{B}^T)^+$
2. $\mathbf{B} = \mathbf{X}^T(\mathbf{A}^T)^+$
3. Go to step one until convergence

The above algorithm uses problem **GLOBAL** in both the update of \mathbf{A} and \mathbf{B} . Suppose that the first mode loading \mathbf{A} is to be updated row-wise. An algorithm for fitting the model could then look as follows

1. For every i
 1. $\mathbf{A}_{(i,:)} = \mathbf{X}_{(i,:)}(\mathbf{B}^T)^+$
 2. $\mathbf{B} = \mathbf{X}^T(\mathbf{A}^T)^+$
 3. Go to step one until convergence

Finally assume that a column-wise update is sought. This can be achieved by the following algorithm

1. For every f
 1. $\mathbf{a}_f = (\mathbf{X} - \mathbf{A}_{-f}\mathbf{B}_{-f}^T)\mathbf{b}_f(\mathbf{b}_f^T\mathbf{b}_f)^{-1}$
 2. $\mathbf{B} = \mathbf{X}^T(\mathbf{A}^T)^+$
 3. Go to step one until convergence

In step one, one may alternatively iterate over the columns, e.g., until convergence in order to obtain the overall least squares solution.

Some constraints are best implemented using problem **GLOBAL**, others problem **ROWWISE**, and others again problem **COLUMNWISE**. These three problems will thus be the starting point for the algorithms described in the following. In Box 15 it is shown how to implement the different problems in an ALS algorithm.

To summarize the typical problem in ALS algorithms is problem

GLOBAL

$$\min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{Z}\mathbf{A}^T\|_F^2. \quad (215)$$

This problem is easily solved in its unconstrained version using ordinary least squares regression. Problem **GLOBAL** can be partitioned into smaller subproblems either estimating the rows or the columns of **A**. The problem **ROWWISE** which is

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Z}\mathbf{a}\|_F^2 \quad (216)$$

or problem **COLUMNWISE** which is

$$\min_{\mathbf{a}} \|\mathbf{T} - \mathbf{z}\mathbf{a}^T\|_F^2 = \min_{\mathbf{a}} \|\boldsymbol{\alpha} - \mathbf{a}\|_F^2. \quad (217)$$

The attractive feature of problem **ROWWISE** is that it also solves problem **GLOBAL** if estimated over all rows unless some constraint over rows is imposed. For problem **COLUMNWISE** on the other hand it holds that the elements of the column can be determined independently of each other, and hence many types of constrained problems are easily solved. Finally it has been shown that all three problems can be posed as a standard multiple regression problem

$$\min_{\mathbf{a}} \|\mathbf{t} - \mathbf{Q}\mathbf{a}\|_F^2$$

which means that the same types of algorithms can be used for all three problems. Depending on the size of the array and the type of model, special algorithms taking advantage of the sparsity of **Q** may be adequate.

6.4 ALGORITHMS

In the following algorithms for some of the discussed constrained problems

will be given. Speed of the algorithms is of major concern as the algorithms are typically solved many times during the iterative overall ALS-algorithm. It will be shown that for the non-negativity and unimodality constrained problems, the algorithms suggested here are among the fastest currently available.

FIXED COEFFICIENTS CONSTRAINED REGRESSION

Consider a multiple regression problem subject to some elements are predefined. Let the index fx indicate the indices of fixed parameters and nf indices of parameters that are not fixed. Let the vector \mathbf{g} hold the predefined parameters, i.e., $\mathbf{a}_{fx} = \mathbf{g}$. Then

Problem Fixed Parameter Constrained Regression

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Z}\mathbf{a}\|_F^2$$

$$\text{subject to: } \mathbf{a}_{fx} = \mathbf{g}$$

defines the fixed parameter constrained regression problem. As

$$\mathbf{y} - \mathbf{Z}\mathbf{a} = \mathbf{y} - \mathbf{Z}_{fx}\mathbf{a}_{fx} - \mathbf{Z}_{nf}\mathbf{a}_{nf} = \mathbf{r} - \mathbf{Z}_{nf}\mathbf{a}_{nf} \quad (218)$$

where $\mathbf{r} = \mathbf{y} - \mathbf{Z}_{fx}\mathbf{a}_{fx}$, the solution is simply

$$\operatorname{argmin}_{\mathbf{a}_{nf}} \|\mathbf{r} - \mathbf{Z}_{nf}\mathbf{a}_{nf}\|_F^2 = \mathbf{Z}_{nf}^+ \mathbf{r} . \quad (219)$$

The choice of which problem to use for implementing the above algorithm depends on the patterns of the fixed parameters. If whole rows of the matrix to estimate are fixed, then simply ignoring these rows is possible, hence giving a smaller problem which can, e.g., be solved using problem **GLOBAL**. If whole columns are fixed (for example because certain spectra are known), it is more wise to use problem **COLUMNWISE** for estimating the parameters. Other situations may also occur.

A related problem arises if the least squares solution is sought not under the constraint that the parameters are fixed but under the constraint that the parameters should not deviate too much from a set of target values. The problem is only defined when the term 'not deviate too much' is quantified. One way of doing this is by the use of a penalty approach. Instead of the above loss function the deviation of the coefficients from the target is incorporated into the loss function. Define the matrix \mathbf{K} is a binary diagonal matrix, with the element k_{ff} equal to one if the f th element of \mathbf{a} is fixed and otherwise zero. Some rows will thus only contain zeros and may be eliminated. Then

$$\begin{aligned} \min_{\mathbf{a}} \left[\|\mathbf{y} - \mathbf{Za}\|_F^2 + \lambda^2 \|\mathbf{g} - \mathbf{a}_{\text{fx}}\|_F^2 \right] = \\ \min_{\mathbf{a}} \left[\|\mathbf{y} - \mathbf{Za}\|_F^2 + \lambda^2 \|\mathbf{g} - \mathbf{Ka}\|_F^2 \right] = \\ \min_{\mathbf{a}} \|\mathbf{s} - \mathbf{Ga}\|_F^2 \end{aligned} \quad (220)$$

where

$$\mathbf{s} = \begin{bmatrix} \lambda \mathbf{g} \\ \mathbf{y} \end{bmatrix} \quad (221)$$

and

$$\mathbf{G} = \begin{bmatrix} \lambda \mathbf{K} \\ \mathbf{Z} \end{bmatrix}. \quad (222)$$

The solution is thus

$$\mathbf{a} = \mathbf{G}^+ \mathbf{s}. \quad (223)$$

The weight λ defines how much the constrained coefficients should resemble the target values. As λ approaches infinity the solution approaches the solution with exact equality and as λ approaches zero the solution approaches the unconstrained solution. Some care has to be taken in avoiding numerical problems if very large values of λ are used (Haskell & Hanson 1981). Especially solving the regression problem using the normal equations can be problematic (Björck 1996, p. 192) and methods based on, e.g., QR-decompositions should be preferred. However, when using the penalty approach, exact fulfillment of the constraint is seldom sought and hence large values of λ are not wanted. Choosing λ is mostly based on either subjective assessment of solutions using different values or quantitative assessment using cross-validation or similar.

NON-NEGATIVITY CONSTRAINED REGRESSION

The time required for fitting true least squares non-negativity constrained models is typically many times longer than for fitting unconstrained models. In Bro & de Jong (1997) a modification of the current standard algorithm is proposed, that cuts execution time considerably. The original non-negativity constrained least squares regression (NNLS) algorithm of Lawson & Hanson (1974) will be described first and then it is shown how to accommodate the NNLS algorithm to an ALS algorithm in a sensible way.

The NNLS problem is equivalent to, e.g., problem **ROWWISE** with non-negativity constraints:

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Za}\|_F^2 \quad (224)$$

subject to $a_m \geq 0$, for all m ,

where a_m is the m th element of \mathbf{a} .

One may also use problem **GLOBAL** or **COLUMNWISE** instead of problem **ROWWISE**. For problem **COLUMNWISE** an easy solution admits itself by the virtue of Lemma 1. It follows directly from Lemma 1 that to solve for a column vector \mathbf{a} of \mathbf{A} it is only necessary to solve the unconstrained problem and subsequently set negative values to zero. Though the

algorithm for imposing non-negativity in problem **COLUMNWISE** is thus simple and may be advantageous in some situations, it is not pursued here. Since problem **COLUMNWISE** optimizes a smaller subset of parameters than the other approaches it may be unstable in difficult situations.

ALGORITHM NNLS**BOX 16***A. Initialization*A1. $P = \emptyset, R = \{1, 2, \dots, M\}$ A2. $\mathbf{a} = \mathbf{0}, \mathbf{w} = \mathbf{Z}^T(\mathbf{y} - \mathbf{Z}\mathbf{a})$ *B. Main loop*B1. While $R \neq \emptyset \wedge [\max_{n \in R}(w_n) > \text{tolerance}]$ B2. $m = \underset{n \in R}{\operatorname{argmax}}(w_n),$ B3. Include the index m in P and remove it from R B4. $\mathbf{s}^P = ((\mathbf{Z}^P)^T \mathbf{Z}^P)^{-1} (\mathbf{Z}^P)^T \mathbf{y}$ *C. Inner loop*C1. While $\min(\mathbf{s}^P) < 0$ C2. $\mathbf{s} := \mathbf{a} + \alpha(\mathbf{s} - \mathbf{a}), \alpha = - \min_{n \in P} \frac{a_n}{a_n - s_n}$ C3. Update R and P C4. $\mathbf{s}^P = ((\mathbf{Z}^P)^T \mathbf{Z}^P)^{-1} (\mathbf{Z}^P)^T \mathbf{y}, \mathbf{s}^R = \mathbf{0}$

end C

B5. $\mathbf{a} = \mathbf{s}, \mathbf{w} = \mathbf{Z}^T(\mathbf{y} - \mathbf{Z}\mathbf{a})$

end B

The solution to the non-negativity constrained problem **ROWWISE** can be found by the algorithm NNLS which is an *active set algorithm*. There are M

inequality constraints in the above stated problem. The m th constraint is said to be active if the m th regression coefficient will be negative if estimated unconstrained, otherwise the constraint is passive. An active set algorithm is based on the fact that if the true active set is known the solution to the least squares problem will simply be the unconstrained least squares solution to the problem using only the variables corresponding to the passive set, setting the regression coefficients of the active set to zero. Or stated more generally: if the active set is known, the solution to the NNLS problem is obtained by treating the active constraints as equality constraints, rather than inequality constraints. To find the true active set an ALS algorithm is applied. An initial feasible set of regression coefficients is found. A feasible vector is a vector ($M \times 1$) with no elements violating the constraints. In this case the vector containing only zeros is a feasible starting vector as it contains no negative values. In each step of the algorithm variables are identified and removed from the active set in such a way that the fit strictly improves. After a finite number of iterations the true active set is found and the solution is found by simple linear regression on the unconstrained subset of the variables. The algorithm is given in Box 16.

The set P comprises all indices of the M variables which are currently not fixed: the passive set. Likewise, R holds indices of those coefficients that are currently fixed at the value zero: the active set. To ensure that the initial solution vector, \mathbf{a} , is feasible, it is set equal to the $M \times 1$ zero vector (step A2), and all constraints are active (step A1). The vector \mathbf{w} defined in step A2 can be interpreted as a set of Lagrange multipliers or (half) the negative partial derivative of the loss function. When the optimal solution has been found the following holds:

$$w_m = 0, \quad m \in P$$

$$w_m < 0, \quad m \in R, \tag{225}$$

since the partial derivative of an unconstrained parameter is zero per definition and since the partial derivative of a constrained parameter must be negative. Otherwise by letting the regression coefficient be unconstrained a positive value would result yielding a lower loss function value.

If the set R is empty all constraints are passive and all coefficients must be positive and thus the problem is solved. If R is not empty, but all w_m , $m \in R$ are negative, no fixed variable can be set free as the regression coefficient of that variable would then have to be negative to lower the error sum of squares. If any w_m , $m \in R$ is positive, transferring the corresponding variable to the set of free variables will yield a new positive regression coefficient. The variable with the highest w_m is included in the set P (step B3) and the intermediate regression vector, \mathbf{s} , is calculated using this new set (step B4). The superscript P indicates the passive set. The matrix \mathbf{Z}^P is a matrix containing only the variables currently in the passive set P . In practice the elements of \mathbf{w} are not tested to be positive but to be above a certain low tolerance to ensure that numerical inaccuracies do not cause a non-feasible variable to enter the passive set. If all regression coefficients in \mathbf{s} are non-negative the regression vector \mathbf{a} is set equal to \mathbf{s} (step B5) and a new, \mathbf{w} , is calculated (step B5). The main loop is continued until no more active constraints can be set free.

When a new variable has been included in the passive set P there is a chance that in the unconstrained solution to the new least squares problem (step B4) some of the regression coefficients of the free passive variables will turn negative (step C1). Calling the new estimate \mathbf{s} and the former \mathbf{a} it is possible to adjust the new estimate to satisfy the constraints. The old estimate \mathbf{a} is feasible but with a worse fit (higher loss) than the new estimate \mathbf{s} which, however, is not feasible. Somewhere along the line segment $\mathbf{a} + \alpha(\mathbf{s} - \mathbf{a})$, $0 \leq \alpha \leq 1$, there is an interval for which the inequalities are not violated. As the fit is strictly decreasing as $\alpha \rightarrow 1$, it is possible to find that particular value of α which minimizes the loss function yet retains as many variables in the passive set as possible (step C2). For the optimal α one or several variables will become zero and hence they are removed from the set P (step C3). After adjusting the active and passive sets the unconstrained solution using the current passive set is calculated (step C4). The regression coefficients of variables removed from the set P are set to zero (step C4). The inner loop is repeated until all violating variables have been moved to the set R . In practice the NNLS algorithm seldom enters the inner loop C .

It can be proved that the algorithm will consist of only a finite number of

iterations as both the inner and outer loop converge after finite iterations. Further comments and variations of this algorithm can be found in the literature (Stoer 1971, Schittkowski & Stoer 1979, Haskell & Hanson 1981, Hanson 1986).

In most cases where the NNLS algorithm is used the number of observations is higher than the number of variables. It is customary to use a QR-decomposition to reduce the problem. In the context of, e.g., PARAFAC another type of reduction seems more appropriate. Most ALS algorithms are based on solving a problem of type $\min\|\mathbf{y} - \mathbf{Z}\mathbf{a}\|$ but in efficient implementations only $\mathbf{Z}^T\mathbf{y}$ and $\mathbf{Z}^T\mathbf{Z}$ are computed, not \mathbf{Z} directly (cf. page 61). This, however, turns out to be an advantage, as these cross-product matrices are smaller in size than the original data and the NNLS algorithm can be modified to handle the cross-product matrices instead of the raw data.

To modify the NNLS algorithm so that it accepts the cross-products as input instead of the raw data, the following changes in algorithm NNLS are required: Steps A2 and B5 are changed to

$$\mathbf{w} = (\mathbf{Z}^T\mathbf{y}) - (\mathbf{Z}^T\mathbf{Z})\mathbf{a}, \quad (226)$$

and steps B4 and C4 are changed to

$$\mathbf{s}^P = ((\mathbf{Z}^T\mathbf{Z})^P)^{-1}(\mathbf{Z}^T\mathbf{y})^P, \quad (227)$$

where $(\mathbf{Z}^T\mathbf{Z})^P$ is the submatrix of $\mathbf{Z}^T\mathbf{Z}$ containing only the rows and columns corresponding to the set P . Likewise, the column vector $(\mathbf{Z}^T\mathbf{y})^P$ contains the elements of $\mathbf{Z}^T\mathbf{y}$ corresponding to the set P . It is easily verified that the vector \mathbf{s}^P defined this way equals \mathbf{s}^P as defined in algorithm NNLS. When \mathbf{Z} has many more rows than columns using the cross-products instead of the raw data improves the speed dramatically.

To further decrease the time spent in the algorithm it is worth noting that during the estimation of loadings in, e.g., the PARAFAC algorithm, typically only small changes occur from iteration to iteration. Elements that are forced to zero will usually stay zero in subsequent iterations. Instead of starting each estimation with all variables forced to zero (step A2) it is

therefore sensible to define R and P according to their optimal values in the previous iteration of that particular set of loadings. To do this the algorithm must be changed so that the feasibility of these predefined sets of active and passive variables is guaranteed before entering the main loop. This is accomplished by calculating the constrained regression vector corresponding to the current passive set P and then entering a loop similar to the inner loop in NNLS before entering the main loop. If the sets R and P initially given are correct this algorithm will converge fast because after estimation of the regression vector only a check for negative regression coefficients and a check for positive w -values has to be performed.

The above algorithm is called Fast NNLS (FNNLS) to distinguish it from the ordinary NNLS algorithm. For completeness some modifications of FNNLS will be discussed. It is possible to do weighted non-negativity constrained linear least squares regression by simply using $\mathbf{Z}^T \mathbf{V} \mathbf{Z}$ and $\mathbf{Z}^T \mathbf{V} \mathbf{y}$ instead of $\mathbf{Z}^T \mathbf{Z}$ and $\mathbf{Z}^T \mathbf{y}$. The diagonal matrix \mathbf{V} contains the weights. It is also possible to only require non-negativity of individual regression coefficients by simply keeping unconstrained regression coefficients passive for good and ignoring their value when testing for negative regression coefficients (step C1). It is possible to use FNNLS for imposing approximate non-negativity. One approach sometimes seen in the literature is to use the bounded problem

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{Z}\mathbf{a}\|_F^2 \quad (228)$$

$$a_m \geq -\rho, \text{ for all } m,$$

where ρ is a small positive number. This way of solving the problem, however, seems not to fulfill its goal, as instead of an approximate solution an exact solution is obtained, only to a slightly shifted problem. Instead approximate non-negativity may be obtained, e.g., by penalizing deviations from the non-negativity by an approach similar to the one described on page 179.

In Bro & de Jong (1997) the algorithm FNNLS is tested on both real and simulated data and it is shown, that compared to using the algorithm of

Lawson & Hanson a 5- to 20-fold gain in time is obtained. Also interesting is that for some models the overall algorithm using FNNLS is actually faster than a corresponding algorithm for a completely unconstrained problem. This can be explained by the fact that if many variables are forced to zero smaller regression problems are to be solved in the non-negativity constrained algorithm than in the unconstrained. Hence, the more elements are forced to zero the faster the algorithm will be in an ALS setting.

MONOTONE REGRESSION

In monotone regression the problem is that of minimizing

$$\min_{\mathbf{a}} \|\boldsymbol{\alpha} - \mathbf{a}\|_F^2 \quad (229)$$

subject to \mathbf{a} is monotone increasing, i.e.

$$a_j \geq a_{j-1}, j = 2, \dots, \text{size}(\mathbf{a}).$$

Conversely the problem may be transformed into one where \mathbf{a} is constrained to be monotonically decreasing. The problem is mostly stated as above, but through the use of Lemma 1 (page 162), it is seen that solving this problem solves problem **COLUMNWISE** subject to the loading vectors being monotone.

Consider a $J \times 1$ vector $\boldsymbol{\alpha}$ with typical element α_j . A vector \mathbf{a} is sought that minimizes the sum-squared difference between $\boldsymbol{\alpha}$ and \mathbf{a} , subject to the requirement that \mathbf{a} is monotone increasing. Only the situation where all elements of \mathbf{a} are free to attain any value whatsoever will be considered. This is a situation with no ties according to Kruskal (64). Consider two consecutive elements α_j and α_{j+1} . Suppose $\alpha_{j+1} > \alpha_j$ then what should the values of \mathbf{a} be to give the best monotone estimate of $\boldsymbol{\alpha}$? If no other elements are violating the constraints implied by the monotonicity then all elements except the j th and the $j+1$ th should equal $\boldsymbol{\alpha}$ as this will naturally lead to a zero-contribution to the sum-squared error. It further holds that the elements a_j and a_{j+1} should be set to the mean of α_j and α_{j+1} (for simplicity assuming the mean is higher than α_{j-1} and lower than α_{j+2}). From the

geometry of the problem it follows that any other set of values will produce a higher sum-squared error. This observation is the cornerstone of monotone regression (Kruskal 1964, Barlow et al. 1972, de Leeuw 1977).

EXAMPLE ON MONOTONE REGRESSION

BOX 17

Consider the vector

$$\boldsymbol{\alpha} = [3 \ 1 \ 6 \ 2 \ 5]^T. \quad (230)$$

A vector \mathbf{a} is sought which is a monotone increasing regression on $\boldsymbol{\alpha}$. First let \mathbf{a} equal $\boldsymbol{\alpha}$ and let the first element of \mathbf{a} be active. This element is down-satisfied per definition, but not up-satisfied as $3 > 1$. The average of these two elements is computed and the interim solution is now

$$\mathbf{a} = [2 \ 2 \ 6 \ 2 \ 5]^T. \quad (231)$$

Now the first block is constituted by the first *and* the second element. This block is down-satisfied and up-satisfied, so the next block is now considered (the third element). This block is down-satisfied but not up-satisfied ($6 > 2$), hence the two corresponding elements are averaged yielding

$$\mathbf{a} = [2 \ 2 \ 4 \ 4 \ 5]^T. \quad (232)$$

Now the block is both up- and down-satisfied. As there is only one final block (the fifth element) and this is down-satisfied and per definition up-satisfied, the monotone regression is computed.

Define a *block* as a set of consecutive elements of \mathbf{a} all having assigned the same value. Initially let \mathbf{a} equal $\boldsymbol{\alpha}$ and let every element of \mathbf{a} be a block. Let the first leftmost block be *active*. If the common value of elements of an active block is higher than or equal to the common value of the block to the left the block is *down-satisfied*; otherwise concatenate the two blocks into

one block whose elements have common value equal to the mean of elements of the two blocks. If the new common value of the block is lower or equal to the value of the block to the right the block is *up-satisfied*; otherwise the two blocks are averaged. Continue checking up- and downwards until the resulting block is both up- and down-satisfied, and then continue to the next block, i.e., the next block becomes active. When the last block is both up- and down-satisfied \mathbf{a} will hold the solution to the monotone increasing least squares regression problem (Box 17). This result is due to Kruskal (1964). Note that, by convention, the last block is automatically up-satisfied and the first block automatically down-satisfied. Monotone decreasing regression can be performed in a similar fashion.

UNIMODAL LEAST SQUARES REGRESSION

The unimodal least squares regression (**ULSR**) problem is defined as the solution to problem **COLUMNWISE** subject to \mathbf{a} is unimodal, which is equivalent to

$$\min_{\mathbf{a}} \|\boldsymbol{\alpha} - \mathbf{a}\|_{\mathbb{F}}^2 \quad (230)$$

subject to \mathbf{a} is unimodal.

The proof of this follows from the proof of Lemma 1. For a non-negativity constrained problem the unimodality constraint can be expressed as follows:

$$\begin{aligned} a_1 &\geq 0, \\ a_j &\geq a_{j-1}, j = 2, \dots, n; \\ a_J &\geq 0 \\ a_j &\leq a_{j-1}, j = n+1, \dots, J = \text{size}(\mathbf{a}) \end{aligned} \quad (231)$$

for some mode location, n , which is *itself* subject to optimization. In the sequel **ULSR** means non-negative **ULSR**. An algorithm for solving this problem is based on the following. Suppose a monotone increasing regression is performed on $\boldsymbol{\alpha}$. While calculating this regression vector the

monotone increasing regressions are obtained for $\alpha_{(1:j)}$, $j = 1, \dots, J$, $\alpha_{(1:j)}$ being a vector containing the first j elements of α . This can be derived as a side-benefit of Kruskal's monotone regression algorithm. Similarly a monotone decreasing regression for α will produce all monotone decreasing regressions for $\alpha_{(1:j)}$. The algorithm now proceeds as follows:

1. Calculate \mathbf{a}^I as the monotone increasing regression on α and \mathbf{a}^D as the monotone increasing regression on $\text{rev}(\alpha)$, where $\text{rev}(\alpha)$ is the vector obtained by reversing the order of the elements of α . Let $\mathbf{a}^{I,n}$ be the monotone increasing regression on the first $n-1$ elements of α , and $\mathbf{a}^{D,n}$ be the monotone decreasing regression on the last $J - n$ elements of α .

2. For all n , $n = 1, \dots, J$ define $\mathbf{c}^{(n)} \equiv \begin{bmatrix} \mathbf{a}^{I,n} \\ \alpha_n \\ \mathbf{a}^{D,n} \end{bmatrix}$.

3. Then $\mathbf{a} = \underset{\mathbf{c}^{(n)}}{\text{argmin}} \left(\|\alpha - \mathbf{c}^{(n)}\|_F^2 \mid \max(\mathbf{c}^{(n)}) = \alpha_n \right)$.

That is, among only those $\mathbf{c}^{(n)}$ satisfying $\alpha_n = \max(\mathbf{c}^{(n)})$, select a vector $\mathbf{c}^{(n)}$ which minimizes $\|\alpha - \mathbf{c}^{(n)}\|$. This algorithm will provide a solution to problem **ULSR** as proven in Bro & Sidiropoulos (1998). To make the above a little more transparent see the example in figure 17. In Bro & Sidiropoulos (1998) it is shown that the complexity of the algorithm only corresponds to two full-size monotone regressions.

One aspect not yet covered is how to implement non-negativity, but it follows immediately from Lemma 1 and the proof of Kruskal's monotone regression that one can simply set all negative values in the regression vector to zero. This will automatically lead to the optimal solution under non-negativity constraints.

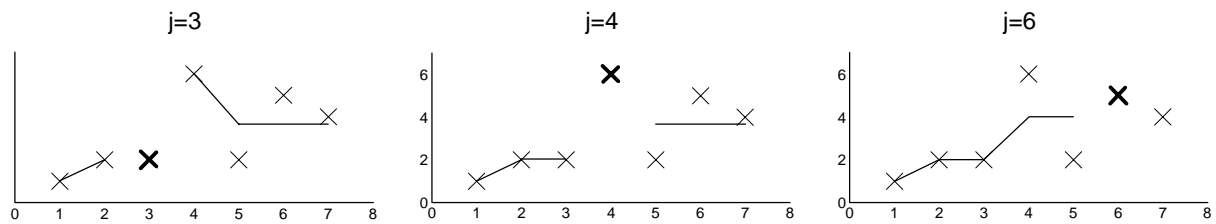


Figure 17. Hypothetical 7×1 input vector α shown with crosses. Three hypothesized mode locations are shown: 3, 4, and 6, though all seven possible modes have to be considered in reality. The lines shown are the monotone increasing and decreasing regressions to the left and right of the mode locations. Only those modes for which the cross of the mode (marked fat) is not below any of the lines are possible candidates (4 and 6). The one of these with the best fit (mode = 4) is the solution vector.

In certain cases more complicated constraints than unimodality and non-negativity is desired. In Bro & Sidiropoulos (1998) and Sidiropoulos & Bro (1998) weighted ULSR, robust unimodal and oligomodal regression is discussed. Suppose a target is given that the vector being estimated should resemble or possibly be equal to. Such a situation can occur if for example the spectra of some analytes are known beforehand, but it can also occur in situations where the vectors being estimated are subject to equality constraints (page 207). A simplified example occurs if the matrix to be estimated, \mathbf{A} , consists of two unimodal column vectors that should be equal. This can be expressed algebraically as the following equality constraint:

$$\mathbf{CA} = \mathbf{d}, \quad (232)$$

where

$$\mathbf{C} = [1 \ -1] \quad (233)$$

and

$$\mathbf{d} = [0]. \quad (234)$$

General methods have been developed for solving linear problems under equality constraints, but for more complicated equality constraints no closed-form solution is available due to the unimodality constraints. Following is a generic example, which could be solved simpler by other approaches, but illustrates how to proceed in general.

For a column of \mathbf{A} , say the first, \mathbf{a} , being estimated the equality constraint can be expressed in terms of the current estimate of the other column vector. This vector will be called \mathbf{g} as it is the goal that \mathbf{a} is similar to \mathbf{g} . For the equality and unimodality constrained problem, one may consider the following soft constraint formulation:

$$\min_{\mathbf{a}} \left[\|\mathbf{Y} - \mathbf{z}\mathbf{a}^T\|_F^2 + \lambda \|\mathbf{g} - \mathbf{a}\|_F^2 \right] \quad (235)$$

subject to \mathbf{a} being unimodal. Note that here \mathbf{g} is not the desired goal of the total problem, though in other situations \mathbf{g} may be known in advance. The parameter λ controls the penalty levied for deviation from \mathbf{g} . A low value of λ means that the solution \mathbf{a} may deviate considerably from \mathbf{g} . A large value of λ would mean that the constraint should be essentially exactly fulfilled. For a given value of λ , the solution of the above hybrid problem may be obtained as follows (below $\boldsymbol{\alpha}$ is the unconstrained solution):

$$\min \left[\|\mathbf{Y} - \mathbf{z}\mathbf{a}^T\|_F^2 + \lambda \|\mathbf{g} - \mathbf{a}\|_F^2 \right] =$$

$$\min \left[\|\boldsymbol{\alpha} - \mathbf{a}\|_F^2 + \lambda \|\mathbf{g} - \mathbf{a}\|_F^2 \right] =$$

$$\min \{ \frac{1}{2} \mathbf{a}^T \mathbf{a} - \boldsymbol{\alpha}^T \mathbf{a} + \frac{1}{2} \lambda \mathbf{a}^T \mathbf{a} - \lambda \mathbf{g}^T \mathbf{a} \} =$$

$$\min \left[\frac{\mathbf{a}^T \mathbf{a}}{2} - \left(\frac{\boldsymbol{\alpha}^T + \lambda \mathbf{g}^T}{1 + \lambda} \right) \mathbf{a} \right]_F^2 =$$

$$\min \left[\left\| \frac{\boldsymbol{\alpha} + \lambda \mathbf{g}}{1 + \lambda} - \mathbf{a} \right\|_{\text{F}}^2 \right] =$$

$$\min \|\mathbf{p} - \mathbf{a}\|_{\text{F}}^2, \quad (236)$$

where

$$\mathbf{p} = \frac{\boldsymbol{\alpha} + \lambda \mathbf{g}}{1 + \lambda}. \quad (237)$$

As can be seen from the above the same algorithm can be used for solving the equality constrained problem as for solving problem **ULSR**, by simply exchanging $\boldsymbol{\alpha}$ with \mathbf{p} . Using this approach it is also possible to impose approximate unimodality by exchanging \mathbf{g} with the least squares unimodal solution and calculating the unconstrained solution to equation 236 which is equal to \mathbf{p} .

