



**CENATAV**

Centro de Aplicaciones de  
Tecnologías de Avanzada  
MINISTERIO DE LA INDUSTRIA BÁSICA

RNPS No. 2142  
ISSN 2072-6287  
Versión Digital

REPORTE TÉCNICO  
**Reconocimiento  
de Patrones**

**SERIE AZUL**

**Multi-way data analysis**

Ing. Diana Porro Muñoz,  
Dr. C. Isneri Talavera Bustamante  
Dr. C. Robert W. Duin

**RT\_014**

**Noviembre 2009**





**CENATAV**

Centro de Aplicaciones de  
Tecnologías de Avanzada  
MINISTERIO DE LA INDUSTRIA BÁSICA

RNPS No. 2142  
ISSN 2072-6287  
Versión Digital

**SERIE AZUL**

REPORTE TÉCNICO  
**Reconocimiento  
de Patrones**

**Multi-way data analysis**

Ing. Diana Porro Muñoz,  
Dr. C. Isneri Talavera Bustamante  
Dr. C. Robert W. Duin

**RT\_014**

**Noviembre 2009**



# Multi-way data analysis

Ing. Diana Porro Muñoz, Dr. C. Isneri Talavera Bustamante, Dr. C. Robert W. Duin

Centro de Aplicaciones de Tecnología de Avanzada, 7a #21812 e/ 218 y 222, Siboney, Playa, Habana, Cuba  
dporro@cenatav.co.cu

RT\_014 CENATAV

Fecha del camera ready: 31 de julio de 2009

**Abstract:** Although for some research areas it is a new thing, multi-way data analysis is a multivariate analysis technique that has had a wide appliance in a lot of fields. In dependence of the problem at hand, different types of multi-way data and specific multi-way models already exist. In this report, we give a general panoramic of multi-way data analysis, and some particular aspects needed to be taken into account when modeling such type of data. We describe some of the most used multi-way models and algorithms, outlining advantages and limitations that have been treated in new versions and others that still not. We also dedicate a chapter to describe in a sort of way, the research areas where multi-way analysis is most common, of a widely range of applications this type of data has. It is conclude with an analysis of the open problems that we found in these techniques and some proposals we think can work to solve them.

**Keyword:** Chemometrics; Multi-way Analysis

## 1. Introduction

Multi-way analysis, originating when Raymond Cattell in 1952 [1] first introduced the very important term of multi-way arrays and later in psychometrics when Tucker published in 1964 [2] the three-mode component analysis, is the extension of multivariate analysis when the analyzed data is in the form of higher order arrays. It is often used for extracting hidden structures, explore the interrelations in the data, etc., since it has been shown in other investigations that this kind of information from the data may not be gotten accurately or identified uniquely by two-way analysis methods because they do not respect the multi-way design of the data.

Two-way analysis methods, e.g. factor models, suffer from rotational freedom unless specific constraints such as statistical independence, orthogonality, etc. are enforced. These constraints requiring prior knowledge or unrealistic assumptions are not often necessary for multi-way models, most of these methods have a strongly exploratory character, which means that one tries to find the patterns among the elements of the three ways, without a priori postulating specific configurations and without applying tests to these patterns. Consequently, multi-way analysis, with advantages over two-way analysis in terms of uniqueness, robustness to noise, although not too ease of interpretation, etc. has been a popular exploratory analysis tool in a variety of application areas.

Important aspects of multi-way models and methods, especially in the social and behavioral sciences, is that they allow for the analysis of individual differences in a variety of conditions. The subjects do not disappear in means, (co)variances, or correlations, and possibly higher-order moments, but they are examined in their own right unless the researchers convert it for a special matter. This implies that, the data set is often taken as it is, and not necessarily as a random sample from a larger population in which the subjects are in

principle exchangeable. Naturally, this affects the possible generalization, but that is considered inevitable. Nevertheless, there are still a lot of scientists that instead of using the entire multi-way array, create a two-way matrix out of it to analyze it later with two-way models, by either laying out all the slices in one long row, so that the relation between similar variables at different time points is neglected, or laying out all slices in one tall column, so that the connections between samples' scores at different moments in time are lost. The technical term for this procedure is matricization. Sometimes this may do no harm, because it is possible that a three-way analysis leads to the conclusion that no three-way analysis is necessary: for instance, if nothing changes over time, or if all subjects may be viewed as having been randomly drawn from one single population, but most of the time it is inefficient because it can lead to overfitting as the model does not represent the natural composition of the data, two of the multi-way array modes are always confounded and no independent parameters for these modes are present in the model itself.

As will be seen in the following subchapters, the multi-way models and algorithms development is not poor at all, an increasing range of methods have been created for different configurations and types of the multi-way arrays, but still they are something far from maturing, so it is needed to improve the models and algorithms now available.

There are also some other philosophical and technical things to be taken into account when doing multi-way analysis. Matters as detection limits, outliers, missing data, if preprocessing is needed before fitting the multi-way model, the optimal number of components to use in the case of multi-linear decomposition models, as well as the term of uniqueness of the model, has to be handled in the process.

The objective of this work is to make a study of the current state of multi-way analysis and discuss over some things that might be undone yet in this field. The most significant aspects of multi-way analysis will be analyzed in the following chapters, along with the description of the composition of a general multi-way array. For this report we have backed up in existing reports and books on multi-way analysis which are very consistent [3], [4], [5]. As will be seen, in the different research areas can be found unlike multi-way array configurations and a great part of the multi-way models have been proposed for a specific type of array. In the next chapters we will refer to the most common configuration, defined in [5] as profile data, and describe the models and algorithms that apply to this type of data. These models were also chosen because they are the most general, and can be applied to almost all the configurations of multi-way arrays. For more details you can refer to [5]. For each model described here we will not only refer to its purpose and mathematical description, but to some important aspects as uniqueness of the solution, advantages and disadvantages, as well as for the algorithms. At the end of this work, some applications of multi-way analysis will be described making some emphasis in chemometrics applications, which is the purpose of research for what this work has been done.

## 2. Multi-way data

The multivariate data prototype is a two way structure (matrix), where there are a number of objects (rows) characterized by measured variables (columns). For a wide variety of problems, the structure of the data can often be more complex than this; you can have several sets of variables measured on different samples, as for example, data collected at different

times or conditions. Sometimes there are two types of objects and only one set of variables, as is the case of a multivariate image where the x-coordinate and y-coordinate of the pixels are taken as the objects and wavelengths as the variables, and the purpose of the analysis is to find the relationship between the objects. These data would be appropriately represented by higher-order generalization of vectors and matrices where  $X \in R^{I_1 \times I_2 \times I_3}$  and the order of  $X$  is  $N$  ( $N > 2$ ).

The most common is the three dimension array (box)  $X (I \times J \times K)$ , but it is even possible to generate four and five-dimension data, where each dimension (way or mode) will correspond with the different set of objects or variables and is called multi-way array.

In this three-way case the first way has index  $i$  running along the vertical axis, the second way has index  $j$  running along the horizontal axis, and the third way has index  $k$  running along the “depth” axis of the box [6] (see Fig. 1).

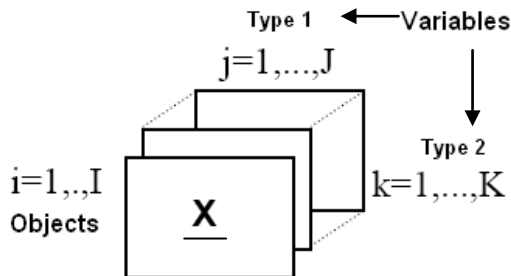
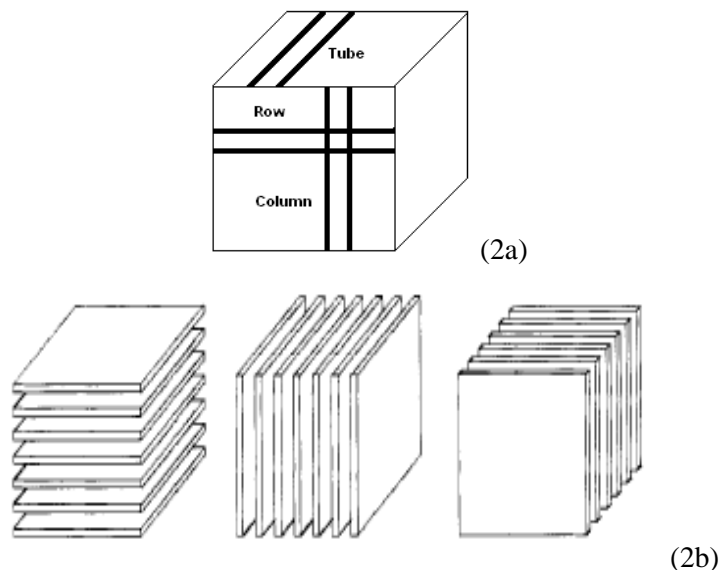


Fig. 1. Design of a typical three-way array

Multi-way and multimode is not exactly the same thing, although for some data they can be used indifferently. The word way is considered more general, referring to the multidimensional arrangement irrespective of the content of the data, while the word mode is more specific and refers to the content of each of the ways. Thus, objects, variables, and conditions can be the modes of a three-way data array.

When the same entities occur in two different ways, as is the case in a correlation matrix, the data are one-mode two-way data. When correlation matrices for the same variables are available from several different samples, one often speaks of a two-mode three-way data array, where the variables and the samples are the two modes.



**Fig. 2.:** (2a) Rows, columns and tubes of the three-way array, (2b) Horizontal, Vertical and Frontal slices of the three-way array respectively

Although the terms rows and columns of the two-way arrays are still used, together with the term tube (see Fig. 2a) which refers to the third mode (three-way), the multi-way is usually conceptualized as a collection of two-way matrices and then the columns and rows replaced by slices or slabs (see Fig. 2b). Each horizontal slice ( $J \times K$ ) of the block represents the data of one object; each vertical slice ( $I \times K$ ) holds the data of a specific type of variable and the back to front slices ( $I \times J$ ), variables of other type [3].

