ELSEVIER

Chemometrics and Intelligent Laboratory Systems 38 (1997) 149–171

Chemometrics and intelligent laboratory systems

Tutorial

PARAFAC. Tutorial and applications

Rasmus Bro *

Chemometrics Group, Food Technology, Royal Veterinary & Agricultural University, Rolighedsvej 30, III, DK-1958 Frederiksberg C, Denmark

Received 4 April 1996; revised 30 July 1996; accepted 8 March 1997

Abstract

This paper explains the multi-way decomposition method PARAFAC and its use in chemometrics. PARAFAC is a generalization of PCA to higher order arrays, but some of the characteristics of the method are quite different from the ordinary two-way case. There is no rotation problem in PARAFAC, and e.g., pure spectra can be recovered from multi-way spectral data. One cannot as in PCA estimate components successively as this will give a model with poorer fit, than if the simultaneous solution is estimated. Finally scaling and centering is not as straightforward in the multi-way case as in the two-way case. An important advantage of using multi-way methods instead of unfolding methods is that the estimated models are very simple in a mathematical sense, and therefore more robust and easier to interpret. All these aspects plus more are explained in this tutorial and an implementation in Matlab code is available, that contains most of the features explained in the text. Three examples show how PARAFAC can be used for specific problems. The applications include subjects as: Analysis of variance by PARAFAC, a five-way application of PARAFAC, PARAFAC with half the elements missing, PARAFAC constrained to positive solutions and PARAFAC for regression as in principal component regression.

Contents

* E-mail: rasmus.bro@pop.foodsci.kvl.dk.

PII S0169-7439(97)00032-4

1. Introduction

PARAFAC is a multi-way method originating from psychometrics [1,2]. It is gaining more and more interest in chemometrics and associated areas for many reasons: Simply increased awareness of the method and its possibilities, the increased complexity of the data dealt with in science and industry, and increased computational power [3,4].

Multi-way data are characterized by several *sets* of variables that are measured in a crossed fashion. Chemical examples could be fluorescence emission spectra measured at several excitation wavelengths for several samples, fluorescence lifetime measured at several excitation and emission wavelengths or any kind of spectrum measured chromatographically for several samples. Determining such variables will give rise to three-way data; i.e., the data can be arranged in a cube instead of a matrix as in standard multivariate data sets. In psychometrics a typical data set could be a set of variables measured on several persons/subjects on several occasions. Similar configurations can be imagined in for example sensometrics. In practice many other types of data might be multi-way: two-way data determined for several chemical treatments, pH's, times, locations, etc. An important way of generating multi-way data is of course images in all its bearings.

PARAFAC is one of several decomposition methods for multi-way data. The two main competitors are the Tucker3 method [5], and simply unfolding of the multi-way array to a matrix and then performing standard two-way methods as PCA. The Tucker3 method should rightfully be called three-mode principal component analysis (or N-mode principal component analysis), but here the term Tucker3 or just Tucker will be used instead. PARAFAC, Tucker and two-way PCA are all multi- or bi-linear decomposition methods, which decompose the array into sets of scores and loadings, that hopefully describe the data in a more condensed form than the original data array. There are advantages and disadvantages with all the methods, and often several methods must be tried to find the most appropriate.

Without going into details of two-way PCA and Tucker it is important to have a feeling for the hierarchy among these methods. Kiers [6] shows that PARAFAC can be considered a constrained version of Tucker3, and Tucker3 a constrained version of two-way PCA. Any data set that can be modeled adequately with PARAFAC can thus also be modeled by Tucker3 or two-way PCA, but PARAFAC uses fewer degrees of freedom. A two-way PCA 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 two-way PCA models will tend to use the excess degrees of freedom to model noise or model the systematic variation in a redundant way (see the last application). Therefore one will generally prefer to use the simplest possible 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 [7]. In the sense that it uses most degrees of freedom the PCA model can be considered the most complex and flexible model, while PARAFAC is the most simple and restricted model.

Conceptually some may find two-way PCA more simple than the multi-linear methods, but in a multi-way context this is not so. Because the array has to be unfolded to a matrix before two-way analysis, the variables in the unfolded modes get mixed up, so that the effect of one variable is not associated with one but many elements of a loading vector. Consider an even more complex model than two-way PCA, e.g. a model that does not assume any structure at all but models each data element individually. This model would equal the data and obviously use all degrees of freedom, giving a perfect fit. Thus, the more structure the poorer the fit is and the simpler the model is.

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 can to some extent be compared to the difference between using multiple linear regression (MLR) and partial least squares regression (PLS) for multivariate calibration. MLR is known to give the best fit to the dependent variable of the calibration data, but in most cases PLS has better predictive power. PLS can be seen as a constrained version of MLR, where the constraints helps the model focusing on the systematic part of the data. In the same way multi-way methods are less sensitive to noise and further give loadings that can be directly related to the different modes of the multi-way array. That two-way PCA can give very complex models can be illustrated with an example. For an F-component PCA solution to an $I \times J \times K$ array unfolded to an $I \times JK$ matrix, the PCA model consists of $F(I+JK)$ parameters (scores and loading elements). A corresponding Tucker 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 a hypothetical example consider a $10 \times 100 \times 20$ array modeled by a 5 component solution. A two-way PCA model of the 10×2000 unfolded array consists of 10050 parameters, a Tucker model of 775 and a PARAFAC model of 650 parameters. Clearly, the PCA model will be more difficult to interpret than the multi-way models.

In this paper a tutorial of how to use PARAFAC is given. The interest in PARAFAC and related methods is often hampered by practical considerations regarding how to implement the algorithm, how to do sound analysis etc. Many excellent papers on PARAFAC are not published in readily available papers. The essence of some of these papers is presented. A very annoying characteristic of PARAFAC is the long time required to calculate the models. The algorithms used are most often based on alternating least squares (ALS) initialized by either random values or values calculated by a direct trilinear decomposition based on the generalized eigenvalue problem. Here the ALS algorithm of PARAFAC is modified in simple manners, which brings about a decrease in the number of iterations and time required to calculate the models of up to 20 times.

In the following, the discussion will be limited to three-way data for simplicity, but most results are valid for data and models of any (higher) order. Three applications will show some typical applications of PARAFAC and also include higher order models.

2. Nomenclature

In the following, scalars are indicated by lower-case italics, vectors by bold lower-case characters, bold capitals are used for two-way matrices, and underlined bold capitals for three-way arrays. The letters I, J, K, L and M are reserved for indicating the dimension of different modes. The ijkth element of $\underline{\mathbf{X}}$ is called x_{ijk}. The terms mode, way and order are used more or less interchangeably though a distinction is sometimes made between the geometrical dimension of the hypercube — the number of ways — and the number of independent ways — which is the order/mode [3,6]. An ordinary two-way covariance matrix is only a one-mode array, because the variables are identical in the two ways. Likewise there will not be distinguished between the terms factor and component. When three-way arrays are unfolded to matrices the following notation will be used: If $\underline{\mathbf{X}}$ is an $I \times J \times K$ array and is unfolded to an $I \times \underline{JK}$ matrix the order of J and K indicates which indices are running fastest. In this case the indices of J are running fastest, meaning that the first J rows of $\mathbf{X}$ contain all variables for $k = 1$ and for $j = 1$ to $j = J$.

3. The model

PARAFAC is a decomposition method, which conceptually can be compared to bilinear PCA, or rather it is *one* generalization of bilinear PCA, while the Tucker3 decomposition is another generalization of PCA to higher orders [8,9]. The model was independently proposed by Harshman [1] and by Carroll and Chang [2] who named the model CANDECOMP (canonical decomposition). A decomposition of the data is made into triads or trilinear components, but 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.

A PARAFAC model of a three-way array is given by three loading matrices, **A**, **B**, and **C** with elements a_{if}, b_{jf}, and c_{kf}. The trilinear model is found to minimize the sum of squares of the residuals, e_{ijk} in the model

$$x_{ijk} = \sum_{f=1}^{F} a_{ij} b_{jf} c_{kf} + e_{ijk} \qquad (1)$$

This equation is shown graphically in Fig. 1 for two components ($F = 2$).

The model can also be written

$$\underline{\mathbf{X}} = \sum_{f=1}^{F} \boldsymbol{a}_f \otimes \boldsymbol{b}_f \otimes \boldsymbol{c}_f$$

where $\boldsymbol{a}_f$, $\boldsymbol{b}_f$ and $\boldsymbol{c}_f$ are the fth columns of the loading matrices **A**, **B** and **C** respectively [9].

3.1. Uniqueness

An obvious advantage of the PARAFAC model is the uniqueness of the solution. In bilinear methods there is a 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 [10–12], 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 in PARAFAC. If the data is indeed trilinear, the true underlying spectra (or whatever constitute the variables) will be found if the right number of components is used and the signal-to-noise ratio is appropriate [13–15]. This important fact is what originally initiated R. A. Harshman to develop the method based on an idea from 1944 [16]. It is a very strong feature, which gives the PARAFAC model an unsurpassed advantage.

Leurgans et al. [17] among others have shown, that

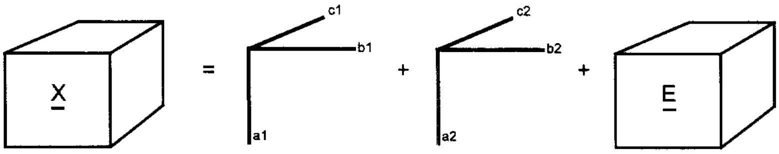


Fig. 1. A graphical representation of a two-component PARAFAC model of the data array $\underline{\mathbf{X}}$.