SMOOTHNESS CONSTRAINED REGRESSION

The implementation of smoothness as described generically page 152 depends on whether the smoothness is imposed on the columns or the rows of the loading matrix. Assuming that it is the columns of the loading matrix \mathbf{A} that are to be estimated subject to being smooth then through the use of Lemma 1 the following model holds

$$\min_{\mathbf{a}} \left[\|\boldsymbol{\alpha} - \mathbf{a}\|_{\text{F}}^2 + \lambda \|\mathbf{P}\mathbf{a}\|_{\text{F}}^2 \right] \quad (238)$$

where $\boldsymbol{\alpha}$ is the unconstrained estimate of the loading vector \mathbf{a} and \mathbf{P} is defined as in equation 187. The scalar λ defines the influence of the penalty term on the overall loss function. For a specific λ the norm of $\|\mathbf{P}\mathbf{a}\|_{\text{F}}^2$ is exactly ρ . Thus by adjusting λ in each update one may impose smoothness in the sense that $\|\mathbf{P}\mathbf{a}\|_{\text{F}}^2 \leq \rho$. Alternatively, one may consider the

penalized expression as the loss function, in which case λ is fixed, while ρ will vary. It is important in the context of an ALS algorithm that the norm of \mathbf{a} be fixed algorithmically. Due to the scaling indeterminacy of multilinear models it is possible to scale down any loading vector by scaling up a loading vector of the same component in another mode. Hence, if the norm of \mathbf{a} is not fixed, then it is possible to make the norm of \mathbf{a} so small that the influence of the last term above has no significant influence on the loss function. Thus, the choice of λ and the norm of \mathbf{a} are closely related and this must be respected algorithmically.

The solution to the above problem is simply

$$\mathbf{a} = (\mathbf{I} + \lambda \mathbf{P}^T \mathbf{P})^{-1} \boldsymbol{\alpha}, \quad (239)$$

where \mathbf{I} is an identity matrix.

A semi-simulated example will be given here indicating that smoothness can be helpful in obtaining meaningful results. The fluorescence data described on page 115 and 142 consist of data from five samples with different amounts of three different amino acids. Here the data have been reduced in the excitation and emission modes giving an array of size 5 (sample) \times 41 (emission) \times 31 (excitation). In order to make smoothing necessary a very large amount of noise was added: normally distributed heteroscedastic noise of magnitude twice the magnitude of the signal. A three-component model is adequate for these data, but the unconstrained model did not succeed very well. In the upper part of Figure 18 the parameters of the unconstrained model are shown. Compared to a model of the data without noise (the lower figures) the differences are significant. In the emission mode it seems that the model has difficulties in distinguishing between two components. This is evidenced by the loading vector shown with a thicker line looks like the derivative of one of the other vectors. As a consequence the estimated relative concentrations are far from good. If the excitation and emission mode loadings are estimated under smoothness constraints (middle row in Figure 18) the model is forced to find loadings deviating from the very unsmooth loadings of the unconstrained model. Comparing the smoothness constrained model with the model of the data without noise added (Figure 18 lowest) the models are

quite close even though the smoothness constrained model is fitted to much noisier data.

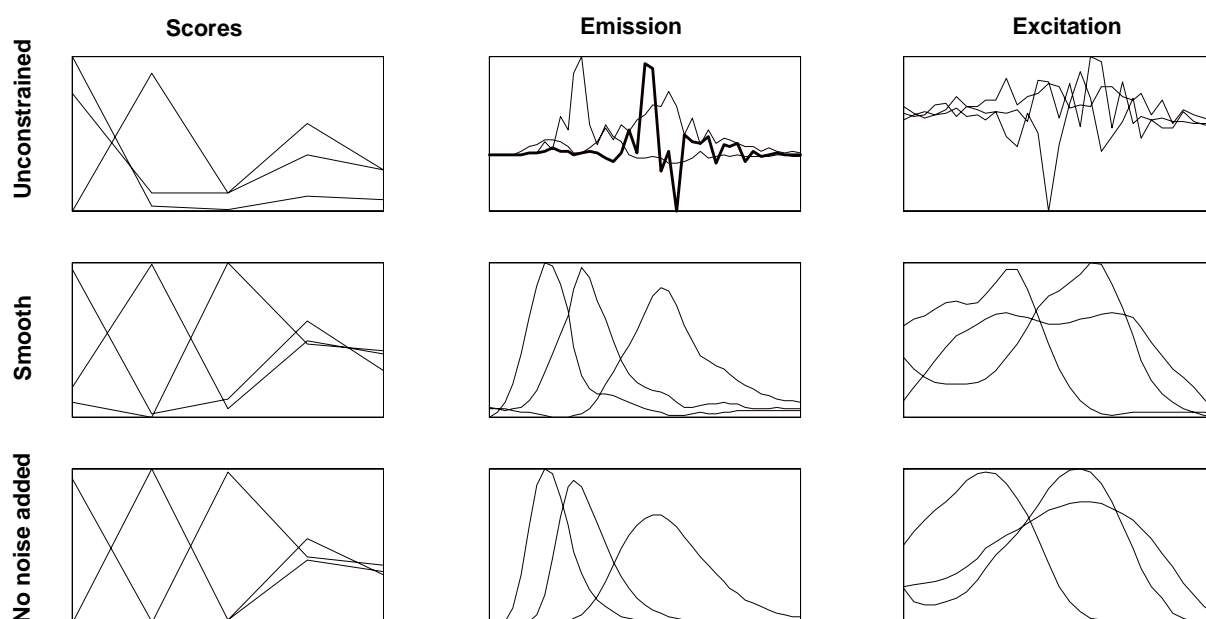


Figure 18. Three three-component PARAFAC models. The top figures show the loadings of an unconstrained model of the data added noise. The 'derivative loading' mentioned in the text is shown with a thicker line. Below the same model is shown with smoothness of the excitation and emission mode loadings imposed. The lower figures show the model of the data with no noise added

It is interesting to see how well the three models estimate the concentrations. Below the correlations between the known reference concentrations and the estimated concentrations are given. It is evident that the smoothness constraint helps the model distinguishing between the components.

Table 1. Correlations between analyte concentrations and PARAFAC scores.

Model	Tryptophan	Tyrosine	Phenylalanine
Unconstrained	.9782	.6512	.5509
Smoothness	.9972	.9871	.9941
No noise	.9998	.9999	.9982

6.5 SUMMARY

The use of constraints has been discussed on several levels. The purposes for using constraints have been discussed. Constraints have been characterized according to the extent to which they are imposed (exact or approximate). The incorporation of constraints has been discussed with respect to ALS algorithms. It has been shown that different approaches of updating lead to different advantages. For the column-wise update it has been shown that constraints often turn out to be simple to implement because the parameters of the problem are independent. A number of constraints have been discussed that may be appropriate in different situations. For some of these constraints algorithms have also been devised.

In the next chapter it will be shown in practice, that using constraints can help in obtaining uniqueness, improving parameter estimates, making models more easy to interpret etc.

Two important characteristics of the use of constraints has to be stressed. Using selectivity uniqueness can often be obtained, and as non-negativity can provide selectivity implicitly, it may also provide uniqueness. Generally though, constraints seldom provide uniqueness. Another important aspect of constraints is that they have to be used with caution. Care must be taken not to overrestrict the model and assessment of the validity of a constrained model is very important.

CHAPTER 7

APPLICATIONS

7.1 INTRODUCTION

In this final chapter several applications will be shown illustrating many of the aspects covered in the prior chapters. The focus will be on the modeling aspects while details concerning the experimental conditions can mostly be found elsewhere. The types of problems described vary from simple laboratory data to complex process data. It is the hope, that the diversity of problems described will enable people to generalize also to other types of data and problems than those actually shown. A review of multi-way analysis applications in the literature especially in spectral analysis is given in Bro et al. (1997). The following examples will be discussed.

7.2 BREAD

Noisy data with no *a priori* known structure exemplified with sensory data. Comparing PARAFAC, consensus-PCA, and PCA and comparing unfold PLS with three-way PLS. ■

7.3 AMINO-N

Prediction of amino-N from fluorescence of sugar samples. Comparing unfold and multi-way calibration models for precise data. ■

7.4 FIA

Flow injection analysis data with structure as a restricted PARATUCK2 model due to rank-deficiency. Shows the benefit of using constraints (non-negativity, unimodality, fixed parameters, and equality). ■

7.5 SUGAR

In this example sugar samples are collected directly from a sugar plant and measured spectrofluorometrically. By the use of non-negativity and unimodality constraints it is possible to develop a PARAFAC model, that seems to reflect chemically meaningful analytes. The thereby obtained estimated concentrations of chemical species are shown to give valid information for predicting the sugar quality with or without incorporating the automatically sampled process data. ■

7.6 ENZYMATIC ACTIVITY

An experimental design was used for exploring the influence of different factors on the activity of polyphenol oxidase, an enzyme responsible for enzymatic coloring of vegetables and fruits. The data thus obtained can be arranged as a five-way array. It is shown that, in this particular case, the PARAFAC model offers a fruitful alternative to traditional analysis of variance models, and gives a model that is simple to interpret. ■

7.7 RETENTION TIME SHIFTS

A chromatographic data set with retention time shifts is shown to be difficult to model using PARAFAC, while very good results are obtained using the more appropriate PARAFAC2 model. ■

The purpose of modeling can for example be exploratory analysis, calibration, curve resolution, or analysis of variance. Before describing the applications these four basic types of analysis will be shortly described.

EXPLORATORY ANALYSIS

In exploratory analysis the main purpose of the analysis is to learn from the data about interrelationships between variables and objects. Exploratory analysis is not a model; it is a process, where each model made gives new insight and knowledge of what to do. The process of exploratory analysis can be understood by the selection cycle described in detail by Munck (1991) and Munck et al. (1998). Initially data are generated according to a hypothesis based on a general idea, e.g., that fluorescence data may be useful to obtain an overall view on the chemistry of a sugar process. The results and information arising from the data give rise to a new set of ideas, which lead either to generation of new data or different analyses of the current data. This process corresponds to one revolution of the selection cycle. By repeated revolutions generating new hypotheses, new data and new models, the knowledge of the problem increases and the analysis may become more focused.

However, inductive exploratory analysis hardly stands alone because of interpretation problems. It constitutes a special language which works most efficiently in a dialogue with the present scientific language; the deductive confirmatory analysis based on *a priori* hypotheses.

The selection cycle can be compared to how a food consumer selects food. By selection of manifest food qualities the consumer influences hidden latent factors such as food taste and long term health. Just as the scientist, the consumer has difficulties in overviewing the result of the complex selection process. Exploratory analysis is essential because it acknowledges both the manifest and the latent components and thus provides means for overviewing and interpreting complicated problems.

It is extremely important to use exploratory tools that enable detailed facets of the data to be revealed. This touches upon the cognitive aspects of modeling: the structure and nature of the data, the properties of the model used, the experience and skills of the analyst etc. Data are typically quite complicated reflecting a complex reality, while the perception of the

analyst is limited by the poverty of the human brain. The model and the (graphical) presentation of the model is the window into the complex data. It is thus essential, that the model used is flexible enough, not to discard important aspects, yet simple enough to be interpretable. These often contradicting requirements can be difficult to fulfill in practice. Some general guidelines can be given though. The most basic feature of a model is that it should reflect and approximate the nature and structure of the data. While this may not be possible to fulfill exactly, the alternative is that the model is of a different structure than the data. This will complicate understanding as the analyst must then distinguish in the analysis between results pertaining to the nature of the data and results pertaining to the nature of the model.

The latent variable models are excellent exploratory tools as they provide condensed models of variable and sample spaces for graphical inspection as well as model residuals for diagnostic purposes. The validity and appropriateness of latent variable models is of course dependent on the nature of the data. For spectral data the appropriateness is almost self-evident, while for other types of data the latent variable models can at most be regarded as good feature extractors.

For most multi-way data it holds, that not using the structure of the data in the modeling will provide models that are less robust and more difficult to interpret, because the nature of the data and the model differ. Which multi-way model should then be used for a given multi-way data set? This has already been discussed on page 107. In practice several competing models will often be investigated, because a comparison of results can provide important information on the nature of the data. When working with spectral data the PARAFAC model will often be a good starting point, because the intrinsic approximate linearity of spectral data is close to the PARAFAC model. For other data types the Tucker3 model may be a more appropriate starting point for an exploratory analysis. Using unfolding techniques can also be helpful, but valuable information may be obscured in the combined modes. In any case, when using a model for exploratory purposes it is important to scrutinize the model for indications of model mis-specifications. If a Tucker3 model is used it is therefore sensible to test if the PARAFAC model is as good a candidate model and vice versa.

FLUORESCENCE EXCITATION-EMISSION**BOX 18**

For one fluorophore the emission intensity at a specific wavelength, j , when excited with light at a wavelength k , can be described:

$$x_{jk} = ab_j c_k \quad (240)$$

where x_{jk} is the intensity of the light emitted at emission wavelength j and excitation wavelength k , a is proportional to the concentration of the analyte, b_j is the relative emission-intensity (quantum yield) at wavelength j , and c_k is the extinction coefficient at excitation wavelength k . This relation holds approximately for diluted solutions (Ewing 1985), and it further holds that b_j is independent of c_k . If several, F , fluorophores contribute for the intensity can be written

$$x_{jk} = \sum_{f=1}^F x_{jkf} = \sum_{f=1}^F a_f b_{jf} c_{kf} \quad (241)$$

implying that the contribution to the emission from each analyte is independent of the contributions of the remaining analytes. In the above equation a_f is the concentration of analyte f . For several samples, and a_{if} being the concentration of the f th analyte in the i th sample, the model becomes

$$x_{ijk} = \sum_{f=1}^F a_{if} b_{jf} c_{kf} \quad (242)$$

which is the three-way PARAFAC model.

The underlying important aspect is to use a model that can give interesting auxiliary information, that can lead to better experiments, models, hypotheses etc. As opposed to for example many types of neural networks, the possibility of getting auxiliary information on how and why the model is

obtained represents a major advantage of latent variable models. In later stages of an analysis, when implementing, e.g., a prediction model in a process, more dedicated models may be used, because then it is known, that the samples and variables are important and valid.

CURVE RESOLUTION

Curve resolution deals with the problem of identifying the parameters of the structural model governing a data set. It relies on the hypothesis that besides a noise contribution the data are generated according to a (structurally known) process expressed mathematically by an algebraic model. Curve resolution is for example a tool for determining the spectra of pure analytes measured in mixtures and finds applications mainly in chromatography but also kinetics, fluorescence spectroscopy, etc.

The list of curve resolution techniques is long: iterative target transformation factor analysis, heuristic evolving latent projection (Liang & Kvalheim 1993), alternating regression (Karjalainen 1989, Karjalainen & Karjalainen 1991), SIMPLISMA (Windig 1994), self modeling curve resolution (Tauler & Barceló 1993), window factor analysis (Brereton et al. 1995), evolving factor analysis (Maeder & Zilian 1988) etc. Most of these techniques are primarily applicable in two-way analysis, and focus on finding selective channels in the data (variables that reflect only one analyte). Establishing selectivity is *the* most important way of obtaining (partial) uniqueness. The uniqueness obtainable from selectivity has been discussed on page 143. Characteristic of many curve resolution techniques is that the fitted models often have no well-defined loss function. Typically the most selective channels are used to peel of components one at a time.

For multi-way data the possible structural uniqueness coupled with other external constraints is a powerful add-on to the current techniques for curve resolution. Using techniques described in this thesis together with traditional curve resolution tools for assessing selectivity it is possible to formulate a curve resolution problem as *one* global problem stated as a structural model with constraints. This is likely to be stabilizing in situations where the traditional algorithms fail to give results.

Leurgans & Ross (1992), Leurgans et al. (1993), Ross & Leurgans (1995), and Nørgaard (1995a) describe in detail the rationale behind using

multilinear models for resolving fluorescence data (see Box 18).

CALIBRATION

Calibration is an important area of application in multivariate analysis (Martens & Næs 1989). For multi-way analysis, the main type of multi-way calibration in the literature has been the so-called second-order calibration also mentioned in Box 14 page 142. In analytical chemistry simple calibration problems are common for which a model is sought for predicting the concentration of a chemical species. If the analyte gives a simple (rank one mostly) contribution to the measured signal, second-order calibration is possible.

However, most problems can not be solved with such an approach, because the dependent variable is not merely a causal factor for one particular spectral phenomenon. And even though calibration is possible using the second-order advantage, this is likely to be suboptimal in many situations. If it is possible to have access to samples similar to the ones to predict it is most likely better to use all sample-mode variation for predicting the sought parameters. This can be done in a PCR/PLS like fashion using linear regression approaches or it can be done using more flexible methods.

One important application for calibration models is for developing *screening analyses*. In calibration the typical setup is to exchange an expensive, unethical, or otherwise problematic analysis with more easily obtained data. Using the calibration model it is possible to predict the sought variable more easily than else making it possible to screen a large amount of objects. This provides a powerful tool for optimizing, understanding, deducing etc. This is the case in, e.g., process analysis where it is common to measure important parameters infrequently because they have to be determined in the laboratory. If these parameters can be determined on-line or at-line by the use of, e.g., NIR or fluorescence measurements and a calibration model, instant on-line determinations of the important parameters are available. A similar situation arises in the medical industry where it is expensive and time-consuming to test the biological activity of new drugs. Predicting the activity from, e.g., physical or quantum-mechanical properties, it is possible to test a larger amount of chemicals also

reducing the use of test animals.

ANALYSIS OF VARIANCE

The use of multiplicative models for analysis of variance (ANOVA) is little known. However, as the responses of a factorial design with more than one factor give rise to data that have the structure of an array it is only natural to suppose that a decomposition model may in some cases be more appropriate for modeling the data than a traditional ANOVA model. Already Fisher (1923) proposed the use of a PCA-like method for obtaining more interpretable models of two-factor experiments, and several other authors have later proposed similar approaches (Gollob 1968, Mandel 1969 & 1971, Hegemann & Johnson 1976, Yochmowitz 1983). Kettenring (1983) has suggested the use of three-way PARAFAC for decomposing three-factor experiments. Here, a general multiplicative model is suggested partly in line with the above references, extending them to a higher level of generality. The model described was first described in Bro (1997) but also Heiser & Kroonenberg (1997) have described a similar method.

It is often stated that higher-order interactions are hard to interpret in ANOVA, and the reason is simple. Given a set of experimental data where two factors are varied on I and J levels respectively, the responses of one variable can be represented by an $I \times J$ matrix, the ij th element, y_{ij} , representing the response when the first factor is on the i th level and the second factor is on the j th level. If no interaction is present the standard ANOVA model for qualitative factors (Montgomery 1991) is

$$y_{ij} = \mu + a_i + b_j + e_{ij}. \quad (243)$$

Here μ is the grand level, a_i is the effect of the first factor at level i , b_j the effect of the second factor at level j and e_{ij} the residual. This model of the data is a simplification. Instead of IJ elements in the original data array, only $1+I+J$ terms of which $I+J-1$ are independent has to be understood and interpreted. If interaction between the two factors is present the model is

$$y_{ij} = \mu + a_i + b_j + (ab)_{ij} + e_{ij}. \quad (244)$$

This model consists of $1+I+J+IJ$ parameters, of which IJ are independent. Clearly, no reduction in the complexity of the representation has been achieved, and therefore the interaction is hard to interpret. For third- and higher-order interactions the problem is even worse.

Another model is normally used for experiments with quantitative or continuous factors. For quantitative factors a linear effect of the factor settings over all levels is estimated. Mathematically the corresponding model underlying this approach is

$$y_{ij} = b_0 + b_1x_{1i} + b_2x_{2j} + b_{12}x_{1i}x_{2j} + e_{ij} \quad (245)$$

Here x_{1i} refer to the value of the first factor at the i th level etc. These values are fixed as they are determined by the experimental design, hence only the parameters b have to be estimated. For the first main effect only the scalar b_1 must be estimated and so on.

For models involving both quantitative and qualitative factors the two models are easily merged. The qualitative ANOVA model is quite flexible, and thus quite prone to overfit, especially for interactions. The quantitative model on the other hand is very restricted in its model formulation. The drawback of both models is that, even though the models can theoretically handle interactions of any order and complexity, data that are mainly generated by interactions can be difficult to model adequately. The model proposed here can be seen as a complement of intermediate complexity.

The GEneral Multiplicative ANOVA (GEMANOVA) model is defined for a two-factor experiment as

$$y_{ij} = \mu + a_{i1} + b_{j1} + \sum_{f=1}^F c_{if}d_{jf} + e_{ij} \quad (246)$$

For a three-factor experiment a full model would contain main effects, two-way interactions as well as three-way interactions. As for the standard ANOVA model only significant terms should be included in the model. It may be noted that the model contains the qualitative ANOVA model as a limiting case. If for example the following ANOVA model is found for a two-factor experiment

$$y_{ij} = (ab)_{ij} \quad (247)$$

the exact same model can be fitted by a full rank bilinear model as

$$y_{ij} = \sum_{f=1}^F a_{if} b_{jf} \quad (248)$$

where F equals the rank of the matrix \mathbf{Y} , with typical elements y_{ij} . The GEMANOVA model also contains more specialized models as the shifted multiplicative model (Cornelius et al. 1992, van Eeuwijk 1996) and thus theoretically gives a unifying algorithm for fitting these models. An important distinction between the GEMANOVA model and other suggested multiplicative models is that the GEMANOVA model is a least squares model. This cannot in general be guaranteed for earlier proposed multiplicative ANOVA models as evidenced in Kruskal (1977b). Any GEMANOVA model can be fitted with a PARAFAC algorithm with certain elements fixed to ones. This points to the other significant advantages of the GEMANOVA model. It generalizes to any number of factors, and due to its close relation to the PARAFAC model many GEMANOVA models are identified by the structural model alone.

Consider a three-factor experiment with response y_{ijk} for factor one at level x_i , factor two at level x_j , and factor three at level x_k . A main effect for factor one will be modeled as

a_i	ANOVA	qualitative
bx_i	ANOVA	quantitative
a_i	GEMANOVA	

A three-way interaction will be modeled as

$(abc)_{ijk}$	ANOVA	qualitative
$bx_i x_j x_k$	ANOVA	quantitative
$a_i b_j c_k$	GEMANOVA	one-component
$\sum_{f=1}^F a_{if} b_{jf} c_{kf}$	GEMANOVA	F-component

The GEMANOVA model obviously has stronger similarity to the qualitative ANOVA model than the quantitative model being parametrically equivalent for main effects. However, for interactions, the GEMANOVA model fills a gap between the very flexible qualitative model – $(abc)_{ijk}$ using a total of IJK parameters – and the very restricted quantitative model – $bx_i x_j x_k$ using one parameter. A one-component three-way GEMANOVA effect is $a_i b_j c_k$ using $I+J+K$ parameters. When used to model experiments with qualitative factors the possible advantage is similar to the gain in insight and robustness obtained by using PCA for exploring two-way data. The qualitative ANOVA interaction terms can often be expected to overfit as no structure is imposed at all between levels and factors. For quantitative experiments the GEMANOVA model may also be advantageous especially if many experiments are performed. The GEMANOVA model is intrinsically more flexible than the quantitative ANOVA model, i.e., compare the main effect for both models. If the variation in response is primarily caused by simple main effects of quantitative factors the quantitative ANOVA model is most likely to provide adequate answers, but if several interactions or cross-products are present, it is quite possible that they can be better modeled by a multiplicative model.

How then, to choose between the different possible modeling strategies? The most sensible way to start is to use the standard ANOVA model. *If* there are indications that the model is mainly governed by interactions or complicated cross-products or *a priori* knowledge suggests a multiplicative model would be more appropriate, it is possible that the GEMANOVA model can be a fruitful alternative.

Several possibilities exist for validating a GEMANOVA model, i.e., determining the complexity of the model. F-like tests, resampling techniqu-

es like bootstrapping and cross-validation, or technological and methodological insight may be used. The details of these approaches will not be touched upon here. The only important point that should be made, is that degrees of freedom are not available for estimating variances in PARAFAC/GEMANOVA models. To circumvent this problem approximate estimated degrees of freedom can be obtained from Monte Carlo studies as described for a similar problem by Mandel (1969 & 1971).

ASPECTS OF GEMANOVA

BOX 19

Fractional factorial designs. Fractional designs correspond to full factorial designs with missing elements and are easily handled as such.

Multiple responses. Throughout it has been assumed that there is only one response variable, but the model also holds for multiple responses. If the responses are not correlated it is of course possible to analyze the responses separately, but it is likely that the model can be stabilized by decomposing several responses simultaneously. This can be accomplished by introducing an extra mode called the response mode with its m th level being the m th response. Appropriate scaling may be necessary for handling differences in scale of different responses.

Heteroscedasticity. If data are heteroscedastic this may be dealt with by using scaling or a weighted loss function.

7.2 SENSORY ANALYSIS OF BREAD

PROBLEM

The best known aspect of multi-way analysis is uniqueness. Uniqueness means that it is possible, e.g., to resolve spectral data into the spectra of the pure components and to estimate analyte concentrations in the presence of unknown interferences. Here some of the other advantages of using multi-way methods will be shown by an example from sensory analysis. This application should be seen more as an example of the benefit of using multi-way analysis as opposed to unfolding, than as an in-

depth sensory analysis¹⁵.

DATA

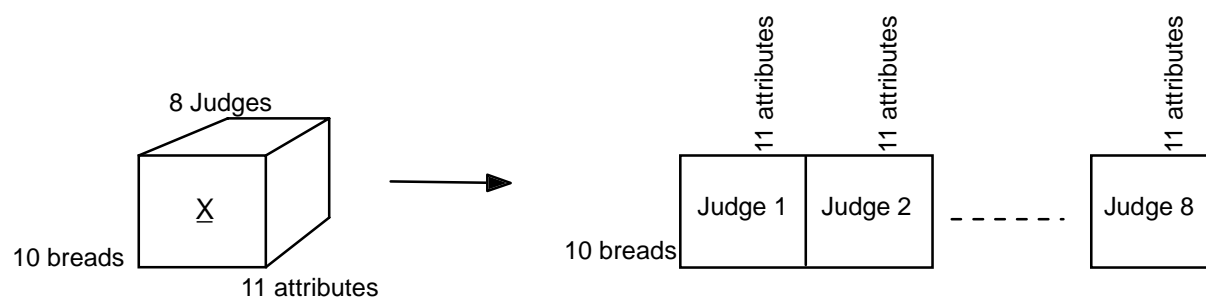


Figure 19. The sensory data.

Five different breads were baked in replicates giving a total of ten samples. Eight different judges assessed the breads with respect to eleven different attributes. The data can be regarded as a three-way array ($10 \times 11 \times 8$) or alternatively as an ordinary two-way matrix (10×88) as shown in Figure 19. Characteristic of the data is that there is no theory stating the basic nature of the data. Neither two-way PCA nor three-way PARAFAC are thus appropriate from that point of view; they are merely ways of reducing data in a sensible way. Also the data are quite noisy as opposed to, e.g., spectral data. It may be speculated, that for unfolding to be appropriate it must hold that the judges use individually defined latent variables. Using a PARAFAC model, the approximation is that the judges use the same latent variables, only in different proportions. In the following three main properties of choosing the right structure will be evaluated: Noise reduction, interpretability and internal validity (in terms of cross-validated predictions).

NOISE REDUCTION

The bread/sample mode of the sensory data can be explored by means of PCA on the unfolded array or by PARAFAC on the three-way data. In both cases a two-component solution of the data centered across the sample mode seems appropriate. No scaling was used since all attributes are in

¹⁵. This analysis was carried out in conjunction with Magni Martens, who also kindly provided the data.

the same units and range. The scores are shown in Figure 20. Notice how in both plots replicates (1-2,3-4, etc) lie close together, but notice also that there are larger discrepancies in the PCA plot than in the PARAFAC plot (replicates 5-6 and 9-10). The more parsimonious PARAFAC model is less influenced by the noise. The discrepancies between replicates that cause the larger differences in the PCA scores could be important in some respects. However, the score plot shows that the multi-way model imposes more structure than the unfolded model, and hence filters away more variation/noise; and replicates should be close together in score plots.

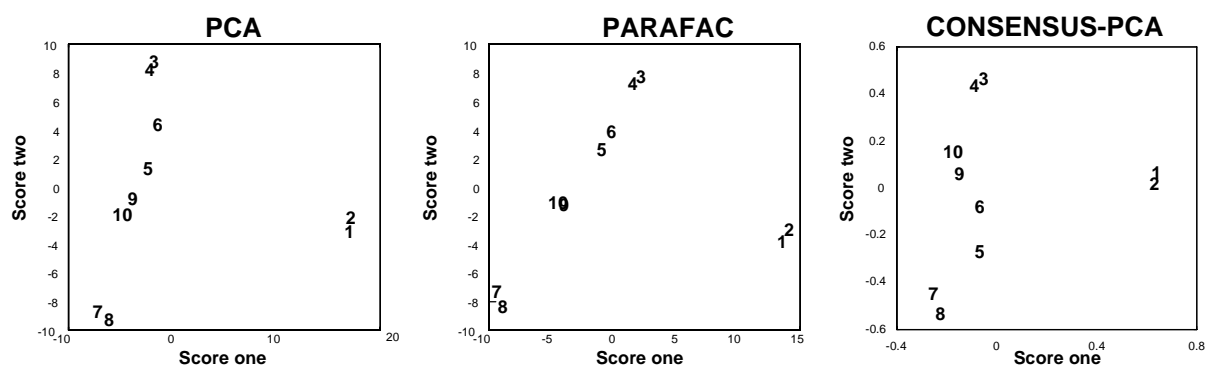


Figure 20. Scores from unfold-PCA, PARAFAC and consensus-PCA.

An important note can be made here regarding another type of model sometimes advocated for in chemometrics, namely the hierarchical or consensus models (Wold et al. 1996)¹⁶. Consensus models are excellent for overviewing huge, but otherwise good, models. They do not, however, impose additional structure to any significant degree compared to unfolding. Therefore consensus models should only be used when data are strictly inappropriate for multi-way modeling. That little structure is added compared to unfolding can be seen by the score plot from a consensus-PCA model¹⁷ of the data shown to the right in Figure 20. If anything, the

¹⁶. Consensus-PCA is based on finding a common structure in several multivariate data sets. This is done by fitting a PCA model based on the scores of local PCA models of each data set. There are several algorithmic nontrivial problem to consensus-PCA which will not be discussed here. Conceptually, the consensus-PCA model is computed by a PCA on a matrix of scores from the individual PCA models.

¹⁷. The algorithm for consensus-PCA is based on an algorithm kindly provided by H. Martens.

consistency of the scores of the consensus-PCA model are worse than those obtained from unfold-PCA with respect to the closeness of the replicates.

INTERPRETATION

The PCA model gives a set of loadings in the combined attribute/judge mode, while the PARAFAC model gives separate loadings for each mode (Figure 21). In the PARAFAC loadings several interesting aspects are easily seen. For example, toughness and saltiness are correlated because salt affects the texture. The breads coded one and two are seen from the score and attribute plots to be correlated to sweetness and off-flavor. Indeed, these breads are special-made sweet buns, while the remaining ones are more ordinary wheat breads. Clearly the PCA loadings are difficult to interpret because each judge is associated with eleven different points (attributes) and each attribute is associated with eight different points (judges).