This type of multi-way data is defined as “profile data” in the taxonomy of designs of multi-way data presented by Kroonenberg in [5]. In dependence of the problem in hand, many types of configurations or designs of the multi-way arrays can be found. In the following chapters we will refer to the “profile data” type, which was the chosen to be analyzed here as is the most common in all research areas, especially in the one that brought us to this investigation, Chemometrics.

## 2.1. Preprocessing

Preprocessing is as an important step in data analysis, and it is nothing else but the transformation of the original data that must be done before fitting a model to it, to improve its results. There are two basic types of data adjustment used before direct fitting of profile data: (a) additive adjustments (centering); and (b) multiplicative adjustments (scaling) [7]. This two preprocessing are the most used in two-way bilinear data analysis, and they can be easily extended to higher-way data. Although some authors say it is quite not easy to preprocess multi-way data, its difficulty is not more that the one it drags from the two-way analysis.

It is also very significant to know if it is necessary to preprocess the data and if it is so, which one is the right to do, most of all for the multi-way case because there are many alternatives, otherwise the effect can be the opposite. In [7] the authors propose some theoretical analysis to discover which of the possible preprocessing techniques is the appropriate for two-way and extended to three-way; [5] presents a summary of some works that relate the selection of the preprocessing with specific research areas. In [8], [3], [7] and [5] are also argument which combinations of centering and scaling can and can't be done, due to the interdependence that exists between these two.

### 2.1.1. Centering

To explain the idea of centering it is important first to know the term 'offset'- also called intercept- which is used for a part of the model that is constant across one or several modes that functions as the zero point of an interval scale [7],[5]. This offsets can be found constant in one mode or in various (for multi-way data), they can also appear combined. So, centering preprocessing deals with this phenomenon, and although it can be use to estimate the offsets sometimes, most of the time it is used to remove them to obtain a reasonable model. It could be said, that centering is performed to make interval-scale data behave as ratio-scale data, which is the type of data assumed in most multivariate models. [8]. Deeper information on the reasons for centering can be found on [3]. There are some aspects, also explained in the previous references that demonstrate some benefits of making this type of preprocessing, of course, if it is right:

1. Reduced rank of the model
2. Increased fit of the data
3. Specific removal of offsets
4. Avoidance of numerical problems

Mathematically centering can be expressed as:

$$y_{ij} = x_{ij} - \frac{\sum_{i=1}^I x_{ij}}{I} \quad (1)$$

or

$$y_{ij} = x_{ij} - \frac{\sum_{j=1}^J x_{ij}}{J} \quad (2)$$

In Eq.1 and Eq.2  $y_{ij}$  is the centered element,  $x_{ij}$  is the element in the  $i$ th row and the  $j$ th column,  $I$  the number of rows and  $J$  the number of columns. In the case of Eq.1 is referred the data are centered by subtracting the column average from every element of the column and it is known as centering across the first mode. Eq.2 is the expression for centering across the second mode, and the idea is the same as in the previous equation, but applied to the rows. For both cases, if this preprocessing is only applied to one mode it is known as single

centering, since there is also the double and triple centering, which means centering across one mode and then center the outputs across the other modes, the order in which they are done is not important, but it must be sequentially.

These types of centering by columns, rows or tubes in the case of multi-way arrays, is known as fiber centering (one-way), and if is done across one of the modes it will remove the offset from all the modes that involve this mode. But these are not the only ones, there are two other approaches that can be applied to three-way data: in the slab (two-way) centering the average of one slab is subtracted from each slab of the three-way array, and as in the case of fiber centering there are three types of slab centering corresponding to the vertical, horizontal and frontal and slices. At last, the grand mean (three-way) centering, where it is subtracted the overall mean of the three-way array from each element of the data. In [7] there are more details of this types of preprocessing and in [5] can be found some other types of preprocessing, besides the others found in the reference mentioned previously. But although there are all these types, as it was argued by [7], following the idea that the appropriate preprocessing should preserve the appropriateness of any factor structure underlying the data, the fiber preprocessing are the most indicated, as slab and global (grand mean) preprocessing have many disadvantages. In the case of single centering, it cannot solve the problem of removing all the constants and its effects itself, but its application to all three modes can do so. Anyway, these triple-centering can sometimes over fit the data, so it is better to use only the double-centering, which eliminates the offsets of all the modes except one and it yields preferable results.

Saying all this, then the expression for preprocessing a two-way array can be extended for three-way arrays, by matricizing the array and centering across one or the two original modes, but the new column-mode that arises from the combination of the original ones should not be transformed as this is not real and won't help in the model at all:

$$y_{ij} = x_{ijk} - \frac{\sum_{i=1}^I x_{ijk}}{I} \quad (3)$$

In the case of Eq.3 the three-way array has been matricized to a  $I \times JK$  matrix and the centering is performed across the first mode as in the two-way preprocessing.

### 2.1.2. Scaling

The scaling or multiplicative adjustment does not have the same effect of changing the structure of the model as centering, and of course the reasons for doing it are neither the same, but one thing they sure have in common is that it also has to be performed in a specific way in order not to introduce artificial structures that need to be modeled. This becomes more apparent when going to three way models [3]. It is used often to adjust the data so the variance of each variable is unity, or what is the same to change the importance attached to different parts of the data in fitting the model, since most of the models assume that variables with large variation are the most important for the model and in some cases these large variations are irrelevant as they are due to noise or to the use of different scale and it shouldn't influence the model more than the necessary.



There are three important reasons among others to scale a data; they are more detailed in [3], [8]:

1. To adjust scale differences
2. To accommodate for heteroscedasticity
3. To allow for different sizes of subsets of data (scaling block)

The idea of scaling is to multiply a determine scalar to the data, which is usually objectively calculated: a standard deviation, an interquartile distance, a range or a robustified range, or sometimes subjective weights are given to the variables. To refer to this type of preprocessing a different terminology to the one of centering is used. When each row of the matrix is multiplied by this scalar it is known as scaling within the first mode. In the case of doing the same operation to the columns, it is referred then as scaling within the second mode. These types of scaling are known as slab-centering and it is demonstrated why they are the most appropriate for three-way arrays. Mathematically it can be expressed as follows:

$$y_{ij} = x_{ij}w_i \quad \text{or} \quad y_{ij} = x_{ij}w_j \quad (4)$$

$$Y = XW \quad \text{or} \quad Y = WX \quad (5)$$

In Eq.4 the term  $y_{ij}$  now refers to the scaled element,  $x_{ij}$  is the same as in the previous equations, and  $w_i$  and  $w_j$  are the weights or scalars multiplied to the rows or the columns respectively. The Eq.5 is the matrix representation where  $Y$  is the entire scaled data,  $X$  the original data and  $W$  is a  $I \times I$  or  $J \times J$  diagonal matrix that holds the scalar  $w_i$  or  $w_j$  in its diagonal elements, for scaling within the first and second mode respectively. This last scaling within the second mode, if it is combined with a previous centering across the first mode, is nothing else but the known autoscaling technique for the two-way data.

The term fiber-scaling also exists in three-way arrays and could be done in the three usual directions: scaling each row of each slab by the coefficient  $w_{ik}$ , each column of each slab by the coefficient  $w_{jk}$  and each tube by the coefficient  $w_{ij}$  [7]. Mathematically the scaling of columns would be expressed as:

$$y_{ijk} = x_{ijk}w_{ijk} \quad (6)$$

In this equation the meaning of each variable is the same as explained before but now representing an element of a three-way array. But it has been also demonstrated in [7] that this type of scaling is not appropriate since by doing this operation on just one mode, will add more than three extra dimensions to the data structure and decrease the fit value at the original true dimensionality.

There is also the case of scaling the entire three-way array, multiplying each entry of it by the same coefficient, but it won't have any effect on the results than this.

So, after concluding that scaling a three-way array is better to scale the whole matrices or slabs that compose the array, the mathematical representation would be the extension of the

one used for two-way arrays . The following expression corresponds to the scaling within the first mode:

$$y_{ijk} = x_{ijk} w_i \quad (7)$$

with

$$w_i = \frac{1}{\sqrt{\sum_{j=1}^J \sum_{k=1}^K \frac{x_{ijk}^2}{JK}}} \quad (8)$$

or

$$Y_{(I \times JK)} = W X_{(I \times JK)} \quad (9)$$

In Eq.8 it is proposed scaling the data to a unit mean within the first mode, it will be explained later why this one is the most common one. The Eq.9 is the representation when using the matricized array and for these cases  $W$  would also be a  $I \times I$  diagonal matrix that contains the scaling values in the diagonal. An important thing to take into account with this representation is that the scaling shouldn't be performed on the resulting array from the combination of two modes, it has to be done by transforming the original data within a given mode or unwanted artificial components will appear in the data. In [5] some other types of scaling can be found.

Another complicated thing shows up when scaling several modes, as scaling one mode after another one has been previously scale will affect the first preprocessing. For this purpose Lundy [7] has proposed to do it iteratively until convergence [9], no matter the order. But this convergence can be only guaranteed when it is desired to scale to a mean square of one within several modes, unlike scaling to standard deviation of one that won't generally converge.

The advantage of several scaling is that one deals simultaneously with different types of measurement scales and with undesired variability between subjects. However, there has been very little experience with this procedure and it is somewhat worrying that a large amount of informative variability from the data is eliminated.

## 2.2. Unfolding

Unfolding, usually known as matricization, is an important concept in multi-way. It is just the conversion of a multi-way array into a two-away array by concatenating the slices from one mode next to each other e.g. a  $I \times JK$  matrix, with mode  $B$  entities ( $j = 1, \dots, J$ ) nested within mode  $C$  entities ( $k = 1, \dots, K$ ), and will be denoted as  $X_a$ . This matrix simply contains all the frontal slices of the array next to each other (see Figure). Other matricizations are those that form the super-matrices  $X_b$  (of order  $J \times KI$ , with mode  $C$  entities nested within mode  $A$  entities) and  $X_c$  (of order  $K \times IJ$ , with mode  $A$  entities nested within mode  $B$  entities). Other nesting are possible, but without further specification, matricization pertains to one of the above procedures [6].