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 linearly dependent. A good example of this is given in the second application below. Kruskal [15,18] gives even less restrictive conditions for when unique solutions can be expected. He uses the k-rank of the loading matrices, which is a term introduced by Harshman and Lundy [19]. If any combination of k^1 columns of **A** has full column-rank, and this does not hold for $k^1 + 1$, then the k-rank of **A** is k^1. 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

$$k^1 + k^2 + k^3 \geqslant 2F + 2,$$

then the PARAFAC solution is unique. k^1 is the k-rank of **A**, k^2 is the k-rank of **B**, k^3 is the k-rank of **C** and F is the number of PARAFAC components sought.

The mathematical meaning of uniqueness is that the estimated PARAFAC model cannot be rotated without a loss of fit, as opposed to two-way analysis where one may rotate scores and loadings without changing the fit of the model. A unique solution therefore means, that no restrictions are necessary to identify estimate the model apart from trivial variations of scale and column order. For appropriate noise, i.e. random and not to severe, it also holds that the true underlying trilinear model will be the model with the best fit. Therefore the true and estimated models must coincide when the right number of components is chosen.

3.2. *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 [15,18], ten Berge et al. [20] and ten Berge [21] have worked on this issue. A 2 × 2 matrix has the 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, which is given by the tensor product of three vectors [9]. The rank of a three-way array is equal to the minimal number of triads necessary to describe the array. For a 2 × 2 × 2 array it turns out, that the maximal rank is three! This means that there exist 2 × 2 × 2 arrays that cannot be described using only two components. An example can be seen in [20]. For a 3 × 3 × 3 array the maximal rank is five (see for example [18]). These results may seem strange, but are due to the special structure of the multilinear model compared to the bilinear. Furthermore Kruskal has shown that if for example 2 × 2 × 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 has positive volume. The practical implication of this is yet to be seen, but the rank of an array might have importance when one wants to create a multi-way array 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 × 8 × 8 array [1,14,19]. This shows that parsimonious 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 not explicit rules for determining the maximal rank of arrays in general, except for the two-way case, and some simple three-way arrays.

4. Implementation

4.1. *Alternating least squares*

The solution to the PARAFAC model can be found by alternating least squares (ALS) by successively assuming the loadings in two modes known and then estimating the unknown set of parameters of the last mode. This is also how the model was initially proposed to be estimated. Consider a 2 × 2 × 2 array sliced into two 2 × 2 matrices as shown in Fig. 2.

Consider then a one-component PARAFAC model of this array. This model can also be written in terms of two bilinear models as shown in Fig. 3.

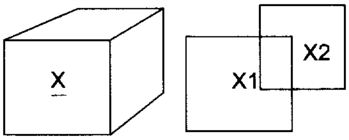


Fig. 2. A 2×2×2 three-way array $\underline{\mathbf{X}}$ can be represented by two matrices **X1** and **X2**.

This way of representing the three-way model as two two-way models can be further modified by simply unfolding the array, i.e., concatenate the two matrices **X1** and **X2** and correspondingly modify the loading vectors (Fig. 4). All three versions are equivalent and merely different graphical formulations of the same model.

If an estimate of $\boldsymbol{b}$ and $\boldsymbol{c}$ is given, it is now easily seen that $\boldsymbol{a}$ can be determined by the least-squares solution to the model $\boldsymbol{a}(\boldsymbol{b} \otimes \boldsymbol{c}) = \mathbf{X}$, where $(\boldsymbol{b} \otimes \boldsymbol{c})$ is interpreted as the row vector obtained as the properly arranged tensor product of the vectors $\boldsymbol{b}$ and $\boldsymbol{c}$ and $\mathbf{X}$ is the unfolded array of size $I \times JK$ as shown in Fig. 4. If the vector $(\boldsymbol{b} \otimes \boldsymbol{c})$ is called z or $\mathbf{Z}$ in case of more than one component, the model defining **A** is

$$\mathbf{X} = \mathbf{AZ}$$

The conditional least squares estimate of **A** is

$$\mathbf{A} = \mathbf{XZ}'(\mathbf{ZZ}')^{-1}$$

The general PARAFAC ALS algorithm can be written

(0) Decide on the number of components, F

(1) Initialize **B** and **C**

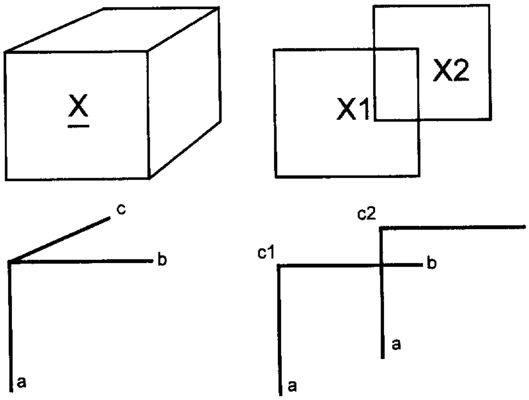
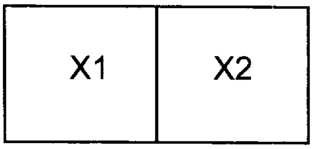


Fig. 3. A trilinear decomposition expressed as either a model of the three-way array or two models of two two-way arrays.



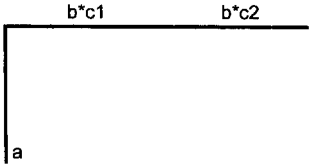


Fig. 4. The principle of unfolding applied to a three-way array (and the corresponding one-component PARAFAC model).

(2) Estimate **A** from $\underline{\mathbf{X}}$, **B** and **C** by least squares regression

(3) Estimate **B** likewise

(4) Estimate **C** likewise

(5) Continue from 2 until convergence (little change in fit or loadings).

A is an $I \times F$ matrix containing in its fth column the fth loading vector. **B** and **C** are defined likewise.

In step 2 $\underline{\mathbf{X}}$ is unfolded to an $I \times JK$ matrix and the fth row in the $F \times JK$ matrix **Z** is defined as

$$z_f = (\boldsymbol{b}_f \otimes \boldsymbol{c}_f).$$

The estimate of **A** is then determined as shown above. For estimating e.g., **B**, $\underline{\mathbf{X}}$ is unfolded to an $J \times IK$ matrix and **Z** becomes an $F \times IK$ matrix calculated from **A** and **C**. **B** is then found as $\mathbf{XZ}'(\mathbf{ZZ}')^{-1}$. For three-way PARAFAC computationally efficient formulations can be seen in e.g. [22]. The ALS algorithm will, in each iteration, improve (or not worsen) the fit of the model. If the algorithm converges to the global minimum, which is most often the case for well-behaved problems, the least-squares solution to the model is found.

ALS is an attractive method because it ensures an improvement of the solution in every iteration, but a major drawback of ALS, is the time required to estimate the models, especially when the number of variables is high. Several hundred or thousands of iterations are sometimes necessary before convergence is achieved. With a data array of size 50 × 50 × 50 a model might very well take hours to calculate on a moderate PC (depending on implementation and convergence criterion of course). 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. In the implementation used here two acceleration methods have been built in (for others see e.g. [1,22,23]).

4.1.1. Compressing

Like in bilinear PCA one most often seeks a low-dimensional representation of a high-dimensional array in PARAFAC. This implies, that the data array is redundant, i.e., there is collinearity between the variables. Consider a 5 × 6 × 500 array, where the third 500-dimensional mode might be spectral. Using ALS on such an array is computationally costly. From the theory of PCA it is known, that the variations in the spectra can be well represented by a low-dimensional score matrix, that contains the main systematic part of the variations. If the data array is unfolded keeping the high-dimensional mode intact, one obtains a 30 × 500 dimensional matrix. By two-way PCA we can describe most of the variation in this matrix by a score matrix, of say, dimension 30 × 5. Folding back the matrix to a three-way array, the array now has the dimensions 5 × 6 × 5. Calculating the PARAFAC model on this low-dimensional array takes only a fraction of the time required to calculate the PARAFAC model of the high-dimensional array. The estimated model is only describing the score matrix and not the original array, but it is only in the compressed mode, that the estimated loadings differ. We can convert the calculated loadings in that mode into the original variable space by multiplying the loadings from the PARAFAC model — which are loadings in a score space — with the loadings from the PCA. The PARAFAC model achieved hopefully equals a PARAFAC model calculated from the original array. To ensure this, ALS is applied to the original array using the calculated loadings and scores as starting values. If the model is good only few extra iterations will be necessary in the high-dimensional space. If several modes are high-dimensional the compressed array can of course be compressed further in another mode.

The compression of modes has been implemented so that compression is done whenever the number of variables in one mode exceeds the number of factors sought with ten. If one mode consists of 20 variables and a 5 factor model is estimated this mode is thus compressed as the dimension of the mode (20) exceeds 5 + 10. The number of principal components to compute is set to the number of factors in the PARAFAC model plus two. These settings work for many types of problems often encountered in our research group, although sometimes other settings may be optimal, because the optimal settings depend on the type of data investigated (primarily the signal-to-noise ratio). When implementing PCA, one has to pay attention to the time demand of the PCA algorithm. Working on cross-product matrices instead of the raw data can speed up the algorithm substantially, if one mode of the unfolded array is very large compared to the other mode.