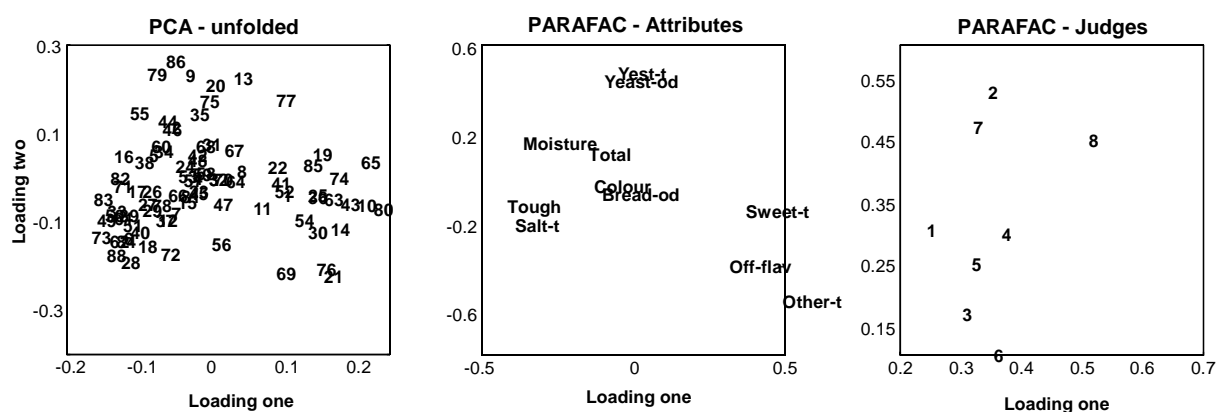


Figure 21. Loading plots from a PCA and a PARAFAC model of the sensory data.


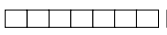

One may infer that the 88 loadings for each component could be averaged in order to get an average picture for, e.g., each judge. Most likely this would lead to a similar result as the PARAFAC result, just like the score plots are quite similar. However, since there is no way of validating which is better it would be impossible to conclude anything from any dissimilarity. Therefore, and because the results throughout this application lead to the

same conclusion anyway, this averaging has not been pursued. Also, an averaging of the loadings can be regarded as a heuristic way of obtaining what consensus methods and PARAFAC achieve. What remains with respect to interpretability is, that the loadings of PCA *are* more difficult to interpret than the three-way loadings as shown by the loading plots. Remedying this for example by averaging or using consensus-PCA is just an ad-hoc way of obtaining what the PARAFAC model is already obtaining in a well-defined way by adding structure.

PREDICTION

For every bread the salt content is known. To predict the salt from the sensory data three different calibration models are calculated: True three-way PLS on the $10 \times 11 \times 8$ data array, unfold-PLS on the 10×88 matrix obtained by unfolding the three-way array and bi-PLS on the 10×11 matrix obtained by averaging over judges (Table 2). Notice that both the unfold-PLS and bi-PLS on the averaged data are ordinary two-way models, only the data differ.

Table 2. Calibration results from three different models: bi-PLS on data averaged over assessors (hence two-way data), unfold-PLS on the whole data set, and tri-PLS on the three-way data. Shown in the table are the percentages variation explained in both X and y during calibration (fit) and full cross-validation. Also shown are root-mean-squared errors. The most appropriate models in terms of cross-validation are marked by a grey background. LV is the number of latent variables.

	LV	Variation explained /%				RMSE	
		X cal.	X val.	Y cal.	Y val.	Y cal.	Y val.
 bi-PLS(ave)	1	61	46	75	60	23	30
	2	83	72	94	85	12	18
	3	94	85	98	94	6	12
	4	95	87	99	88	4	17
	5	97	87	99	82	4	20
	6	98	89	100	76	2	23
 unfold-PLS	1	43	25	80	62	21	29
	2	61	38	95	76	10	23
	3	74	49	100	84	3	19
	4	78	49	100	84	1	19
	5	84	50	100	84	0	19
	6	87	52	100	84	0	19
 Trilinear PLS	1	31	22	75	60	23	30
	2	46	36	93	82	12	20
	3	54	44	98	91	7	15
	4	57	46	100	91	3	14
	5	60	47	100	90	2	15
	6	61	47	100	90	0	15

Several conclusions can be drawn from this table. First, it is important to note, that the calibration problem is not relevant in a real setting. Predicting

physical or chemical properties from expensive sensory evaluations is seldom relevant. However, the quality and differences in the three calibration models tells something about the appropriateness of the structural models of \mathbf{X} and are hence important. Also the problem can be considered a generic example of noisy data where little prior knowledge of the structure of the data is available. Hence a situation similar to, e.g., process analysis.

The most important conclusion is that unfold-PLS is clearly the poorest model of the three. It has the lowest predictability ($Y_{val} - 3 LV$) and it also amply overfits the models of both \mathbf{X} and \mathbf{y} . For example, the three-component model fits 74 % of the variation of \mathbf{X} but explains only 49 % in terms of cross-validation. This is actually a common phenomenon in multi-way analysis using unfolding methods. It also points to the important fact, that interpretation of unfolded models should be performed with great care. The degree of overfit, will most likely lead to misleading model parameters. This is similar to the difficulty in interpreting regression coefficients of multiple linear regression models of collinear data. The variance of the parameters caused by the inappropriateness of the model makes the model parameters uncertain. Similar conclusions will hold for consensus or hierarchical methods, as these do not add much structure in the variable mode.

So what is left is the multi-way model and the averaged two-way model. Theoretically, the choice between these two should be based on how well-trained the panel is. For a well-trained panel each assessor can be regarded as a replicate, and it is thus sensible to average over judges leading to the two-way (not unfolded) model. If on the other hand differences do occur between judges this is not easily handled if data are merely averaged over judges. Methods do exist for assessing such phenomena, but here the focus is not on sensory analysis specifically, but on understanding when it is appropriate to use the multi-way structure of the data.

The two-way and three-way models give a similar degree of overfit of \mathbf{X} , but the two-way model is describing more variation than the trilinear model. This is understandable and appropriate as the prior averaging of the data naturally leads to exclusion or reduction of variation not consistent with the overall variation. Hence the differences in the variance explained do not

help in assessing which method is preferable. Looking instead at the root-mean squared errors of the dependent variable it may be noted that the two-way model outperforms the trilinear model, though the difference between the two is smaller than the difference between tri-PLS and unfold-PLS. However, the change in the root-mean squared errors with the number of latent variables seems to point to tri-PLS, as it is more robust against slight mis-specification of the model. It is not exactly clear why this is so, but a plausible explanation can be that an excessive number of components in the bi-PLS model will lead to more or less arbitrary results because only little structure is imposed in the \mathbf{X} model. Contrary, for the trilinear model, even though most of the sensible variation in \mathbf{y} is explained after three components, the little meaningful variation that is left combined with the more restricted and structured model of \mathbf{X} will not lead to a model completely guided by random variations.

CONCLUSION

Multi-way methods have many advantages not only attributable to uniqueness. The results throughout this analysis consistently confirm that multi-way methods can be superior to unfold methods; essentially because they are more parsimonious. However, it has also been demonstrated, that knowledge of the data being modeled is important. While unfold-methods should generally be avoided or used with caution, the averaging and subsequent two-way analysis of the data yielded an excellent model. The small differences between the averaged two-way model and the trilinear model demonstrate that both can be sensible approaches. In the earlier stages of the analysis multi-way analysis may be preferable in order to be able to assess the individual differences between judges, while at later stages the averaged model may be better, because then differences between judges is perhaps not an issue, and the two-way model is thus easier to interpret.

This application arose from sensory analysis, but in process analysis, QSAR (Nilsson et al. 1997), and many other areas, noisy data abound. Restraining from the use of unfolding can be beneficial there as well.

7.3 COMPARING DIFFERENT REGRESSION MODELS (AMINO-N)

PROBLEM

In order to verify if the quality of refined sugar can be predicted from the fluorescence of dissolved sugar, 98 samples were measured spectrofluorometrically. The amino-N content was determined and a model sought for predicting amino-N from fluorescence. In the following several approaches to construct a calibration model will be tested in order to find the most appropriate complexity of a model for amino-N. In this example traditional second-order calibration in the GRAM-sense is not possible, as there is no causal or indirect relationship between one specific spectral component and the amino-N content. Following more or less the strategy of PCR a model is sought whose scores can predict the amino-N content of the sugar samples from the fluorescence. The scores constitute the independent variables and are related to the amino-N content by multiple linear regression.

DATA

The emission spectra of 98 sugar samples dissolved in phosphate buffered water were measured at four excitation wavelengths (excitation 230, 240, 290, and 340 nm, emission 375-560 nm, 0.5 nm intervals). Hence the data can be arranged in a $98 \times 371 \times 4$ three-way array or a 98×1484 unfolded two-way matrix. The amino-N content was determined by a standard wet-chemical procedure as described in Nørgaard (1995b).

RESULTS

Several calibration models were constructed. The quality of the models was determined by test set validation with 49 samples in the calibration as well as the test set. In each case a multilinear model was fitted to the fluorescence data and a regression model estimated by regressing amino-N on the scores of the fluorescence model. For the PLS models the decomposition and regression steps are performed simultaneously. In order to predict the amino-N content of the test set samples, the scores of the test set samples were calculated from the model of the fluorescence data. Using

the scores the amino-N content was estimated from the regression model.

The following models were investigated: PARAFAC regression using raw fluorescence data¹⁸, tri-PLS as well as two-way unfold-PLS on centered data, Tucker3 regression on raw data, and two-way unfold principal component regression (PCR) on centered data. The Tucker3 regression was performed by decomposing the raw data with a Tucker3 model using the same number of components in each mode and using the scores for regression. PCR was performed using the successively estimated score vectors from an unfold-PCA model for the regression model.

The results from the calibration models are shown in Table 3 as a function of the number components. All models give optimal or near-optimal predictions around five components. Unfold-PLS and tri-PLS seem to perform slightly better than the other methods and furthermore using fewer components. All pure decomposition methods (PARAFAC, Tucker3, PCA) describe approximately 99.8% of the spectral variation in the test set. Even though the PCA and Tucker3 models are more complex and flexible than PARAFAC the flexibility apparently does not contribute to better modeling of the spectra. Combining this with the fact that the PARAFAC regression models outperform both Tucker3 and PCA, illustrates that when PARAFAC is adequate there is no advantage of using more complex models.

The constraints imposed in unfold-PLS and tri-PLS seem to be useful. Both give more predictive models for amino-N. Both models fit the spectral data poorer than the pure decomposition methods, which is expectable due to the additional constraint of the scores having maximal covariance with the dependent variable. Tri-PLS uses only a fraction of the number of parameters that unfold-PLS uses to model the spectral data, so in a mathematical sense, tri-PLS obtains optimal predictions with the simplest model. Therefore it can be argued, that the tri-PLS model is the most appropriate model. However, the tri-PLS model does not possess the uniqueness properties of PARAFAC in the sense that it provides estimates of the pure spectra. It might therefore also be argued that a five-component PARAFAC model is preferable, if the found loadings can be related to specific chemical analytes (see page 230 for an example on this).

¹⁸. A non-negativity constrained model as well as a PARAFAC model on centered data gave similar results.

Table 3. Percentage of variance explained of the dependent variable (amino-N) of the test set. Each column corresponds to a different model and each row to the number of latent variables/components used. Bold numbers indicate % variation explained of amino-N (test set) for candidate models, and the numbers in parentheses are the percentage of variance explained of the three-way array of independent variables in the test set (fluorescence spectra).

LV	PARAFAC	Unfold-PLS	Tri-PLS	Tucker3	PCR
1	84.0	84.4	84.2	84.3	83.9
2	85.4	86.6	86.1	84.8	85.7
3	85.2	88.5	88.9	85.3	86.8
4	87.1	91.6	91.4	88.0	87.2
5	91.2 (99.8)	91.9 (96.0)	92.3 (95.7)	87.7 (99.8)	88.0 (99.9)
6	90.8	92.2	92.2	89.7	87.0

CONCLUSION

It has been shown that, when data can be assumed to be approximately multilinear and the signal-to-noise ratio is high, there is no benefit in using unfolding methods. Even though the unfolding models describe more variation per definition the increased modeling power does not provide more predictive models neither in terms of modeling the independent nor the dependent variables.

Using PARAFAC for regression as shown here has the potential for simultaneously providing a model that predicts the dependent variable, and uniquely describes which latent phenomena are crucial for describing the variations in the dependent variable. However, N-PLS outperforms PARAFAC giving better predictions using fewer components.

7.4 RANK-DEFICIENT SPECTRAL FIA DATA

PROBLEM

Nørgaard & Ridder (1994) have investigated a problem of measuring samples with three different analytes on a flow injection analysis (FIA) system where a pH-gradient is imposed. The data are interesting from a data analytical point of view, especially as an illustration of closure or rank-deficiency and the use of constraints¹⁹.

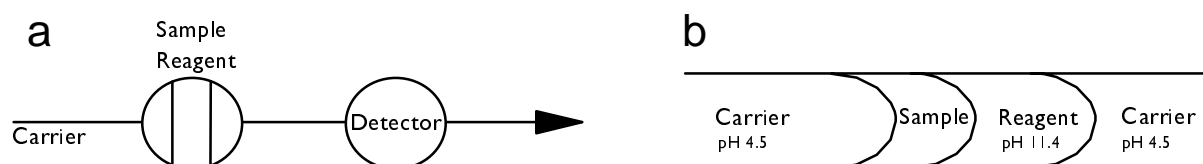


Figure 22. In a) the basic setup of the FIA system is shown, and in b) the sample plug is shown schematically.

DATA

The basic setup of the FIA system is shown in Figure 22a. A carrier stream containing a Britton-Robinson buffer of pH 4.5 is continuously injected into the system with a flow of 0.375 mL/min. The 77 μ L of sample and 770 μ L of reagent (Britton-Robinson buffer pH 11.4) are injected simultaneously into the system by a six-port valve and the absorbance is detected by a diode-array detector (HP 8452A) from 250 to 450 nm in two nanometer intervals²⁰. The absorption spectrum is determined every second 89 times during one injection. By the use of both a carrier and a reagent (Figure 22b) a pH gradient is induced over the sample plug from pH 4.5 to 11.4.

¹⁹. Richard A. Harshman is thanked for important discussions and contributions to the work presented here. The results are partly part of a collaboration between Harshman and Bro.

²⁰. One wavelength was ignored as apparently the part of the grating of the photometer corresponding to this wavelength was not working properly. This does not have any quantitative influence on the results because of the number of variables and the structure imposed. The only influence is that parameters corresponding to that specific variable are somewhat distorted.

The three analytes present in the samples are 2-, 3-, and 4-hydroxybenzaldehyde (HBA). All three analytes have different absorption spectra depending on whether they are in their acidic or basic form. Twelve samples of different constitution (Table 4) are measured. Thus the data set is a 12 (samples) \times 100 (wavelengths) \times 89 (times) array. The time mode of dimension 89 is also a pH profile due to the pH-gradient.

Table 4. The concentrations of the three analytes in the 12 samples.

Sample	2HBA	3HBA	4HBA
1	0.05	0.05	0.06
2	0.05	0.10	0.04
3	0.05	0.10	0.06
4	0.10	0.05	0.04
5	0.10	0.10	0.04
6	0.10	0.10	0.06
7	0	0.10	0.04
8	0	0.10	0.06
9	0.05	0	0.06
10	0.10	0	0.06
11	0.05	0.10	0
12	0.10	0.05	0

For each sample a landscape is obtained showing the spectra for all times, or conversely the time profiles for all wavelengths (Figure 23).

It is characteristic of FIA that there is no physical separation of the sample. All analytes have the same dilution profile due to dispersion, i.e., all analytes will have equally shaped *total* time profile. In Figure 23 this profile is shown with a thick line below to the left. This profile thus maintains its shape at all wavelengths for all samples and for all analytes. The total profile is the profile actually detected by the photometer (the manifest profile) and is the sum of the profiles of protonated and deprotonated analytes. Due to the pH-gradient, and depending on the pK_a of a given

analyte, an analyte will show up with different amounts of its acidic and basic form at different times, and hence will have different acidic and basic profiles in the sample plug. In the figure above these profiles are shown for one analyte. The first part of the sample plug, i.e., the earliest measurements of a sample, is dominated by deprotonated analytes while the end of the sample plug is dominated by protonated analytes.

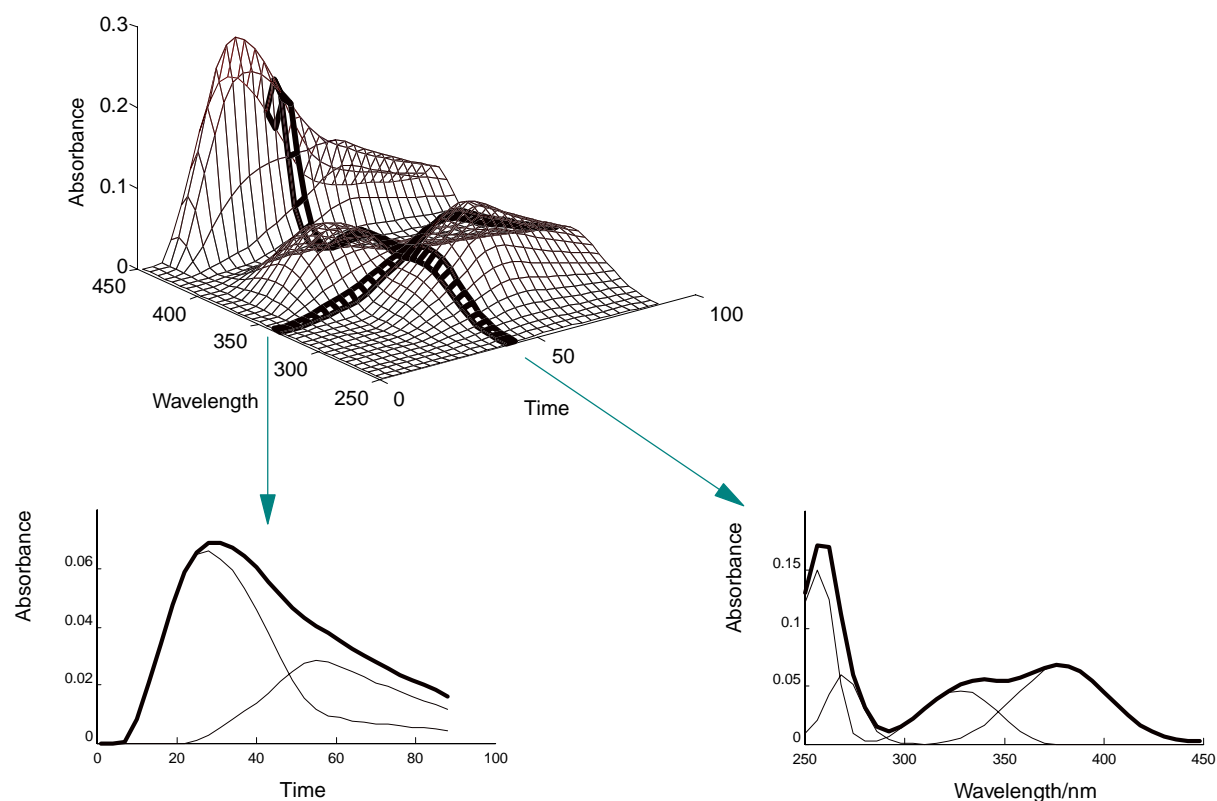


Figure 23. A landscape obtained from a sample containing only 2HBA (top). The measured profile at 340 nm is shown below to the left with a thick line. This measured profile is the sum of the profile of the (unknown) acidic and basic profile of the analyte shown with thinner lines. The same holds for the spectra. Below to the right the measured spectrum at time 43 is shown with a thick line. This spectrum is the sum of the (unknown) acidic and basic spectrum of 2-HBA shown.

STRUCTURAL MODEL

In order to specify a mathematical model for the data array of FIA data initially ignore the time domain and consider only one specific time, i.e., one

specific pH. An $I \times J$ matrix called \mathbf{X}_k is obtained where I is the number of samples (12), J is the number of wavelengths (100), and k indicates the specific pH/time selected.

There are three analytes with three corresponding concentration profiles and there are six spectra, an acidic and a basic for each analyte. A standard bilinear model would be an obvious decomposition method for this matrix, but this is not very descriptive in this case. In the sample mode, a three-dimensional decomposition is preferable, as there are only three different analytes. However, each analyte exists in two forms (acid/base), so there will be six different spectra, to be resolved, requiring a six-dimensional decomposition in the spectral mode. To accommodate these seemingly conflicting requirements, a more general model can be used instead

$$\mathbf{X}_k = \mathbf{A}\mathbf{H}\mathbf{B}^T, \quad (249)$$

where \mathbf{A} is an $I \times 3$ matrix, and the columns are vectors describing the variations in the sample domain (ideally the concentrations in Table 4), \mathbf{B} is a $J \times 6$ vector describing the variations in the spectral domain (ideally the pure spectra), and \mathbf{H} is a 3×6 matrix which defines the interactions between the columns of \mathbf{A} and \mathbf{B} . In this case it is known how the analyte concentrations relate to the spectra, as the acidic and basic spectrum of, e.g., 2HBA only relate to the concentration of 2HBA. Therefore \mathbf{H} reads

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \quad (250)$$

The matrix \mathbf{H} assures that the contribution of the first analyte to the model is given by the sum of $\mathbf{a}_1\mathbf{b}_1^T$ and $\mathbf{a}_1\mathbf{b}_2^T$ etc. By using only ones and zeros any information in \mathbf{H} about the relative size of the interactions is removed; this information is represented in \mathbf{B} . The \mathbf{H} matrix is reserved for coding the interaction *structure* of the model.

So far, only a single time/pH has been considered. To represent the

entire data set, the model must be generalized into a multi-way form. For each time the data can be represented by the model above except that it is necessary to adjust it such that the changes in relative concentration (acidic and basic fraction) can be represented as well. The relative concentration of each of the six acidic and basic analytes can be represented by a 6×1 vector at each time. The relative concentrations at all K times is held in the $K \times 6$ matrix \mathbf{C} . To use the generic model in equation 249 at the k th time it thus is necessary to scale the contribution from each analyte by its corresponding relative concentration. The six weights from the k th row of \mathbf{C} are placed in a 6×6 diagonal matrix \mathbf{D}_k so that the s th diagonal element gives the relative amount of the s th species. The model can be then written

$$\mathbf{X}_k = \mathbf{A} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} c_{k1} & 0 & \dots & 0 \\ 0 & c_{k2} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & & c_{k6} \end{bmatrix} \mathbf{B}^T \quad (251)$$

or, in other words, as

$$\mathbf{X}_k = \mathbf{A} \mathbf{H} \mathbf{D}_k \mathbf{B}^T. \quad k = 1, \dots, K \quad (252)$$

Note how the use of a distinct \mathbf{H} and \mathbf{C} (\mathbf{D}_k) matrix allows the qualitative and quantitative relationships between \mathbf{A} and \mathbf{B} to be expressed separately. The interaction matrix \mathbf{H} , which is globally defined, gives the interaction *structure*; it shows exactly which factors in \mathbf{A} are interacting with which factors in \mathbf{B} . In contrast, the \mathbf{C} matrix gives the interaction *magnitudes*. For every k the k th row of \mathbf{C} (diagonal of \mathbf{D}_k) shows to which extent each interaction is present at the given k . The distinction between qualitative and quantitative aspects is especially important, since knowledge of the exact pattern of interactions is not always available. Not fixing \mathbf{H} as here allows for exploring the type of interaction. This can be helpful for rank-deficient problems in general. The matrix \mathbf{C} also has a straightforward interpretation

as each column in \mathbf{C} will be the estimated FIAGram or time profile of the given analyte in its acidic or basic form.

ALTERNATIVE DERIVATION OF THE FIA MODEL

BOX 20

The data obtained from one sample is a matrix \mathbf{X} of size $J \times K$ (wavelengths \times time). A one analyte sample can in theory be modeled

$$\mathbf{X} = \mathbf{a}\mathbf{b}_1\mathbf{c}_1^T + \mathbf{a}\mathbf{b}_2\mathbf{c}_2^T = \mathbf{a}\mathbf{B}\mathbf{C}^T, \quad (253)$$

where a is the concentration of the analyte, \mathbf{b}_1 is the spectrum of the analyte on acidic form, \mathbf{b}_2 is the spectrum of the basic form, and \mathbf{c}_1 and \mathbf{c}_2 are the respective time profiles of the two forms of the analyte. Extending the model to more than one analyte is done by simply summing over the number of analytes:

$$\mathbf{X} = \sum_{f=1}^F \mathbf{a}_f \mathbf{B}_f \mathbf{C}_f^T, \quad (254)$$

where \mathbf{B}_f is a $J \times 2$ matrix containing the acidic and basic spectrum of analyte f , and \mathbf{C}_f contains the corresponding time profiles of analyte f . This can also be expressed without summation signs by introducing the interaction matrix \mathbf{H} as defined in equation 250 and letting $\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3 \ \mathbf{b}_4 \ \mathbf{b}_5 \ \mathbf{b}_6]$ where \mathbf{b}_1 , \mathbf{b}_3 , and \mathbf{b}_5 are the acidic spectra of the respective analytes and \mathbf{b}_2 , \mathbf{b}_4 , and \mathbf{b}_6 are the basic. The matrix \mathbf{C} is defined likewise. With this notation equation 254 can be expressed

$$\mathbf{X} = \mathbf{a}_1(\mathbf{b}_1\mathbf{c}_1^T + \mathbf{b}_2\mathbf{c}_2^T) + \mathbf{a}_2(\mathbf{b}_3\mathbf{c}_3^T + \mathbf{b}_4\mathbf{c}_4^T) + \mathbf{a}_3(\mathbf{b}_5\mathbf{c}_5^T + \mathbf{b}_6\mathbf{c}_6^T) \Rightarrow (255)$$

$$\text{vec}(\mathbf{X})^T = \mathbf{a}^T \mathbf{H} (\mathbf{C} \otimes \mathbf{B})^T$$

where \mathbf{a} is the vector holding the three elements a_i . For several samples, \mathbf{X}_i , the model can then be expressed

$$\mathbf{X}^{(I \times JK)} = \mathbf{A} \mathbf{H} (\mathbf{C} \otimes \mathbf{B})^T \Leftrightarrow \mathbf{X}_k = \mathbf{A} \mathbf{H} \mathbf{D}_k \mathbf{B}^T. \quad (257)$$

Note that equation 252 bears some resemblance to the PARAFAC model

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{B}^\top, \quad (258)$$

but differs mainly by the introduction of the matrix \mathbf{H} , which enables the interactions between factors in different modes. It also enables \mathbf{A} and \mathbf{B}/\mathbf{C} to have different column dimensions.

The PARATUCK2 model is given

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k^{\mathbf{A}}\mathbf{H}\mathbf{D}_k^{\mathbf{B}}\mathbf{B}^\top, \quad (259)$$

while the FIA model is

$$\mathbf{X}_k = \mathbf{A}\mathbf{H}\mathbf{D}_k\mathbf{B}^\top. \quad (260)$$

The FIA model can thus be fitted a restricted PARATUCK2 algorithm (page 40). The matrix \mathbf{H} remains fixed during the analysis and is set as specified in equation 250 to ensure that every analyte only interact with two spectra/profiles, namely its acidic and its basic counterpart. In Box 20 an alternative derivation of the model is given.

UNIQUENESS OF BASIC FIA MODEL

For quantitative and qualitative analysis of the FIA data it is crucial to know if the model is unique. If so, the first mode scores, \mathbf{A} , can be considered to be estimates of the concentrations of the analytes up to a scaling. Hence, if the true concentrations are known in one or more samples the concentrations in the remaining samples can be estimated (second-order advantage). Also, if the second mode loadings can be uniquely determined, the spectra of the analytes can be recovered, hence providing means to identify the analytes.

In order to investigate the uniqueness properties of the structural model consider the following recasting of the restricted PARATUCK2 model as a PARAFAC model. Let

$$\mathbf{G} = [\mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_3] = \mathbf{A}\mathbf{H} \quad (261)$$

Then the FIA model

$$\mathbf{X}_k = \mathbf{A}\mathbf{H}(\mathbf{C} \otimes \mathbf{B})^T, \quad (262)$$

is equivalent to the PARAFAC model

$$\mathbf{X}_k = \mathbf{G}(\mathbf{C} \otimes \mathbf{B})^T. \quad (263)$$

From the theory of PARAFAC (Harshman 1970) it is known that such a special PARAFAC model with the first mode loadings being pairwise identical will have certain uniqueness properties. Specifically, if the columns in \mathbf{C} and \mathbf{B} are independent, i.e., each span a six-dimensional sub-space then \mathbf{G} is unique, whereas \mathbf{C} and \mathbf{B} are only partially unique. As \mathbf{g}_1 and \mathbf{g}_2 are identical (\mathbf{a}_1) then the corresponding second and third mode loadings will not be unique but the sub-space of these two components in both modes will be uniquely determined. It follows that \mathbf{A} , $\text{span}([\mathbf{c}_1 \mathbf{c}_2])$, $\text{span}([\mathbf{c}_3 \mathbf{c}_4])$, $\text{span}([\mathbf{c}_5 \mathbf{c}_6])$, $\text{span}([\mathbf{b}_1 \mathbf{b}_2])$, $\text{span}([\mathbf{b}_3 \mathbf{b}_4])$, $\text{span}([\mathbf{b}_5 \mathbf{b}_6])$ are unique. Here $\text{span}(\mathbf{M})$ is the span of the column-space of \mathbf{M} . Kiers and Smilde (1997) proved similar uniqueness results using a slightly different restricted Tucker3 model for the same data.

In this particular case, however, there are further restriction on \mathbf{C} in that the two profiles for any analyte sum to the same shape (the sample profile), which means that

$$\mathbf{c}_1 + \mathbf{c}_2 = \mathbf{c}_3 + \mathbf{c}_4 = \mathbf{c}_5 + \mathbf{c}_6 = \mathbf{c}_{\text{tot}}, \quad (264)$$

where \mathbf{c}_{tot} is the total profile (shown with a thick line in the lower left plot in Figure 23). This holds up to a scaling that can, for simplicity, be considered to be in \mathbf{B} . The appropriateness of this constraint depends on the chemical similarity between the analytes. In this case the analytes are very similar and will have the same profile except that the sample profile for 4-HBA differ marginally (Smilde et al. 1998). It is essential to consider if the profile constraint adds anything to the above-mentioned uniqueness properties.

AVOIDING LOCAL MINIMA**BOX 21**

In fitting the unconstrained FIA model it was apparent that often local minima were obtained. In the top part of the table below the value of the loss function (sum-squared errors) is shown for ten runs of the algorithm, here called ALS. Each run was initiated with different random numbers. The model consistently ends up in one of three different solutions all of which appear very similar. Four runs stopped before convergence due to the number of iterations (5000), and only one solution gave minimal loss function value. Hence it was difficult to have confidence in that the global minimum was found. Alternatively, the model was also fitted ten times using approximate orthogonality as described page 154. The first 200 iterations were performed using a decreasing penalty for the spectral loadings deviating from orthogonality. The resulting model results are tabulated below. As can be seen the number of iterations are now lower, and seven of the ten models converge to the same lowest minimum. Apparently, the use of approximate orthogonality helps avoiding the appearance of local minima, and also thereby speeds up the algorithm considerably. Note, though, that the presence of local minima does not disappear by using approximate orthogonality as the local minima are intrinsic to the model not the algorithm.