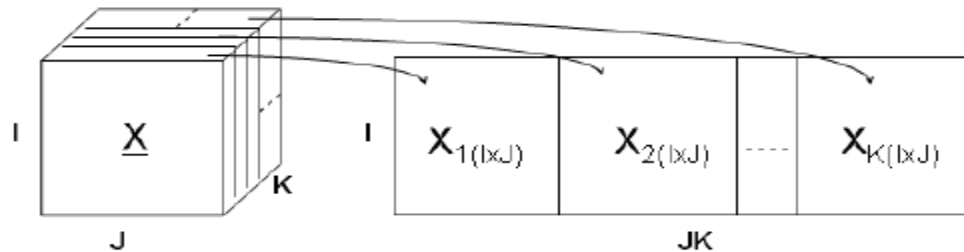


Fig. 3. Unfolding of the first mode of three-way array. Wide combination-mode matrix

Notice, that the column-dimension of the generated matrix becomes quite large in the mode consisting of two prior modes. This is because either the subjects and conditions from the original modes are combined into a single mode (tall combination-mode matrix) or the variables or conditions are so combined (wide combination-mode matrix) (see Fig. 3). There is not a new variable referring to one original variable, but rather a set of variables. Thus, two of the modes are always confounded and no independent parameters for these modes are present in the model itself, except when models are used specifically geared toward this situation.

Matricization is not only useful for three-way arrays. In fact,  $N$ -way data usually are to be read from file in a two-way structure. An  $N$ -way array is ‘matricized’ in essentially the same way as a three-way array e.g.  $X_a$  contains all vertical fibers collected next to each other in an  $I \times JKLM \dots$  matrix, within the columns the mode  $B$  entities nested within mode  $C$  entities, mode  $C$  entities nested within mode  $D$  entities, etc. [6]

Once a three-way array is arranged as a two-way dataset, two-way analysis methods, e.g. Singular Value Decomposition (SVD) [10], [11] and other factor models can be employed in understanding the structure in data, as well as other methods for classification and regression tasks. But this transformation, ignoring the multi-way structure to apply the two-way methods can lead to information loss and misinterpretation especially if the data are noisy, then the models would be:

- less robust
- less interpretable
- less predictive
- non-parsimonious

These problems of course will arise for arrays that need to be approximated by multi-way structures and the noisier the data are, the more beneficial it will be to use the multi-way structure. But in cases where the data does not have an underlying multi-way structure, the matricization process and a two-way analysis afterwards will be enough and easier to interpret.

Anyway, the fact that the data can be approximated by a multi-way structure is somewhat vague. To verify if this is so, in the case of three-way problem for example, it has to be considered a hypothetical two-way matrix with rows and columns equal to the first and second mode of the three way array, and another two matrixes with the mode one with three,

and mode two with three respectively. If all these hypothetical two-way problems are adequately modeled by a bilinear model, then it is suitable to use a multi-way structure. 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 [12].

### 3. Multi-way models

It is something important to define what means the term ‘model’ at least for our field before doing some history, as this is not seen in the same sense in all fields. It is known as a way to describe the structure of the data, an approximation of it; and each one has its own specifications, being a very important one the definition of a loss function to verify how well this model fits the data. Thus, a model usually consists of two parts, one that is “structural” - an algebraic expression describing the modeled pattern(s)-and one that is “stochastic” - usually a single symbol representing the residuals or unmodeled variation. Their sum equals the array.

Although multi-way analysis has a lot of advantages to deal with cases when you have a considerable amount of data and need to make use of all of it and find interrelations, as can be easily found in Chemometrics with the development of complicated analytical instruments, it is not something very usual nowadays, maybe because of its complexity when representing it and to analyze it later. Nevertheless, it is not a new thing; there have been studies on it since 1874 when Camille Jordan was investigating on how to simultaneously diagonalize two-matrices. Another important figure on this field was Raymond Cattell who defined the principle of “parallel proportional files” in 1944 [13] and in 1952 he introduced the very important term of multi-way arrays [1]. In the 1960s started then to emerge three-mode techniques that really obeyed to the principle of three-way analysis and nowadays there are many of them that extend to multi-way analysis. One of the precursors of this techniques was Ledyard Tucker [2], [14], [15] when he came up with the three mode factor analysis, where he introduced the term mode instead of dimension and still in use today. These models are nowadays known as Tucker models or N-mode principal component models (multi-way PCA) for others and the most used is TUCKER3. Based on the principle proposed by Cattell, was introduced the parallel factor analysis as one of the most important models to analyze multi-way data at present, PARAFAC. This model was created by Harshman in 1970 [16], and by the same year Carroll & Chang also proposed the CANDECOMP model [17]. Although they were introduced independently, the model behind the two is the same. In the case of PARAFAC and TUCKER3 the extension to multi-way is straightforward.

There have been proposed some other models for multi-linear regression analysis as N-PLS [18]. In 1970 J. Douglas Carroll in collaboration with Jih-Jie Chang, started the development of multidimensional scaling techniques for individual differences with the individual differences scaling model (INDSCAL) [17], which Kroonenberg names as a milestone in multi-way analysis, and is an extension of existing methods for two-way similarity/dissimilarity data to three-way data formed by sets of this matrices. Some other versions of this method were published later [19]. This type of design of multi-way array was referred in Kroonenberg’s taxonomy as “scaling designs”, and is very common in psychology. Some others have introduced methods for three-way cluster analysis, multi-way

covariance analysis, multivariate longitudinal analysis, etc, but not all of these techniques will be treated here, only the most transcendent ones. In [3] and [5] there are more details about the history of multi-way data analysis and all these techniques. In the following subchapters we will deepen into the theory and algorithms of the main multi-way models (multi-way component models), which are widely used for many types of multi-way data and have shown the best results. We will see also their most known versions and some special aspects that have to be taken into account when choosing these models.

### 3.1. Multi-way component models

In a table of different categories of multi-way models that was presented by Pieter M. Kroonenberg in his book *Applied Multi-way Data Analysis* (2008), is defined a category for the component models, among others. In this subchapter it is planned to deepen into models that were placed into the first category mentioned, which are as was said before, among the most used in a lot of fields; PARAFAC for example has had a violent deployment in analytical chemistry and other fields, in theory as much as in its applications. The objective is then to analyze these main models and in some way its versions, in order to demonstrate its advantages and disadvantages. In this work we will refer only to the three-way models, in case they generalize to n-way the needed specifications will be made.

#### 3.1.1. Parallel factor analysis (PARAFAC)

The philosophy of PARAFAC is very similar to the followed by the Principal Component Analysis (PCA) [10], [11] based on components decomposition. There are of course differences, besides PCA works on two-dimensional data and PARAFAC on three and n-dimensional data. It is obtained now a loading matrix  $(A, B, C)$  for each variable mode and the score matrix for the object (in case it is considered like it), anyway, in three-way analysis has become very common calling loading all the vectors obtained in the decomposition because these are treated equally numerically.

PARAFAC [16] (see also [20], [21], [22]), as was said before, was introduced independently by Harshman and by Carroll and Chang who named the model CANDECOMP (canonical decomposition) in 1970 [17] and it is based on Cattell's principle of Parallel Proportional Profiles:

“The basic assumption is that, if a factor corresponds to some real organic unity, then from one study to another it will retain its pattern, simultaneously raising or lowering all its loadings according to the magnitude of the role of that factor under the different experimental conditions of the second study. No inorganic factor, a mere mathematical abstraction, would behave in this way.....” [13]. Cattell in its article argued extensively that this principle is the most fundamental property for obtaining meaningful decompositions.

So, following this criterion, the basic idea of a PARAFAC model is to use the same factors to describe the variation in several matrices simultaneously but with different proportions or scaled depending on the conditions. This leads to the fact that all the loading matrices obtained in the decomposition will have the same number of factors and the model won't be subject of rotational freedom, of which we will talk about later. This can be a very attractive characteristic of this model, but also a limitation, as by disallowing the interaction

among the factors of the different modes some important variations in the structure of the data can be ignored.

Mathematically, in this decomposition the array is converted in a sum or linear combination of triads (three-way) or N-ads (n-way) [23]. The word triad is used for a 3-array that can be decomposed in an outer product of 3 vectors (a, b, c) or what is the same, in an outer product 3-array (rank  $\leq 1$ ), this definition is the same for any order  $> 2$  array. In the following expression it is shown the representation of an R-component PARAFAC model, which can be also seen as a generalization of the singular value decomposition truncated to R components. An illustration of this decomposition is also shown in Fig. 4.

$$x_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr} + e_{ijk} \tag{11}$$

with  $a_{ir}, b_{jr}$  and  $c_{kr}$  being the elements of the loading matrices  $\mathbf{A}(I \times R)$ ,  $\mathbf{B}(J \times R)$ ,  $\mathbf{C}(K \times R)$  respectively and  $e_{ijk}$  are the elements of the corresponding residual matrix E. In this decomposition it is also obtained a diagonal matrix  $G = \text{diag}(g_{111}, \dots, g_{rrr})$  called the singular value or core matrix containing the R largest singular values of X in decreasing order, which would be the weights of the combinations mentioned before, but as in Eq.11 the values of  $g_{rrr}$  can be absorbed in  $a_{ir}, b_{jr}$  and  $c_{kr}$  or in both.

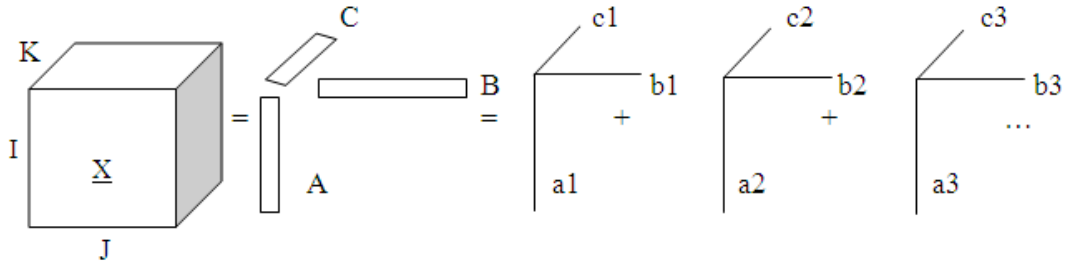


Fig. 4. Three-component PARAFAC model

In the following equation it is also shown the PARAFAC model in terms of the loading matrices, where  $X_k$  is the  $k$ th ( $I \times J$ ) frontal slice of the three-way array  $X(I \times J \times K)$

$$X_k = AD_k(C)B' + E_k \tag{12}$$

$D_k$  is the diagonal matrix with the  $k$ th row of third component matrix  $C$  in its diagonal ( $c_{k1}, \dots, c_{kR}$ );  $a_{ir}, b_{jr}$  are the  $r$ th columns of the component matrices in the first and second mode,  $A$  and  $B$  respectively, and the residual term  $E_k(I \times J)$  is defined

similarly to  $X_k$ . Hence, each  $X_k$  is modeled using the same components  $A$  and  $B$ , but with different weights, represented by  $D_k$ .