When a nonnegativity constraint is used (see later) compressing by PCA is not appropriate. Instead one can use a subset of the original variables to estimate an initial model. This submodel can be found on a smoothed version of the original data to ensure that important aspects are not missing.

4.1.2. Extrapolating

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 elementwise. There are two good reasons for using the temporal information in the iterations of 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. The main fraction of iterations are used for minor modifications of these factors. (ii) The changes in each element of the factors is 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 with quadratic fit and adaptive parameters tend to be so slow, that there is no gain in computing time. Several implementations have been tried ending up with an algorithm by Claus A. Andersson, which works fast in our Matlab code. At the ith iteration the estimated factor loadings **A**, **B** and **C** are saved as **A1**, **B1** and **C1**. After the $(i+1)$th iteration a linear regression is performed for each element to predict the value of that element a certain number of iterations ahead. As only two values of each element are used in the regression, the prediction can simply be written

$$\text{Anew} = \mathbf{A1} + (\mathbf{A} - \mathbf{A1})d,$$

where d is the number of iterations to predict ahead. Making d dependent on the number of iterations have proved useful, and specifically letting

$$d = \text{it}^{1/3}$$

where it is the number of iterations. When applying the extrapolation, it is important not to extrapolate during the first, say five, iterations, because the variations in the elements are very 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 $\text{it}^{1/n}$ to $\text{it}^{1/n+1}$.

4.1.3. Initialization

Good starting values for the ALS algorithm could potentially speed up the algorithm and ensure, that the global minimum is found. Several possible kinds of initializations have been proposed. Harshman and Lundy [19] advocate for using random starting values and starting the algorithm from several different starting points. If the same solution is (essentially) reached several times there is little chance that a local minimum is reached due to an unfortunate initial guess. In [24–27] it is proposed to use starting values based on generalized eigenvalue decompositions. These eigenvalue decompositions are all comparable to the generalized rank annihilation methods, where two samples are used to estimate the loadings in the second and third mode. With respect to speed, however, there is often no advantage of using these initialization methods. Rather, the advantage is if the ALS algorithm tends to get stuck in local minima, a good initialization might help overcoming this problem. Our experience is that local minima is seldom a problem if the data are trilinear, but others have reported differently [28,29]. Another practical problem with these methods is how to extend them to higher orders. This problem has not yet been addressed.

4.2. Stopping criterion

The importance of using a suitable stopping criterion has been mentioned by several authors. It sometimes occurs, that even small changes in the fit can be associated with huge differences in the estimated loadings, because the response surface of the least squares error function is very flat [1]. This is especially true if some underlying phenomena are highly correlated. As a safeguard against this, one can run the algorithm twice. If the algorithm has truly converged, the two solutions will essentially be identical. If the algorithm has not converged it is unlikely, that the estimated solutions are identical if a random initialization has been used. A common criterion to use, is to stop the iterations when the *relative* change in fit between two iterations is below a certain value (e.g., 10^{-6}). In some cases a low change in the relative changes of the loadings is used [30]. The difference between these two approaches is not clear.

4.3. Constraining the solution

Constraining the PARAFAC solution can sometimes be helpful in terms of interpretability or stability of the solution. The fit of a constrained model will always be lower than the fit of an unconstrained model, but if the constrained model is more interpretable and realistic this may justify the decrease in fit. In psychometrics orthogonalizing has been described as a means of overcoming problems with unstable solutions [22]. For the first mode, an orthogonal least squares solution to the PARAFAC model can be estimated as

$$\mathbf{A} = \mathbf{XZ}'(\mathbf{ZX}'\mathbf{XZ}')^{-0.5}$$

X being $I \times JK$ and **Z** being $F \times JK$ as defined before [23,31]. This estimation method also normalizes the loadings. Unless all modes are to be orthogonalized, this is not a problem, but merely a matter of scaling. Models estimated under orthogonality constraints will differ from models estimated without this constraint. The models however, will still be mathematically unique, only the models will be least squares models *under* the orthogonality constraint. Orthogonalization is not often used in chemometrics, because it hinders the straightforward interpretation of the loadings, but for more explorative purposes it can be useful. It also enables more straightforward interpretation of e.g., data arising from experimentally designed data, as the sum-of-squares described by the model can be partitioned into contributions from individual components.

Another and more often used constraint is to require nonnegative loadings in e.g., a spectral data set. While orthogonalizing is based on a purely mathematical basis, a nonnegativity constraint is often chosen on the basis of specific knowledge of the data; for instance that absorbance measurements should be positive if proper blanking is used. To find the least squares loading vector given a nonnegativity constraint is somewhat complicated. A general method has been described by Lawson and Hanson [32]. This method is implemented in Matlab as NNLS. An equivalent but faster algorithm is available from the author on request.

For certain types of data it can be fruitful to apply constraints on the interrelationship between the loading vectors. For closed systems one might for example want to restrict the sum of the $F-1$ first loading vectors to equal the Fth loading vector in one mode to ensure, that the solution follows the known behavior of the underlying phenomena. This can be accomplished using equality constraints in the least squares solutions [32–36].

It is possible to fix certain loadings to predefined values (usually zero or one) by adjusting for these elements during the regression steps in the ALS algorithm. For other knowledge based types of constraints see [37,38].

4.4. Missing values

Missing values in PARAFAC are easily handled by iteratively estimating the missing values. This estimate is given for free when iterating in the ALS algorithm. The estimate of the *ijk*th element of $\underline{\mathbf{X}}$ is

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

The missing elements are consistently replaced with the estimates of the elements, and the ALS is continued until no changes occur in the estimates of these missing elements and the overall convergence criterion is fulfilled. It is also possible to handle missing values by weighted regression setting the weights of missing values to zero.

5. Preprocessing

Preprocessing of three-way 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 the first mode can be done by unfolding the calibration array to an $I \times JK$ matrix, and then center this matrix as in ordinary PCA:

$$x_{ijk}^{\text{cent}}=x_{ijk}-\overline{x_{jk}}$$

where

$$\overline{x_{jk}}=\frac{\sum_{i=1}^{I}x_{ijk}}{I}$$

This is often referred to as single-centering. The centering shown above is called centering *across* the first mode, which is the terminology suggested in [39]. 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, one has to do this 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 [39–41] the effect of both scaling and centering on the trilinear 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 model. Centering one mode at a time essentially removes any constant levels in that particular mode. Centering for example matrices instead of

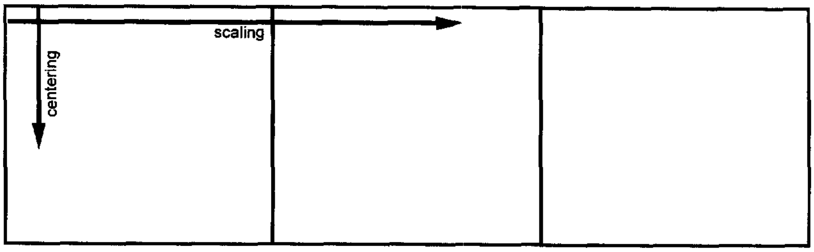


Fig. 5. Three-way unfolded array, with rows constituting one intact mode. Centering must be done across the columns of this matrix, while scaling has to be done on the rows.

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. One may, for instance, know that the true model consists of a set of PARAFAC terms and one overall level, which might incline one to estimate a PARAFAC model on 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. The model obtained as the grand level plus the PARAFAC model is not the global least squares estimate given the required structure. The grand level and the PARAFAC model would have to be estimated simultaneously to obtain the global least squares model. Instead the subtraction of the grand level shifts the data, so that an extra spurious component will be necessary to describe the variation [40]. Scaling in multi-way analysis also has to be done, taking the trilinear model into account. One should not, as with centering, scale column-wise, but rather whole 'slabs' 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 all columns where variable j occurs. This means that one has to scale whole matrices instead of columns. For a four-way array, one would have to scale three-way arrays. Mathematically scaling can be described

$$x_{ijk}^{\text{scal}} = \frac{x_{ijk}}{s_i}$$

where s_i can be defined as

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

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 [39]. Another complicating issue, is the interdependence of centering and scaling. In general 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 [41]. Hence only centering across arbitrary modes or scaling within one mode is straightforward, and not all combinations of iterative scaling and centering will converge. These rules may sound complicated, but in practice it need not influence the outcome much if the iterative approach is not used. Scaling to a sum-of-squares of one is arbitrary anyway and it may be just as defensible 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. In the Matlab code available from the Internet (see materials and methods) an M-file is given to perform the iterative scaling and centering procedures.

A common rule of thumb is to center across the mode of interest, but of course the purpose of centering is to remove constant levels, hence knowledge of the data might guide the proper preprocessing. The appropriate centering and scaling procedures can most easily be summarized in a figure where the array is shown unfolded to a matrix (Fig. 5). Centering must be done across the columns of this matrix, while scaling should be done on the rows of this matrix.