ALS	Loss	12072	12216	12216	12257	12257	12257	12319	12684	15835	16010
	Iter.	628	3204	1352	2608	4456	1872	5000	5000	5000	5000
ORTH	Loss	12072	12072	12072	12072	12072	12072	12072	12216	12257	12257
	Iter.	944	856	792	880	824	1088	808	1492	1956	1716

Note first that *any* subset of two columns of **C** will have full rank, as none of the analytes have identical pK_a , hence similar time profiles. Also note that $\text{span}([\mathbf{c}_1 \ \mathbf{c}_2])$, $\text{span}([\mathbf{c}_3 \ \mathbf{c}_4])$, and $\text{span}([\mathbf{c}_5 \ \mathbf{c}_6])$ will not be identical as this would imply that the rank of **C** was two. The rank of **C** under the equality

constraint can be shown to be four. Even though part of, e.g., $\text{span}([\mathbf{c}_1 \ \mathbf{c}_2])$ is implicitly given by the other profiles, at least part of $\text{span}([\mathbf{c}_1 \ \mathbf{c}_2])$ will be unique to the first analyte. This means, that some of the variable-space is only caused by the first analyte. It is therefore conjectured that the equality constraint does not add anything to the structural uniqueness of the model. This will be verified empirically by fitting the model without specifically imposing the equality constraint.

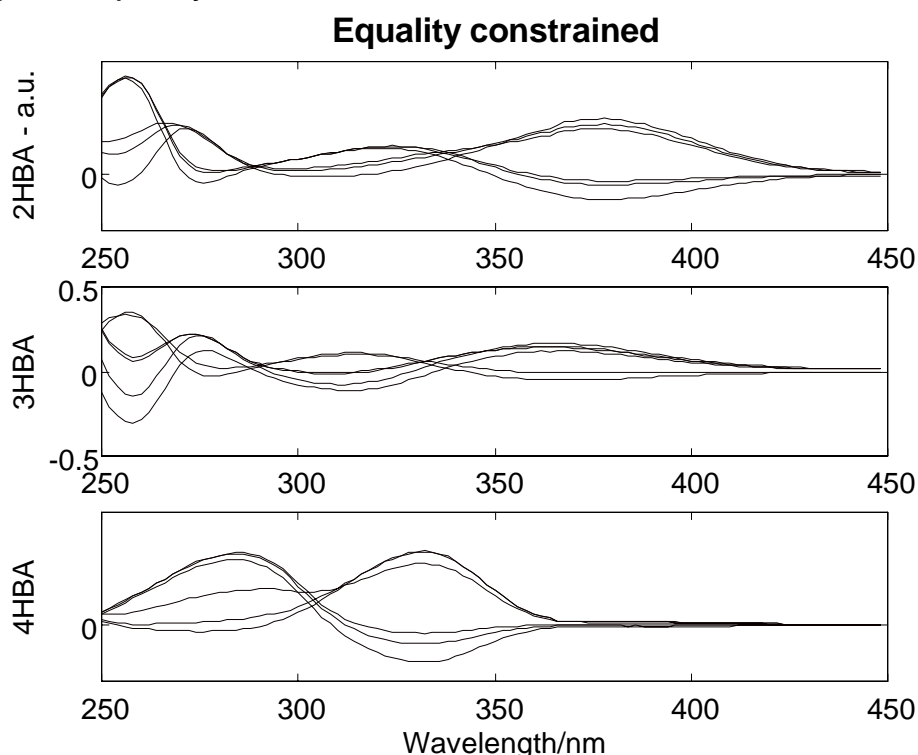


Figure 24. Three superimposed estimated spectral loadings of the three analytes estimated under equality constraint of the time profiles. Each of the plots hold three estimates of the two spectral loading vectors corresponding to one analyte. The model is seen not to be unique, as the parameters differ from estimate to estimate. The fits of the models are identical.

From the fitted model without constraints (Box 21) a model is obtained that readily verifies the above. From different runs of the algorithm different estimates of the 12×3 first mode loadings are obtained. These, however, are identical (correlation of loading vectors > 0.9999999). The spectral and

time mode loadings, on the other hand, are not unique but change from estimate to estimate. The span of the pairs belonging to the same analyte should still be unique which is verified empirically by monitoring the singular values of an SVD of the matrix obtained by concatenating, e.g., the estimated spectral loadings of the first analyte from two different estimates. Each set is of dimension 100×2 and the concatenated matrix thus of size 100×4 . A typical singular vector is $\mathbf{s} = [1.3091 \ 1.1130 \ 0.0002 \ 0.0001]$. Clearly, the two sets of loading vectors span the same space even though the individual loading vectors are unequal.

When fitting the model imposing equality of the summed profiles, not much is added with respect to uniqueness. The concentration mode loadings are still unique, while the remaining parameters are not (Figure 24).

An interesting numerical aspect was revealed in the analysis. For the unconstrained problem, the model was difficult to fit being subject to multiple local minima (Box 21). For the equality constrained problem, no local minima were observed. All fitted models converged to exactly the same solution (when using approximate orthogonalization). Even though the unconstrained model works well, it seems, that applying the additional equality constraint, helps overcoming the problem of local minima.

It is interesting to see if the equality constrained model provides better estimates of the concentrations of the analytes. Since both the constrained and unconstrained model gives unique estimates of the concentration loadings, both should be capable of estimating the concentrations. The quantitative goodness of the models is given below in Table 5 as the correlations between the reference concentrations and the estimated parameters. The equality constrained model provides better results especially for 3HBA.

Table 5. The correlations between reference and estimated concentrations from an unconstrained and an equality constrained model.

Unconstrained			Equality constrained		
2HBA	3HBA	4HBA	2HBA	3HBA	4HBA
.935	.700	.993	.977	.984	.998

To the extent that the equality constraint is considered an intrinsic part of the FIA model, it may be argued that fitting the model without equality constraint is wrong. In any case, it has been shown that the model, be it with or without equality imposed, allows for quantitative analysis but not for uniquely identifying the spectra. Applying equality, however, improves the model as evidenced in the table above. In the following it will be examined to which extent it is possible to obtain uniqueness of the spectral mode by using non-negativity and further if using additional appropriate constraints can help in obtaining better results. First it will be shown how to obtain estimates of the pure spectra.

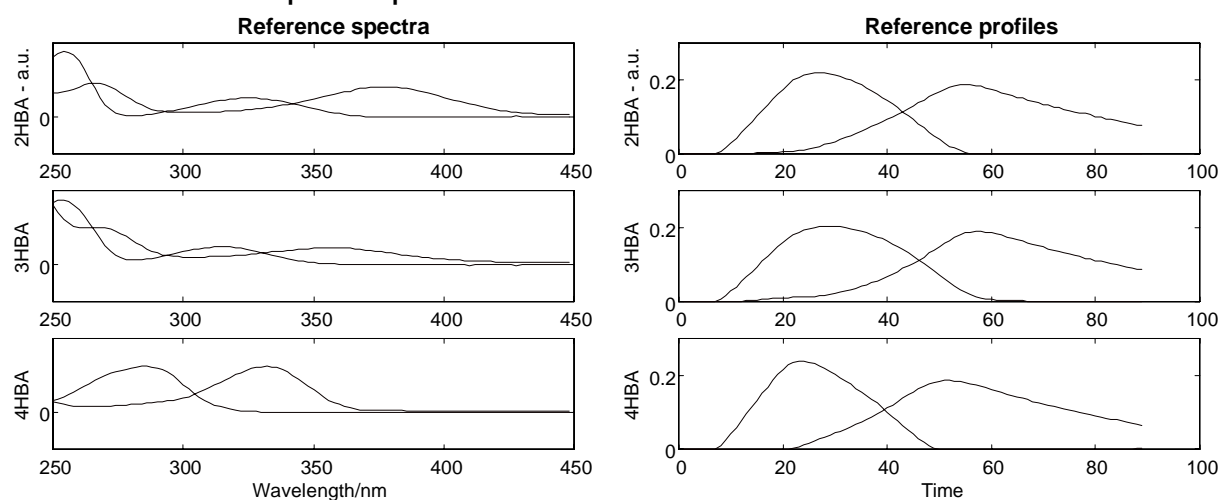


Figure 25. Estimated spectra and profiles of the pure analytes determined from pure samples. Each figure consists of eight consecutive estimated bilinear two-component non-negativity constrained models. All re-estimated models are identical. In each spectrum plot the acidic spectrum is the leftmost spectrum and in the time profile plots the basic profiles are the leftmost.

DETERMINING THE PURE SPECTRA

The spectra of the pure components, protonated 2HBA, deprotonated 2HBA etc. can be determined from samples of pure analytes. Such were additionally measured in triplicates and averaged. For each analyte sample a matrix of size $J \times K$ is obtained. If each such two-way sample is modeled by a two-way bilinear model with non-negativity constraints the resulting models are unique. This is strongly indicated empirically by modeling the same model several times, each time initiating with different random

numbers. The resulting parameters are identical in each run (correlation between reestimated spectra/profiles > 0.999999). Even though the two-way bilinear model calculated in each model is subject to rotational freedom, this apparently is circumvented by the non-negativity constraint, and by the fact that some wavelengths and times in the profiles are selective (see Figure 25). As seen from the time profiles there is a part in the beginning and in the end of the sample plug where only one of the analytes is present. The non-negativity constraint causes the parameters of the analyte not present to be exactly zero and as described in Manne (1995) this effectively leads to uniqueness when there are only two analytes present. Using traditional curve resolution techniques based on selectivity a similar result is obtained.

An interesting aspect of the bilinear models used to estimate the pure spectra is that a very stringent convergence criterion has to be used in order to find the solution. Otherwise slight deviations appear. Also, if the spectra and time profiles are estimated from a data matrix where only every third variable in each mode is used, the decomposition is not unique. For every analyte one of the two spectra is uniquely defined and the other only defined up to a certain span, even though all fitted models have the exact same fit. Even though the models fitted from the complete data are unique, simply reducing the data slightly changes the uniqueness properties 'dramatically'. This shows that structural uniqueness (e.g. as in PARAFAC) is a more robust feature of a model than relying on uniqueness from imposed constraints. Relying on uniqueness from non-negativity can be hazardous. The possible uniqueness obtained from using non-negativity stems from the selectivity implicitly induced by the pattern of zeros enforced. Instead of relying on a fortunate pattern of zeros it may be worthwhile to investigate the selectivity specifically and use this specifically as described page 143. Nevertheless, in this case uniqueness *is* obtained by applying non-negativity on the full data, and the resulting spectra have been confirmed empirically (Smilde et al. 1998) as well as by traditional curve resolution techniques.

A possible complication in fitting the models of mixtures of the analytes may arise from the very correlated spectra and time profiles. For the pure spectra (100 × 6) the correlation matrix reads

Spectra	2HBAA	2HBAb	3HBAA	3HBAb	4HBAA	4HBAb
2HBAA	1.00					
2HBAb	.23	1.00				
3HBAA	.97	.25	1.00			
3HBAb	.75	.62	.79	1.00		
4HBAA	.19	.08	.34	.49	1.00	
4HBAb	.28	-.33	.16	.01	-.08	1.00

where 2HBAA is the acidic spectrum of 2HBA etc. Some of the spectra have very high correlations. For the time profiles (89 × 6) the correlation matrix reads

Time	2HBAA	2HBAb	3HBAA	3HBAb	4HBAA	4HBAb
2HBAA	1.00					
2HBAb	-.48	1.00				
3HBAA	.99	-.38	1.00			
3HBAb	.58	.98	-.50	1.00		
4HBAA	.96	-.60	.91	-.65	1.00	
4HBAb	-.38	.98	-.26	.93	-.54	1.00

As can be seen all time profiles of similar protonation have a correlation higher than 0.9 and typically 0.96-0.99. To clarify the structure of the correlation matrix of the time profiles disregard correlations below 0.65. Then the full correlation matrix has the form

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

This amazing checkerboard pattern is due to the pH profile and the chemical similarity of the analytes. It shows that all acidic profiles are almost identical and all basic profiles are almost identical. Undoubtedly these high correlations complicate the estimation of the model parameters.

UNIQUENESS OF NON-NEGATIVITY CONSTRAINED ANALYTE SUB-SPACE MODELS

The structural uniqueness of **A** and of the sub-space orientations in **B** and **C** coupled with the uniqueness of the non-negativity constrained sub-space models seem to infer that the global PARATUCK2 model is unique if fitted under non-negativity constraints due to the selectivity implicitly enforced by the non-negativity constraints. Five estimates of the fitted model using only non-negativity constraints were calculated. As for the equality constrained model, no problems with local minima are observed. However, the model is extremely difficult to fit. The convergence criterion has to be 10^{-11} (relative change in sum-squared error) before convergence is achieved. As expected, the concentration mode loadings are identical in each fitted model and as good as the estimates obtained from the equality constrained model (Table 6).

Table 6. The correlations between reference and estimated concentrations from a non-negativity and an equality constrained model.

Non-negativity			Equality constrained		
2HBA	3HBA	4HBA	2HBA	3HBA	4HBA
.9988	.9787	.9996	.9769	.9837	.9979

However, the spectral and time mode loadings are not uniquely determined for basic 2HBA as seen in Figure 26.

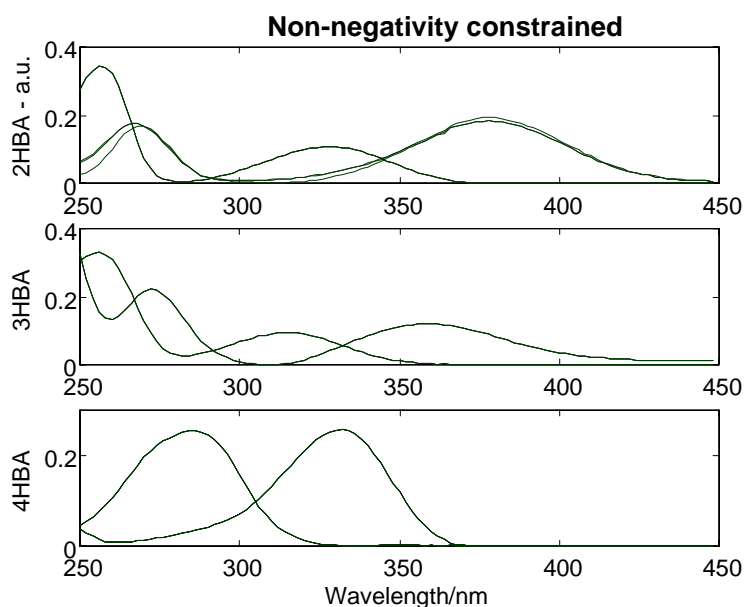


Figure 26. Five superimposed sets of estimated spectra from non-negativity constrained model. All fits are identical but the basic spectrum of 2HBA differs depending on starting point of the algorithm.

That the bilinear models of the pure analyte measurements are unique but the overall model is not must be due to matrix effects that changes the latent phenomena in the mixtures. It remains though, that constraining with non-negativity as opposed to equality provides a model with most parameters apparently being uniquely determined and as good estimates of the concentrations of the analytes.

IMPROVING A MODEL WITH CONSTRAINTS

In the preceding it has been shown that constraints can be helpful for providing (partial) uniqueness for an otherwise non-unique model. A more general problem is whether to use *a priori* knowledge in terms of constraints if available? What would the purpose be? While caution is definitely important it is, however, more important to realize, that any fitted model is subject to deviations from the 'truth' due to sampling variation, numerical aspects, model mis-specifications, and random error. Hence any constraint

which is appropriate (and active) should also be helpful for obtaining a more appropriate solution. This is to some extent comparable to the difference in the solution of a linear regression problem using least squares regression and using rank-reduced regression. Though the same model (the regression vector) is being fitted the use of additional constraints like keeping the norm of the regression vector low (ridge regression) or implicitly only allowing for special structure in the regression vector (PLS) makes the model more predictive and actually often more in accordance with the expectation of the least squares solution.

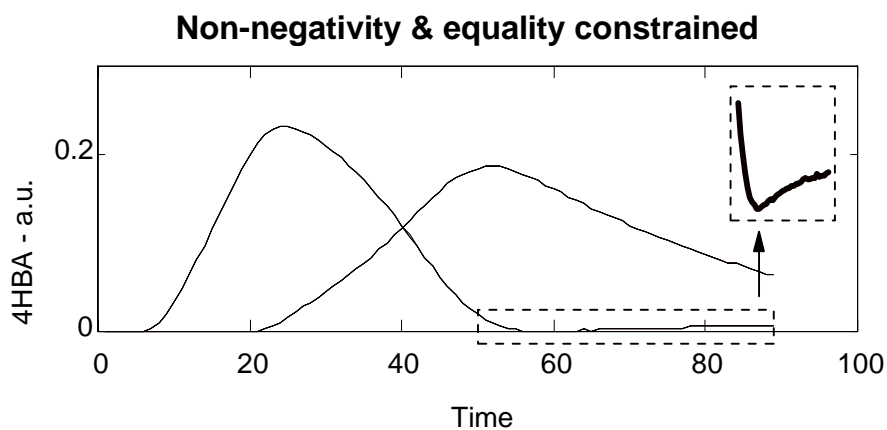


Figure 27. Estimated time profiles of 4HBA from a non-negativity and equality constrained model. A part of the basic 4HBA profile is shown enlarged.

In the following it will be shown that adding additional constraints to the FIA model *will* give better models in terms of how well the spectra and concentrations are estimated. Above the following models were considered:

- UNCONSTRAINED, providing unique estimates of concentrations
- EQUALITY CONSTRAINED, adding improved concentration estimates
- NON-NEGATIVITY CONSTRAINED, adding partially unique spectral estimates

It is natural to combine the non-negativity and equality constraints and indeed this provides a model that gives better estimates of the parameters. In Table 7 especially the concentration estimates of 3HBA are seen to be improved and in Table 8 the basic spectra of 2- and 3HBA are seen to be

improved. The model does not, however, provide uniqueness of the basic 2HBA spectrum.

Exploring the model parameters of the non-negativity and equality constrained model it is seen that the time profile of basic 4HBA increases after time 60 (Figure 27). As the pH profile induced over the sample plug is smooth it is not possible that the amount of deprotonated analyte increases after a decrease. This is also evident from the remaining profiles. It is difficult to explain why the estimated profile has this peculiar shape. It may be the result of a change in the acidic spectrum of 4HBA with pH caused by electronic changes in the chromophoric entity. Such a pH dependent variation of the acidic spectrum may be modeled as an increase in the basic spectrum. In any case, there is no doubt that the profile is not realistic. The time profiles must be unimodal. Hence a model is fitted where in addition to the non-negativity and equality constraints unimodality is required for the time profiles. Again the estimated parameters are improved slightly (Table 7 & 8), and still uniqueness is not obtained for the basic spectrum of 2HBA.

Table 7. Correlation between estimated and reference concentrations. The following abbreviations are used for the models. Eq: equality constrained; NNLS: non-negativity constrained; ULSR: unimodality constrained; Fix: Fixing parameters to zero as described in the text.

Concentrations	2HBA	3HBA	4HBA
Eq	.9769	.9837	.9979
NNLS	.9988	.9787	.9996
NNLS/Eq	.9992	.9987	.9996
NNLS/ULSR/Eq	.9992	.9987	.9996
NNLS/ULSR/Fix/Eq	.9990	.9987	.9996
Fixed parameters	.9975	.9915	.9986

*Table 8. Correlation between estimated and reference spectra. The following abbreviations are used for the models. Eq: equality constrained; NNLS: non-negativity constrained; ULSR: unimodality constrained; Fix: Fixing parameters to zero. The correlation marked with * are typical correlations as these parameters are not uniquely determined.*

SPECTRA	2HBA acidic	2HBA basic	3HBA acidic	3HBA basic	4HBA acidic	4HBA basic
Eq	.9893*	.9871*	.9689*	.7647*	.9106*	.9211*
NNLS	.9944	.9117*	.9952	.9241	.9974	.9977
NNLS/Eq	.9946	.9312*	.9953	.9988	.9965	.9971
NNLS/ULSR/Eq	.9946	.9590*	.9953	.9989	.9966	.9943
NNLS/ULSR/Fix/Eq	.9946	.9989	.9954	.9986	.9961	.9977
Fixed parameters	.9930	.9866	.9950	.9982	.9973	.9988

The most important result from this severely constrained model is that it is not unique. As for the non-negativity constrained model it holds, that the spectra of deprotonated 2HBA is only determined up to a certain span. It is interesting that even adding this many constraints does not change the uniqueness properties of the model. It clearly demonstrates, that uniqueness can be very difficult to obtain by constraints alone. However, even though the additional constraints do not help in obtaining uniqueness, they improve the estimates of the unique parameters.

Looking at the time profiles of the non-negativity, unimodality, and equality constrained model (Figure 28) it is seen that the time profile of basic 4HBA does not reach zero even by the end of the detection. This points to a modeling artifact as by the end of the sample plug where the pH is 4.5 there should not be any basic analytes present at all. This may be enforced in the model by requiring the basic profiles to be zero after, say time 75. Similarly it can be required that the acidic profiles do not appear until time 20 though this seems already to be fulfilled. A model with these additional

constraints was fitted. This model *will* be unique since the uniqueness properties of the restricted PARATUCK2 model states that every sub-space (temporal and spectral) for each analyte is unique and because the zeros induced implicitly ensures that there are selective time variables for both acidic and basic profiles within each sub-space. Indeed several fitted models estimated from different starting points all converge to the same unique solution. Furthermore the algorithm only requires approximately 500-700 iterations as compared to several thousands of iterations for some of the former models. Unlike the former models *all* parameters of this model are uniquely determined. The correlations of the concentration and spectral parameters with the reference values are shown in Table 7 & 8. It is easily verified that especially the estimation of the problematic basic spectrum of 2HBA has improved substantially. For the concentration estimates little differences are found, simply because these estimates are already uniquely determined and probably optimal.

Non-negativity, unimodality & equality constrained

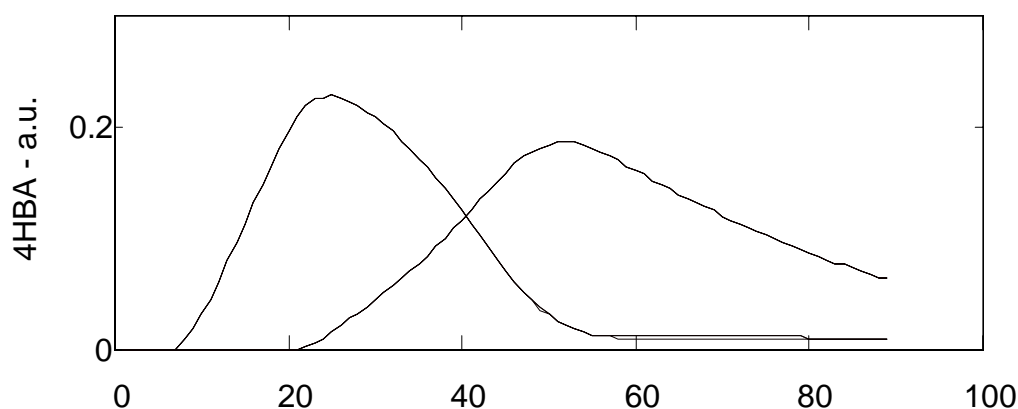


Figure 28. Estimated time profiles of 4HBA from a non-negativity, unimodality and equality constrained model.

That a model with fixed parameters as described above is unique, means that none of the additional constraints (non-negativity, unimodality, and equality) are necessary for obtaining uniqueness. In the above tables the results of a model using only fixed parameters is given. As can be seen the

model is excellent, but still the additional constraints improve the estimates for most parameters.

SECOND-ORDER CALIBRATION

Having established the uniqueness properties of the FIA model, it follows that it is possible to do second-order calibration. In Smilde et al. (1998) this is shown for these data using a single-analyte sample as standard and as unknown sample a sample with the analyte and one interferent. In Smilde et al. (1998) the results of the restricted PARATUCK2 model is compared with the results using restricted Tucker3 and multivariate curve resolution. All three methods give similar results and are shown to be able to predict the concentration of any analyte in a sample with an unknown interferent. Due to the special characteristics of the structural model it is not possible to quantify samples with more than one interferent if only one standard and one unknown sample is used. Using two or more unknown samples simultaneously or increasing the order of the array can remedy this.

CONCLUSION

It has been demonstrated that the restricted PARATUCK2 model is appropriate for a certain type of multi-way problems characterized by having different dimensionalities in different modes. The structure of the restricted PARATUCK2 model is appropriate for many rank-deficient problems and further have certain intrinsic uniqueness properties due to its relation to the PARAFAC model.

Besides the treatment of rank-deficient data, the analysis has also pointed to a number of results of more general importance. The most important findings are:

SWAMPS & LOCAL MINIMA

- It seems that minor swamps and local minima are related at least in the case studied here. For all models investigated estimating a model that ended up in a local minimum took more iterations than when it ended up in the global minimum.
- It seems that if a model is poorly defined, which means that it is not

identified, the appearance of swamps and local minima increases. This happens for example when some loading vectors are very collinear or when a component is hard to distinguish from the random variation.

- The appearance of local minima and swamps can be reduced by the use of initial approximate orthogonality.

UNIQUENESS

- Identifiability or uniqueness is not only an attractive feature; it is also often necessary if the number of iterations is to be kept low. It seems that the more 'poorly identified' the model is the flatter is the derivative of the loss function, and therefore the more iterations are needed for convergence. In this case, if an unconstrained model is sought, it may therefore be feasible to constrain the analyte sub-space models to, e.g., orthogonality in order to fully identify the model.
- Uniqueness of restricted PARATUCK2 can be at least partly investigated by posing the model as a PARAFAC model. Unlike other restricted models, this immediately shows the uniqueness properties of the model. It follows that most often at least one mode of the restricted PARATUCK2 model will be uniquely determined.

CONSTRAINTS

- It has been shown that constraints in general can not *a priori* be expected to provide uniqueness of an otherwise unidentified model. Even though a non-negativity constrained model apparently provides partial uniqueness with only one spectrum unidentified, adding equality and unimodality does not significantly help removing this unidentifiability. The only soft constraint treated here, that has significant potential of providing uniqueness is non-negativity. This is so, because an active non-negativity constraint necessarily implies that certain parameters are forced to be zero. If the pattern of zeros is such that certain variables are only non-zero for certain components uniqueness may come about as a consequence of the selectivity thereby enforced. This has been shown most clearly here for the bilinear models of pure analyte samples and the partial uniqueness obtained of the full model fitted only under

non-negativity constraints.

- It has been shown that *any* valid constraint will improve the estimated parameters if the constraint is active. This is true for curve resolution problems because the model parameters themselves are the goal of the model. For other problems other considerations come in.

A note on selectivity is important here. The model fitted only under selectivity constraints is excellent, though it can be improved by the use of additional constraints. This shows, that when a model is not structurally unique, it is beneficial to draw on the power of current curve resolution techniques. However, many problems exist for which selectivity is absent (e.g. fluorescence excitation emission matrices). In such cases the traditional curve resolution has little to offer, and partial or full uniqueness using other constraints has to be pursued. Then it is important to be able to constrain the model as much as possible in order to reduce the possible solution space as much as possible.

7.5 EXPLORATORY STUDY OF SUGAR PRODUCTION

PROBLEM

There is a need in the sugar industry to rationalize and improve quality and process control in general. One aspect of this is to gain a better understanding of the chemistry involved in the process. This can lead to better guidance of the sugar-beet growers and to a better foundation for controlling the process. Earlier investigations have primarily focused on establishing which chemical analytes are present in the sugar and intermediate products. Winstrøm-Olsen et al. (1979a & b) were able to separate a dozen catecholamines from raw juice of sugar and found that the typical concentration of norepinephrine was about 1-2 ppm, while the color-forming precursor dopa (3,4-dihydroxyphenylalanine) typically appeared in the concentration range 1-5 ppm. In Winstrøm-Olsen (1981a & b) the results were further elaborated on by isolating from beets enzymatic material which was then characterized with respect to how it affected catecholamines in pure solutions producing colored components (melanins). It was noted that if more than one catecholamine was present the enzymatic effect was difficult to predict due to the interrelationship between the catecholamines and the enzymes.

This type of information seldom leads to conclusive suggestions regarding a complicated process like the sugar manufacturing. It is an extremely expensive and reductionistic approach to learning about technological aspects of sugar production. In this case, it is additionally well-known that the enzymatic mis-coloring of sugar is but one of several ways that mis-coloring occurs. For example, non-enzymatic colorforming processes due to reaction of amino-acids and phenols with reducing sugars creating melanoidines (Maillard products) is also known to be a very important factor.

An attempt to use a more exploratory approach could be based on the following alternative strategy, much in line with the exploratory approach suggested by Munck et al. (1998) where fluorescence analysis is used to monitor the beet sugar process from the beet raw-material to the intermediate products and the final product sugar.

-
- Measure sugar samples spectrofluorometrically
 - Decompose the spectra using PARAFAC
 - Identify correlations between scores and quality/process parameters
 - Identify the underlying chemical components generating the relevant PARAFAC components
 - Utilize these components as indicator substances to monitor the process throughout the different production steps as a screening analysis for chemical/physical and process parameters

Thus, the sugar samples are observed and measured almost directly in the process, instead of being 'dissected' in a chemical laboratory analysis. This potentially can bring about more relevant information.

The main advantage of using spectral data for unraveling the information, is that spectral data makes it possible to make efficient and robust multivariate screening models. Within certain limits the spectral data provide a holistic and also non-destructive source of information enabling simultaneous modeling of several parameters. This has traditionally been shown using near infrared (NIR) analysis starting with the pioneering work of K. Norris and later developed into the area of multivariate calibration (Martens & Næs 1989), but the basic principle is by no means restricted to NIR. On the contrary, using fluorescence data instead of or supplementing NIR, another basic type of information is obtained. NIR data reflect chemical bonds more than chemical species present in the sample measured. This makes NIR suitable for modeling overall properties like fat or protein content. Fluorescence data, on the other hand, has a more subtle basis. Only analytes possessing well-defined fluorophores give rise to fluorescence. These analytes may or may not be indicative for important properties of the sample and the process. Evidently fluorescence does not possess the same generality as NIR. It gives selective and sensitive information. Due to the selectivity fluorescence yields precise causal information of potential indicator substances. Even more importantly the nature of the beets and the nature of the processing causes many phenomena to correlate quite closely (Nørgaard 1995b). Therefore the variation of the indicator substances may not only reflect direct causal relationships but also indirectly correlated phenomena as will be shown

based on the results of Bro (1998). It must be emphasized that the candidate indicator substances are not determined from theoretical consideration. They are determined from exploring the process and the data arising from the fluorescence screening analysis in a selection process interpreted in a dialogue with the sugar technologists.

In the following the data will first be described. The primary hypothesis is that the fluorescence screening analysis may reflect a global fingerprint of the chemical conditions in the sugar factory (Munck et al. 1998). A PARAFAC model of the fluorescence data will be developed. The results obtained solely from this model will first be interpreted. The results lead to the secondary hypothesis that fluorescence data reflect chemical variation and are related to the quality and other external parameters. This is investigated and proved by showing how the chemically meaningful fluorescence data model compares to physical, chemical and process data.