Across all slices  $X_k$ , the components  $a_{ir}$  and  $b_{jr}$  remain the same, only their weights  $(d_{k1}, \dots, d_{kR})$  are different. Hence, all slices  $X_k$  are modeled with parallel and proportional profiles  $(d_{k1}a_1b_1, \dots, d_{kR}a_Rb_R)$  [3]. As this representation in terms of the frontal slabs, there are two more for the horizontal and vertical slabs which are equivalent.

In [3] there are some other ways to express the PARAFAC model in terms of the Kronecker, Hadamard and Khatri-Rao [24], [25] products. Here we will only show the Khatri-Rao expression as it will be of interest for the algorithms section (4.1.1):

$$X = A(C \square B)' + E \quad (13)$$

where  $X = [X_1 \ X_2 \ \dots \ X_K]$  is a  $(I \times JK)$  matrix,  $X_k$  is defined as in Eq.12;  $A, B$  and  $C$  are the loading matrices and  $E = [E_1 \ E_2 \ \dots \ E_K]$  is the residual matrix.

Anyway, all of them should be seen as they are chosen or preferred depending on the problem in hand.

### 3.1.1.1. Uniqueness

In bilinear models there is the well-known problem of rotational freedom and this is the main reason of PARAFAC's success. Over the years many researchers have tried different methods more interpretable than PCA or for rotating to more appropriate solutions but most of them have ended in ill-defined properties. PARAFAC on the other hand, can obtain a unique solution (up to permutation, sign and scaling indeterminacy) in terms of the model estimated parameters; that is, the calculated  $A, B$  and  $C$  loading matrices cannot be changed without changing the residuals. When talking about unique solution it is important to know that it doesn't refer at all to the fact that the model obtained the true solution. Anyway, if the structure of the data is approximately tri-linear, the right number of components is chosen and the signal-to-noise ratio is appropriate [23], [26], [27], the true underlying parameters can be estimate.

Kruskal states in [23] the following definition for this:

“A rank  $R$   $N$ -adic decomposition of an  $N$ -array  $\mathbf{X}$  is rotationally unique, often shortened to just unique, if all rank  $R$  decompositions of  $\mathbf{X}$  are equivalent to it.”

The uniqueness properties of the PARAFAC model are sometimes stated as the model having unique axes or intrinsic axis property. As opposed to a bilinear model where the subspace spanned by the data can be uniquely determined but the bases are not unique, the PARAFAC model not only determines the subspace but also the position of the axes defining the subspace. That is why the name of unique axes.

Nevertheless, there are some conditions to achieve the PARAFAC unique solution on what some authors have worked on. Harshman in [26] and Leurgans et al. in [28] 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. On the other hand [23],[29] made a whole investigation based on the ranks of the N-array  $\mathbf{X}$  (smallest number  $\mathbf{R}$  such that  $\mathbf{X}$  has an N-adic decomposition of rank  $\mathbf{R}$ ) of the loading matrices and its relation with uniqueness. For the case when the N-adic has rank 1 he stated:

*Lemma 4i:* “For any N, a rank 1 N-adic decomposition of an array is unique if the array contains no zero slabs in any direction.”

He also based on the mathematical fact that dyadic and triadic decompositions are usually unique for small enough ranks and never unique for large ranks, where the dividing line increases with the size of the array in the cases of triadic decompositions. In the special case of a  $2 \times 2 \times 2$  array it is proven that a rank 2 decomposition of a rank 2 array is always unique, while a rank 3 decomposition is never unique. There have been some special cases like this that have been proven in studies as a ten factor model that was uniquely determined from an  $8 \times 8 \times 8$  array [20]. For an  $R \times R \times R$  array, the Theorem 4a presented also in [23] and later extended by [30] shows that a condition to be accomplished by the PARAFAC model to achieve the unique estimates of the parameters is:

$$k_A + k_B + k_C \geq 2R + 2 \quad (14)$$

where  $k_A$ ,  $k_B$  and  $k_C$  are the  $k$ -ranks of the component matrices  $A$ ,  $B$  and  $C$ , respectively and  $R$  is the number of components in the PARAFAC model. This condition can be met in almost all practicable applications. A more understandable and accessible proof of this condition was proposed in [31]. Nevertheless, there can be seen in Eq.14 that it doesn't hold when  $R = 1$ , but as explained before, the uniqueness for this case was proven in [26].

The  $k$ -rank term was introduced by Kruskal in [32]. It states that considering a matrix  $X (I \times J)$  of rank  $R$ , then  $X$  has some set of  $R$  independent columns. However, some other set of  $R$  columns of  $X$  might not be independent. The largest integer  $k$  for which every subset of  $x$  columns of the matrix  $X$  is linearly independent, is called the  $k$ -rank of  $X$  and is denoted by  $kX$ . Stated otherwise, the  $k$ -rank of  $X$  is the largest subset size for all subsets of columns of  $X$  that always has full rank.

The theorem 4a, which defines the above condition, shows that many decompositions of rank  $\lceil 3R/2 - 1 \rceil$  are unique, but here it is also suggested that none of higher rank are unique. Later in [30] were generalized this Kruskal's fundamental result on the uniqueness of tri-linear decomposition of three-way arrays, to the case of n-way arrays as follows:

$$\sum_{n=1}^N k_{ranks} \geq 2R + (N - 1) \quad (15)$$



In [33] it is demonstrated that Kruskal's conditions are necessary for  $R$  smaller than four and made a conjecture that it could be also accomplished when  $R > 3$ , but in [34] the authors refuted this conjecture. The necessary conditions for uniqueness of PARAFAC models with more than three components are still unknown.

Besides there are some data where the Kruskal's condition cannot be met, and the PARAFAC model cannot estimate the unique solution, it might still be possible to obtain partial uniqueness, which means that some parameters are uniquely determined for those components that do have 'adequate' variation across all three modes [26]. The only well known cases are the mentioned before and those when more components are extracted than the required for the perfect fit. In [35] are shown results of studies that have been done on special 3-way arrays as  $5 \times 3 \times 3$  where it is demonstrated how they have partial uniqueness solution, but anyway the study of this type of uniqueness have fallen behind.

Although there have been all this research on PARAFAC uniqueness and have been demonstrated its achievement on a several number of cases, there are also cases where it cannot be proven. Besides, all the demonstrations have been made on models where it is used the same number of factors for the three modes, as a consequence of the fact that factors in different modes can only interact factor-wise.

### 3.1.1.2. Degeneracy problems

When trying to fit a PARAFAC model to the data, sometimes the algorithm can have difficulties to find the right fitting and the estimated model parameters are hence often unstable and unreliable. This is known as a problem of degeneracy and then the solutions found are called degenerate solutions.

There reasons for this problem are not easy to determine but some causes are known e.g. too many components are extracted because the 'noise' components can be correlated, poor preprocessing has been applied. Another reason is related to the tri-linearity constraint, if the data is not appropriately modeled by a tri-linear model which is also referred as two-factor degeneracies. There is also degeneracy-like situation related with the algorithm and is called swamp, which is an area of the solution space where it advances too slowly. In this case the factor matrices become ill-conditioned when the solution approaches to the high-rank domain (degenerate solution) and the algorithm gets stuck making large changes in the factor matrices to improve the loss function, it could disappear after a number of iterations.

An indication of degenerate solutions can thus be obtained by monitoring the correlation between all pairs of components [3]. The measure used for this is called Tucker's congruence coefficient and is also sometimes referred to as the uncorrected correlation coefficient [36], but it is not our objective to deepen in on it. For other degeneracy indicators you can refer to [37] More details about this measure and how analyze its value to infer the possibility of a degenerate solution can be found in [3], [20].

There have been some proposals to affront this problem. Some authors suggest that Tucker models are more suitable than PARAFAC models when two factors are interrelated. In the case of swamps some authors as in [38] suggest the use of some amount of regularization to keep the algorithms away from them. Others, as [34] suggest using those runs that are not probable to give degenerate solutions from a lot of runs made in a few iterations. Another way of circumventing degenerate solutions, although it doesn't mean that the cause of the problem has been removed, is by applying orthogonality or non-negativity

constraints on the model as this effectively prohibits negative correlations. For more studies on degeneracy cases in multi-way analysis and how to deal with them you can also refer to [39], [40] and [41]. In the later the authors proof that the algorithm will lead to degeneracies when there is not an optimal solution. Recently Stegeman and De Lathauwer in [42] based on the proof they made in [39], they propose a way to avoid degenerate solutions in a  $(I \times J \times 2)$  three-way array.

This is a problem that might be found in a lot of cases, but the main ones as mentioned before, are related with the processing of the data and the initialization parameters of the algorithms, so this leads to the conclusion that the analysis of the data have to be done carefully. The use of the presented uncorrected correlation coefficient would also save a lot of work. This is seen as a drawback of the method, so more investigations on this area should be done.