6. Assessing the solution

6.1. Postprocessing

As in two-way PCA different scalings of the solution can be used. Scaling one loading vector by a constant does not change the model, if another loading vector of the same component is scaled accordingly by the inverse of the same constant. The loading vectors of the second and third mode can be normalized to length one. The scores or loadings of the first mode will then show the sum-of-squares, SS, of each component as

$$SS_f = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \left(a_{ij} b_{jf} c_{kf}\right)^2 = \sum_{i=t}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} a_{if}^2$$

in the same way as in bilinear PCA. Due to the nonorthogonality of PARAFAC solutions in general, one cannot simply add the sum-of-squares for all components to get the total sum-of-squares. To judge a component's influence one should compare the sum-of-squares of the original data with the sum-of-squares of the data subtracted the specific component.

It is also common practice to scale the loading vectors, so the maximal loading is one to enhance visual interpretability. Other scalings can be applied guided by the problem. As there are no predefined order of the components the order of the components might not be the same in two estimates of the same data set, even though the models are identical. This is just a matter of permutations. One can of course build in a sorting in the algorithm, so that components are sorted e.g., in order of their descriptiveness of the data as in two-way PCA.

6.2. Leverages and residuals

Leverages and residuals can be used for influence and residual analysis. As the loading vectors are not orthogonal the leverages have to be calculated as

$$\boldsymbol{v} = \mathrm{diag}\left(\mathbf{A}(\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\right),$$

A being replaced with the proper loading matrix (**A**, **B** or **C**), and diag meaning the diagonal of the matrix. The leverage for the ith sample or variable, v_i, is the ith element of $\boldsymbol{v}$ and has a value between zero and one [42]. 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 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. As leverages are actually developed for regression analysis, the term squared Mahalanobis distance might be more appropriate for a decomposition method as PARAFAC, but as leverages are also widely used in two-way PCA, the term leverage is preferred here.

Residuals are easily calculated by subtracting the model from the data. These residuals can be used for calculating variance-like estimates [43] or they can be plotted in various ways to detect trends and systematic variation.

6.3. Number of components

It is difficult to decide the best rank of a PARAFAC model. This area is not very well founded yet, and research is absolutely called for. It is not in general profitable to use cross-validation as in bilinear PCA. In PCA one deflates the **X** matrix after calculation of each component, and therefore eventually the components describe noise instead of systematic variation. This is seen as an increase in the residuals of modeling independent samples. This is the basis of using cross-validation or jackknifing. Sometimes the increase in the residual variance is not very pronounced which makes it difficult to correctly estimate the proper rank of the model. This situation can be even worse in PARAFAC. In PARAFAC, one does not deflate the array, because the trilinear model calculated simultaneously for all components can be shown to fit the array better, than if the components were calculated successively as is possible in PCA [28]. As a consequence, extracting too many components does not only mean that noise is being increasingly modeled, but also that the true factors are being modeled by more (correlated) components. In general, one will therefore not see as steep an increase in a cross-validation procedure as in the bilinear models.

There are three main ways of determining the correct number of components: (1) Split-half experiments, (2) judging residuals and, (3) compare with external knowledge of the data being modeled. If the PARAFAC model is to be used for e.g., calibration one can of course do cross-validation on the predictions of the dependent variable to find the optimal model.

Ad. (1). Harshman and Lundy [19] advocate for using split-half experiments. The idea is to divide the data into two halves and then make a PARAFAC model on both halves. Due to the uniqueness of the PARAFAC model, one will obtain the same result — same loadings in the nonsplitted modes — on both data sets, if the correct number of components is chosen. To judge if two models are equal one must remember the intrinsic indeterminacy in PARAFAC: The order and scale of a model may change if not fixed algorithmically. If a wrong number of components is chosen in the split-half experiment, there is a good chance, that the two models will not be equal, due to the differences in the different samples. When doing a split-half experiment one has to decide which mode to split. In general one should split the data in a mode with a sufficient number of *independent* variables/samples. If one has 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. Any number of components would yield the same result if the spectra behave additively.

Ad. (2). As in bilinear models, one can judge a model on the fit. 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 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 [44] give the following degrees of freedom for a trilinear PARAFAC model

$$\mathrm{dof}(F) = IJK - F(I + J + K - 2),$$

and

$$\mathrm{dof}(F) = IJKL - F(I + J + K + L - 3),$$

for a quadrilinear PARAFAC model. I, J, K and L are the dimensions of the first, second, third and fourth mode respectively and F is the number of components in the model. These degrees of freedom might be used for explorative purposes, but they are not to be taken as statistically exact numbers of degrees of freedom. Such are currently not available.

Ad. (3). With experience one gets a feeling for which results are good and which results are bad. This can be very important for making good models. The use of experience and intuition can also be more systematically used. Often one knows certain things about the underlying phenomena in the data. Spectra of certain analytes might be known, the shape of chromatographic profiles might be known or the nonnegativity of certain phenomena might be known. These kinds of hard facts can be very informative when comparing different models. In [38,44] some examples on how to use residuals and external knowledge to choose the appropriate number of components are shown.

6.4. Degenerate solutions

Degenerate solutions are sometimes encountered. Degenerate solutions are solutions hard to handle for the PARAFAC model. The estimated models are often unstable and unreliable. A typical sign of a degenerate solution, is that loading vectors of the same mode have high correlations. Most often a degenerate solution is characterized by two PARAFAC components showing equally shaped loading vectors in all modes with two or none of the pairs of loading vectors of each mode positively correlated and one or three negatively correlated. An indication of degenerate solutions can thus be obtained by monitoring the correlation between all pairs of loading vectors. In practice the triple cosine, TC, of all combinations of components is used. TC is defined as

$$TC_{ij} = \cos(\boldsymbol{a}_i, \boldsymbol{a}_j)\cos(\boldsymbol{b}_i, \boldsymbol{b}_j)\cos(\boldsymbol{c}_i, \boldsymbol{c}_j)$$
$$= \frac{\boldsymbol{a}'_i\boldsymbol{a}_j}{\|\boldsymbol{a}_i\|\|\boldsymbol{a}_j\|}\frac{\boldsymbol{b}'_i\boldsymbol{b}_j}{\|\boldsymbol{b}_i\|\|\boldsymbol{b}_j\|}\frac{\boldsymbol{c}'_i\boldsymbol{c}_j}{\|\boldsymbol{c}_i\|\|\boldsymbol{c}_j\|}$$

i and j indicating the ith and jth component. Mitchell and Burdick [29,45] refer to TC as the uncorrected correlation coefficient (UCC). The TC value can be shown to correspond to the cosine of the angle between two vectors, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, where $\boldsymbol{x}_i$ is the vector obtained by properly unfolding the tensor product of all loading vectors with index i (like the $\boldsymbol{b}$ and $\boldsymbol{c}$ vector in Fig. 4 only $\boldsymbol{a}$ should also be included).

A TC value close to -1 indicates a degenerate solution. A TC value lower than -0.85 is an indication of a troublesome model according to [46], but this can just be taken as a rule of thumb. Furthermore, for degenerate solutions the TC value will typically continue to worsen for more iterations. If the numerically high TC value is just caused by a poor initialization, the TC value will decrease again numerically. If several new estimations of the same model are consistent and not degenerate, the degenerate solution can be discarded as an accidental local minimum. If decreasing the convergence criterion does not eliminate degeneracy the cause is most often one of three [41,47]. (i) Too many components are extracted. This will be easily recognizable, by the fact that models with fewer components yield nondegenerate solutions. Often extracting too many components will give high positive TC values just as well as negative, which is not the case for real degenerate solutions. Split-half experiments will also help to distinguish this situation from more serious problems. (ii) Poor preprocessing has been applied, which can be characterized by degenerate solutions even for a low number of components, and when other information indicates that further systematic information is present. (iii) The last situation of degeneracy occurs when the model is simply inappropriate, for example, when the data are not trilinear as the model. In [48] some of these situations are referred to as *two-factor degeneracies*. When two factors are interrelated a Tucker3 model is appropriate and estimating PARAFAC models with too few factors can yield degenerate models that can be shown not to converge to a minimum, while estimating models with a higher number of components is difficult due to the correlations between the components. An indication of this situation might be, that the estimated two-way rank of the unfolded array is different depending on which mode is unfolded. A PARAFAC model may still be appropriate but if the differences are large, this indicates that some latent variables do not vary across some of the ways, or perhaps vary interdependently. In such a case the Tucker (or restricted versions) or unfold bilinear models might be better [48–50].

Mitchell and Burdick [29,45] investigate degeneracy and find it profitable to do several runs of a few iterations, and only use those runs that are not subject to degeneracy. Another way of circumventing degenerate solutions is by applying orthogonality constraints on the model.

If the variation in one mode is not exactly obeying the linearity of the PARAFAC model, it is possible to eliminate this mode by using it for calculating covariance or cross-product matrices. Fitting the model to these derived data, is called indirect fitting and has been used for longitudinal data [8,51]. Consider a three-way data array where the third mode could be chromatographic. Perhaps the chromatographic profiles of the same analytes change a little from sample to sample due to analytical properties. The data array is therefore almost trilinear, but the differences from sample to sample in the third mode makes it hard to make a sound PARAFAC model. The $I \times J \times K$ array can be seen as J matrices of size $I \times K$. For each j (1 to J) one can calculate an $I \times I$ covariance matrix as $\mathbf{X}'_j\mathbf{X}_j$, where $\mathbf{X}_j$ is the $I \times K$ submatrix of $\underline{\mathbf{X}}$ on the jth level of the second mode. The thus obtained data array has the size $I \times I \times J$ and consists of covariance matrices. The original third mode has vanished and fitting the PARAFAC model to this array will give the following model (compare Eq. (1)).

$$x_{ijk} = \sum_{f=1}^{F} a_{if} a_{jf} b_{kf}^2$$

disregarding the noise. The a's and b's in this model correspond exactly to the a's and b's in the model obtained from the raw data array (Eq. (1)) *if* the loading vectors in the third mode are orthonormal. This, however, is not very likely and a solution to this problem has been suggested by Harshman. He has developed a model called PARAFAC2 [51]. In this model the loading vectors of the third mode can be oblique — nonorthogonal. The PARAFAC2 model has not yet been used very extensively maybe because the implementations so far have been complicated and slow [52].