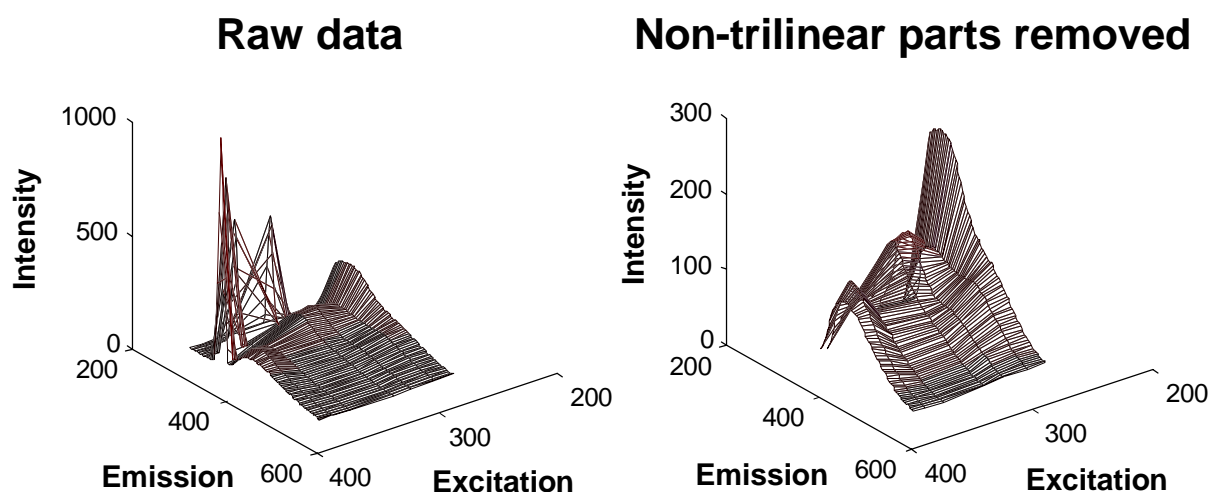


Figure 29. Fluorescence data from one sugar sample. To the left the raw data are shown and to the right the data are shown after removal of emission below excitation wavelength as well as Rayleigh scatter.

DATA

Sugar was sampled continuously during eight hours to make a mean sample representative for one 'shift' (eight hour period). Samples were taken during the three months of operation (the campaign) from a sugar plant in Scandinavia giving a total of 268 samples of which three were

discarded as extreme outliers in this investigation. The sugar was sampled directly from the final unit operation (centrifuge) of the process.

BOX 22**USING NON-NEGATIVITY**

As the parameters of the PARAFAC model reflect concentrations and emission and excitation spectra, non-negativity seems a valid constraint to use. It may be inferred, that non-negativity should not be necessary, since the model should be identified even without using non-negativity. The adequacy of the unconstrained model, however, only holds to the extent that the PARAFAC model is correct for the data. This is not the case here. There is a portion of the data that is missing. Also very likely some of the elements that have not been set to missing may be influenced by Rayleigh scatter to a slight degree. Furthermore heteroscedasticity, quenching and other deviations from the model can cause problems. In short, few data sets can be assumed to follow exactly a mathematical model.

It is preferable that fitting an unconstrained and a constrained model, e.g., to fluorescence data i) give similar results, and ii) that any deviations between the models should be explainable and plausible. Indeed, similar results are obtained by an unconstrained and a non-negativity constrained model. In the sample and excitation modes the loadings of the two models are very correlated (≈ 0.99). The main problem is that some of the emission loading vectors of the unconstrained model have large negative areas in the parts corresponding to Rayleigh scatter, i.e., the area where there are many missing elements. This will also be discussed later.

The sugar was dissolved in un-buffered water (2.25g/15mL) and the solution was measured spectrofluorometrically in a 10 by 10 mm cuvette on a Perkin Elmer LS50 B spectrofluorometer. Raw non-smoothed data was output from the fluorometer. For every sample the emission spectra from 275-560 nm were measured in 0.5 nm intervals (571 wavelengths) at seven excitation wavelengths (230, 240, 255, 290, 305, 325, 340 nm). To the left

in Figure 29 a typical sample is shown.

The data of all the 265 samples can be arranged in an $I \times J \times K$ three-way array of specific size $265 \times 571 \times 7$. The first mode refers to samples, the second to emission wavelengths, and the third to excitation wavelengths. The ijk th element in this array corresponds to the measured emission intensity from sample i , excited at wavelength k , and measured at wavelength j .

Also available were laboratory determinations of the quality of the produced sugar sampled at the same rate. These quality measures are ash content and color. Ash content is determined by conductivity and is a measure of the amount of inorganic impurities in the refined sugar. It is given in percentages. Color is determined as the absorption at 420 nm of a membrane-filtered solution of sugar adjusted to pH 7. The color is given as a unit derived from the absorbance where 45 is the maximum allowed color of standard sugar. It gives an indication of the discoloring of the sugar. This color is by far so low, that it is of no importance for the consumer, but it is of interest for process control and for retailers.

Finally 67 automatically sampled process variables were available of which 10 were sampled so infrequently that they were not included in this investigation. The process variables include temperature, flow, and pH determinations at different points in the process. Typically these variables are noisy and sampled at quite different rates. For this investigation all process measurements have been resampled to the same frequency as the above variables by simply ignoring additional measurements. This quite simple approach is justified, by the fact that the process data are not of primary concern at this stage of the exploratory analysis. Only indications of patterns and relationships which are technologically and chemically explanatory are sought. For similar reasons twenty-two of the 57 process variables were selected in this investigation. This was done to eliminate the irrelevant variables simply by removing those that were almost orthogonal to the ash and color determinations after removal of gross outliers. The twenty-two selected variables are primarily pH measurements from different parts of the process, but also some temperatures, flows and other variables.

A MODEL OF THE FLUORESCENCE DATA

For weak solutions fluorometric data can theoretically be described by a PARAFAC model (see also Box 18 and 22), with the exception that for each sample the measured excitation-emission matrix (size $J \times K$, specifically 571×7) has a part that is systematically missing in the context of the trilinear model (Ewing 1985). Emission is not defined below the excitation wavelength and due to Rayleigh scatter emission slightly above the excitation wavelength does not conform to the trilinear PARAFAC model. As the PARAFAC model only handles regular three-way data it is necessary to set the elements corresponding to the 'non-trilinear' areas to missing, so that the fitted model is not skewed by these data points (Figure 29). It is very important to note, that the elements in this triangular part of the matrix holding the data of each sample can not be replaced with, e.g., zeros. Even though emission well below the excitation wavelength is approximately zero, this part *does not* conform to the trilinear model. Therefore no matter if these data are absent or not, they should be treated as missing. In this case a large part of the data are missing in the emission area from 275 to 340 nm, hence making the model prone to some instability in this area.

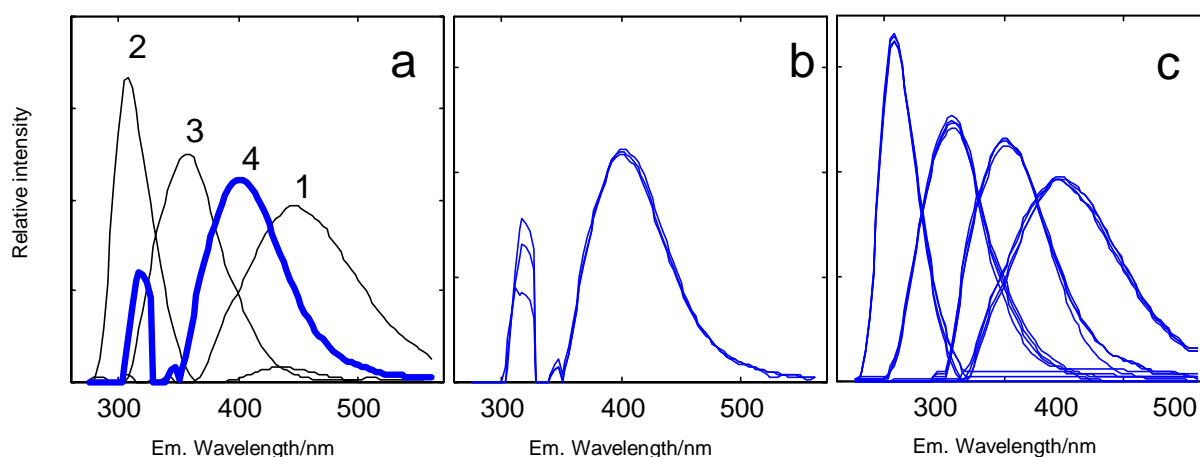


Figure 30. Estimated emission spectra from fluorescence data of sugar samples. a) Four spectra estimated using non-negativity. The 'suspicious' spectrum, four, is marked with a thicker line, b) suspicious spectrum estimated from four different subsets using non-negativity, c) all estimated spectra estimated from different subsets using unimodality as well as non-negativity.

The PARAFAC model is intrinsically unique under mild conditions and the hope with these data is to be able to estimate components that are chemically meaningful, and hence provide a direct connection between the process and the chemical understanding of quality.

The dimension of the PARAFAC model has been judged by split-half analysis. Here only the results for the final complexity will be discussed (see Bro 1998 for details). When fitting a four-component PARAFAC model to the fluorescence data with non-negativity constraints (Box 22) four pseudo-concentration profiles are obtained each with corresponding pseudo-excitation and emission spectra. The components are pseudo spectra and concentration profiles in the sense that they are estimated from the composite fluorescence data but may be estimates of real analytes. In Figure 30a the estimated emission spectra are shown. From visual inspection the spectra seem mainly reasonable, but for spectrum four, the bump slightly above 300 nm seems to be more of a numerical artifact than real. This is plausible since many variables are missing in this area. From 275 to 312 nm only three of the seven excitations are present, hence almost 60% is missing. From 360 nm and above, no variables are missing.

To possibly substantiate the visual judgement a split-half experiment was performed by dividing the samples into four groups as described page 111. Using different sets of samples for fitting the four-component model should give essentially the same estimated emission spectra as the model parameters should not depend on the specific set of samples used as long as the samples span the same population. The resulting model estimates of the problematic emission spectrum are shown in Figure 30b. The area around 300 nm is seen to be unstable in a split-half sense. The estimated parameters in this region change depending on which subset of samples is used for fitting the model, whereas the remaining parameters are more or less insensitive to subset variations. The split-half experiment thus confirms that the area is ill-modeled. The following features all indicate that the 300 nm area is unreliable:

- The parameters are even visually off-the-mark, in the sense that

wavelength-to-wavelength changes are not smooth.

- The split-half experiment shows that the parameters can not be identified in a stable fashion.
- The fact that the data contain many missing values (60%) in the area of the unstable region explains why the instability occurs.

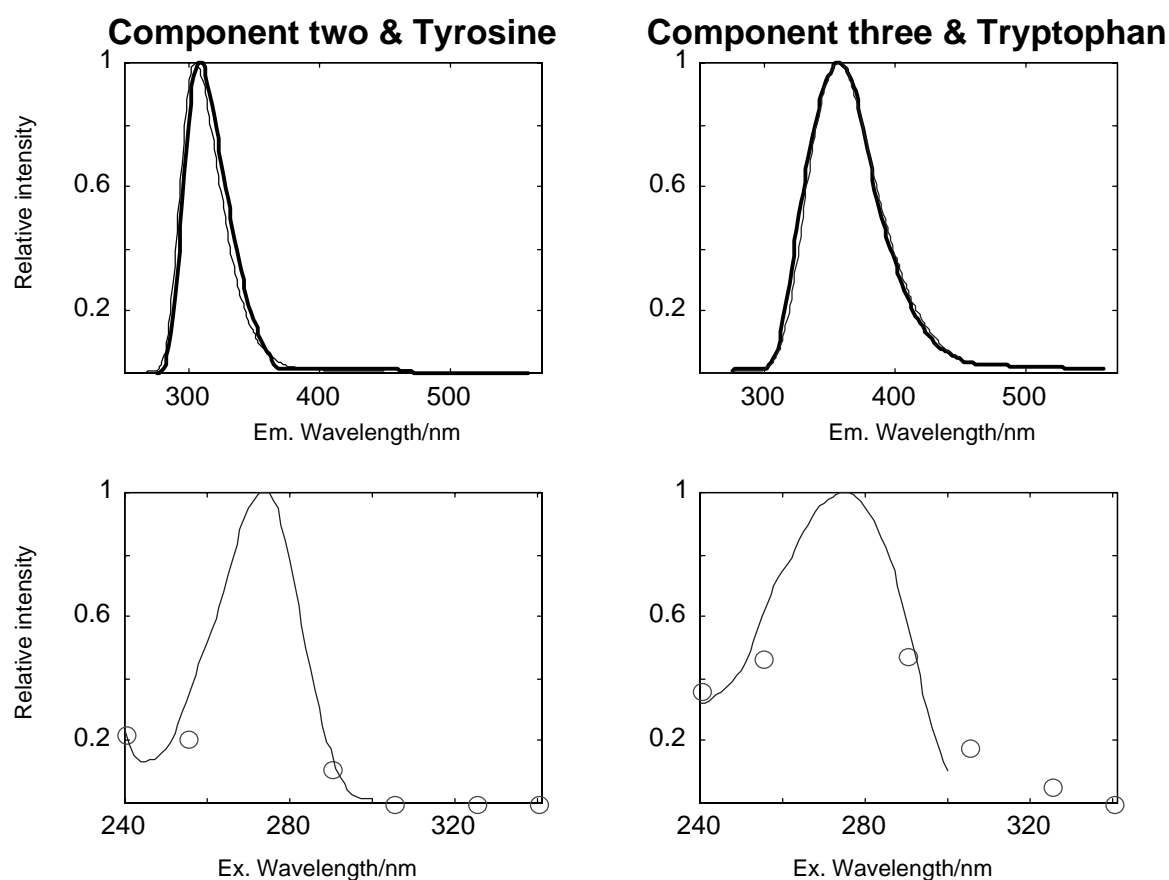


Figure 31. Comparing estimated spectra from sugar samples with selected spectra of pure analytes. PARAFAC emission parameters shown with thick lines (emission)/circles (excitation). Note the good estimates of the emission spectra, while the excitation spectra are less well estimated due to lack of excitation measurements between 255 and 290 nm.

It is important to note that each of the four submodels in the split-half analysis are uniquely estimated. However, the uniqueness of the model is governed partially by random variation and hence not attributable to the real underlying systematic variation. Therefore the solution for a given subset

of data does not generalize to other sets data. The question then is what to do? As the most probable cause for the problem is that too few excitation wavelengths have been used (seven), the best thing to do would probably be to remeasure the samples using more excitation wavelengths. However, the measurements as they are currently being performed require a substantial amount of work, and remeasuring is therefore not realistic. For future samples more excitation wavelengths may be used, but for these data the only possibility is to remedy the artifact by means of data processing. Several aspects indicate that the spectrum should be unimodal:

- The spectrum *is* unimodal apart from the unstable part
- The remaining estimated emission spectra are almost unimodal
- The most likely fluorophores in sugar (amino-acids, simple phenols, and derivates) have unimodal emission spectra
- The Kasha rule (Verhoeven 1996) states that a fluorophore will emit light under the same (S_1 - S_0) transition regardless of excitation, i.e, an excited molecule will drop to the lowest vibrational level through radiationless energy transfer, and then from the excited singlet level S_1 return to the ground state S_0 by fluorescence (Ewing 1985). Even though there are exceptions to this rule, it often holds especially for simple molecules. The fact that the emission occurs from the same transition, mostly implies that the corresponding emission spectrum will be unimodal

The above reasoning led to specifying a new model where all emission spectra were estimated under unimodality and non-negativity constraints and remaining parameters under non-negativity constraints. The fitted model was stable in a split-half sense (Figure 30c) and interestingly the estimated excitation spectra and relative concentrations did not change much from that of the non-negativity constrained model. This confirms that the cause of the artifact in Figure 30a is mainly due to the amount of missing data in the specific region. It means, that unimodality is probably a valid constraint, and it also implies, that unimodality is mainly necessary for improving the visual appearance of the emission loadings, hence enabling better identification of the underlying analytes.

Selected estimated spectra are shown in Figure 31 together with the

emission and excitation spectra of tyrosine and tryptophan, two substances of known technological importance. The spectra of tyrosine and tryptophan were acquired in experiments unrelated to this study. Still, the similarity confirms that the PARAFAC model is capturing chemical information.

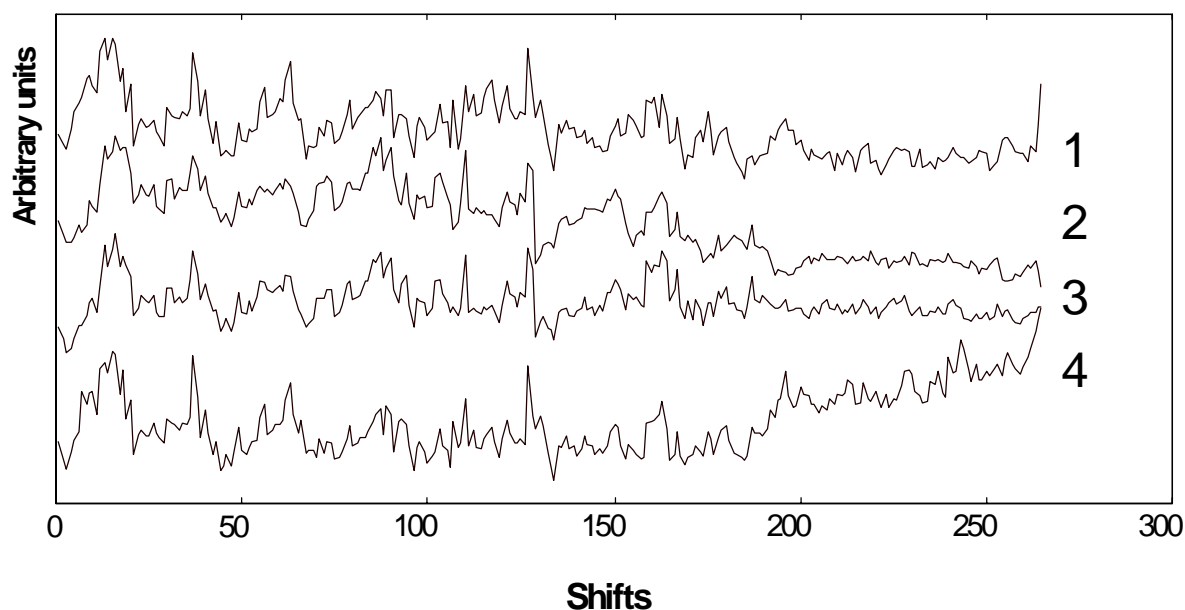


Figure 32. The scores of the four-component PARAFAC model of the sugar fluorescence data. The score vectors have been shifted vertically for visual appearance. As the samples are ordered according to time, the line plots represents the variation of each component over the campaign, each sample representing an eight hour period. Component two is the tyrosine-like and component three the tryptophan-like component (see figure 31).

The scores of the model of the fluorescence data are estimates of relative concentrations (Figure 32), or in a more general setting a window into the chemistry of the sugar beets. In the score plot several interesting features are seen. All four scores seem to follow the same overall pattern. The first half of the campaign they vary with the same low frequency. This frequency follows the weeks – six weeks, six periods. From shift number 150 to 200 the variation is more modest and in the final period from shift 200 only minor variations are seen with the exception that component four increases steadily. These distinct patterns of variation must be reflecting a variation in the chemistry of the sugar during the campaign. A preliminary hypothesis

– to be investigated – that may explain these variations is based on the following observations. The beets are stored before entering the factory. The storage time differs, and there is a pile up of beets during the weekend. During storage a significant increase in temperature is likely to occur possibly leading to increased enzymatic activity which can then be reflected in the weekly patterns of the fluorescence scores seen in the first half of the campaign. In this part of the campaign the weather was relatively warm and all scores follow the same overall pattern.

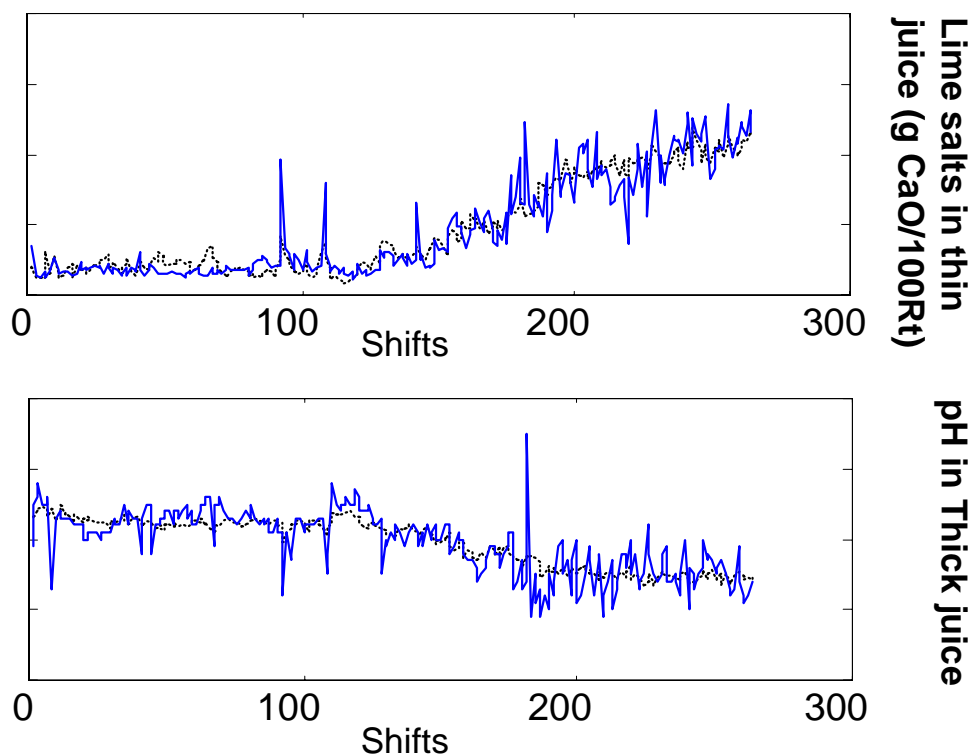


Figure 33. Predictions of selected process parameters from the four PARAFAC scores of the fluorescence data model. Unbroken lines are reference values. Notice the smoothing effect of the predictions due to the combined effect of noisy process measurements and the sugar samples being average samples collected over eight hours.

This also explains why the variations generally decrease in time as the outdoor temperature influencing the stored beets steadily decreases during the campaign. The increase in the amount of compound four from shift 200 (15. November) seems to be correlated with the onset of the frost according

to the process records, hence again a temperature phenomenon. During this time the variation in component four is highly correlated to color (Munck et al. 1998). To the extent that these provisional ideas and hypotheses are correct they indicate that controlling the temperature of the incoming beets is *the* most important factor for maintaining a well-controlled process on a chemical level. To the extent that the chemical information from fluorescence carries information on other process parameters, this conclusion carries over to the process quality as well.

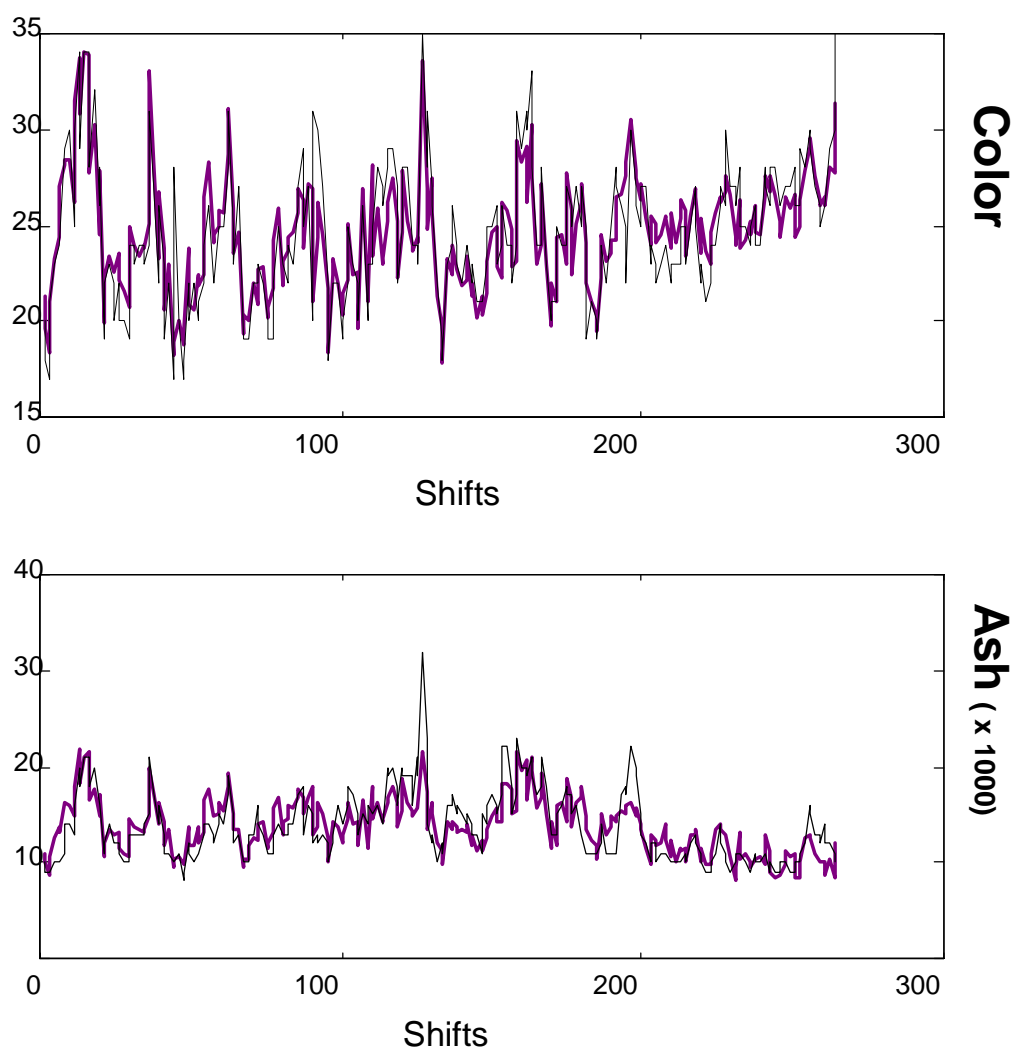


Figure 34. Multiple linear regression predictions of color and ash from PARAFAC scores. Thin lines are reference values.

USING PARAFAC SCORES FOR MODELING PROCESS PARAMETERS AND QUALITY

In order to see whether there is any connection between the variation in the chemistry of the sugar as given by the four PARAFAC scores, the variation in the quality parameters of sugar, and the process variables, several models were investigated. In the sequel only the scores of the PARAFAC model will be used in the models investigated. Even though it may be feasible to use the raw fluorescence data directly, the idea here is to explore whether the chemical representation of the fluorescence data, i.e., the PARAFAC model, can be related to process and laboratory data.

Initially the correlation between the PARAFAC scores and the process variables was investigated. For some process variables there were almost no correlation, but for a large number of process variables excellent correlations were obtained. For examples of this, see Figure 33. Here the fitted values obtained using multiple linear regression (MLR) and only paying attention to the fitted data are shown. MLR was chosen because the condition number of the matrix of independent variables (265 samples \times 4 PARAFAC scores) is low, hence no problems arising from collinearity is expected. Secondly, because the aim here is not to establish the exact predictability little attention was paid to assessing predictability in terms of cross-validation etc. Instead the statistics of the MLR models were observed in order not to obtain misleading results.

Multiple linear regression models were also made for predicting the quality parameters ash content and color from PARAFAC scores. The models for predicting ash and color of the sugar were excellent. The predicted values are shown together with the reference values in Figure 34. Though no cross- or test set validation has been performed, the prediction models are only based on *four* regression coefficients each. With these models it is confirmed, that it is possible to use fluorescence for on-line or at-line monitoring of sugar quality. This is important as currently these parameters are only determined every eighth hour and with a certain lag as the laboratory analysis takes some time. Furthermore, by scrutinizing the calibration models important clues to the chemical background for the variations in color and ash can be obtained. For example, component one ($r = 0.60$) and four ($r = 0.81$) seems to be correlated to color (Munck et al. 1998). Only component four, as well as color, seems to be influenced by

the increasing amount of frozen beets during the last part of the campaign. On the other hand, component one ($r = 0.69$) and the tryptophan-like component three ($r = 0.64$) show some correlation with ash. The tyrosine-like component two does not correlate significantly with neither ash nor color. Thus, the four PARAFAC components show different patterns in connection with the two important quality parameters color and ash.

It is interesting to compare the predictions of ash and color from fluorescence data with predictions obtained using the process variables. Calibration models were constructed for ash and color separately using more thorough validation than above. The following three sets of independent variables were tested: process data²¹ of size 265×22 , the fluorescence data given by the four score vectors of the PARAFAC model (265×4), or both (265×26). The purpose of using the PARAFAC scores instead of using the fluorescence data directly is to show that it is possible to use the chemically meaningful PARAFAC components for process control and prediction, rather than obtaining a model with abstract mathematically derived latent variables from the raw fluorescence data. Thereby a direct connection between the chemistry of the sugar processing and the quality and process parameters is obtained.

The independent variables were autoscaled and lagged twice (one and three lags were also tried), thereby giving a three-way array of size $265 \times 22 \times 3$ in case of the process data.

For each of the above settings a model was fitted using either the first 50 or 150 samples. Recall, that the samples are obtained contiguously every eighth hour so that, e.g., 50 samples correspond to approximately 17 days. The model used for calculating the calibration model was N-PLS (unfold-PLS was also tried giving similar predictions though more complicated models).

For, e.g., the model of ash predicted from process data using the first 50 samples the size of the independent calibration data are thus 50 (samples) $\times 22$ (variables) $\times 3$ (lag mode). The first two samples were excluded as two thirds of the data elements were missing due to lagging. The

²¹. It was also tried to smooth the process variables with both median filters and wavelets to remove spikes, but this had no significant influence on the results.

number of latent variables was determined by minimum cross-validation error and the model then used for predicting the remaining left-out samples.