### 3.1.2. PARAFAC 2

As was said before, the PARAFAC model is based on Catell's principle of parallel proportional profile, which applied to this method means that, even when in a three-way array there is a variation between the observations in the first mode in the different occasions, they will be determined by the same set of factors. The problem with this assumption is presented when the slabs of the array have not the same dimensions (either in the samples or variables mode), maybe because all the samples could not be measured in all occasions, etc. For cases like this is why PARAFAC2 [43] was introduced, applying the parallel proportional profile principal in an adjusted way as follows:

$$X_k = A_k D_k B' + E_k \quad (16)$$

where  $X_k$  would be a matrix of  $(N_k \times J)$  corresponding with the  $k$ th slab,  $A_k$  would be the factor score matrix for the same slab, taking this way into account the variances that may exist between the slabs, and here it is the difference with the PARAFAC model. The rest of the members of the equation have the same interpretation as in PARAFAC. But in this representation, although  $D_k B$  are proportional there cannot be guaranteed uniqueness as the factor scores are not the same for all matrices. To achieve this uniqueness Harshman proposed in [41] the invariance constraint in the factor scores that the cross product  $A_k A_k'$  is constant over all  $k$ th slab. The final PARAFAC2 model would be then Eq.16 under the mentioned constraint.

There were two proposals for the fitting of the PARAFAC2 model: indirect and direct. The precursor of this method proposed fitting the cross product version of Eq.16 model to the observed cross products as follows:

$$C_k = B D_k \Phi D_k B' + E_k \quad (17)$$

where  $C_k$  denotes the cross product matrix associated with  $X_k$ ,  $\Phi$  denotes the invariant matrix  $A_k A_k'$  and  $E_k$  the residuals for the cross product matrix. But with this model fitting it is difficult to see directly what types of deviations from the tri-linear model are then allowed, and it is computationally complex and inefficient.

Therefore in [44] it is proposed a way to fit the PARAFAC 2 model directly to the data, instead of the derived model presented in Eq.16. This is called the direct fitting and besides the main advantage mentioned, it has some other advantages like: is easier to adjust when constraints are imposed, is better handling missing data and it is also easier to extend from three-way to n-way arrays. Then, as it is easily verified that  $A_k$  from Eq. 16 can be expressed with a column-wise orthonormal basis matrix of size  $N_k \times R$  ( $N_k \geq R$ ),  $P_k$  a  $R \times R$  matrix as following:

$$A_k = P_k A \quad (18)$$

and can be proven that the cross product is constant over the  $k$ th slab:

$$A_k' A_k = A' P_k' P_k A = A' A \quad (19)$$

Then the direct fitting of PARAFAC2 model is mathematically expressed as:

$$X_k = P_k A D_k B' + E_k \quad (20)$$

where  $B$  and  $D_k$  are the same as in Eq.16,  $A$  and  $P_k$  are from Eq.18. More details on this demonstration can be found in [44].

### 3.1.2.1. Uniqueness

Although PARAFAC2 uniqueness has been analyzed, there have not been so profound studies as for PARAFAC. In [ten Berge 1996] have been obtained some results demonstrating that the parameters estimated in the indirect fitting ( $B, D_k$  and  $\Phi$ ) are “essentially unique” up to some conditions (see [44]):

1. joint permutations of the columns of  $B$  and the columns and rows of  $D_1 \dots D_k$  and  $\Phi$ .
2. arbitrary scalings/reflections of the columns of  $B$  and of the supermatrix  $(D_1 \dots D_k)$ , combined with the inverse scalings/reflections of the columns and rows of  $\Phi$ .
3. reflections of any subset of the matrices  $D_1 \dots D_k$ .

There have been also studies for the uniqueness conditions of special cases as when  $R = 2$  (see harshman1996, kiers) where they present that a matrix with  $K \geq 3$  is unique if non-negativity is imposed to  $D_1 \dots D_k$  matrices and  $B$  has full column rank. If

$K < 3$  although there is the non-negativity constraint, the model won't be unique, the same will occur for  $K < 4$  when the model is unconstrained. There are other proven sufficient conditions in [43], [44] but not necessities as they have some incongruence with the previous condition.

In the case of the direct fitting, the conditions for essential uniqueness are directly related to the ones for indirect fitting, only that they are applied to the parameters of this model, and the simulations made on the paper suggest that the model gives unique solution as soon as  $K \geq 4$ , for arbitrary values of  $R$ . There is a Theorem in this paper that says:

“The parameters of the direct PARAFAC2 model are essentially unique if and only if the parameters of the indirect PARAFAC2 model are essentially unique ...”

As it can be seen, PARAFAC2 is a good alternative for solving problems with which a PARAFAC model cannot deal, like recovering underlying structure because the observations units vary from dataset to dataset (slabs, occasions), or there are different dimensionalities in one mode. In the case of the direct fitting it has even more advantages for imposing constraints, handling missing data and generalization of the model to n-way arrays. But even when this direct fitting is much easier to implement than the indirect and has more advantages, both are computing time consuming and that's maybe it hasn't been so exploited as PARAFAC. Also, the unique solutions can be achieved, but requires more level of variation and the presence of a sufficient number of levels in the third mode. So, there still some research to do in this topic.

### 3.1.3. Shifted factor analysis (S-PARAFAC)

S-PARAFAC was introduced by Harshman in [45], with the idea of dealing with the problem of factor shifting in the sequence of the measurements, which are mainly encountered in some data as time series, spectrums, etc. When it is spoken about shifting, it means that the values of the measures made on a sample, change systematically their relative position up or down on the sequence: if  $\tilde{v}$  is a shifted version of  $v$  then  $\tilde{v}_i = v_{i+s}$ , and  $s$  represents the amount of shift. This method is very similar to PARAFAC2 and has the same purpose in this sense, trying to relax the factor models, in this case PARAFAC. The PARAFAC model cannot be applied in cases like this where introducing the shift term disturbs the multilinearity, most of all when it is independent for each factor i.e. the shifts of one factor differ from those of another.

The idea of S-PARAFAC is to introduce into the PARAFAC model an explicit mathematical representation of any factor shifts present in a data set, so the model can be fitted even when the shifts are present and then describe the data correctly. The mathematical expression in Array Index Notation (AIN) would be:

$$x_{IJK} = a_{[I+s_{jr}]R} b_{JR} c_{KR} \quad (21)$$

The terms of Eq.21 are the equivalents to the PARAFAC expression in Eq.11 where  $x_{IJK} = X$ ,  $a_{[I+s_{jr}]R} = A$  and the same for  $b_{JR}$  and  $c_{KR}$ . The difference of this model is

that it allows for independent shifts of each factor at each level  $j$  of the data, so  $s_{jr}$  is the shift parameter that gives the shift at column  $j$  that is exhibited by factor or component  $r$  [45], being then  $A$  the sequential shifted mode as here it is assumed that it contains the sequentially ordered levels (e.g. different variables, time points) and  $B$  the shifting mode, which refers to the sequential profiles (e.g. objects). The modes above  $B$  reweight the factors.

The model can be also expressed in terms of the “shift operator”  $\mathfrak{S}$ , which was defined explicitly to facilitate the representation of these models. When using it, the elements of an array are shifted to a new position by adding a fixed amount to its subscripts; the amount is given to the operator. The representation would be then:

$$x_j = \mathfrak{S} s_j(A) \langle b_j \rangle C' \quad (22)$$

This expression is similar to the one for lateral slices of PARAFAC model, but now including the shift factor. In this representation, if it is kept  $A$  as shifted and  $B$  as shifting, the model cannot be in terms of the frontal slices, as shifts in  $A$  change from one column of  $B$  to the next, which makes impossible to use a fixed  $A$  for all columns of  $B$  simultaneously. S-PARAFAC can be also generalized to  $N$ -way arrays [46].

There are cases where shifting problems can be found in more than one mode, also known as multiple-shifting, but this method has not yet incorporate how to deal with it, what could be considered as a limitation specially now with the instrumental equipment development e.g the combination of equipments for analytical chemistry as GC-MS (Gas-Chromatography and Mass-Spectra) where there can be a shifting problem in both chromatography and mass mode. Another limitation is that sometimes the shift or peaks position is combined with shape changes and this method doesn't stand these changes.

As can be seen S-PARAFAC is a version of PARAFAC2 as both share the same purpose in what to shifting problems refer, but besides the already mentioned limitation, its main difference with PARAFAC2 and also considered as a limitation is that it cannot capture the structure of the data, it can only handle the shifting factors. Nevertheless, it may be sometimes more effective than PARAFAC2, as this depends on the inner product of the factors.

In [45] there are more details about shifting factor analysis, starting from two-way data and then generalized to three-way. One of the aspects treated is the uniqueness of the model, where a conjecture of what could be a proof of essential uniqueness for the bilinear model is presented.

Details about the algorithm will be presented in Section 4.

#### 3.1.4. Sparse Non-Negative Tensor Factor Double Deconvolution (SNTF2D) (cPARAFAC)

The Sparse Non-Negative Factor Double Deconvolution (SNTF2D) [47] or presented in [4] as cPARAFAC, is a generalization of Non-negative Matrix Factor Deconvolution (NMF2D) [48] for multi-way spectral data to measure convolutive mixtures, so the process of finding the loading matrices is seen as a deconvolution process as each row of the multi-way  $X$  matrix is considered as a convolutive mixture of the factor matrices' rows. As will be seen in Eq.23, the cPARAFAC model, which is only convolutive in two of the three modes (and this

is one of the limitations), is very similar to PARAFAC model and becomes equivalent to it when some parameters take determine values, this is why the model has been defined as 2-D Convolutional PARAFAC model and is defined as:

$$x_{ijk} = \sum_{r=1}^R a_{ir} b_{(j-\theta)r}^{\tau} c_{(k-\tau)r}^{\phi} + e_{ijk} \quad (23)$$

or in terms of the Khatri-Rao product:

$$X = A \left( \sum C^{\phi} \square B^{\tau} \right)' + E \quad (24)$$

Where  $A$  would represent the instantaneous linear mix of the original rows e.g. channel in a multi-channel time-frequency analysis. The terms  $\phi$  and  $\tau$  represent the factor shifting at the second and third mode respectively. In Eq.24, the  $\downarrow$  is the downward shift operator which indicates how many rows down will be moved in the matrices, thus  $B^{\tau}$  denotes that each element  $B$  will be moved  $\phi$  rows down and the same for  $C^{\phi}$  and this is how the relation between the different modes is seen. Any other combination of the component matrices is equivalent to the one of Eq.24.