7. Types of data suitable for PARAFAC analysis

There are three different types of data, that are more or less commonly analyzed by PARAFAC: PCA-like data, analysis of variance — ANOVA — data and multidimensional scaling data. In chemometrics the most well-known application of PARAFAC is for PCA-like data, spectral data for example. The use of PARAFAC for these kind of data should be rather simple following the same strategy as for decomposing bilinear data.

The use of PARAFAC for analysis of variance is rare [53]. However, the use of PCA and related methods for ANOVA has been known for several years (see [54,55] and references in these). The advantage of using PARAFAC for ANOVA is in the way interaction terms are modeled. In a standard ANOVA an interaction between three factors (A, B and C) would be estimated as E_{ijk}, while in a trilinear model, this effect would be estimated as $a_i b_j c_k$ or as a sum of such expressions if more PARAFAC components are estimated. The interaction is not only estimated as a whole, but is modeled as a multiplicative effect of the three different factors. If the multiplicative model is appropriate, the applied restriction ($a_i b_j c_k$ instead of merely E_{ijk}) will give a more interpretable model.

Carroll and Chang [2] who developed PARAFAC (CANDECOMP) simultaneously with Harshman developed it for its application to multidimensional scaling. In psychometrics it has gained widespread use for this purpose, but this will not be touched upon specifically here.

8. Application I: Analysis of variance

8.1. Data

This data set is obtained for exploring the influence and properties of enzymatic browning of vegetables. The primary contributor to enzymatic browning is PPO, polyphenol oxidase [56]. The relationship between PPO activity (expressed as oxygen consumption) and experimental conditions is investigated. For five O_2 levels, three CO_2 levels, three pH values, three different temperatures and three substrate types — all varied independently — the activity of PPO was determined in replicate. Building a calibration model to predict the activity from the experimental conditions would 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 1. The number of samples in the replicated full factorial design is 5 × 3 × 3 × 3 × 3 × 2 = 810. For details on the experimental conditions and a more in-depth discussion on the technological aspects see [57,58].

Table 1
Experimental design

PPO activity	
factor	levels
O_2 (%)	0, 5, 10, 20, 80
CO_2 (%)	0, 10, 20
pH	3.0, 4.5, 6.0
Temp (°C)	5, 20, 30
Substrate	CG, EPI, MIX
Replicates	I, II

In [58] the results obtained with PARAFAC are compared with ANOVA, locally weighted regression and nonlinear methods based on PLS and feedforward neural networks, but here the focus is on PARAFAC and partially ANOVA.

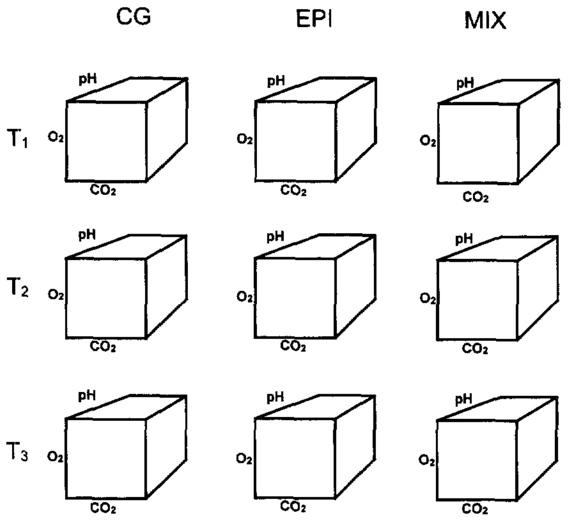


Fig. 6. A graphical representation of the five-way array of activities.

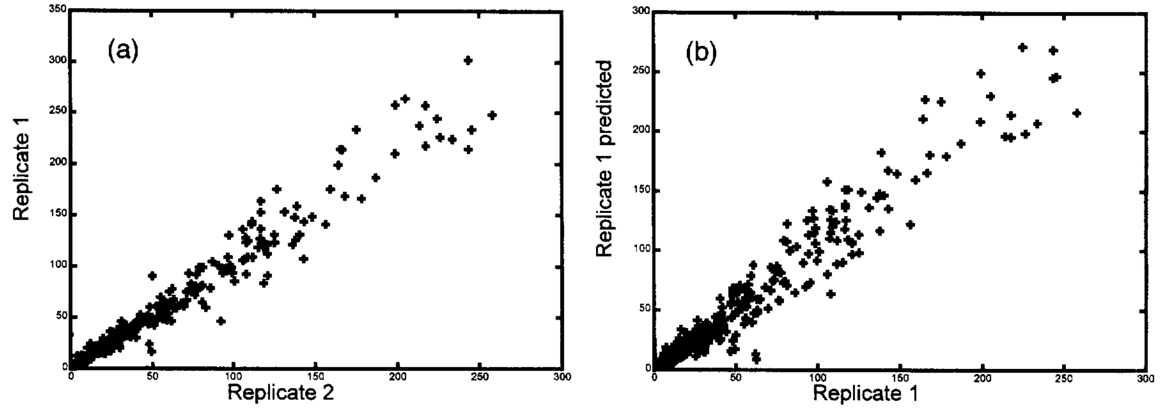


Fig. 7. Activities of one replicate set versus the other (a). Predictions of the activities of one replicate set from a model of the other replicate set (b).

In the PARAFAC model, the data is interpreted as a multi-way array of activities, specifically a five-way array. In a sense, the whole array is seen as one sample namely PPO activity, which is measured at different conditions. The five different modes (ways) are: O_2 (dimension five), CO_2 (dimension three), pH (dimension three), temperature (dimension three), and substrate type (dimension three). The *ijklm*th 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 temperature, the *l*th level of substrate type, for the *m*th pH. The five-way array is depicted in Fig. 6. Each element in the array is the average of the two replicates.

8.2. Results and discussion

Preprocessing of ANOVA data is somewhat difficult and no general guidelines can be given. One must either try different scalings and centerings or use external knowledge for guidance. In this case, the only preprocessing thought to be of potential importance was scaling the oxygen mode. From the residuals of the two sets of replicates, some heteroscedasticty was observed in the oxygen mode. However, scaling the data according to this did not improve the predictions of one replicate set predicted from the other, simply because the elements with high residuals were downweighted and henceforth modeled with even higher error. Other kinds of meancentering and scaling were tried, but without improving the solution [58].

To decide on the number of components, a five-way PARAFAC model was made using the first of the two replicate sets instead of just using the mean of these. The model from this analysis given by the loadings **A**, **B**, **C**, **D** and **E** was compared to the other replicate set. The number of components was chosen to minimize the sum squared prediction error calculated as

$$SS = \left[\left(\sum_{f=1}^{F} a_{ij} b_{jf} c_{kf} d_{lf} e_{mf} \right) - x_{ijklm} \right]^2$$

x_{ijklm} being the *ijklm*th element of the replicate set not used for estimating the model. One component gave the lowest prediction error, which furthermore was in the neighborhood of the intrinsic error of the reference value (standard deviation between replicates 11.9 and standard deviation between the model and the test set 13.4 corresponding to 94% variance explained). The predictions are shown in Fig. 7, where one clearly sees, that the model is very good and comparable to the intrinsic error of the data.

The activity can hence be modeled by a very simple one-component model. The loading vectors of this model are shown in Fig. 8. To predict the activity at a certain setting of the different factors, one simply read the ordinate-values of the five different factors and multiply these. If a low activity is sought it is very easy to see how this can be obtained, i.e. by keeping the temperature, oxygen and pH levels as low as possible.

A one-component solution was also found from a

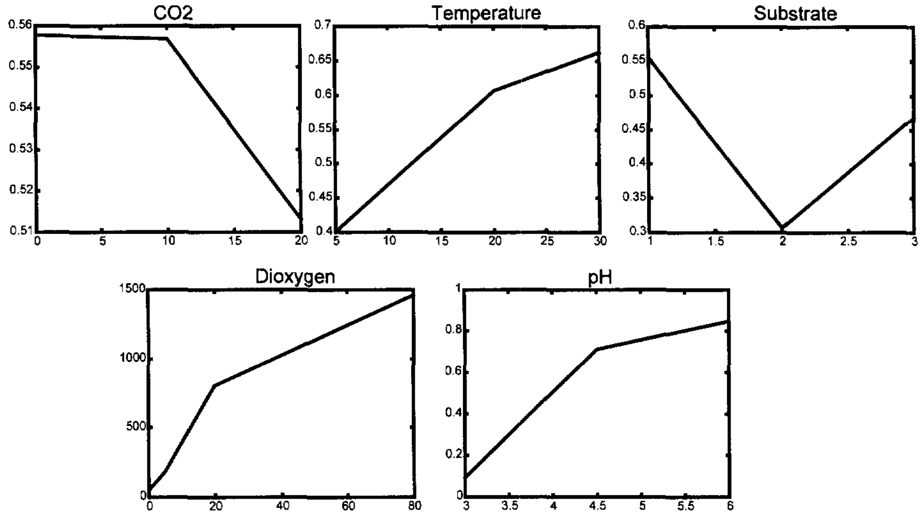


Fig. 8. Loading vectors from a one-component PARAFAC model of enzymatic data.

test set procedure where half the elements of the five-way array were eliminated. The elements were eliminated according to a fractional factorial design. For the remaining data, a PARAFAC model was built with an algorithm that handles missing data. With the model the activity of the left-out samples was predicted for both a one- and a two-component PARAFAC model. The root mean square error of prediction, RMSEP, for a one-component model was 12.6, while for two components an RMSEP of 51.9 was found.