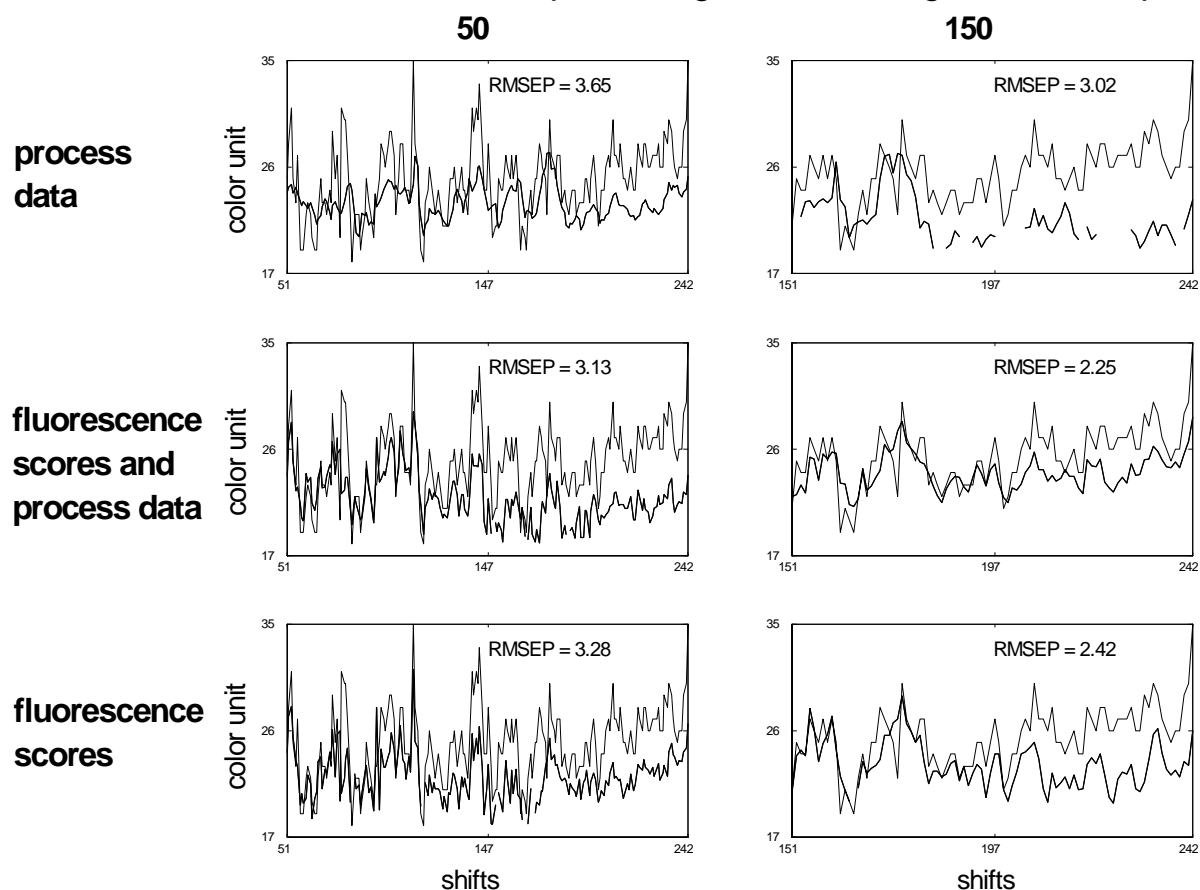


Figure 35. Predictions of color using N-PLS of lagged data. Reference values are shown with thin lines and predictions with thick lines. Top two figures show the predictions using only the process data as independent variables; middle figures show the predictions using both process variables and fluorescence PARAFAC scores as independent variables, and the lower ones show the predictions using only the fluorescence PARAFAC scores as independent variables. The leftmost three figures give the results using the first 50 samples and predicting the remaining ones. The rightmost give the predictions using the first 150 samples and predicting the rest. Above in each figure the root mean squared error of predicting the first 30 (10 days) samples is shown. Predictions outside the ordinate scale have been removed for consistency.

For the calibration models using the first 50 samples, this means that the models are based on the first half month of the campaign and tested on the last two and a half month. It is worth mentioning that the models are suboptimal with respect to variable selection and lagging. More ingenious variable selection and more in-depth analysis of the variation in time of the variables may lead to better models. However, as stated before, the goal is not to find *the* model, but to explore the relevance in and the patterns of the variations in the data. In essence this is an example of a first shot analysis. Depending on the relevance of the results, further experiments, data, and analyses can lead to better models.

The results of the predictions are shown in Figure 35 for the color determinations. The ash determinations were similar. All models are capable of predicting ash and color several days after the models have been fitted. Up to 50 days in some cases. For ash there is little difference in the quality of the predictions obtained from the process data and the fluorescence data, while for color it seems that the fluorescence data provide more information than the process data. This is very clearly seen from the quality of the predictions in the first fifty days ahead. It suggests that the fluorescence data not only give information already present in the process data, but supplements the process data with extra chemical information important for predicting the color of sugar. This is expectable and illustrates that spectroscopic techniques provide a general source of information in process control and quality monitoring.

CONCLUSION

The models described in this application are quite extraordinary. They give a direct connection between the state of the process, the product (from a scientific point of view the chemistry of the sugar), and the quality (as defined by laboratory measurements defining the internal as well as the external consumer quality). As such, the conceptual idea behind the results reaches far beyond the specific data treated here. It provides means for combining process analytical chemistry and multivariate statistical process control. This is possible through the use of the unique PARAFAC model in conjunction with using fluorescence data as a screening analysis.

The refined sugar can be considered a data-logger, in that it contains

hidden information on the history of both the raw-material – the beets – and the processing of the beets. The fluorescence of the sugar gives a window into this information. The loadings provide means to identify the PARAFAC components as specific chemical compounds as indicated above, and the scores reveal the variation in time of these components.

In this work the trust that the fluorescence analysis may work is the primary hypothesis. By data and model selection (Munck et al. 1998) four chemically identifiable components are found. Together they give highly representative information regarding both sugar quality and process parameters. Thus four valid indicator substances for quality and process parameters are identified in this preliminary screening. The results have to be checked and followed up in several product seasons if one wants to develop a process control system based on fluorescence.

That it is possible to capture and identify specific chemical variation is somewhat surprising considering the fact that the samples are simply taken directly from the process and dissolved in water. Further, the sample matrix is very complex. Approximately 99.999% of the sugar is sucrose, which is not fluorescent. It is the very small fraction of impurities such as amino acids, phenols and their reaction products that are detected by the fluorometer. Hence it is the complex mixture of a very small amount of natural and process related components that are being measured with the sensitive fluorescence method.

When more certain conclusions have been drawn it becomes relevant from an academic and technological point of view to identify what the estimated fluorescence spectra represent with more confidence using, e.g., standard addition or chromatography. In this work this has only been hinted at, since there is little sense in spending too much efforts in elucidating the chemical background until the relevance and usefulness of the models are established. This is one of the key benefits of using exploratory analysis. It has been established that the variations in the fluorescence data are closely correlated to the variations in the quality of the sugar as well as important process parameters. Coupling this with the results from studying the fluorescence data alone, this could indicate that by controlling the temperature of the beet stores more precisely it may be possible to avoid large fluctuations in the sugar quality.

The predictive models obtained from using the process variables in conjunction with the chemical information are important and should be further elaborated on, when new data are available. In the PARAFAC models there are signs of more components in the data. It is also plausible from a chemical point of view, that more fluorophores may be represented. However, it is not possible with the given data to estimate more components reliably. In the near future samples from the 1996 campaign will be measured using more excitation wavelengths, in the hope, that more components can be extracted.

Citing Munck et al. (1998): "*At the roots of science lies observation and data collection from the world as is and from which conclusions can be induced after classification. This is far from the present theory-driven deductive, normative stage of science which depends heavily on modeling discrete functional factors in laboratory experiments and suppresses the aspect of interaction. ... In traditional chemical analysis one starts by defining the hundreds of chemical substances involved in a process, as was done for the sugar industry by Madsen et al (1978). If the target hypothesis is to find easily identifiable indicator substances by which to model quality and process characteristics, we suggest that our exploratory, inductive method by introducing a multivariate screening method in the global area of the sugar factory would be more economical than a normative, deductive strategy based only on a priori chemical knowledge, chromatography and classical statistics as studied in the local area - the research laboratory.*"

7.6 ENZYMATIC ACTIVITY

PROBLEM

A major contributor to undesirable browning effects in fruit and vegetables is the enzymatic browning caused by PPO, polyphenol oxidase (Martinez & Whitaker 1995). Enzymatic browning can henceforth be expressed by the dioxygen consumption of PPO, as this is directly related to the activity of PPO. To avoid enzymatic browning of fruits and vegetables it is important to store them under conditions that suppress the enzymatic activity.

The activity of PPO was investigated as a function of different levels of O₂, CO₂, temperature, pH, and substrate according to a factorial design. A

traditional ANOVA model of the data did not shed much light on the relation between these factors and the activity, while a multiplicative model based on GEMANOVA/PARAFAC (page 192) was easy to interpret²².

DATA

The PPO used was obtained from fresh iceberg lettuce and extracted and purified according to Heimdal et al. (1997). PPO activity was measured in nanomoles of O₂ consumed per second by a polarographic polyphenol oxidase assay as described in Heimdal et al. (1994). For five O₂ levels, three CO₂ levels, three substrate types, three pH values and three different temperatures – all varied independently – the activity of PPO was determined in replicate. The substrates used were 0.01M chlorogenic acid (CG), 0.01 M epicatechin (EPI) and an equimolar mixture of both (MIX), where MIX = 0.005M CG + 0.005M EPI, hence the substrate concentration is also 0.01 M.

Table 9. Experimental design for the experiment.

Factor	Code	Levels
O ₂ /%	O	0, 5, 10, 20, 80
CO ₂ /%	C	0, 10, 20
Substrate	S	CG, EPI, MIX
pH	P	3.0, 4.5, 6.0
Temperature /°C	T	5, 20, 30

Building a calibration model to predict the activity from the experimental conditions can give important information on how the PPO activity - and therefore the color formation - is influenced by the different factors. The different levels of the factors are shown in Table 9. The number of samples

²². The problem presented here is part of an investigation conducted by Hanne Heimdal and described in detail in Bro & Heimdal (1996) and Heimdal et al. (1997).

in the replicated full factorial design is $5 \times 3 \times 3 \times 3 \times 3 \times 2 = 810$. The data constitute a full five-factor factorial design, but in the PARAFAC/GEM-ANOVA model, the data are interpreted as a multi-way array of activities, specifically a five-way array. The five different modes are: O_2 (dimension five), CO_2 (dimension three), pH (dimension three), temperature (dimension three), and substrate type (dimension three). The $ijklmth$ element of the five-way array contains the activity at the i th O_2 level, the j th CO_2 level, the k th level of pH, the l th level of temperature, for the m th substrate type. The five-way array is depicted in Figure 36.

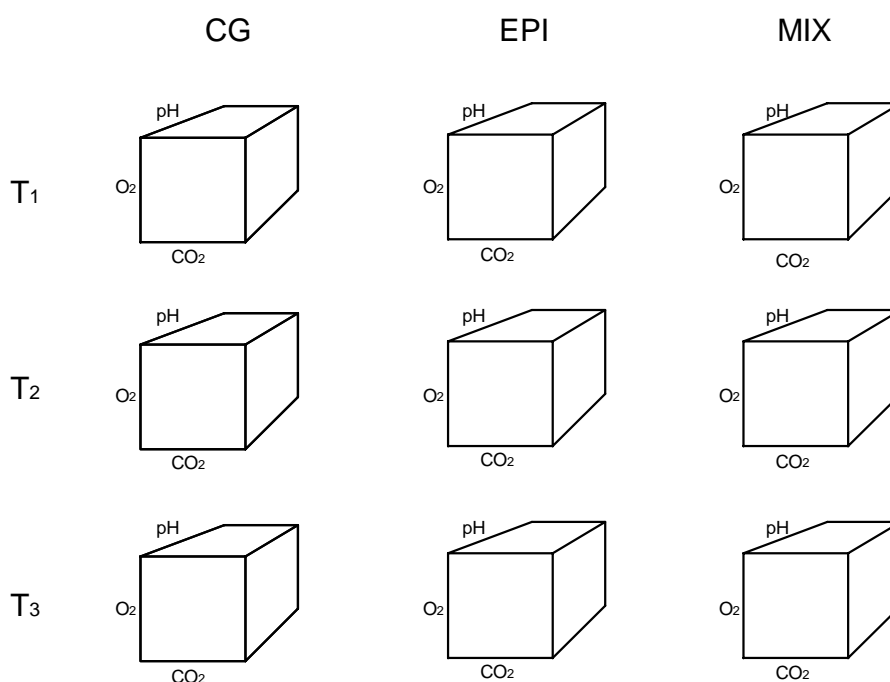


Figure 36. A graphical representation of the five-way array of enzymatic activities.

RESULTS

To choose the model, i.e., the number of components, GEMANOVA models (page 192) were fitted using half the data as calibration set and half as a test set. The predictions of activity from each model given by the loadings **O**, **C**, **S**, **P** and **T** were compared to the test set activities. The number of components, F , was chosen to minimize the predicted residual

sum of squares, PRESS, calculated as

$$\text{PRESS} = \left[\left(\sum_{f=1}^F o_{if} c_{jf} s_{kf} p_{lf} t_{mf} \right) - \text{act}_{ijklm} \right]^2 \quad (265)$$

act_{ijklm} being the $ijklm$ th element/activity of the replicate set not used to build the model. One component with no fixed elements gave the lowest prediction error, which furthermore was in the neighborhood of the intrinsic error of the reference value. The resulting GEMANOVA model is thus very simple, namely a one-component PARAFAC model, which is equivalent to a five-way multiplicative ANOVA model

$$\text{act}_{ijklm} = o_i c_j s_k p_l t_m + e_{ijklm} \quad (266)$$

as compared for example to a traditional ANOVA model which after pruning of insignificant terms reads

$$\begin{aligned} \log(\text{act}_{ijklm}) = & \\ & b_0 + b_1 x^{o,i} + b_2 x^{s,k} + b_3 x^{p,l} + b_4 x^{t,m} + b_5 (x^{o,i})^2 + \\ & b_6 (x^{p,l})^2 + b_7 x^{o,i} x^{p,l} + b_8 x^{s,k} x^{t,m} + e_{ijklm} \end{aligned} \quad (267)$$

where $x^{o,i}$ means the i th setting of oxygen (scaled appropriately). The PARAFAC model is given by the five loading vectors depicted in Figure 37. The loadings are interpreted in the following way. For given settings of the factors simply read the corresponding five loading elements on the respective plots and multiply these five numbers. The product is the estimated PPO activity.

To determine how to obtain low PPO activity, hence low enzymatic browning, it is clear from the figures, that the setting for each factor should be the one with the lowest accompanying loading. As the model is multiplicative this will be the setting yielding the lowest enzymatic activity. The main conclusion derived from the model is therefore that by keeping temperature, oxygen, and pH as low as technologically possible the enzymatic browning will be minimized. The effect of CO_2 is small, but

ignoring it leads to a model with significantly poorer predictability. The consequence of this is elaborated on in Heimdal et al. (1997).

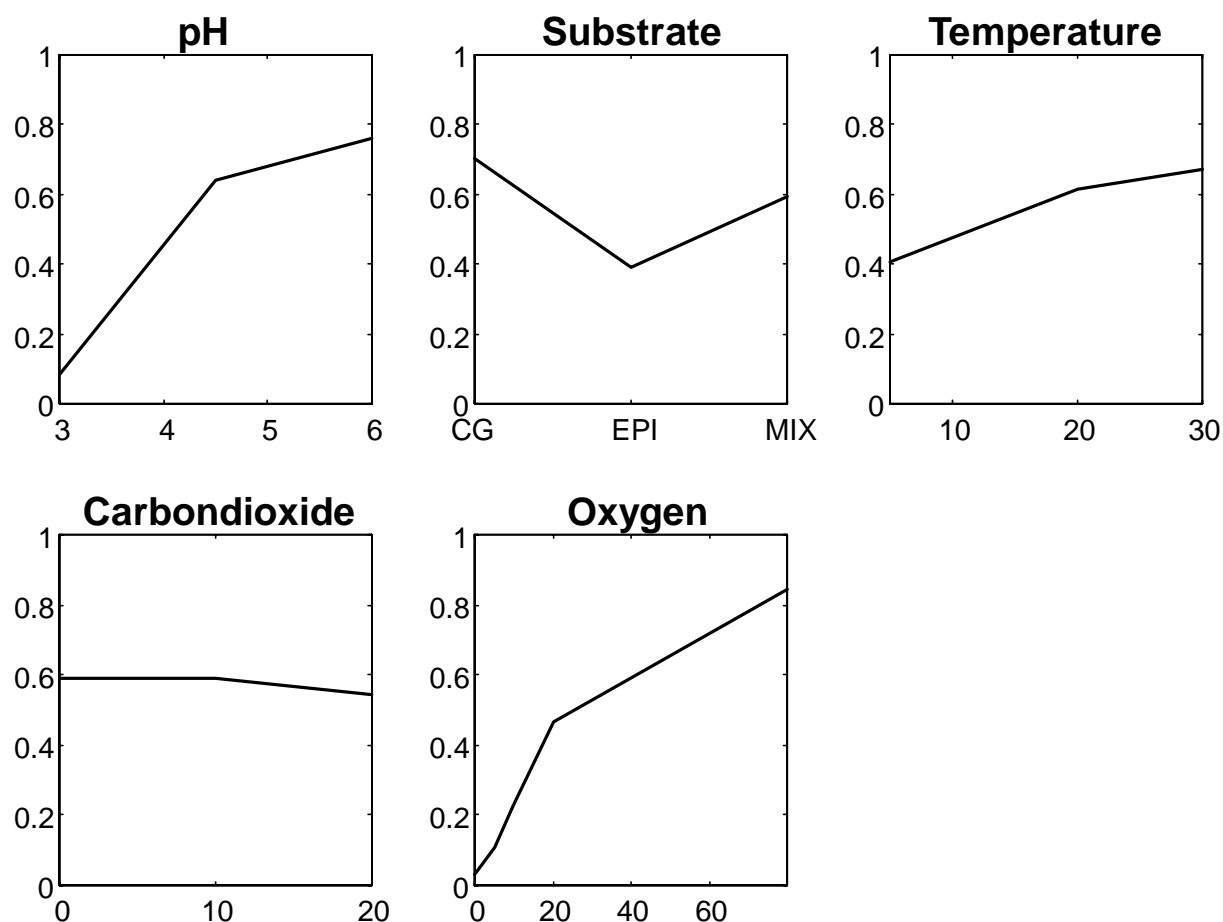


Figure 37. Loading vectors of the one-component/one effect GEMANOVA model of the enzymatic activity data.

The multiplicative model could also have been obtained by traditional ANOVA by using the logarithm of activity *and* modeling all factors as qualitative variables. However, even though this would give the same structural model, the loss function would not be the same, and the corresponding model would be poorer with respect to predicting activity. Specifically the root mean squared error of predicting one replicate set from a model built from the other replicate set is 18.0 using this approach whereas it is only 14.3 using PARAFAC/GEMANOVA. A traditional ANOVA model based on a logarithmic transform of activity and treating all factors

except substrate as quantitative factors is also possible. Though such a model give almost as good predictions as the GEMANOVA model, it contains more effects, and is somewhat more difficult to interpret. To compare the two different models both models were used to predict the activities of the test set samples. The resulting predictions are shown in Figure 38.

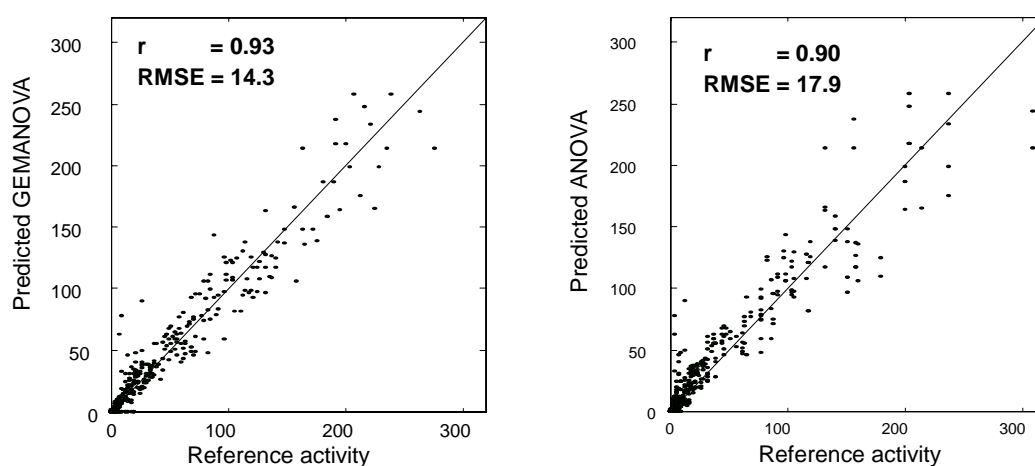


Figure 38. Predictions of independent test set obtained from GEMANOVA (left) and ANOVA (right).

CONCLUSION

For the data treated here a simple one-component PARAFAC model was sufficient to obtain a good model. In that respect this is not a good example on the flexibility of the GEMANOVA model. However, it illustrates that it is sometimes worthwhile using other than standard ANOVA models. The important aspect of the present GEMANOVA model is that it is easy to understand, because the structural model is a plausible basis for describing the data. There may be many possible ways to model the data. In fact for these data it is possible to make an excellent model using a traditional ANOVA model with no logarithmic transformation of the activity. Such a model is hard to interpret, however, since purely additive effects of the factors make little sense for enzymatic activity. Thus the statistical significance alone is not sufficient for validating a model. Even though,

there are many ways to describe data mathematically, the model must also be in line with the nature of the data modeled in order to be useful.

7.7 MODELING CHROMATOGRAPHIC RETENTION TIME SHIFTS

PROBLEM

In order to understand the chemistry of the color formation during sugar processing an experiment was conducted to explore the presence and amount of chemical analytes in thick juice, an intermediate product in the sugar production. The molecular entities of thick juice samples were separated by size on a chromatographic system and detected by fluorescence in the hope that the individual fluorophores could be separated and detected.

The only aspect considered here, is the problem of modeling chromatographic data with retention time shifts. It is not uncommon that analysis of chromatographic data is hampered by retention time shifts. Retention time shifts causes the chromatographic profiles of specific analytes to be dissimilar from run to run. This is problematic because it severely prevents or degrades the application of multilinear models. The data given here are ideally suited for showing that using PARAFAC2 it is possible to maintain the uniqueness and robustness of multi-way models with certain types of shifted data.

DATA

Fifteen samples of thick juice from different sugar factories were introduced into a sephadex G25 low pressure chromatographic system using a $\text{NH}_4\text{Cl}/\text{NH}_3$ buffer (pH 9.00) as carrier. In this way the high molecular reaction products between reducing sugar and amino acids/phenols are separated from the low molecular free amino acids and phenols. The high molecular substances elute first followed by the low molecular species. Aromatic components are retarded the most. The sample size was 300 μL and a flow of 0.4 mL/min was used. Twenty-eight discrete fractions were sampled and measured spectrofluorometrically on a Perkin Elmer LS50 B spectrofluorometer.

The column was a 20 cm long glass cylinder with an inner radius of 10

mm packed with Sephadex G25-fine gel. The water used was doubly-ion exchanged and milli-pore filtrated upon degassing. The excitation-emission matrices were collected using a standard 10 mm by 10 mm quartz cuvette, scanning at 1500 nm/min with 10 nm slit widths in both excitation and emission monochromators (250 - 440 nm excitation, 250 - 560 nm emission). The size of the four-way data set is 28 (fractions) \times 20 (excitation) \times 78 (emission) \times 15 (samples).

RESULTS

These chromatographic data are four-way. Ideally, they are quadrilinear, the components of the modes corresponding to time profiles (28), excitation spectra (20), emission spectra (78), and sample concentration profiles (15). Hence a four-way PARAFAC model should be capable of uniquely and meaningfully describing the variation. In this case, a four-component model seems adequate.

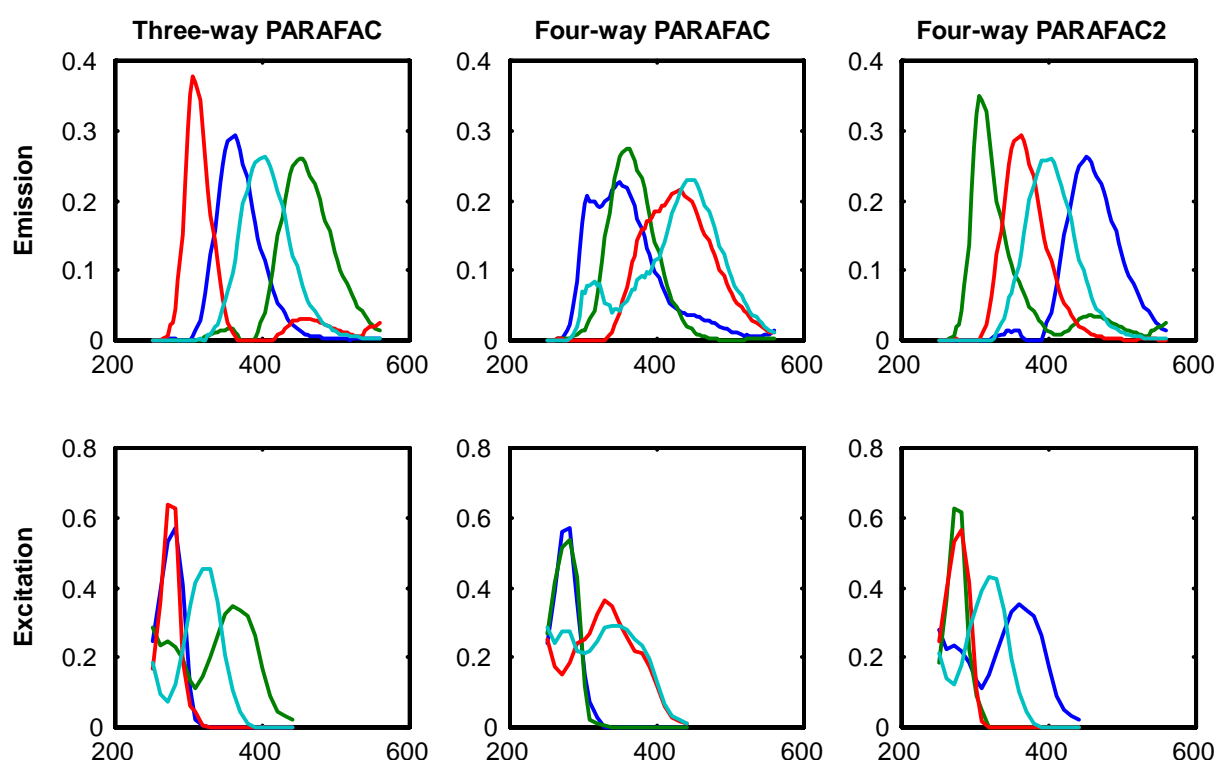


Figure 39. Estimated emission (top) and excitation (bottom) spectra from three-way PARAFAC with sample and elution mode combined (left), four-way PARAFAC (middle), and four-way PARAFAC2 (right).

In Figure 39 the excitation and emission mode loadings of a four-component non-negativity constrained model is shown. For the PARAFAC2 model non-negativity was not imposed in the elution profile mode for practical reasons. Apparently the parameters are reasonable. Hypothesizing that the model is correct and valid, it must hold that a four-way PARAFAC model is also valid, since it is equivalent to the PARAFAC model expect that it puts less restrictions on the variation in elution mode. In Figure 39 the excitation and emission mode loadings of a non-negativity constrained PARAFAC2 model is shown. These also look reasonable but different from the PARAFAC loadings especially in the emission mode. The PARAFAC2 model seems to be slightly better as judged from the smoothness of the loadings. One may also, though, anticipate that the models differ due to that the PARAFAC2 model is more influenced by noise and model errors, since it imposes less structure than the PARAFAC model. However, for these data, a very simple way of validating which model is better admits itself. Often, chromatographic data are at most three-way, since there is often only one spectral mode. The fact that these data are four-way means that even if the data are unfolded to a three-way structure a PARAFAC model of such three-way data is unique. Therefore, the sample and elution modes may be confounded and the subsequent three-way array be uniquely modeled by a three-way PARAFAC model. Since each elution mode will then be modeled separately for each sample, no problems arising from possible retention time shifts will affect the model. This means, that these data provide an extraordinary simple, exceptional and very elegant way to validate the four-way models. Namely to compare the two candidate solutions with the three-way PARAFAC model. Note, that this is not possible for the more frequently occurring three-way data, since unfolding would lead to two-way data that can not be uniquely modeled in general.

For the three-way data the excitation and emission mode loadings are shown in Figure 39. Note the extreme closeness of the three-way PARAFAC and four-way PARAFAC2 solution. Hence, the PARAFAC2 model gives an adequate four-way description of the data, while the non-smooth and less appealing PARAFAC loadings can only be explained by the model being too strict and inappropriate.

From the three-way model of the data a set of loadings is also obtained in the combined elution/sample mode. Reshaping the loading for one specific component to a matrix a set of elution profiles for this 'analyte' is obtained; one for each sample. In Figure 40 this is shown for component one.

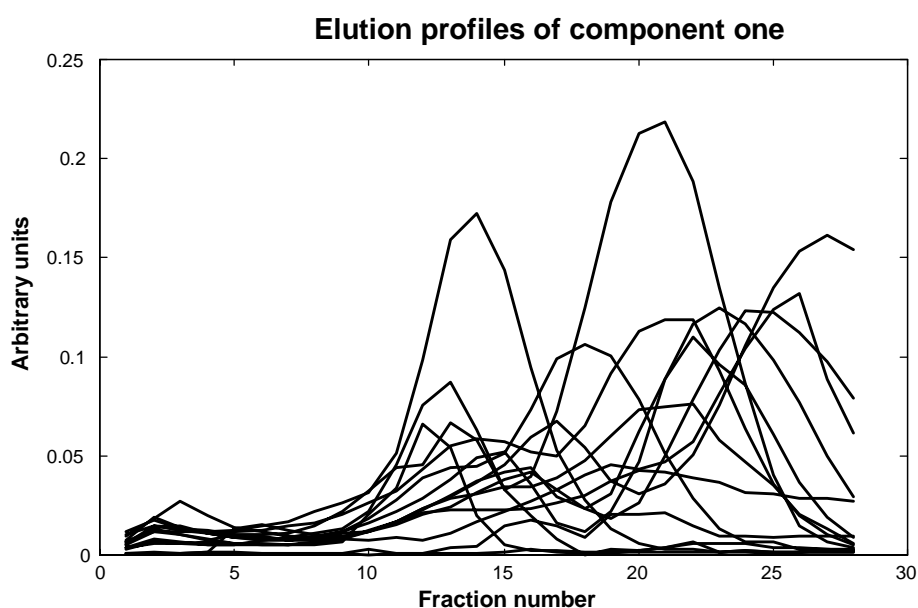


Figure 40. Estimated elution profiles of component one. Estimated from a three-way PARAFAC model. Each line is the estimated profile of the component in one specific sample.

It is readily seen that even though the elution profiles should be identical in each run, this is certainly not the case. There are huge shifts in the retention times from sample to sample, probably caused by the very different contents of the samples. The gel in the column is known to be sensitive towards the concentration of phenolic compounds and certain amino acids. Thus, the inter-sample variation in the elution profiles is probably due to different contents of such compounds with high affinity for the chosen gel. Obviously, using four-way PARAFAC in this case, it is expectable that a valid model can not be obtained.

CONCLUSION

In this application, a suggestion has been given for the solution of a very

important and frequently arising problem, namely shifted data. It has been shown that even though the data are severely shifted PARAFAC2 apparently is capable of modeling the data. In this case validation could be very elegantly performed by unfolding the data to a three-way structure, but mostly shifted chromatographic data are at most three-way and it is therefore essential to maintain the order of the data in the model if uniqueness is to be guaranteed.