This model is equivalent to the traditional PARAFAC model [49] if there was no shifting in any of the modes, and also equivalent to the S-PARAFAC model in case the shifting was analyzed in only one mode. It can be also extended to n-way data and incorporate the convolutional mixtures in the  $n-1$  modes.

The sparseness constraint has been imposed to the NMF2D due to this model, as a generalization of factor decomposition, carries the non-uniqueness problems and therefore sometimes fails to find the correct solution, and although it is already a constrained version of factor decomposition, where the non-negativity constraint has been imposed precisely to avoid this problem in which these models can incur, this isn't enough. Thus, the sparseness constraint has been proposed as a better alternative to obtain unique solutions [50]. Details on the imposed sparseness cost to the model and the algorithm can be found in Section 4.1.3.

### 3.1.5 Parallel Profiles with Linear Dependencies (PARALIND) (restricted PARATUCK2)

In real chemical and other types of data, sometimes are variations in their underlying structure that cannot be analyzed without taking into account the interaction among the different factors of the different modes. Although PARAFAC model has the very attractive uniqueness properties, the structure of this type of data is too complex to be modeled by it; some important information might be ignored if the interactions are not taken into account. Tucker (see Section 3.1.6) models on the other hand, are more flexible, they allow to analyze all these interactions, but under the cost of rotational freedom problems and less interpretable

solutions. Thus PARATUCK2 [51], [12] models emerge trying to combine both methods advantages: flexibility and uniqueness properties. The model is then defined as:

$$X_k = AD_k^A HD_k^B B' + E_k \quad (25)$$

$D_k^A$  is an  $R$  by  $R$  diagonal matrix containing the weights for the columns of  $A$  at level  $k$  of the third mode. Similarly,  $D_k^B$  is an  $S$  by  $S$  diagonal matrix giving the weights at level  $k$  for the columns of  $B$ .  $H$  is an  $R$  by  $S$  and is similar to the core matrix of Tucker models, which permits the interaction among the factors. From here it can be seen that PARATUCK2 is general in that it requires neither symmetry of the data or model nor the same dimensionality for Modes  $A$  and  $B$  [51].

In this reference are also given some proofs of PARATUCK2 models' uniqueness, although only in detail when  $R = S$ , for which is needed that all loading matrices and  $H$  are of full rank,  $H$  has no zero elements, and there is an "adequate variation" in the weight matrices  $D_k^A$  and  $D_k^B$  in the way is described in [50]. They also made some studies for  $R > S$  and suspect that for this cases unique solutions may be found, but deeper investigations must be done.

This PARATUCK2 model is nevertheless not suitable for some phenomenon rank-deficiency, which is found in some types of chemical data. Sometimes is possible to find dependencies in the underlying structure which can produce that the loadings of one or more modes of the  $n$ -way array become linearly dependent, such that the rank of the component matrix is lower than the number of columns, so they become rank-deficient. This phenomenon puts in risk the unique solution of the PARAFAC model for the factors involved as the Kruskal's conditions will not be accomplished. The Parallel Profiles with Linear Dependencies (PARALIND) [52], [53] models were specifically developed for this purpose as a constrained version of PARAFAC or restricted version of PARATUCK2. The  $H$  matrix is called now the "dependency matrix" and is used so the intrinsic rank-deficiency can be explicitly incorporated into the model as in Eq.25 where the linear dependencies are only present in one mode:

$$X_k = \tilde{A} D_k (C) B' + E_k \quad (26)$$

or the matricized version:

$$X = \tilde{A} (C \square B)' + E \quad (27)$$

with

$$\tilde{A} = AH \quad (28)$$

where  $A$  would represent the  $(I \times R)$  component matrix of the mode with linear dependencies (lower rank than the others) and  $H$  a  $(R \times S)$  binary matrix by which this

dependencies between the different factor are represented;  $P$  is the number of independent columns of  $A$ . In a  $(2 \times 3)$  matrix for example, where the last two factors of  $A$  are linearly dependent the matrix  $H$  would have the form:

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad (29)$$

This representation can be adapted for cases when there are linear dependencies in more than one mode. Also, in cases where the dependency is between more than two factors, the  $H$  matrix can be more complex (and non-binary), and it doesn't force the user to have previous knowledge of the data's structure, it can be calculated from the data itself (similar to PARATUCK2).

Although PARAFAC models is not prepared to recover uniquely the factors involved in the linear dependencies, the authors propose that the uniqueness or partially uniqueness of the PARALIND model would give unique solutions when determine constraints are imposed e.g. non-negativity, unimodality, but sometimes are not enough. They assume that uniqueness or partially uniqueness of this model would follow from the PARAFAC model, depending on the combination of its tri-linearity and added structural constraints imposed onto it.

This model is a very good option for this type of problems, and apparently it really inherit PARAFAC uniqueness under some constraints, in fact the authors have demonstrated it for flow injection analysis (FIA) [53] and Fluorescence data [52], but further studies most be done for other types of applications.

As the years go by, the use of new instrumental equipments which give as outcome multi-way data increases, and the researchers apply multi-way models to make their analysis, so more versions of the existing models and new models will arise for specific problems in all the research areas e.g. PARAFAC with splines [54], PARAFASCA [55]. In these subchapters, the most reported versions of the PARAFAC model in the literature have been presented, trying to bring out the advantages and disadvantages of each of them, their usefulness and other important issues related. In the following we will refer to other remarking multi-way methods that have had also a big impact in multi-way analysis: Tucker models.

### 3.1.6. TUCKER models

The Tucker models were of the first models introduced for multi-way data analysis by Ledyard Tucker [2], [15]. These are also known as N-mode principal component analysis as are in some way an extension of PCA. The most used of Tucker models for multi-way analysis and will be mainly described here is the Tucker3 model, whose mathematical expression is very similar to PARAFAC's, but with some changes that mark the main differences between both methods. The expression is presented hereafter:

$$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R a_{ip} b_{jq} c_{kr} g_{pqr} + e_{ijk} \quad (30)$$



where  $a_{ip}$  is an element of the  $A(I \times P)$  loading matrix,  $b_{jq}$  an element of  $B(J \times Q)$ ,  $c_{kr}$  of  $C(K \times R)$ ,  $e_{ijk}$  an element of the  $E(I \times J \times K)$  error array and  $g_{pqr}$  an element of the so called G-core array  $G(P \times Q \times R)$ . As can be seen, unlike the PARAFAC model, in this expression the G-core array cannot be absorbed by the other loading matrices. In this model the core array is non-diagonal, allowing the different loading matrices to differ in the number of components. Thus, there will be an interaction (complex) among the distinct components in the different modes, which is an important characteristic of this model, its flexibility, giving it some attractive features for the analysis of the underlying structure of the data. This is the main difference with the PARAFAC model, where the component vectors only interact factor-wise, and what makes also difficult the interpretation of the results. So, this property of the core-array can be seen as an advantage of the model from some point of view, but a limitation from another.

The Tucker3 model can be also expressed in a more efficient way in terms of the Kronecker product:

$$X = AG(C \otimes B)' + E \quad (31)$$

where  $X$  and  $E$  are defined as in Eq.13;  $A$ ,  $B$  and  $C$  are the loading matrices and  $G = [G_1 \ G_2 \ \dots \ G_R]$  is the matricized core array of size  $(P \times QR)$  where  $G_r$  is the  $r$ th slab of  $G$ .

Besides Tucker3 model, in which the three modes are reduced with the different number of components for each one, there are models where only two or even one of the models is reduced i.e. the loading matrix that is not reduced is chosen to be the identity matrix and has the dimensions e.g.  $(K \times K)$  for the third mode. These models were called Tucker2 and Tucker1 in dependence of whether two or one mode was not reduced, respectively. The mathematical expression of Tucker2 and Tucker1 models are presented in the following equations in the same order, in the form of the Kronecker product:

$$X = IG(B \otimes A)' + E = G(B \otimes A)' + E \quad (32)$$

and

$$X = IG(I \otimes A)' + E = AG + E \quad (33)$$

All the terms of the equations above have the same meaning as in Eq.31. In Eq.32 it is assumed that the third mode ( $C$ ) does not change, and in Eq.33 the second mode ( $B$ ) is neither reduced. These expressions can of course change in dependence of which modes are reduced or not. As can be seen in Eq. 33, Tucker1 model can be easily interpreted as PCA model, so the same algorithms can be used for it.

### 3.1.6.1. Uniqueness

As in PCA, the Tucker3 model has the problem of rotational freedom, the subspaces provided by the model is unique but the parameters cannot be uniquely estimated i.e. if a reflection/rotation transformation (orthogonal) is made on the loading matrices there is no change in the model's fit. This is considered as a limitation and is one of the aspects that make researchers prefer PARAFAC over Tucker3 model. Nevertheless, this property of the Tucker3 model can be useful to facilitate the interpretation of the core-array and the loading matrices, by constraining the loading matrices to be orthogonal, so all the variation in the data is contained in  $G$ . This constraint also avoids problems with the degenerate solutions. But this is not enough to avoid the rotational freedom, extra constraints must be imposed as: the loading matrix  $A$  needs to contain the eigenvectors of  $X(CC' \otimes BB)'$  in order of decreasing eigenvalues and similar ones must be imposed to  $B$  and  $C$ . More details on this aspect and about how to simplify the core-array can be found in [3], [56], [57]. This problem of rotational freedom is also generalized to Tucker2 and Tucker1 models.

### 3.1.7. Other Tucker versions

As for PARAFAC, in [48] was also proposed a Shifted version for Tucker3 (S-T3) and Tucker2 (S-T2) models. So, assuming as before that mode  $A$  is the shifted mode and  $B$  is the shifting mode, the S-T3 model would have the form:

$$x_{IJK} = a_{[I+s_{jp}]P} b_{JQ} c_{KR} g_{PQR} \quad (34)$$

where the terms have the same interpretation as in Eq.21. A model for S-T2 was also developed:

$$x_{IJK} = a_{[I+s_{jp}]P} c_{KR} g_{PQR} \quad (35)$$

In these models it is only considered shifting in one mode; other possibilities have not yet been explored. The authors have strong conjectures that these shifted versions can be unique, unlike the original Tucker models.