From the PARAFAC loadings it is possible to predict the effect at any level of the factors investigated. To validate this, a PARAFAC model was made leaving out all samples with 20% oxygen. A curvefit of the oxygen loading vector makes it possible to find the oxygen effect of any level between 0 and 80%. For 20% the loading was estimated from a quadratic curvefit of the loading vector. From this value and the loadings of the remaining modes the 27 $(1 \times 3 \times 3 \times 3)$ left-out samples were predicted with an RMSEP of 13.1. This shows that a completely general model is obtained showing the effect of each factors as simply a loading vector.

The data constitute a full factorial design, and analysis of variance is thus an obvious tool for investigating the influence of different factors. However, the problem is highly nonlinear and the results from ANOVA are hard to interpret. A standard ANOVA performed in SAS pointed to the following model.

$$\begin{aligned}\text{Activity} = {} & A + B + C + D + E + AB + AC \\ & + AD + AE + BC + BC + BE + CD \\ & + CE + DE + ABC + ABD + ABE \\ & + ACE + ADC + ABDC + ABCE \\ & + ADE + ABDE + DCE + ADCE,\end{aligned}$$

A, B, C, D and E being the main effect of O_2, CO_2, temperature, substrate and pH, and e.g. ACD the interaction between O_2, temperature and substrate. PARAFAC on the other hand suggested that the five-way multiplicative interaction term is sufficient to model the variations

$$\text{Activity} = ABCDE,$$

or rather

$$\text{Activity} = a_i b_j c_k d_l e_m.$$

It is interesting, that one of the few interactions not significant in the ANOVA model is the five-way interaction term! Even though more sophisticated ANOVA methods can be used, this example illustrates, that choosing the right mathematical method can greatly influence the outcome, both with respect to prediction and interpretability. When using half the samples to estimate a model and predict the left-out samples an RMSEP of 35.3 and 12.6 was achieved for the ANOVA and the PARAFAC model respectively. The reason for the better results with PARAFAC is simply that the underlying multiplicative model is more appropriate for the enzymatic data, than is the mathematical model underlying ANOVA. It is to be expected that the main variation in the activity is caused by something, that could be approximately multiplicative. If pH is three little activity is observed no matter the oxygen level, but if pH is 6, the activity of PPO is extremely dependent on oxygen.

8.3. Further modification of the model

Mathematically the one-component PARAFAC model could also be obtained by ANOVA by a logarithmic transformation of the data, but PARAFAC offers an even more general model and is furthermore unique which is not the case for the ANOVA model. In an effort to make a better model a constrained modification of PARAFAC was used where some loadings were forced to ones, thereby permitting modeling of lower-order interactions and main effects. The best model was found to consist of one main additive effect, one four-way and one five-way interaction. This model, specifically

$$x_{ijk} = a_{i1} + a_{i2}b_{j2}c_{k2}d_{l2} + a_{i3}b_{j3}c_{k3}d_{l3}e_{m3}$$

estimated from one of the replicate sets gave a model, that predicted the other replicate set with an RMSEP of 12.3. This as compared to the intrinsic error between the two replicates, 11.9, and the error obtained predicting with the one-component five-way interaction model, 13.4. However, the model only explained one percent more of the variance than the one-component model, and hence the increased complexity was judged not to be beneficial enough to justify the model. The model was estimated by a three-component PARAFAC where the loadings of the first component were all fixed at the value one except in the first mode. In the second component the loadings of the fifth mode were forced to ones. Further investigation is now in progress trying to develop a general multiplicative ANOVA model using PARAFAC as sketched here. Problems with defining degrees of freedom and the numerical obstacles in estimating the constrained PARAFAC models are the most obvious problems to be dealt with. It is noteworthy that this approach can also be used for estimating constant baselines or other lower-order effects in a spectral decomposition.

9. Application II: Unique decomposition of sparse fluorescence data

9.1. Data

This problem is an illustrative example of the unique decomposition obtained by PARAFAC using a nonnegativity constraint. The data set is part of an investigation conducted by Claus A. Andersson at our laboratory. Two samples containing different amounts of tyrosine, tryptophane and phenylalanine were measured by fluorescence (excitation 250–300 nm, emission 250–450 nm, 1 nm intervals). The array to be decomposed is hence 2 × 51 × 201. The samples were measured on a PE LS50B spectrofluorometer with excitation slitwidth of 2.5 nm, an emission slitwidth of 10 nm and a scanspeed of 1500 nm/s. Originally five samples were used, which could be decomposed by unconstrained PARAFAC without problems. However, to show how to incorporate external knowledge in the decomposition only two of the samples are used here.

The theoretical multilinearity of fluorescence has been described in [17,28,59]. In Fig. 9 one of the two samples is shown. Notice the Rayleigh scatter in the left part, which is not multilinear in its nature [60]. Rayleigh scatter should be avoided in a multilinear decomposition if possible, and there are three ways of doing that: (i) Only measure the emission above the excitation, if this wavelength area contains sufficient

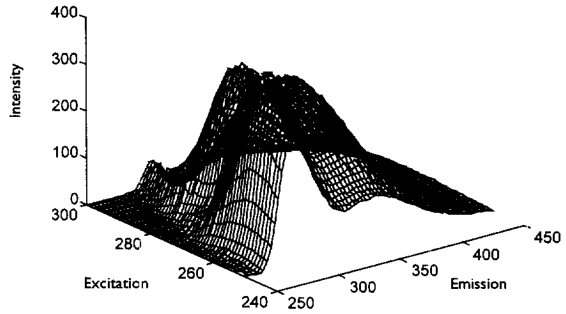


Fig. 9. A plot of the fluorescence of a sample containing Tyr, Trp and Phe. Notice the Rayleigh scatter in the left corner.

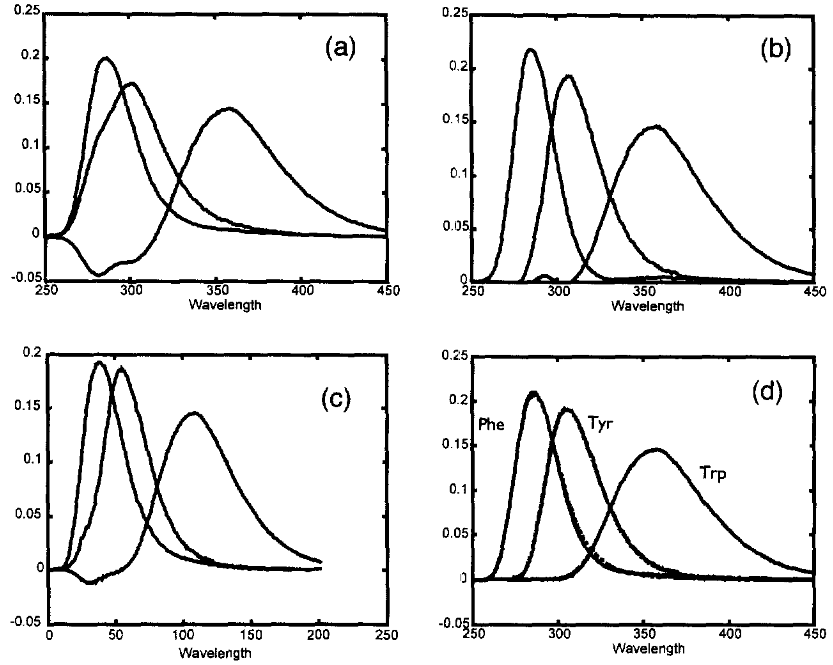


Fig. 10. Estimated and true emission spectra: (a) Unconstrained PARAFAC, (b) NNLS PARAFAC, (c) PARAFAC with missing elements and (d) missing elements and NNLS. In (d) the true spectra are also shown.

information; (ii) Use curvefitting in some form to estimate the emission in the neighborhood of the excitation wavelengths; (iii) Measure a blank and subtract this measurement from the sample measurement. This, however, can be problematic if the Rayleigh scatter is mainly caused by particles in the sample. In this experiment nothing was done to eliminate the Rayleigh scatter initially.