The general problem solved in PARAFAC is the ideal model

$$\mathbf{X} = \mathbf{A}(\mathbf{C} \otimes \mathbf{B})^T \quad (268)$$

but for shifted data the loadings of one mode, e.g. the first, will not be the same at every occasion k . Thus a model of shifted data may generically be stated

$$\mathbf{X} = \mathbf{A}_k(\mathbf{C} \otimes \mathbf{B})^T. \quad (269)$$

It is the choice of the structure of \mathbf{A}_k that determines the structure of the model. Imposing no structure, will result in an unfolded model while in PARAFAC2 the constraint is imposed that

$$\mathbf{A}_k = \mathbf{P}_k \mathbf{A}, \quad k = 1, \dots, K \quad (270)$$

where \mathbf{P}_k is an orthogonal rotation matrix and \mathbf{A} can be interpreted as a kind of basic set of profiles. Thus, the flexibility offered by PARAFAC2 can to some extent be compared with a Procrustes analysis. One may of course also envision other ways of imposing structure in \mathbf{A}_k but it seems that the rotational freedom provided by the PARAFAC2 model is adequate for approximating many occurring deviations from the strict linearity required in the standard PARAFAC model. Importantly, the PARAFAC2 model has the advantage of intrinsic structural uniqueness.

PARAFAC2 also has another distinguishing feature, which has not been touched upon in this application. Note that in equation 270 \mathbf{A}_k is of size $I \times F$, \mathbf{P}_k is of size $I \times F$ and \mathbf{A} is of size $F \times F$. There is no hindrance that the dimension of the first mode differs from slab to slab, hence each slab k has

its own specific first mode dimension l_k . For process data where, e.g., different batches run for different times or in free choice profiling in sensory analysis, slabs are obtained that can not meaningfully be arranged in a three-way table, but where some relation is bound to be occurring in the first mode anyway. The PARAFAC2 model seems to be a very promising candidate for modeling such data in a meaningful way.

CHAPTER 8

CONCLUSION

8.1 CONCLUSION

An important part of this thesis has been to describe multi-way analysis and develop it into a useful tool for chemometricians. A number of theoretical aspects of multi-way analysis have been explained, exploited, and extended. The major contributions are:

- Introducing the Khatri-Rao product which enables more transparent notation of especially higher-order PARAFAC models (page 20)
- Extending PLS to higher orders (page 51)
- Proposing the PARAFAC2 model for solving problems with shifted data as well as data where different slabs have different dimensionality (process data, longitudinal data, free choice profiling data etc.) (page 33)
- Proposing the PARATUCK2 model for modeling rank-deficient data. Unlike other related models, this model has been shown to have some intrinsic uniqueness properties due to its relationship to the PARAFAC model (page 37)
- Extending PARAFAC2 and PARATUCK2 to higher orders (page 68 and 71)
- Suggesting the GEMANOVA model for multiplicative ANOVA models (page 192)
- Optimal compression of arrays prior to model estimation (page 88)

- Core consistency diagnostic for determining model complexity (page 113)
- Describing non-negativity constraints in compressed spaces (page 149)
- Using approximate orthogonality for speeding up algorithms and avoiding local minima and swamps (page 154)
- Providing a fast non-negativity constrained least squares algorithm (page 169)
- Providing a fast least squares unimodality constrained algorithm (page 177)
- Providing smoothness constraints for ALS algorithms (page 152)

The most important conclusion from the work reported here is, that multi-way analysis combined with the use of constraints is feasible in chemistry and related areas. A number of applications of different origin and nature have shown that multi-way analysis can give good models which are easy to use and interpret. Improved alternatives to existing model have been demonstrated and in some cases new models have been developed which could not otherwise have been obtained.

An interesting aspect shown in the applications is that it is not necessary to be reluctant in the use of constraints. *If* these are appropriate and well-motivated in the context they will improve the model. This also points to the fact that there is no such thing as *the* model, i.e., a model that can be optimal in any situation. For a given data set *and* context choosing an appropriate model is a matter of exploring the mathematical universe, selecting the most appropriate mathematical representation of the data on the conditions of the context. Guidelines for how to perform such a selection without risking spurious results from chance correlations have been given. In fact, it has been shown that multi-way analysis in itself provides a guard against chance correlations due to the structure imposed on the model. The multi-way structure acts like a strong noise filter by requiring the structural part of the data to be of a very specific nature.

Multi-way analysis and the use of constraints provide mathematical tools which have amply been shown to be beneficial in many problems treated

differently today. Even though data are not intrinsically multi-linear the use of multi-linear components is often a more sensible approach than using unfolding techniques, because the information lost by over-simplifying is smaller than the variability introduced by not imposing structure.

The advantage of multi-way analysis can be summarized in two aspects. Multi-way models provide uniqueness and better structural models. Uniqueness makes it possible to do mathematical chromatography (pure spectra) and do second-order calibration, i.e., calibrate in presence of unknown interferences. Better structural models increase the robustness and increase the noise reduction, provide simpler models using fewer parameters, give more interpretable models because of parsimony and correspondence between the nature of the data and the model, and give better predictions in many cases.

Specific applications illustrating the above have been discussed:

- Qualitative and quantitative comparison of multi-way and unfolding techniques for noisy sensory data illustrating the benefits of PARAFAC and N-PLS in particular (page 196)
- Quantitative comparison of multi-way and unfolding calibration models using fluorescence data, showing that nothing is gained by unfolding data with little model error (page 204)
- Application of PARAFAC (GEMANOVA) for ANOVA-modeling of enzymatic activity (page 247)
- Application of unique PARAFAC modeling of fluorescence data of process samples using non-negativity and unimodality constraints (page 230)
- Application of a modified PARATUCK2 model for rank-deficient FIA data using non-negativity, unimodality, fixed parameters, and equality constraints (page 207)
- Application of PARAFAC2 for modeling data with retention time shifts (page 253)
- Application of N-PLS for several multi-way calibration problems showing that N-PLS is often the best alternative for obtaining predictive models

of multi-way data (page 196, 204 and 230)

8.2 DISCUSSION AND FUTURE WORK

The work presented here points to a number of areas that still need to be considered in more detail.

Some scattered initiatives has shed light on the mathematical properties of three-way arrays, but much is missing. Subjects such as rank of arrays and degrees of freedom for models could benefit from further investigation.

Except for Tucker3 and N-PLS much is needed in order to improve the speed and robustness of the algorithms. Some suggestions have been given here and others in the literature, but there is still a need for improvements in this direction in order to make the models useful for other than 'high-end' users. The appearance and characteristics of local minima and swamps should be better understood. Problems as how to detect and treat nonlinearities and how to assess and obtain uniqueness for 'arbitrary' models are also important problems that have not yet been thoroughly addressed.

It has been mentioned that weighted regression makes it possible to incorporate the uncertainties in the loss function directly (page 145). If knowledge of the correlation structure of the uncertainty is available this may also be incorporated into the loss function by weighted regression. This has not yet been thoroughly addressed. It is, however, likely that for, e.g., spectral data, where the noise structure is often easily determined, this structure can be beneficially used.

Variable selection is important in certain situations. Consider a situation where a model is sought to predict the concentration of an analyte in a large set of samples from the fluorescence excitation-emission of the analyte. Suppose also that it has been chosen that the model to be used is N-PLS. If a smaller subset of the samples have been measured spectrofluorometrically it is possible to explore, e.g., which emission spectra are the most important for getting a prediction model. Thereby a sound yet easier implemented measuring procedure for the predictive model can be obtained. A similar situation may occur when creating a filter-based spectrophotometer and the most descriptive wavelength areas for describing the spectral variation expected in certain type of samples are to

be found. Little has been done in developing procedures for such problems. Mostly, accommodated two-way approaches have been used. Extending the theory of principal variables (Höskuldsson 1993) to multi-way analysis it is possible to make a procedure for choosing subsets of variables based on either external or internal information. The use of experimental design as thoroughly elaborated on by Abel (1991) is also advantageous.

The use of multi-way analysis has only just begun. Undoubtedly, multi-way analysis can be beneficially used in, for example, analytical chemistry for developing fast and cheap calibration methods for a variety of chemical analytes. Such methods may well reduce the cost and use of additional chemicals while providing more robust and accurate estimations. Developing such methods require skills on both the analytical, mathematical, and instrumental side, but holds promise in a variety of settings, e.g., medical diagnostics, ecological studies, food technology, environmental studies etc.

Besides pure chemical and spectral data multi-way problems are often encountered in other areas. Examples are analysis of quantitative structure activity relationships for designing medical drugs or assessing environmental and health effects, batch and continuous process analysis, electronic nose data, consumer analysis, sensory analysis, image analysis, blind source separation in, e.g., telecommunication and speech recognition systems, time series analysis and signal processing for example related to medical instruments or process analysis, etc. In most of these areas no or very few applications of multi-way analysis have yet appeared. The use of multi-way analysis may provide huge economic savings and increased insights. For example, blind source separation, deals with how to uniquely determine the individual contributions to a set of measured signals, for example signals arising from telecommunication. Today, many complicated tricks have to be used in order to be able to separate the individual signals. However, being able to specify the problem as a multi-way problem a unique solution may be obtained simply by virtue of the multi-way nature of the data.

Widespread use of multi-way analysis, especially for on-line analysis will only evolve properly if better programs are made. Thus, a close collaboration with the field of numerical analysis is needed in order to specify better algorithms, that may eventually be incorporated into commercial software.

For inferences, e.g., in building control charts in process analysis a stronger theoretical theory on error propagation and uncertainties is mandatory. Statistical insight is needed in that respect (see Faber et al. 1997).

In the work presented here, some of the most fruitful results arose from collaborating with fields outside of chemometrics. Specifically psychometrics and signal processing. Such cross-disciplinary work often brings about useful information, that would otherwise not have appeared. For example, in developing the fast algorithm for unimodality constrained regression (page 177), many intermediate steps would not have been possible without knowledge of dynamic programming and multidimensional scaling. Proposing the solution to the problem of shifted (chromatographic) data (page 253) or the model of rank-deficient spectral data (page 207) would not be possible without extensive knowledge of the psychometrics literature. Further interdisciplinary collaboration can help provide models and algorithms which are faster and specifically aimed at dealing with particular problems, e.g., such as modeling multi-way time-series arrays.

Multi-way analysis started in the social science in the sixties. It was re-invented in the hard sciences in chemistry twenty years later. Today the two branches have started merging yielding many exciting results. This thesis is a contribution to the maturing of the field. The multitude of problems that have been shown to be handled efficiently with multi-way analysis holds promise for the future work. Getting a grasp of complex situations and data is a limiting factor for any sound problem solution in science and technology. Multi-way data analysis may help here.

APPENDICES

APPENDIX A THE MULTI-WAY TOOLBOX FOR MATLAB

A set of MATLAB™ M-files has been made for most problems discussed in this thesis. These are collectively called The *N*-way Toolbox for MATLAB™. The following functions are currently available and most can be downloaded at <http://newton.foodsci.kvl.dk>.

MODELS

PARAFAC	N-way PARAFAC with weighted or unweighted loss function. Optional: Non-negativity, orthogonality, unimodality, inequality, smoothness. Optimal compression. Missing values. Fixed parameters. Iteratively reweighting
NPLS	N-way PLS. Handles missing values
PARAFAC2	PARAFAC2. Handles missing values. Non-negativity
PARATUCK2	Restricted PARATUCK2. Handles missing values. Non-negativity. Fixed interaction matrix.
TUCKER3	N-way Tucker3. Handles missing values. Non-negativity. By Claus A. Andersson
NPRED	Predicting new samples using existing N-PLS model

NUMERICAL ANALYSIS

ULSR	Unimodal least squares regression
ULSRFIX	Unimodal least squares regression with fixed mode location
UNIMODAL	Solves $\min\ \mathbf{Y} - \mathbf{X}\mathbf{B}^T\ ^2$ subject to \mathbf{b}_f is unimodal $\forall f$
FNNLS	Non-negativity constrained regression, fast version
MONREG	Monotone regression
SLS	Smooth regression
SULSR	Smooth unimodal regression

OTHER

PPP	Khatri-Rao product
NSHAPE	Rearrange an N -way array
NCOSINE	Multiple cosine (Tucker's congruence coefficient) between several sets of loading matrices
NMODEL	Calculates the model from the model parameters
NPROCESS	Pre- and postprocess array
CORCOND	Core consistency diagnostics

APPENDIX B

RELEVANT PAPERS BY THE AUTHOR

The following publications by the author have all been important for this thesis.

- R. Bro, Multi-way calibration. Multi-linear PLS, *J. Chemom.*, **10** (1996) 47.
- R. Bro, H. Heimdal, Enzymatic browning of vegetables. Calibration and analysis of variance by multi-way methods., *Chemom. Intell. Lab. Syst.*, **34** (1996) 85.
- R. Bro, Håndbog i multivariabel kalibrering, Jordbrugsforlaget, Copenhagen, 1996
- H. Heimdal, R. Bro, L. M. Larsen, L. Poll, Prediction of polyphenol oxidase activity in model solutions containing various combinations of chlorogenic acid, (-)-epicatechin, O₂, CO₂, temperature, and pH by multiway data analysis, *J. Agric. Food Chem.*, **45** (1997), 2399.
- R. Bro, S. de Jong, A fast non-negativity constrained linear least squares algorithm for use in multi-way algorithms, *J. Chemom.*, **11** (1997) 393.
- R. Bro, PARAFAC: Tutorial & applications, *Chemom. Intell. Lab. Syst.*, **38** (1997) 149.
- R. Bro, P.R. Mobley, B.R. Kowalski, and J.J. Workman Jr., Review of chemometrics applied to spectroscopy: 1985-1995, part III – multi-way analysis, *Appl. Spectrosc. Rev.*, **32** (1997) 237.
- C. A. Andersson, R. Bro, Improving the speed of multi-way algorithms. Part i: Tucker3, *Chemom. Intell. Lab. Syst.*, **42** (1998) 93.
- R. Bro, C. A. Andersson, Improving the speed of multi-way algorithms. Part ii: compression, *Chemom. Intell. Lab. Syst.*, **42** (1998) 105.
- R. Bro, N. Sidiropoulos, Least squares algorithms under unimodality and non-negativity constraints, *J. Chemom.*, **12** (1998) 223.
- N. Sidiropoulos, R. Bro, Mathematical programming algorithms for

regression-based nonlinear filtering in \mathbb{R}^N , *IEEE Trans. Signal Proc.*, In press.

- A. K. Smilde, R. Tauler, J. Saurina, R. Bro, Calibration methods for complex second-order data, Submitted.
- L. Munck, L. Nørgaard, S. B. Engelsen, R. Bro, C. A. Andersson, Chemometrics in food science – a demonstration of the feasibility of a highly exploratory, inductive evaluation strategy of fundamental scientific significance, *Chemom. Intell. Lab. Syst.*, In press.
- R. Bro, Exploratory study of sugar production using fluorescence spectroscopy and multi-way analysis, *Chemom. Intell. Lab. Syst.*, In press.
- H. A. L. Kiers, J. M. F. ten Berge, R. Bro, PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model, *J. Chemom.*, Submitted.
- R. Bro, H. A. L. Kiers, C. A. Andersson, PARAFAC2 - Part II. Modeling chromatographic data with retention time shifts, *J. Chemom.*, Submitted.
- R. Bro, H. A. L. Kiers, A new efficient method for determining the number of components in PARAFAC models, *J. Chemom.*, Submitted.

BIBLIOGRAPHY

R. B. Abel, Experimental design for multilinear models in spectroscopy, Ph.D. dissertation, Ohio State University, 1991.

B. K. Alsberg, O. M. Kvalheim, Speed improvement of multivariate algorithms by the method of postponed basis matrix multiplications. Part I. Principal component analysis, *Chemom. Intell. Lab. Syst.*, **24** (1994a) 31.

B. K. Alsberg, O. M. Kvalheim, Speed improvement of multivariate algorithms by the method of postponed basis matrix multiplications. Part II. Three-mode principal component analysis, *Chemom. Intell. Lab. Syst.*, **24** (1994b) 43.

M. Amrhein, B. Srinivasan, D. Bonvin, M. M. Schumacher, On the rank deficiency and rank augmentation of the spectral measurement matrix, *Chemom. Intell. Lab. Syst.*, **33** (1996) 17.

C. A. Andersson, R. Bro, Improving the speed of multi-way algorithms. Part i: Tucker3, *Chemom. Intell. Lab. Syst.*, **42** (1998) 93.

C. J. Appellof, E. R. Davidson, Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents, *Anal. Chem.*, **53** (1981) 2053.

J. L. Barlow, Error analysis and implementation aspects of deferred correction for equality constrained least squares problems, *SIAM J. Numer. Anal.*, **25** (1988) 1340.

J. L. Barlow, N. K. Nichols, R. J. Plemmons, Iterative methods for equality-constrained least squares problems, *SIAM J. Sci. Stat. Comput.*, **9** (1988) 892.

R. E. Barlow, D. J. Bartholomew, J. M. Bremner, H. D. Brunk, Statistical inference under order restrictions, J. Wiley & Sons, NY, 1972.

F. L. Bauer, Das Verfahren der Treppeniterationen und verwandte Verfahren zur Lösung algebraischer Eigenwertprobleme, *ZAMP*, **8** (1957) 214.

Å. Björck, A direct method for sparse least squares problems with lower and upper bounds, *Numer. Math.*, **54** (1988) 19-32.

Å. Björck, Numerical Methods for Least Squares Problems, SIAM, Philadelphia, 1996.

K. S. Booksh, B. R. Kowalski, Comments on the data ANalysis (DATAN) algorithm and rank annihilation factor analysis for the analysis of correlated spectral data, *J. Chemom.*, **8** (1994) 287.

K. S. Booksh, Z. Lin, Z. Wang, Kowalski B. R., Extension of trilinear decomposition method with an application to the flow probe sensor, *Anal. Chem.*, **66** (1994) 2561.

-
- G. B. Brereton, S. P. Gurden, J. A. Groves, Use of eigenvalues for determining the number of components in window factor analysis of spectroscopy and chromatographic data, *Chemom. Intell. Lab. Syst.*, **27** (1995) 73.
- R. Bro, Multi-way calibration. Multi-linear PLS, *J. Chemom.*, **10** (1996) 47.
- R. Bro, H. Heimdal, Enzymatic browning of vegetables. Calibration and analysis of variance by multi-way methods, *Chemom. Intell. Lab. Syst.*, **34** (1996) 85.
- R. Bro, PARAFAC: Tutorial & applications, *Chemom. Intell. Lab. Syst.*, **38** (1997) 149.
- R. Bro, S. de Jong, A fast non-negativity constrained linear least squares algorithm for use in multi-way algorithms, *J. Chemom.*, **11** (1997) 393.
- R. Bro, N. Sidiropoulos, Least squares algorithms under unimodality and non-negativity constraints, *J. Chemom.*, **12** (1998) 223.
- R. Bro, C. A. Andersson, Improving the speed of multi-way algorithms. Part ii: compression, *Chemom. Intell. Lab. Syst.*, **42** (1998) 105.
- R. Bro, P.R. Mobley, B.R. Kowalski, and J.J. Workman Jr., Review of chemometrics applied to spectroscopy: 1985-1995, part III – multi-way analysis, *Appl. Spectrosc. Rev.*, **32** (1997) 237.
- R. Bro, Exploratory study of the sugar production using fluorescence spectroscopy and multi-way analysis, *Chemom. Intell. Lab. Syst.*, In press.
- R. Bro, H. A. L. Kiers, C. A. Andersson, PARAFAC2 - Part II. Modeling chromatographic data with retention time shifts, *J. Chemom.*, Submitted.
- R. Bro, H. A. L. Kiers, A new efficient method for determining the number of components in PARAFAC models, *J. Chemom.*, Submitted.
- P. Brouwer, P. M. Kroonenberg, Some notes on the diagonalization of the extended three-mode core matrix, *J. Classification.*, **8** (1991) 93.
- D. S. Burdick, X. M. Tu, L. B. McGown, D. W. Millican, Resolution of multicomponent fluorescent mixtures by analysis of excitation-emission-frequency array, *J. Chemom.*, **4** (1990) 15.
- D. S. Burdick, An introduction to tensor products with applications to multiway data analysis, *Chemom. Intell. Lab. Syst.*, **28** (1995) 229.
- J. D. Carroll, J. Chang, Analysis of individual differences in multidimensional scaling via an

-
- N-way generalization of "Eckart-Young" decomposition, *Psychometrika*, **35** (1970) 283.
- J. D. Carroll, P. Arabie, Multidimensional scaling, *Ann. Rev. Psychol.*, **31** (1980) 607.
- J. D. Carroll, S. Pruzansky, J. B. Kruskal, Candelinc: A general approach to multidimensional analysis of many-ways arrays with linear constraints on parameters, *Psychometrika*, **45** (1980) 3.
- R. B. Cattell, "Parallel proportional profiles" and other principles for determining the choice of factors by rotation, *Psychometrika*, **9** (1944) 267.
- N. Cliff, Orthogonal rotation to congruence, *Psychometrika*, **31** (1966) 33.
- R. D. Cook, S. Weisberg, Residuals and influence in regression, Chapman and Hall Ltd., New York, 1982.
- P. L. Cornelius, M. Seyedsadr, J. Crossa, Using the shifted multiplicative model to search for "separability" in crop cultivar trials, *Theor. Appl. Genet.*, **84** (1992) 161.
- S. de Jong, H. A. L. Kiers, Principal covariates regression. Part I. Theory, *Chemom. Intell. Lab. Syst.*, **14** (1992) 155.
- S. de Jong, Regression coefficients in multi-linear PLS, *J. Chemom.*, **12** (1998) 77.
- S. de Jong, A. Phatak, Partial least squares regression, In: Recent advances in total least squares techniques and errors-in variables modeling, (Ed. van Huffel), SIAM, Philadelphia, 1997, 25.
- J. de Leeuw, F. W. Young, Y. Takane, Additive structure in qualitative data: an alternating least squares method with optimal scaling features, *Psychometrika*, **41** (1976) 471.
- J. de Leeuw, Correctness of Kruskal's algorithms for monotone regression with ties, *Psychometrika*, **42** (1977) 141.
- J. de Leeuw, S. Pruzansky, A new computational method to fit the weighted Euclidian distance model, *Psychometrika*, **43** (1978) 479.
- C. L. de Ligny, M. Spanjer, J. C. van Houwelingen, H. M. Weesie, Three-mode factor analysis of data on retention in normal-phase high-performance liquid chromatography, *J. Chromatography*, **301** (1984) 311.
- N. R. Draper, H. Smith, Applied regression analysis, J. Wiley & Sons, NY, 1981
- S. R. Durell, C. Lee, R. T. Ross, E. L. Gross, Factor analysis of the near-ultraviolet absorption

spectrum of plastocyanin using bilinear, trilinear and quadrilinear models, *Archives of biochemistry and biophysics*, **278** (1990) 148.

H. T. Eastment, W. J. Krzanowski, Cross-validatory choice of the number of components from a principal component analysis, *Technometrics*, **24** (1982) 73.

G. W. Ewing, Instrumental methods of chemical analysis, McGraw-Hill Int. Ed., NY, 1985.

N. M. Faber, M. C. Buydens, G. Kateman, Generalized rank annihilation method. I: derivation of eigenvalue problems, *J. Chemom.*, **8** (1994) 147.

N. M. Faber, L. M. C. Buydens, G. J. Kateman, Generalized rank annihilation method. III: Practical implementation, *J. Chemom.*, **8** (1994) 273.

K. Faber, A. Lorber, B. R. Kowalski, Analytical figures of merit for tensorial calibration, *J. Chemom.*, **11** (1997) 419.

S. Fang, S. Puthenpura, Linear optimization and extensions: theory and algorithms, AT&T - Prentice-Hall, Englewood Cliffs, NJ, 1993.

M. J. Fay, A. Proctor, D. P. Hoffman, D. M. Hercules, Improved principal component analysis of noisy data, *Anal. Chem.*, **63** (1991) 1058.

A. S. Field, D. Graupe, Topographic component (Parallel Factor) analysis of multichannel evoked potentials: practical issues in trilinear spatiotemporal decomposition, *Brain Topography*, **3** (1991) 407.

R. A. Fisher, W. A. MacKenzie, Studies in crop variation. II. The manurial response of different potato varieties, *J. Agric. Soc.*, **13** (1923) 311.

S. D. Frans, M. L. McConnel, J. M. Harris, Multiwavelength and reiterative least squares resolution of overlapped liquid chromatographic peaks, *Anal. Chem.*, **57** (1985) 1552.

M. Frisén, Unimodal regression, *The Statistician*, **35** (1986) 479.

K. R. Gabriel, The biplot display of matrices with application to principal component analysis, *Biometrika*, **58** (1971) 453.

P. Geladi, Analysis of multi-way (multi-mode) data, *Chemom. Intell. Lab. Syst.*, **7** (1989) 11.

P. J. Gemperline, Target transformation factor analysis with linear inequality constraints applied to spectroscopic-chromatographic data, *Anal. Chem.*, **58** (1986) 2656.

P. J. Gemperline, A. Salt, Principal components regression for routine multicomponent UV

-
- determinations: a validation protocol, *J. Chemom.*, **3** (1989) 343.
- P. J. Gemperline, K. H. Miller, T. L. West, J. E. Weinstein, J. C. Hamilton, and J. T. Bray, Principal component analysis, trace elements, and blue crab shell disease, *Anal. Chem.*, **64** (1992) 523A.
- Z. Geng, N. Shi, Isotonic regression for umbrella orderings, *Appl. Statist.*, **39** (1990) 397.
- A. Gifi, Nonlinear multivariate analysis, J. Wiley & Sons, Chichester, 1990.
- P.E. Gill, W. Murray, M.H. Wright, Practical optimization, Academic Press, London, 1981.
- H. F. Gollob, A statistical model which combines features of factor analytic and analysis of variance techniques, *Psychometrika*, **33** (1968) 73.
- M. I. Griep, I. N. Wakeling, P. Vankeerberghen, D. L. Massart, Comparison of semirobust and robust partial least squares procedures, *Chemom. Intell. Lab. Syst.*, **29** (1995) 37.
- R. J. Hanson, Linear least squares with bounds and linear constraints, *SIAM J. Sci. Stat. Comput.*, **7** (1986) 826.
- R. A. Harshman, Foundations of the PARAFAC procedure: model and conditions for an 'explanatory' multi-mode factor analysis, *UCLA Working Papers in phonetics*, **16** (1970) 1.
- R. A. Harshman, Determination and proof of minimum uniqueness conditions for PARAFAC1, *UCLA Working Papers in phonetics*, **22** (1972) 111.
- R. A. Harshman, S. A. Berenbaum, Basic concepts underlying the PARAFACCANDECOMP three-way factor analysis model and its application to longitudinal data, In: Present and past in middle life, (Eds. D. H. Eichorn, J. A. Clausen, N. Haan, M. P. Honzik, P. H. Mussen), Academic Press, NY, 1981, 435.
- R. A. Harshman, M. E. Lundy, The PARAFAC model for three-way factor analysis and multidimensional scaling, In: Research methods for multimode data analysis, (Eds. H. G. Law, C. W. Snyder, J. A. Hattie, R. P. McDonald) Praeger, New York, 1984a, 122.
- R. A. Harshman, M. E. Lundy, Data preprocessing and the extended PARAFAC model, In: Research methods for multimode data analysis, (Eds. H. G. Law, C. W. Snyder, J. A. Hattie, R. P. McDonald) Praeger, New York, 1984b, 216.
- R. A. Harshman, W. S. de Sarbo, An application of PARAFAC to a small sample problem, demonstrating preprocessing, orthogonality constraints, and split-half diagnostic techniques, In: Research methods for multimode data analysis, (Eds. H. G. Law, C. W. Snyder, J. A. Hattie, R. P. McDonald) Praeger, New York, 1984, 602.

R. A. Harshman, "how can I know if it's 'real'?" A catalog of diagnostics for use with three-mode factor analysis and multidimensional scaling, In: Research methods for multimode data analysis, (Eds. H. G. Law, C. W. Snyder, J. A. Hattie, R. P. McDonald) Praeger, New York, 566, 1984.

R. A. Harshman, M. E. Lundy, PARAFAC: Parallel factor analysis, *Comp. Stat. Data Anal.*, **18** (1994) 39.

R. A. Harshman, M. E. Lundy, Uniqueness proof for a family of models sharing features of Tucker's three-mode factor analysis and PARAFAC/CANDECOMP, *Psychometrika*, **61** (1996) 133.

K. H. Haskell, R. J. Hanson, An algorithm for linear least squares problems with equality and nonnegativity constraints, *Math. Prog.*, **21** (1981) 98.

T. J. Hastie, R. J. Tibshirani, Generalized additive models, Chapman & Hall, London, 1990.

C. Hayashi, F. Hayashi, A new algorithm to solve PARAFAC-model, *Behaviormetrika*, **11** (1982) 49.

V. Hegeman, D. E. Johnson, On analyzing two-way AoV data with interaction, *Technometrics*, **18** (1976) 273.

H. Heimdal, L. M. Larsen, L. Poll, Characterization of Polyphenol Oxidase from photosynthetic and vascular lettuce tissues (*lactuca sativa*), *J. Agric. Food Chem.*, **42** (1994) 1428.

H. Heimdal, R. Bro, L. M. Larsen, L. Poll, Prediction of polyphenol oxidase activity in model solutions containing various combinations of chlorogenic acid, (-)-epicatechin, O₂, CO₂, temperature, and pH by multiway data analysis, *J. Agric. Food Chem.*, **45** (1997), 2399.

W. J. Heiser, P. M. Kroonenberg, Dimensionwise fitting in PARAFAC-CANDECOMP with missing data and constrained parameters, Leiden Psychological Reports, PRM 97-01, 1997.

H. V. Henderson, S. R. Searle, The vec-permutation matrix, the vec operator and Kronecker products: a review, *Lin. Multilin. Algebra*, **9** (1981) 271.

R. Henrion, G. Henrion, G. C. Onuoha, Multi-way principal components analysis of a complex data array resulting from physicochemical characterization of natural waters, *Chemom. Intell. Lab. Syst.*, **16** (1992) 87.

R. Henrion, Body diagonalization of core matrices in three-way principal component analysis: theoretical bounds and simulation, *J. Chemom.*, **7** (1993) 477.

R. Henrion, N-way principal component analysis. Theory algorithms and applications,

Chemometrics Intell. Lab. Syst., **25** (1994) 1.

G. Henrion, D. Nass, G. Michael, R. Henrion, Multivariate 3-way data analysis of amino acid patterns of lakes, *Fresenius J. Anal. Chem.*, **352** (1995) 431.

R. Henrion, C. A. Andersson, Diagonality versus variance of squares for simple-structure transformations of N-way core arrays, *J. Chemom.*, (1998), In press.

C. Ho, G. D. Christian, E. R. Davidson, Application of the method of rank annihilation to quantitative analyses of multicomponent fluorescence data from the video fluorometer, *Anal. Chem.*, **50** (1978) 1108.