Recently a Sparse Non-Negative Tucker (SN-TUCKER) factorization was proposed in [58]. The authors developed ways to impose sparseness in any combination of modes and proposed algorithms as for cPARAFAC based on least squares minimization and KL divergence minimization. They demonstrate how these new algorithms are superior to the proposed ones when data and interactions can be considered non-negative.

### 3.1.8. Multi-linear partial least squares regression (N-PLS)

In multi-way analysis, regression problems can be found as in two-way analysis for the prediction of unknown quantitative information from the information available. In this

kind of problems the arrays are composed by two blocks: the multi-way array  $X$  with the independent measured variables (prior knowledge), and a vector, matrix or multi-way array  $Y$  for the dependent variables.

For this purpose some methods have been already proposed. Such is the case of PARAFAC and Tucker, though are mainly used for exploratory analysis, they can be used for regression by decomposing the three-way array and then regress the dependent variables on the obtained loadings. But these procedures have some problems with the necessary components for the decomposition and the regression, as they are done separately. In [59], the authors proposed the method known as Multi-way covariates regression to compensate this situation. More details on this can be also found on [3].

The most popular of the methods proposed for multi-way regression is the Multi-linear PLS (N-PLS) [18] , [60], which is an extension of the bilinear PLS for multi-way arrays. The idea is the same as in PLS, to find the components by describing the covariance of the dependent and independent variables. But, unlike the other multi-way regression methods, in N-PLS the multi-linear models for the dependent  $Y(I \times L \times M)$  and independent variables, and the regression model that relates the two decompositions are done simultaneously; all of these constitute the regression model. In the decomposition models to be presented afterwards the notation will be different to the used in the previous models to be consequent with the notation used in traditional PLS models:

$$X = T(W^K \otimes W^J)' + E \quad (36)$$

and

$$Y = U(Q^M \otimes Q^L)' + E \quad (37)$$

The  $T$  and  $U$  matrices represent the scores of  $X$  and  $Y$  decompositions respectively. Assuming that  $X$  and  $Y$  are two three-way matrices with  $(I \times J \times K)$  and  $(I \times L \times M)$  dimensions respectively, then  $W$  and  $Q$  matrices refer to the loadings of both decompositions and the superscript indicates to which mode belongs each one of them. The models above are under the restriction that the loading vectors found most conduce to the scores vectors of maximal covariance. The regression model would be then as follows:

$$U = TB + E \quad (38)$$

where  $B$  contains the regression coefficients obtained from the regression of  $Y$  on  $T$ . Although the presented model refers to three-way arrays, then N-PLS model, as its name indicates, can be generalized to multi-way arrays, and as in this case a similar model to PARAFAC's is assumed for the decomposition steps.

As the N-PLS method consists in finding successively one-component model (rank-one tri-linear model) (see Section 4.2.), and as discussed in Section 3.1.1.1., it is

demonstrated that rank-one multi-linear models are always unique, the author deduce that the N-PLS solutions are unique.

### 3.1.9. Selecting the number of components

One of the important issues of component models is the selection of the right number of components to use in the model. For the selection of this number exist several tools, but its use depends on what the purpose of the model is. In [3] some data analytical validation tools are described: scree plots, split-half analysis, cross-validation and visually checking residuals; the authors do not advise to be confident in only a particular one, as not all datasets have to accomplish the assumptions of this rule.

Besides the general tools, there have been proposed some others for specific models, such as: DIFFIT [61] for the Tucker3 models, and later in [62] they proposed DIFFIT on approximate fit, which improves the speed of this method with similar results in the optimal fit. Also in [63] the authors proposed a new way of finding the number of components for tri-linear decomposition (the authors made emphasis in PARAFAC model) by incorporating the tri-linear character of the data into the component-determining procedure. This new method is called ADD-ONE-UP and the authors demonstrated it is of easy implementation and has a very good performance. Bro and Kiers also proposed in [64], a method for determining the number of components for PARAFAC models, although it can be also used in restricted Tucker3 models. These core consistency diagnostic method (CORCONDIA), also quantifies subjective results, it helps to determine if the model is stable or not. Nevertheless, the authors suggest that the determination of the components should not be done either on this solo criterion.

Although all these methods are very helpful to determine the most suitable dimensionality of the model, the definition of the final one is responsibility of the analyst in dependence of the problem at hand and the background knowledge he has on it.

## 4. Algorithms

For the estimation of the multi-way component models' optimized parameters, i.e. the parameters that minimize most the loss function:

$$L(parameters) = \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K (x_{ijk} - \hat{x}_{ijk})^2 \quad (39)$$

there have been proposed several algorithms which are divided in three main groups: The first one is formed by the alternating algorithms, which update only a subset of the parameters at each step. In this group are among others: Alternating least squares (PARAFAC-ALS, TUCKALS-3) [16], [65], a alternating slice-wise diagonalization (ASD) [66], alternating tri-linear decomposition (ATLD) [67], self-weighted alternating tri-linear decomposition (SWATLD) [68], pseudo alternating least squares (PALS) [69], alternating

coupled vectors resolution (ACOVER) [70], and alternating coupled matrices resolution (ACOMAR) [71]. In the second group are the derivative based methods, seeking an update for all the parameters simultaneously by successive approximations: Positive Matrix Factorization for 3-way arrays (PMF3) [72] and damped Gauss–Newton (dGN), also known as Levenberg–Marquadt [72]; [73]. And in the third group are the direct (non-iterative) procedures where the most known are: the Generalized Rank Annihilation Method (GRAM) [74] and the Direct Tri-Linear Decomposition method (DTLD) [75], both based on a generalized eigenvalue problem, but GRAM works only when there are only two slices in one mode and DTLD is a generalization of it to data with more than two slices by generating two pseudoslices as differently weighted averages of all the slices. There are other algorithms created to resolve particular problems as is the case of [76] which was introduced to deal with the influence of model deviations on the predictive accuracy in second-order calibration.

All this relation of algorithms we took it from In [77] and [compare fitting [78], and in both a comparison of the mentioned algorithms is made, in terms of the complexity to incorporate constraints in, robustness, convergence, handling of missing values, sensibility to over-factoring, etc. Besides, the mentioned algorithms are of course more detailed in the respective papers.

As it can be seen in the referenced papers, algorithms that are variations of the ALS or not, are trying to solve its speed problems, but it results that even when they are faster, when the model is over-factored specially, their capability for fitting the model is poorer and most of them can only deal with unconstrained three-way models. In this case is also [79] who proposes a modified version of the PARAFAC algorithm using penalty diagonalization error (PDE) to relieve the slow convergence caused by the random initialization and it is insensitive to the over-factoring, but only for three-way data. So, although ALS has its slow convergence problems, it is guaranteed, and still the only one that handles higher order arrays, constrained models, missing data and weighted least square loss functions. Due to this convergence characteristics and other theoretical implications related to the measurement noise, is that Vega Montoto created an extension of the Maximum Likelihood PCA (MLPCA) [80] for the PARAFAC-ALS algorithm, Maximum Likelihood PARAFAC (ML-PARAFAC) [81], [82], [83], [84], which accounts for measurement errors in the estimation of model parameters.

For all mentioned before, the ALS algorithm is preferred over the other recently introduced when a better quality of the estimated model is searched, and this is why we chose it to deepen into.

Nevertheless, although when using an algorithm it is always recommended to look for the alternative that fits the most to the problem at hand; in cases where the priority is the computation time, this is definitely not the advised one. Recently a way to improve this problem in PARAFAC-ALS algorithm was introduced by [85], which proposes to accelerate the algorithm for three and four-way data and reduce storage requirements by utilizing multi-dimensional wavelet compression.

#### 4.1. Alternating Least Squares

The Alternating Least Squares basics were introduced in 1933 by Yates and the main idea is to resolve a bigger optimization problem with smaller sub-problems iteratively. In the algorithm the parameters to estimate are separated in different sets (as few as possible); this separation will make possible the use of simpler algorithms for the estimation of the parameters. In each iteration these sets will be fixed except one that will be left free to vary, with which will be minimized a new loss function depending only on it. The algorithm will iterate, alternating from one set to another until there is no change observed in the loss function or the parameter values, or their variation is less than a predefined convergence criterion. The fewer the sets are, decreases the possibility of finding local minima and slow convergence. If the algorithm converges to the global minimum, the least squares model is then found. In the least square sense, the algorithm should not get a worse fit at any step, the loss function tends asymptotically to a minimum so if it doesn't improve the least it can do is to keep the same convergence as in previous steps.

Following, it will be presented the steps of a generic ALS algorithm as in [12].

Given an array  $X$  consider the general model

$$X = f(A, B, C, \dots) + E \quad (40)$$

To estimate the parameters  $A, B, C$ , etc. an ALS algorithm can be as:

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

where  $f$  is the model of  $X$  and is a function of the parameters  $A, B, C, \dots$ . It is a matrix of the same size as the data held in  $X$ .  $\|\cdot\|_F$  refers to the Frobenius norm. The first step will be treated in each of the following specific algorithms of PARAFAC, TUCKER, etc. In the last step it is determined whether the algorithm has already converge or not and it will depend on the conditions we mentioned before.

This algorithm as already mentioned has the advantages that besides it is easy to implement and simple comparing to the algorithms that work simultaneously, can handle missing data, constraints, and can be extended to n-order arrays and guarantees to converge. This last property is very attractive and one of the reasons for its popularity. Nevertheless it has problems with slow convergence when the called swamps (explained in Section 3.1.1.2) or high co-linearity are present, cannot withstand the presence of outliers which are very common in many research lines e.g. Chemometrics. It also carries other disadvantages of the iterative algorithms with: 1. the fact that a stopping criterion have to be chosen beforehand, 2.

the starting point that would lead to the best solution has also to be chosen, etc. More details on the iterative algorithms can be found in [5].

So we can conclude that, besides the alternating least squares algorithms are actually the most used in multi-way analysis because of all the advantages they have, and that the problems that can arise also depend on the data, there are very important disadvantages that still have to be treated.