9.2. Results and discussion

A three-component PARAFAC solution should give the correct solution if the trilinear model is appropriate. The emission loadings of a three component PARAFAC solution is shown in Fig. 10a. The spectrum corresponding to tryptophane has large negative areas. It was concluded that the decomposition was difficult due to the low variability (two samples) and knowing that the fluorescence spectra and concentrations should be positive, it was natural to constrain the PARAFAC loadings to positive values. In Fig. 10b the estimated emission loadings are shown using nonnegativity constraints. The spectra are quite similar to the pure spectra of the analytes, but for tryptophane there is a small hump below 300 nm caused by the non-multilinear Rayleigh scatter. To avoid this it was tried to set all variables influenced by Rayleigh scatter to missing values and then estimate the corresponding PARAFAC model. The result can be seen in Fig. 10c. Apparently this alone is not sufficient to ensure a good curve resolution for the tryptophane spectrum. Combining the missing elements approach with the nonnegativity constraint helps the model focuses on the right aspects of the data and the estimated loadings in Fig. 10d are shown together with the pure spectra. As seen the estimated loadings are now quite similar to the pure spectra. The estimated excitation spectra are shown in Fig. 11.

The model precisely estimates the three pure spectra, even though there are only two independent samples, and the excitation spectra of tyrosine and tryptophane are very alike (correlation 0.93). According to the rule mentioned in the paragraph on uniqueness, it is theoretically possible to estimate these three different spectra correctly if only the concentrations vary independently pairwise and no spectra are lin-

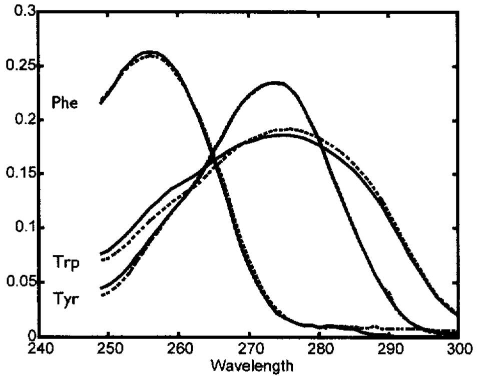


Fig. 11. Estimated excitation spectra using missing elements and nonnegativity constraints. The thick lines are the pure spectra of Trp, Tyr and Phe.

early dependent on any of the others. However, due to Rayleigh scatter, noise and spectral alikeness unconstrained PARAFAC was not sufficient in this case for resolving the spectra.

The four different models differ mainly in the area of the Rayleigh scatter, but for all models the fit to the non-Rayleigh part of the data is almost identical. The unconstrained PARAFAC explains 99.957% of the variation, the nonnegativity constrained model explains 99.958% of the variation, the missing elements model explains 99.974%, and the combined missing and nonnegativity model explains 99.973%. The little difference in explained variance clearly supports that the preconceived assumptions of nonnegativity and inappropriateness of the Rayleigh-scatter are valid. Otherwise the altered models would have had significantly poorer fit than the unconstrained model. It was also tried to resolve the spectra by using generalized rank annihilation as described in [26], but the result was similar to, though worse than, the result using PARAFAC with missing values.

The loadings of the sample mode are estimates of the concentrations of the analytes if the right number of components has been chosen. Due to the scaling indeterminacy in PARAFAC we cannot estimate the concentration of any of the analytes without knowing the concentration in one sample. Suppose the concentrations of the analytes in the first sample are known, we can then scale the PARAFAC solution and compare the concentration estimates of the second sample with the true concentrations. The result is shown in Table 2.

Table 2
Predictions of concentrations in the second sample

True concentration	Predicted concentration
8.8×10^{-7}	7.8×10^{-7}
4.4×10^{-6}	3.5×10^{-6}
3.0×10^{-4}	2.3×10^{-4}

Though the errors are large relatively, they are in the right neighborhood.

10. Application III: Prediction of amino-N in sugar samples from fluorescence

10.1. Data

As for bilinear PCA, the outcome of a PARAFAC model can be used as input to other models, most often for regression. In this example 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). The amino-N content was also determined by a standard wet-chemical procedure as described in [61]. Following more or less the strategy of PCR a PARAFAC 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.

10.2. Results and discussion

Three different PARAFAC calibration models were made: One using raw fluorescence data and an unconstrained PARAFAC model, one using raw data and nonnegativity constraints on the emission mode, and one using meancentered data and unconstrained PARAFAC. The models were made using test set validation with 49 samples in each set. A PARAFAC model was estimated and a regression model estimated from the scores of the PARAFAC model. The scores of the test set samples were calculated from the

Table 3
Percentage of variance explained of the dependent variable (amino-N) of the test set. Each column correspond to a different model and each row to the number of latent variables/components used. Bold numbers indicate variance explained 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)

	PARAF AC (raw)	PARAF AC (meancentered)	PARAF AC (NNLS)	Two-way PLS	Three-way PLS	Tucker	PCR
1 LV	84.0	84.1	84.0	84.4	84.2	84.3	83.9
2 LV	85.4	85.4	85.5	86.6	86.1	84.8	85.7
3 LV	85.2	85.4	85.2	88.5	88.9	85.3	86.8
4 LV	87.1	85.1	86.8	**91.6**	**91.4**	**88.0**	87.2
5 LV	**91.2** (99.8)	**90.7** (99.9)	**91.1** (99.8)	**91.9** (96.0)	**92.3** (95.7)	87.7 (99.8)	**88.0** (99.9)

excitation and emission loadings from the PARAFAC model and from these the estimated amino-N content was determined from the regression model. For comparison the results of using multi-way PLS (N-PLS, [62]), ordinary two-way PLS, Tucker3 regression, and two-way principal component regression (PCR) are also calculated. N-PLS is a multi-way calibration model. By N-PLS the array of independent variables is sequentially decomposed to a multi-linear model, in such a way that the scores have maximal covariance with the yet unexplained variation of the dependent variable. The Tucker regression was performed by decomposing the raw data with a Tucker3 model using the same number of components in each mode and using the loadings of the sample mode for regression. PCR was performed using the successively estimated score vectors from a PCA model for regression. The spectral data was meancentered prior to the PCA decomposition.

The results from the calibration models are shown in Table 3. Several interesting aspects are illustrated here. All models obtain optimal or near-optimal predictions around five components. PLS and N-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 using five components. 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, very well illustrates that when PARAFAC is adequate there is no advantage of using more complex models. The constraints imposed in PLS and N-PLS seem to be more adequate. 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 constraints of the scores having maximal covariance with the dependent variable. N-PLS uses only a fraction of the number of parameters that PLS uses to model the spectral data, so in a mathematical sense, N-PLS obtains optimal predictions with the most simple model. Therefore one can argue, that the N-PLS model is the most appropriate model. However, the N-PLS model does not possess the uniqueness properties of PARAFAC. One might therefore also argue that the five-component nonnegativity constrained PARAFAC model is preferable, if the found loadings can be related to specific chemical analytes; an issue that will not be further investigated here.

Using PARAFAC for regression as shown here has the potential for simultaneously providing a model, that predicts the dependent variable, and precisely describes which phenomena in the independent variables, that are crucial for describing the variations in the dependent variable. However the little experience obtained so far in our laboratory indicates that often, one is better of by focusing on either decomposition (PARAFAC) or calibration (N-PLS). Purely spectral data as here is the only type of data, where there seems to be little differences in the predictive ability.

10.3. *M'n'M*

All calculations were done on a 133 MHz Dell PC with 32 Mb RAM. The PARAFAC algorithm was made in the mathematical software Matlab for Windows 4.2c.1 (Mathworks). This implementation works with arrays up to ten ways. It also contains the possibility to constrain loadings to be orthogonal or nonnegative and handles missing data. The algorithm is available from the Internet at http:\\ newton.foodsci.kvl.dk \ foodtech.html. Also available are M-files for PARAFAC and Tucker3 made by Claus A. Andersson, and N-PLS by R. Bro. Other programs for PARAFAC modeling are also available. Richard A. Harshman, Dept. Psychology, Social Science Center, University Western Ontario, London, Ontario, Canada N6A 5C2 has a very general PARAFAC program for three-way analysis which runs in batch mode on PC's. Rob Ross, at http://www.biosci.ohio-state.edu/ ~ rtr/multilin/ muldoc.html offers fortran code for PARAFAC, and Pentti Paatero, Dept. Physics, University of Helsinki, BOX 9, FIN-00014 University, Helsinki, Finland, has made a program for two- and three-way PARAFAC which incorporates nonnegativity constraints and weighted regression. P.M. Kroonenburg's latest version of his three-mode toolbox now contains the PARAFAC model. The program runs in DOS mode. Orders should be sent to P.M. Kroonenburg, Dept. Education, Leiden University, Wassenaarseweg 52, 2333 AK Leiden, The Netherlands.

11. Conclusion

The PARAFAC model and its estimation has been described and its application for ANOVA, curve-resolution and calibration has been exemplified. It is my hope that this tutorial might encourage others to investigate multi-way methods. Multi-way methods have many advantages (and of course shortcomings) that have not yet been fully acknowledged.

Acknowledgements

Rasmus Bro is grateful for support and inspiration from and funds to Professor Lars Munck from Nordic Industry Foundation project P93149 and the FØTEK foundation. Claus A. Andersson, Age K. Smilde and Henk Kiers are thanked for numerous suggestions during this work. Lars Nørgaard, Hanne Heimdal and Claus A. Andersson are thanked for letting me use their data sets. Anonymous referees are thanked for helpful suggestions.

References

[1] 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.

[2] J.D. Carroll, J. Chang, Analysis of individual differences in multidimensional scaling via an N-way generalization of and Eckart–Young decomposition, Psychometrika 35 (1970) 283.