C. Ho, G. D. Christian, E. R. Davidson, Application of the method of rank annihilation to fluorescent multicomponent mixtures of polynuclear aromatic hydrocarbons, *Anal. Chem.*, **52** (1980) 1071.

C. Ho, G. D. Christian, E. R. Davidson, Simultaneous multicomponent rank annihilation and applications to multicomponent fluorescent data acquired by the video fluorometer, *Anal. Chem.*, **53** (1981) 92.

P. K. Hopke, P. Paatero, H. Jia, R. T. Ross, R. A. Harshman, Three-way (PARAFAC) factor analysis: examination and comparison of alternative computational methods as applied to ill-conditioned data, *Chemom. Intell. Lab. Syst.*, In press.

A. Höskuldsson, The H-principle: new ideas, algorithms, and methods in applied mathematics and statistics, *Chemom. Intell. Lab. Syst.*, **23** (1993) 1.

E. J. Karjalainen, The spectrum reconstruction problem. Use of alternating regression for unexpected spectral components in two-dimensional spectroscopies, *Chemom. Intell. Lab. Syst.*, **7** (1989) 31.

E. J. Karjalainen, U. P. Karjalainen, Component reconstruction in the primary space of spectra and concentrations. Alternating regression and related direct methods, *Anal. Chim. Acta*, **250** (1991) 169.

A. Kapteyn, H. Neudecker, T. Wansbeck, An approach to *n*-mode components analysis, *Psychometrika*, **91** (1986) 269-275.

J. R. Kettenring, A case study in data analysis, *Proc. Symp. Appl. Math.*, **28** (1983) 105.

H. A. L. Kiers, Three-way methods for the analysis of qualitative and quantitative two-way data, DSWO Press, Leiden, 1989.

H. A. L. Kiers, Hierarchical relations among three-way methods, *Psychometrika*, **56** (1991)

449.

H. A. L. Kiers, W. P. Krijnen, An efficient algorithm for PARAFAC of three-way data with large numbers of observation units, *Psychometrika*, **56** (1991) 147.

H. A. L. Kiers, Tuckals core rotations and constrained TUCKALS modelling, *Statistica Applicata*, **4** (1992) 659.

H. A. L. Kiers, P. M. Kroonenberg, J. M. T. ten Berge, An efficient algorithm for TUCKALS on data with large number of observation units, *Psychometrika*, **33** (1992) 415.

H. A. L. Kiers, An alternating least squares algorithm for PARAFAC2 and three-way DEDICOM, *Comp. Stat. Data anal.*, **16** (1993) 103.

H. A. L. Kiers, A. K. Smilde, Some theoretical results on second-order calibration methods for data with and without rank overlap, *J. Chemom.*, **9** (1995) 179.

H. A. L. Kiers, R. A. Harshman, Relating two proposed method for speedup of algorithms for fitting two- and three-way principal component and related multilinear methods, *Chemom. Intell. Lab. Syst.*, **39** (1997) 31.

H. A. L. Kiers, Weighted least squares fitting using ordinary least squares algorithms, *Psychometrika*, **62** (1997) 251.

H. A. L. Kiers, Recent developments in three-mode factor analysis: Constrained three-mode factor analysis and core rotations, In: Data science, classification and related methods, (Ed. C. Hayashi), Springer, Tokyo, 1998a, 563.

H. A. L. Kiers, A three-step algorithm for CANDECOP/PARAFAC analysis of large data sets with multicollinearity, *J. Chemom.*, In press, 1998b

H. A. L. Kiers, J. M. F. ten Berge, R. Rocci, Uniqueness of three mode factor models with sparse cores: The 3 by 3 by 3 case, *Psychometrika*, **62** (1997) 349.

H. A. L. Kiers, A. K. Smilde, Constrained three-mode factor analysis as a tool for parameter estimation with applications in chemistry, *J. Chemom.*, **12** (1998) 125.

H. A. L. Kiers, J. M. F. ten Berge, R. Bro, PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model, *J. Chemom.*, Submitted.

F. J. Knorr, H. R. Thorsheim, J. M. Harris, Multichannel and numerical resolution of overlapping chromatographic peaks, *Anal. Chem.*, **53** (1981) 821.

W. P. Krijnen, The analysis of three-way arrays by constrained PARAFAC methods, DSWO

Press, M&T Series, **23** (1993).

W. P. Krijnen, H. A. L. Kiers, Clustered variables in PARAFAC, In: Advances in longitudinal and multivariate analysis in the behavioral sciences: Proceedings of the SMABS 1992 conference (Eds. J. H. L. Oud, R. A. W. van Blokland-Vogelesang), Nijmegen: ITS, 1993, 166.

P. M. Kroonenberg, J. de Leeuw, Principal components analysis of three-mode data by means of alternating least squares algorithms, *Psychometrika*, **45** (1980) 69.

P. M. Kroonenburg, Three-mode principal component analysis. Theory and applications, DSWO Press, Leiden, 1983.

P. M. Kroonenberg, J. M. F ten Berge, P. Brouwer, H. A. L. Kiers, Gram-Schmidt versus Bauer-Rutishauser in alternating least squares algorithms for three-mode principal component analysis, *Comp. Stat. Quarterly*, **2** (1989) 81.

J. B. Kruskal, Nonmetric multidimensional scaling: a numerical method, *Psychometrika*, **29** (1964) 115.

J. B. Kruskal, More factors than subjects, tests and treatments: an indeterminacy theorem for canonical decomposition and individual differences scaling, *Psychometrika*, **41** (1976) 281.

J. B. Kruskal, Three-way arrays: rank and uniqueness of trilinear decomposition, with application to arithmetic complexity and statistics, *Linear algebra and its applications*, **18** (1977a) 95.

J. B. Kruskal, Some least-squares theorems for matrices and N-way arrays, Unpublished manuscript, Bell Laboratories, Murray Hill, N. J., 1977b.

J. B. Kruskal, Factor analysis and principal components, *Int. Encyclopedia Stat.*, **1** (1978) 307.

J. B. Kruskal, Multilinear methods, *Proc. Symp. Appl. Math.*, **28** (1983) 75.

J. B. Kruskal, Multilinear methods, In: Research methods for multimode data analysis, (Eds. H. G. Law, C. W. Snyder, J. A. Hattie, R. P. McDonald), Praeger, New York, 1984, 36.

J. B. Kruskal, Rank, decomposition, and uniqueness for 3-way and N-way arrays, In: Multiway data analysis, (Eds. R. Coppi, S. Bolasco), Elsevier Science Pub. (North-Holland), 1989, 7.

J. B. Kruskal, R. A. Harshman, M. E. Lundy, How 3-MFA data can cause degenerate PARAFAC solutions, among other relationships, In: Multiway data analysis, (Eds. R. Coppi, S. Bolasco), Elsevier Science Pub. (North-Holland), 1989, 115.

C. L. Lawson, R. J. Hanson, Solving least squares problems. Prentice Hall Series in Automatic Computation, Prentice Hall, Inc, Englewood Cliffs, 1974.

-
- W. H. Lawton, E. A. Sylvestre, Self modeling curve resolution, *Technometrics*, **13** (1971) 617.
- S. Leurgans, R. T. Ross, Multilinear models in spectroscopy, *Statist. Sci.*, **7** (1992) 289.
- S. Leurgans, R. T. Ross, R. B. Abel, A decomposition for three-way arrays, *SIAM J. Matrix Anal. Appl.*, **14** (1993) 1064.
- S. Li, J. C. Hamilton, P. J. Gemperline, Generalized rank annihilation using similarity transformations, *Anal. Chem.*, **64** (1992) 599.
- S. Li, P. J. Gemperline, Eliminating complex eigenvectors and eigenvalues in multiway analyses using the direct trilinear decomposition method, *J. Chemom.*, **7** (1993) 77.
- Y. Liang, O. M. Kvalheim, Heuristic evolving latent projections: resolving hyphenated chromatographic profiles by component stripping, *Chemom. Intell. Lab. Syst.*, **20** (1993) 115.
- C. K. Liew, Inequality constrained least-squares estimation, *JASA*, **71** (1976) 746.
- J. W. Longley, Least squares computations using orthogonalization methods, Lecture notes in pure and applied mathematics, Dekker Inc. New York, 1984.
- A. Lorber, Quantifying chemical composition from two-dimensional data arrays, *Anal. Chim. Acta*, **164** (1984) 293.
- A. Lorber, Features of quantifying chemical composition from two-dimensional data array by the rank annihilation factor analysis method, *Anal. Chem.*, **57** (1985) 2395.
- D. G. Luenberger, Introduction to linear and nonlinear programming, Addison-Wesley, Reading, Massachusetts, 1973.
- M. E. Lundy, R. A. Harshman, J. B. Kruskal, A two-stage procedure incorporating good features of both trilinear and quadrilinear models, In: Multiway data analysis, (Eds. R. Coppi, S. Bolasco), Elsevier Science Pub. (North-Holland), 123, 1989.
- R. F. Madsen, W. K. Nielsen, B. Winstrøm-Olsen, T. E. Nielsen, Formation of colour compounds in production of sugar from sugar beets, *Sugar Technology Reviews*, **6** (1978/79) 49.
- M. Maeder, A. Zilian, Evolving factor analysis, a new multivariate technique in chromatography, *Chemom. Intell. Lab. Syst.*, **3** (1988) 205.
- E. R. Malinowski, Factor analysis in chemistry, J. Wiley & Sons, NY, 1991.
- J. Mandel, The partitioning of interaction in analysis of variance, *J. Res. National Bureau*

Stand. B. Math. Sc., **73B** (1969) 309.

J. Mandel, A new analysis of variance model for non-additive data, *Technometrics*, **13** (1971) 1.

R. Manne, On the resolution problem in hyphenated chromatography, *Chemom. Intell. Lab. Syst.*, **27** (1995) 89.

H. Martens, T. Næs, Multivariate calibration, J. Wiley & Sons, Chichester, 1989.

M. V. Martinez, J. R. Whitaker, The biochemistry and control of enzymatic browning, *Trends in Food Science & Technology*, **6** (1995) 195.

R. P. McDonald, A simple comprehensive model for the analysis of covariance structures: Some remarks on applications, *British Journal of Mathematical and Statistical Psychology*, **33** (1980) 161.

W. Meredith, Notes on factorial invariance, *Psychometrika*, **29** (1964) 177.

B. C. Mitchell, D. S. Burdick, An empirical comparison of resolution methods for three-way arrays, *Chemom. Intell. Lab. Syst.*, **20** (1993) 149.

B. C. Mitchell, D. S. Burdick, Slowly converging PARAFAC sequences: Swamps and two-factor degeneracies, *J. Chemom.*, **8** (1994) 155.

D. C. Montgomery, Design and analysis of experiments, J. Wiley & Sons, NY, 1991

L. Munck, Man as selector – a Darwinian boomerang striking through natural selection, In: Environmental concerns. An interdisciplinary exercise, (Ed. J. Aa. Hansen), Elsevier, London, 1991, 211

L. Munck, L. Nørgaard, S. B. Engelsen, R. Bro, C. A. Andersson, Chemometrics in food science – a demonstration of the feasibility of a highly exploratory, inductive evaluation strategy of fundamental scientific significance, *Chemom. Intell. Lab. Syst.*, In press.

T. Murakami, J. M. F. ten Berge, H. A. L. Kiers, A case of extreme simplicity of the core matrix in three-mode principal component analysis, *Psychometrika*, (1998), In press.

J. Möcks, The influence of latency jitter in principal component analysis of event-related potentials, *Psychophysiology*, **23** (1986) 480.

P. R. C. Nelson, P. A. Taylor, J. F. MacGregor, Missing data methods in PCA and PLS: Score calculations with incomplete observations, *Chemom. Intell. Lab. Syst.*, **35** (1996), 45.

-
- J. Nilsson, S. de Jong, A. K. Smilde, Multiway calibration in 3D QSAR, *J. Chemom.*, **11** (1997) 511.
- L. Nørgaard, C. Ridder, Rank annihilation factor analysis applied to flow injection analysis with photodiode-array detection, *Chemom. Intell. Lab. Syst.*, **23** (1994) 107.
- L. Nørgaard, A multivariate chemometric approach to fluorescence spectroscopy, *Talanta*, **42** (1995a) 1305.
- L. Nørgaard, Classification and prediction of quality and process parameters of thick juice and beet sugar by fluorescence spectroscopy and chemometrics, *Zuckerind.*, **120** (1995b) 970.
- L. Nørgaard, Spectral resolution and prediction of slit widths in fluorescence spectroscopy by two and three way methods, *J. Chemom.*, **10** (1996) 615.
- P. Paatero, Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values, *Environmetrics*, **5** (1994) 111.
- P. Paatero, A weighted non-negative least squares algorithm for three-way "PARAFAC" factor analysis, *Chemometrics Intell. Lab. Syst.*, **38** (1997) 223.
- P. Paatero, The multilinear engine - a table-driven least squares program for solving all kinds of multilinear problems, including the n-way factor analytical PARAFAC model, In prep.
- G. R. Phillips, M. E. Edward, Comparison of conventional and robust regression in analysis of chemical data, *Anal. Chem.*, **55** (1983) 1134.
- J. R. Piggot, K. Sharman, Methods to aid interpretation of multidimensional data, In: Statistical procedures in food research, (Ed. J. R. Piggott), Elsevier Applied Science, London, 1986, 181.
- C. R. Rao, S. Mitra, Generalized inverse of matrices and its applications, Wiley, New York, 1971.
- W. S. Rayens, B. C. Mitchell, Two-factor degeneracies and a stabilization of PARAFAC, *Chemom. Intell. Lab. Syst.*, **38** (1997) 173.
- R. T. Ross, S. Leurgans, Component resolution using multilinear models, *Methods in Enzymology*, **246** (1995) 679.
- M. D. Russell, M. Gouterman, Excitation-emission-lifetime analysis of multicomponent systems - I. Principal component factor analysis, *Spectrochimia Acta*, **44A** (1988a) 857.
- M. D. Russell, M. Gouterman, Excitation-emission-lifetime analysis of multicomponent systems - II. Synthetic model data, *Spectrochimia Acta*, **44A** (1988b) 863.

-
- M. D. Russell, M. Gouterman, J. A. van Zee, Excitation-emission-lifetime analysis of multi-component systems - III. Platinum, palladium and rhodium porphyrins, *Spectrochimica Acta*, **44A** (1988) 882.
- H. Rutishauser, Computational aspects of F.L. Bauer's simultaneous iteration method, *Numer. Math.*, **13** (1969) 4.
- J. Röhmel, B. Streitberg, W. M. Herrmann, The COMSTAT algorithm for multimodal factor analysis: An improvement of Tucker's three-mode factor analysis method, *Neuropsychobiology*, **10** (1983) 157.
- E. Sanchez, B. R. Kowalski, Generalized rank annihilation factor analysis, *Anal. Chem.*, **58** (1986) 496.
- E. Sanchez, B. R. Kowalski, Tensorial calibration II. Second-order calibration, *J. Chemom.*, **2** (1988) 265.
- E. Sanchez, B. R. Kowalski, Tensorial resolution: a direct trilinear decomposition, *J. Chemom.*, **4** (1990) 29.
- R. Sands, F. W. Young, Component models for three-way data: an alternating least squares algorithm with optimal scaling features, *Psychometrika*, **45** (1980) 39.
- L. Sarabia, M. C. Ortiz, R. Leardi, G. Drava, A program for non-orthogonal factor analysis, *Trends Anal. Chem.*, **12** (1993) 226.
- I. Scarminio, M. Kubista, Analysis of correlated spectral data, *Anal. Chem.*, **65** (1993) 409.
- K. Schittkowski, J. Stoer, A factorization method for the solution of constrained linear least squares problems allowing subsequent data changes, *Numer. Math.* **31**, (1979), 431.
- P. H. Schönemann, An algebraic solution for a class of subjective metrics models, *Psychometrika*, **37** (1972) 441.
- M. B. Seasholz, B. R. Kowalski, The parsimony principle applied to multivariate calibration, *Anal. Chim. Acta*, **277** (1993) 165
- N. D. Sidiropoulos, R. Bro, Mathematical programming algorithms for regression-based nonlinear filtering in \mathbb{R}^N , *IEEE Trans. Signal Proc.*, In press.
- A. K. Smilde, D. A. Doornbos, Simple validity tools for judging the predictive performance of PARAFAC and three-way PLS, *J. Chemom.*, **6** (1992) 11.
- A. K. Smilde, Y. Wang, B. R. Kowalski, Theory of medium-rank second-order calibration with

restricted-Tucker models, *J. Chemom.*, **8** (1994a) 21.

A. K. Smilde, R. Tauler, J. M. Henshaw, L. W. Burgess, B. R. Kowalski, Multicomponent determination of chlorinated hydrocarbons using a reaction-based chemical sensor. 3. Medium-rank second-order calibration with restricted Tucker models, *Anal. Chem.*, **66** (1994b) 3345.

A. K. Smilde, Comments on multilinear PLS, *J. Chemom.* **11** (1997a) 367.

A. K. Smilde, Multi-way regression models, Internal report, University of Amsterdam, 1997b.

A. K. Smilde, R. Tauler, J. Saurina, R. Bro, Calibration methods for complex second-order data, Submitted.

H. Späth, Mathematical algorithms for linear regression, Academic Press, Inc., Boston, 1987.

S. S. Stevens, On the theory of scales of measurement, *Science*, **103** (1946) 677.

J. Stoer, On the numerical solution of constrained least-squares problems, *SIAM J. Numer. Anal.*, **8** (1971) 382.

G. P. H. Styan, Hadamard products and multivariate statistical analysis, *Linear Algebra and its Applications*, **6** (1973) 217.

L. Ståhle, Aspects of analysis of three-way data, *Chemom. Intell. Lab. Syst.*, **7** (1989) 95.

Y. Takane, F. W. Young, J. de Leeuw, Nonmetric individual differences multidimensional scaling: an alternating least squares method with optimal scaling features, *Psychometrika*, **42** (1977) 7.

R. Tauler, D. Barceló, Multivariate curve resolution applied to liquid chromatography-diode array detection, *Trends Anal. Chem.*, **12** (1993) 319.

J. M. F. ten Berge, J. de Leeuw, P. M. Kroonenberg, Some additional results on principal components analysis of three-mode data by means of alternating least squares, *Psychometrika*, **52** (1987) 183.

J. M. F. ten Berge, H. A. L. Kiers, J. de Leeuw, Explicit Candecomp/PARAFAC solution for a contrived 2 x 2 x 2 array of rank three, *Psychometrika*, **53** (1988) 579.

J. M. F. ten Berge, Convergence of PARAFAC preprocessing procedures and the Deming-Stephan method of iterative proportional fitting, In: Multiway data analyses, (Eds. R. Coppi, S. Bolasco), Elsevier Science Pub., North-Holland, 1989, 53

-
- J. M. F. ten Berge, Kruskal's polynomial for $2 \times 2 \times 2$ arrays and a generalization to $2 \times n \times n$ arrays, *Psychometrika*, **56** (1991) 631.
- J. M. F. ten Berge, H. A. L. Kiers, W. P. Krijnen, Computational solutions for the problem of negative saliences and nonsymmetry in INDSCAL, *J. Classification* **10** (1993) 115.
- J. M. F. ten Berge, H. A. L. Kiers, J. de Leeuw, Some uniqueness results for PARAFAC2, *Psychometrika*, **61** (1996) 123.
- J. M. F. ten Berge, H. A. L. Kiers, Simplicity of core arrays in three-way component analysis with applications to the typical rank of $P \times Q \times 2$, *Psychometrika*, (1998) Submitted.
- V. Tomišić and V. Simeon, Assessment of the effective rank of a (co)variance matrix: A non-parametric goodness-of-fit test, *J. Chemom.*, **7** (1993) 381.
- X. M. Tu, D. S. Burdick, Resolution of trilinear mixtures: application in spectroscopy, *Statistica Sinica*, **2** (1992) 577.
- L. R. Tucker, *A method for synthesis of factor analysis studies (Personnel Research Section Report No.984)*, Dept. of the Army, Washington D.C. (1951).
- L. R. Tucker, Problems in measuring change (Ed. C. W. Harris), University of Wisconsin Press, Madison, 1963, 122.
- L. R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika*, **31** (1966) 279.
- W. A. van der Kloot, P. M. Kroonenberg, External analysis with three-mode principal component models, *Psychometrika*, **50** (1985) 479.
- F. A. van Eeuwijk, Between and beyond additivity and non-additivity; the statistical modeling of genotype by environment interaction in plant breeding, Ph.D. dissertation, University of Wageningen, 1996.
- Y. Wang, O. S. Borgen, B. R. Kowalski, M. Gu., F. Turecek, Advances in second-order calibration, *J. Chemom.*, **7** (1993) 117.
- S. Weisberg, Applied linear regression, J. Wiley & Sons, NY, 1985.
- J. W. Verhoeven, Glossary of terms used in photochemistry, *Pure & Appl. Chem.*, **68** (1996) 2223.
- H. J. Werner, On equality constrained generalized least-squares estimation, *Linear Algebra and its Applications*, **127** (1990) 379.

- B. E. Wilson, E. Sanchez, B. R. Kowalski, An improved algorithm for the generalized rank annihilation method, *J. Chemom.*, **3** (1989) 493
- B. E. Wilson, W. Lindberg, B. R. Kowalski, Multicomponent quantitative analysis using second-order nonbilinear data: theory and simulations, *Anal. Chem.*, **111** (1989) 3797.
- W. Windig, The use of second-derivative spectra for pure-variable based self-modeling mixture analysis techniques, *Chemom. Intell. Lab. Syst.*, **23** (1994) 71.
- B. Winstrøm-Olsen, R. F. Madsen, W. K. Nielsen, Sugar beet phenols, part I, *Int. Sugar J.*, **81** (1979a) 332.
- B. Winstrøm-Olsen, R. F. Madsen, W. K. Nielsen, Sugar beet phenols, part II, *Int. Sugar J.*, **81** (1979b) 362.
- B. Winstrøm-Olsen, Enzymic colour formation in sugar beet. Characterization of enzymes using catecholamines. part I, *Int. Sugar J.*, **83** (1981a) 102.
- B. Winstrøm-Olsen, Enzymic colour formation in sugar beet. Characterization of enzymes using catecholamines. part I, *Int. Sugar J.*, **83** (1981b) 137.
- S. Wold, Cross-validatory estimation of the number of components in factor and principal components models, *Technometrics*, **20** (1978) 397.
- S. Wold, N. Kettaneh, K. Tjessem, Hierarchical multiblock PLS and PC models for easier model interpretation and as an alternative to variable selection, *J. Chemom.*, **10** (1996) 463.
- F. Yates, The analysis of replicated experiments when the field results are incomplete, *The empire journal of experimental agriculture*, **1** (1933) 129.
- M. G. Yochmowitz, Factor analysis-of-variance (FANOVA), In: *Enc. Stat. Sc.* (Eds. S. Kotz, N. L. Johnson), J. Wiley & Sons, NY, **3** (1983) 8.
- F. W. Young, Quantitative analysis of qualitative data, *Psychometrika*, **46** (1981) 357.

INDEX

- A priori knowledge . . . 4, 18, 107, 125, 222
- Active set algorithm 170
- ALS 57
- Compression 88
- Improving 86
- Improving ALS 87
- In detail 158
- PARAFAC 62
- PARAFAC2 65
- PARATUCK2 68
- Regularization 87
- Tucker 72
- Alternating least squares 158
- See ALS 57
- Alternating regression 190
- Amino-N 204
- Analysis of variance 192, 248
- Analytical model 88
- ANOVA
- See Analysis of variance 248
- Approximate orthogonality 216
- Baseline
- constant 143
- Bauer-Rutishauser 76
- Biplots 127
- Bootstrapping 196
- Britton-Robinson buffer 207
- Calibration 191, 204
- CANDECOMP 23
- CANDELINC 89, 150
- Catecholamine 230
- Centering 102
- Centering data with missing values 106
- Closure 150
- Component 10
- Components
- number of 110
- Compression 88
- Exact 91
- Compression model 88
- Computational validation 99
- congruence coefficient 123
- Conjugate gradient 97
- Consensus models 198
- Constraints 18
- Approximate 140
- Equality 150
- Exact 140
- Extent of 140
- Functional 156
- Inequality 149
- Linear 150
- Monotonicity 151
- Non-negativity 148
- Orthogonality 154
- Smoothness 152
- Unimodality 151
- Continuous factor 193
- Convergence 124, 132
- assessing 121
- Core 45
- Computation 72
- PARATUCK2 38
- Core 50
- Extended 49
- Zero elements 48
- Core consistency 113
- Core consistency plot 117
- Correlation coefficient
- Uncorrected 123
- Cross-validation 113
- Curve resolution 190, 218
- Curveresolution 190
- Degeneracy 122
- Two-factor 124
- Degrees of freedom . . . 113, 127, 156,

- 196
- Diagnostics 99
- Dihydroxyphenylalanine 230
- Dimension 8
- Dimensionality
 - Determining 116
- Diode-array detector 207
- Direct fitting 60, 65
- Direct trilinear decomposition 30
- Dopa 230, 239
- Enzymatic browning 247
- Equality constraints 150
- Evolving factor analysis 190
- Experimental design 248, 263
- Explanatory validation 99
- Exploratory analysis 20, 50, 187, 230
- Extended core 49
- Extrapolation 95
- Factor 10
- Fixed coefficients regression 167
- Fixed parameters 142
- Flow injection analysis 207
- Fluorescence 2, 26, 41, 115, 142, 182, 191, 204, 231
- Fractional factorial designs 196
- Functional constraints 156
- Gauss-Newton 96
- GEMANOVA 193, 248
- GRAM 30
- Gram-Schmidt orthogonalization 76
- Heteroscedasticity 196
- Heuristic evolving latent projection 190
- Hierarchical models 198
- Hierarchy
 - of models 108
- Indeterminacy
 - PARAFAC 28
- Inequality constraints 149
- Influence analysis 126
- Interactions 192
 - Factors 37, 40, 47, 119, 127, 210
- Interval-scale 101
- Iteratively reweighted regression 145
- Jack-knife analysis 111
- Joint plots 127
- k-rank 27, 126
- Kasha rule 238
- Kathri-Rao product 20
- Latent variable 10, 188
- Layer 8
- Leverage 126
 - corrected residuals 146
- Line search 95
- Linear constraints 150
- Loading 10
- Logarithmic transform 251
- longitudinal 33
- Lower-order effects 143
- Mahalanobis 127
- Many-fit diagnostics 100
- Measurement conditionality 156
- Measurement level 156
- Measurement process 156
- Median filter 243
- Missing data 146, 235
- MLR
 - See Multiple linear regression 242
- Mode 8
- Model
 - Choosing 107, 188
 - Structural 17
- Model hierarchy 108
- Modeling
 - cognitive aspects 187
- Monotone regression 175
- Monotonicity 151
- Multi-linear engine 29, 97
- Multi-way analysis 1, 2, 129
- Multi-way data 7
- Multi-way PCA 129
- Multiple cosine 123
- Multiple linear regression 108, 130, 159, 202, 204, 242
- Multiplicative ANOVA model 192

N-PLS		say	248
Algorithm	78	Polyphenol oxidase	247
Notation	53	Preprocessing	101
Nested model	19	Problem COLUMNWISE	161
NIPALS	58, 76	Problem GLOBAL	158
Noise reduction	197	Problem ROWWISE	160
Nominal	156	Pseudo-rank	89, 111
Non-linearity		QR orthogonalization	76
signs of	113	Qualitative data	156
Non-negativity	37, 124, 131, 148, 178, 233	Quantitative factor	193
Non-negativity constrained regression	169	RAFA	30
Norepinephrine	230	Rank	12, 51, 127, 131
Notation		Rank analysis	111
N-PLS	53	Rank-deficiency	207
Numerical	157	Ratio-scale	101
Oligomodal	179	Rayleigh scatter	136, 232, 233, 235
One-fit diagnostics	100	Regression	
Optimal scaling	156	Iteratively reweighted	145
Ordinal	156	Weighted	145
Orthogonality	124, 154	Regularization	87, 124
approximate	155	Relaxation	95
PARAFAC		Residual analysis	113, 126
ALS algorithm	61	Residuals	
Initialization	62	leverage-corrected	146
PARAFAC	23	Restricted PARATUCK2	40
PARAFAC2	33, 207	Restricted Tucker3 models	50
ALS algorithm	65	Retention shifts	
Uniqueness	37	Modeling	33
Parallel proportional profiles	20	Ridge regression	124
PARATUCK2	37	Robustness	128
ALS algorithm	68	Score	10
Fluorescence	42	Scree-plot	110, 113
Initialization	71	Screening	191, 231
Uniqueness	39	Second-order advantage	31, 142, 191, 213
Parsimony	108	Second-order calibration	119, 143, 191, 204, 227, 261
Partial least squares		Selectivity	143, 190
See PLS	51	Self modeling curve resolution	190
PLS		Sensory analysis	196
Multilinear	51	Shifted multiplicative model	194
N-PLS	51	Shifts	
PMF3	29, 96	Modeling	33
Polarographic polyphenol oxidase as-		SIMPLISMA	190

Single-centering	102	Variable selection	262
slab	8	Wavelets	243
Slice	8	Way	8
Smoothness	152	Weighted regression	145
Split-half analysis	111, 124, 141, 236	Window factor analysis	190
Statistical validation	99	Zero-fit diagnostics	99
Sugar	204, 230		
Swamp	124		
Symmetry	151		
Target	143, 179		
Tryptophane	239		
Tucker			
ALS algorithm	72		
Tucker models	44		
Tucker1	49		
Tucker1 based compression	92		
Tucker2	49		
Tucker3			
Constrained	50		
Uniqueness	48		
Tucker3 based compression	92		
Tucker3 core analysis	110		
Tucker3 initialization	76		
Two-factor degeneracy	124		
Uncorrected correlation coefficient			
.....	123		
Unfolding	10		
Unimodality	151, 238		
Unimodality constrained regression			
.....	177		
Unique axis	26		
Uniqueness	18, 190, 213		
assessing	124		
from constraints	140		
k-rank	26		
N-PLS	53		
PARAFAC	25		
PARAFAC2	37		
Rotation	28		
Tucker3	48		
Uniqueness	3		
PARATUCK2	39		
Validation	99		

Multi-way Analysis in the Food Industry
Models, Algorithms, and Applications

ACADEMISH PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam,
op gezag van de Rector Magnificus
prof.dr J. J. M. Franse
ten overstaan van een door het college voor promoties
ingestelde commissie in het openbaar te verdedigen
in de Aula der Universiteit
op vrijdag 20 november 1998 te 11.00 uur

door

Rasmus Bro

geboren te Holbæk, Denemarken

Promotores

Prof.dr A. K. Smilde, University of Amsterdam

Prof.dr L. Munck, The Royal Veterinary and Agricultural University, Denmark

Promotiecommissie

Prof.dr P. D. Iedema, University of Amsterdam

Prof.dr H. Poppe, University of Amsterdam

dr S. de Jong, Unilever Research

dr H. A. L. Kiers, University of Groningen

Prof.dr R. J. M. M. Does, University of Amsterdam

dr P. M. Kroonenberg, Leiden University