[3] P. Geladi, Analysis of multi-way (multi-mode) data, Chemom. Intell. Lab. Syst. 7 (1989) 11.

[4] A.K. Smilde, Three-way analyses. Problems and prospects, Chemom. Intell. Lab. Syst. 5 (1992) 143.

[5] P.M. Kroonenburg, Three-mode principal component analysis, Theory and applic ations, DSWO Press, Leiden, 1983.

[6] H.A.L. Kiers, Hierarchical relations among three-way methods, Psychometrika 56 (1991) 449.

[7] M.B. Seasholz, B.R. Kowalski, The parsimony principle applied to multivariate calibration, Anal. Chim. Acta 277 (1993) 165.

[8] R.A. Harshman, S.A. Berenbaum, Basic concepts underlying the PARAFAC-CANDECOMP three-way factor analysis model and its application to longitudinal data, in: D.H. Eichorn, J.A. Clausen, N. Haan, M.P. Honzik, P.H. Mussen (Eds.), Present and past in middle life, Academic Press, NY, 1981, pp. 435-459.

[9] D.S. Burdick, An introduction to tensor products with applications to multiway data analysis, Chemom. Intell. Lab. Syst. 28 (1995) 229.

[10] I. Scarminio, M. Kubista, Analysis of correlated spectral data, Anal. Chem. 65 (1993) 409.

[11] L. Sarabia, M.C. Ortiz, R. Leardi, G. Drava, A program for non-orthogonal factor analysis, Trends Anal. Chem. 12 (1993) 226.

[12] N.M. Faber, M.C. Buydens, G. Kateman, Generalized rank annihilation method I: derivation of eignevalue problems, J. Chemom. 8 (1994) 147.

[13] R.A. Harshman, Determination and proof of minimum uniqueness conditions for PARAFAC1, UCLA Working Papers in phonetics 22 (1972) 111.

[14] J.B. Kruskal, More factors than subjects, tests and treatments: An indeterminacy theorem for canonical decomposition and individual differences scaling, Psychometrika 41 (1976) 281.

[15] J.B. Kruskal, Three-way arrays: Rank and uniqueness of trilinear decomposition, with application to arithmetic complexity and statistics, Linear Algebra Appl. 18 (1977) 95.

[16] R.B. Cattell, Parallel proportional profiles and other principles for determining the choice of factors by rotation, Psychometrika 9 (1944) 267.

[17] S. Leurgans, R.T. Ross, R.B. Abel, A decomposition for three-way arrays, SIAM J. Matrix Anal. Appl. 14 (1993) 1064.

[18] J.B. Kruskal, Rank, decomposition, and uniqueness for 3-way and N-way arrays, in: R. Coppi, S. Bolasco (Eds.), Multiway data analyses, Elsevier Science Pub., North-Holland, 1989.

[19] R.A. Harshman, M.E. Lundy, The PARAFAC model for three-way factor analysis and multidimensional scaling, in: H.G. Law, C.W. Snyder, J.A. Hattie, R.P. McDonald (Eds.), Research methods for Multimode data analysis, Praeger, New York, 1984.

[20] J.M.F. ten Berge, H.A.L. Kiers, J. de Leeuw, Explicit Candecomp/PARAFAC solution for a contrived $2 \times 2 \times 2$ array of rank three, Psychometrika 53 (1988) 579.

[21] 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.

[22] 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.

[23] R.A. Harshman, M.E. Lundy, PARAFAC: Parallel factor analysis, Comp. Stat. Data Anal. 18 (1994) 39.

[24] R. Sands, F.W. Young, Component models for three-way data: An alternating least squares algorithm with optimal scaling features, Psychometrika 45 (1980) 39.

[25] 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.

[26] E. Sanchez, B.R. Kowalski, Tensorial resolution: A direct trilinear decomposition, J. Chemom. 4 (1990) 29.

[27] S. Li, P.J. Gemperline, Eliminating complex eigenvectors and eigenvalues in multiway analyses using the direct trilinear decomposition method, J. Chemom. 7 (1993) 77.

[28] S. Leurgans, R.T. Ross, Multilinear models in spectroscopy, Statist. Sci. 7 (1992) 289.

[29] B.C. Mitchell, D.S. Burdick, An empirical comparison of resolution methods for three-way arrays, Chemom. Intell. Lab. Syst. 20 (1993) 149.

[30] X.M. Tu, D.S. Burdick, Resolution of trilinear mixtures: Application in spectroscopy, Stat. Sinica 2 (1992) 577.

[31] N. Cliff, Orthogonal rotation to congruence, Psychometrika 31 (1966) 33.

[32] C.L. Lawson, R.J. Hanson, Solving least squares problems, Classics in Appl. Math., No. 15, SIAM, Philadelphia, 1995.

[33] R.J. Hanson, Linear least squares with bounds and linear constraints, SIAM J. Sci. Stat. Comput. 7 (1986) 826.

[34] H. Späth, Mathematical algorithms for linear regression, Academic Press, Inc., Boston, 1987.

[35] J.L. Barlow, Error analysis and implementation aspects of deferred correction for equality constrained least squares problems, SIAM J. Numer. Anal. 25 (1988) 1340.

[36] 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.

[37] 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.

[38] R.T. Ross, S. Leurgans, Component resolution using multilinear models, Methods Enzymol. 246 (1995) 679.

[39] J.M.F. ten Berge, Convergence of PARAFAC preprocessing procedures and the Deming-Stephan method of iterative proportional fitting, in: R. Coppi, S. Bolasco (Eds.), Multiway data analyses, Elsevier Science Pub., North-Holland, 1989.

[40] J.B. Kruskal, Multilinear methods , Proc. Symp. Appl. Math. 28 (1983) 75.

[41] R.A. Harshman, M.E. Lundy, Data preprocessing and the extended PARAFAC model, in: H.G. Law, C.W. Snyder, J.A. Hattie, R.P. McDonald (Eds.), Research methods for Mulitmode data analysis, Praeger, New York, 1984.

[42] R.D. Cook, S. Weisberg, Residuals and influence in regression, Chapman and Hall Ltd, New York, 1982.

[43] A.K. Smilde, D.A. Doornbos, Simple validatory tools for judging the predictive performance of PARAFAC and three-way PLS, J. Chemom. 6 (1992) 11.

[44] 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, Arch. Biochem. Biophys. 278 (1990) 148.

[45] B.C. Mitchell, D.S. Burdick, Slowly converging PARAFAC sequences: Swamps and two-fator degeneracies, J. Chemom. 8 (1994) 155.

[46] W.P. Krijnen, The analysis of three-way arrays by constrained PARAFAC methods, Ph.D. thesis, University of Groningen, 1993.

[47] J.B. Kruskal, Multilinear methods, in: H.G. Law, C.W. Snyder, J.A. Hattie, R.P. McDonald (Eds.), Research methods for Mulitmode data analysis, Praeger, New York, 1984.

[48] J.B. Kruskal, R.A. Harshman, M.E. Lundy, How 3-MFA data can cause degenerate PARAFAC solutions, among other relationships, in: R. Coppi, S. Bolasco (Eds.), Multiway data analyses, Elsevier Science Pub., North-Holland, 1989.

[49] A.K. Smilde, Y. Wang, B.R. Kowalski, Theory of medium-rank second-order calibration with restricted-Tucker models, J. Chemom. (1994) 21.

[50] 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 (1994) 3345.

[51] R.A. Harshman, PARAFAC2: Mathematical and technical notes, UCLA Working Papers in Phonetics 22 (1972) 30.

[52] H.A.L. Kiers, An alternating least squares algorithm for PARAFAC2 and three-way DEDICOM, Comp. Stat. Data Anal. 16 (1993) 103.

[53] J.R. Kettenring, A case study in data analysis, Proc. Symp. Appl. Math. 28 (1983) 105.

[54] A. Aastveit, H. Martens, ANOVA interactions by partial least squares regression, Biometrics 42 (1984) 829.

[55] H. Martens, L. Izquierdo, M. Thomassen, M. Martens, Partial least squares regression on design variables as an alternative to analysis of variance, Anal. Chim. Acta 191 (1986) 133.

[56] M.V. Martinez, J.R. Whitaker, The biochemistry and control of enzymatic browning, Trends Food Sci. Technol. 6 (1995) 195.

[57] H. Heimdal, L.M. Larsen, L. Poll, R. Bro, Oxidation of chlorogenic acid and (−)-epicatiechin by letucce polyphenol

oxidase in model solutions at various combinations of O_2, CO_2, Temperature and pH, J. Agric. Food Chem., in press.

[58] R. Bro, H. Heimdal, Enzymatic browning of vegetables, Calibration and analysis of variance by multiway methods, Chemom. Intell. Lab. Syst, 34 (1996) 85.

[59] L. Nørgaard, Classification and prediction of quality and process parameters of thick juice and beet sugar by fluorescence spectroscopy and chemometrics, Zuckerind. 120 (1995) 970.

[60] G.W. Ewing, Instrumental methods of chemical analysis, McGraw-Hill Book Company, NY, 1985.

[61] L. Nørgaard, A multivariate chemometric approach to fluorescence spectroscopy, Talanta 42 (1995) 1305.

[62] R. Bro, Multiway calibration, Multilinear PLS, J. Chemom. 10 (1996) 47.