#### 4.1.1 PARAFAC-ALS

The PARAFAC-ALS was introduced by [16], [17] where  $X$  is a three-way array, so there are three sets of parameters ( $A$ ,  $B$ , and  $C$ ) and the loss function :

$$L(a_{11}, a_{12}, \dots, c_{kr}) = \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K (x_{ijk} - \sum_{r=1}^R a_{ir} b_{jr} c_{kr})^2 \quad (41)$$

or with the Eq. 13

$$\min_{A,B,C} \left\| X - A(C \square B) \right\|^2 \quad (42)$$

In this algorithm as was said before, the parameters are updated in each operation until the convergence, alternating:  $A$  given  $B$  and  $C$ ,  $B$  given  $A$  and  $C$ , and  $C$  given  $A$  and  $B$ . Due to the symmetry of the model, an update of one mode is essentially identical to an update for any of the modes with the role of the different loading matrices shifted [3]. To estimate  $A$  conditionally on  $B$  and  $C$  results in the optimization problem:

$$\min_A \| X - AZ \|^2 \quad (43)$$

with

$$Z = C \odot B = [D_1 B' D_2 B' \dots D_k B']' \quad (44)$$

and the diagonal matrix  $D_k$  is the same as in Eq.12. Then when given  $B$  and  $C$  the least squares optimal  $A$  can be found as:

$$A = X(Z')^+ = XZ(Z'Z)^{-1} \quad (45)$$

The updating of  $B$  and  $C$  can be done in the same way due to the mentioned symmetry property of the model. So now it will be presented the PARAFAC-ALS algorithm where  $X(I \times J \times K)$  is unfolded to an  $I \times JK$  array.

1. Initialize  $B$  and  $C$

2.  $Z = C \square B$   
 $A = X_{I \times J \times K} = Z(Z'Z)^{-1}$
3.  $Z = C \square A$   
 $B = X_{J \times I \times K} = Z(Z'Z)^{-1}$
4.  $Z = B \square A$   
 $C = X_{K \times J \times I} = Z(Z'Z)^{-1}$
5. Go to step 1 until relative change in fit is small

This version anyway is computationally costly for large arrays and the continuous rearranging of the array in the three unfolded versions requires excessive memory. Harshman [16] and Carroll [17] noted that a simpler updating scheme is possible due to the special structure of the problem, formulated in terms of the frontal slices,  $X_k$ , then the multiplication of  $XZ$  and  $ZZ'$  is done straight out from  $B$ ,  $C$  and  $X$ . So making use of Eq.12:

$$XZ = [X_1 X_2 \dots X_k][D_1 B' D_2 B' \dots D_k B']' = X_1 B D_1 + X_2 B D_2 + \dots + X_k B D_k \quad (46)$$

and

$$ZZ' = (C' C) * (B' B) \quad (47)$$

we can define  $A$  as:

$$A = \left( \sum_{k=1}^K X_k B D_k \right) \left\{ (C' C) * (B' B) \right\}^{-1} \quad (48)$$

The following algorithm is more efficient and although the calculation for all the modes is not identical as in the previously shown, it is still symmetric and equivalent to the previous one.

1. Initialize  $B$  and  $C$
2.  $A = \left( \sum_{k=1}^K X_k B D_k \right) \left\{ (C' C) * (B' B) \right\}^{-1}$
3.  $B = \left( \sum_{k=1}^K X_k' A D_k \right) \left\{ (C' C) * (A' A) \right\}^{-1}$
4.  $d_k = \left\{ (B' B) * (A' A) \right\}^{-1} \left\{ (A' X_k B) * I \right\}, k = 1, \dots, K$
5. Go to step 2 until relative change in fit is small

There are some other ways to represent this algorithm, these were taken from [3] and there are more details about it.



As introduced previously, the ALS algorithms have the limitation that the starting point (initialization values) and stopping criterion have to be chosen. These require a lot of work to find the ones that would lead to a good solution and they are very important as any error can lead to slow-convergence, a local minima and finally not getting the most fitted model.

A good initialization of the parameters can influence in the speeding up of the algorithm and decreases the chances of getting to a local minimum, most of all in difficult cases like: loading vectors are intrinsically correlated or there is over-factoring and then the noise is included in the modeling. There have been some proposals for the initialization points e.g. [21] suggests a random initialization and starting the algorithm from different points but this usually do not lead to a good estimation. There have been suggested also the term for a “good initialization as possible”, rational start and a similar one semi-rational start, which of course should be as close as possible to the true solution and therefore save computational time. There are more details in [3] on how to select these points. At the end the advice is to fit the model from the three mentioned points of view and if the same solution is obtained, the chances of a local minimum are very low.

In the case of the stopping criterion, as the convergence of the fit model does not mean that the parameters have converged, then the small changes between the fitness of one iteration and another can be a criterion. Anyway, following the changes in the parameters instead, can be another criterion, it won't be equivalent to the previous one as it will converge simultaneously and changes in the parameters can still be detected. But in cases when there is two-factor degeneracy it can end in infinite iterations searching for the convergence of the parameters estimates even when the fit has already converge, this can only be fixed imposing the number of iterations of the algorithm.

#### 4.1.2. PARAFAC2-ALS

Because of the structure of the PARAFAC2 model (see Section 3.1.2), although it was introduced by Harshman in 1972 [43] it was not until 1993 that Kiers [86] proposed an algorithm for this fitting (indirect). This is, as demonstrated, computationally complex and time consuming, so in [44] it is proposed a simpler one(direct fitting) and it will be the presented here. If it is used the PARAFAC2 model defined in Eq.20, the optimization function would be then:

$$\min_{A,C,B,P_k} \sum_{k=1}^K \left\| X_k - P_k A D_k B' \right\|_F^2 \quad (50)$$

minimizing all its arguments, subject to the constraints that  $P_k' P_k = I_R$  and  $D_k$  as was said before is diagonal. If  $A D_k B' X_k'$  is decomposed by singular value decomposition then  $P_k$  can be expressed as:

$$P_k = V_k U_k' \quad (50)$$

The minimization problem shown is then equivalent to:

$$\min_{A,C,B,P_k} \sum_{k=1}^K \left\| P_k' X_k - AD_k B' \right\|_F^2 \quad (51)$$

$P_k$  can be calculated also as:

$$P_k = X_k B D_k A' (A D_k B' X_k' X_k B D_k A')^{-1/2} \quad (52)$$

So finally the algorithm steps are:

1. If  $J < N_k$ , replace  $X_k$  by  $H_k (M_k \times J)$  from the Cholesky decomposition of  $X_k' X_k = H_k' H_k$ , and  $M_k$  is considerably smaller than  $N_k$ .
2. Initialize  $A$ ,  $B$  and  $D_k$ .
3. Compute the SVD of  $A D_k B' X_k'$  and update  $P_k = V_k U_k'$ ,  $k = 1, \dots, K$ .
4. Update  $A$ ,  $B$  and  $D_k$  by one iteration of PARAFAC-ALS applied to  $P_k' X_k$ .
5. Go to step 2 until relative change in the fit is small.
6. If  $X_k$  has been replaced by  $H_k$  in Step 1, replace  $H_k$  by  $X_k$  again and compute  $P_k$  according to Step 3,  $k = 1, \dots, K$ .

More details about this algorithm can be found in [44]. Here it is also suggested to initialize  $B$  as the loading matrix from PCA on  $\sum_k X_k' X_k$ , and  $A$  and  $D_k$  ( $k = 1, \dots, K$ )

as  $I_R$ . As in the previous algorithms it can be also suggested to try using this initialization and other with random values to verify convergence and assess uniqueness.

This algorithm for the direct fitting is less complex computationally than the indirect fitting one, but as can be seen there are operations that can explain the computation time required by PARAFAC2 and this must be one of the main reasons of why it hasn't been so accepted or applied unless it is really necessary. Nevertheless, in the last years its use has been increased mainly in the Chemometrics field.

#### 4.1.3. cPARAFAC

Finding an efficient algorithm for matrix factor deconvolution has been a difficult task, although some have been proposed [87], [88], most of them cannot reach the global minimum of the cost function, but only a local minimum. Fast and easy to implement algorithms, based on the developed ones for the traditional matrix factor decomposition, have been proposed and proof to converge for the NMF2D [48], but again, only for a local minimum of the cost function. Such is the case of the least squares minimization algorithm, and the other based on the Kullback-Leibler (KL) divergence minimization, with a transformation matrix to form the convolution. These two algorithms were also extended for sparse non negative matrix factor double deconvolution (SNMF2D) [48] and later for

SNTF2D [47] in which a sparseness penalty is added to the cost function, like in the following equation for the Least Squares algorithm:

$$C_{SparseLS} = \frac{1}{2} \left\| X_{(I \times P)} - \hat{X}_{(I \times P)} \right\|_F^2 + C_{Sparse}(C) \quad (53)$$

with

$$C_{Sparse}(C) = \beta \|C\|_1 = \beta \sum_{p,\theta,r} C_{k,r}^\phi + C_{Sparse}(C) \quad (54)$$

where  $p = jk$  and  $\beta$  the weight of sparseness to the reconstruction error. The sparseness cost is imposed to restrict  $C$  to be sparse in order for the underlying structure of the data to be present in  $B$  and it can be any function with positive derivative. The selection of  $\beta$  could become a drawback of the method as it is not trivial.

The Alternating Least Squares for cPARAFAC is presented:

1. Initialize  $A, B$  and  $C$  randomly

$$2. \hat{X} = A \left( \sum_{\tau} \sum_{\phi} C^{\phi} \square B^{\tau} \right)'$$

$$3. A \leftarrow A \bullet \frac{XZ + \text{Adiag}(1(AZ'Z) \bullet A)}{AZ'Z + \text{Adiag}(1(XZ) \bullet A)} \quad \text{where } Z = \left( \sum_{\tau} \sum_{\phi} B^{\tau} \square C^{\phi} \right)$$

$$4. A_{i,r}^{\tau} = \frac{A_{i,r}^{\tau}}{\|A_r\|_F}, \hat{X} = \sum_{\tau} \sum_{\phi} B^{\tau} (C^{\phi} \square A)'$$

$$5. B^{\tau} \leftarrow B^{\tau} \bullet \frac{\sum_{\phi} \hat{X} (C^{\phi} \square A) + B^{\tau} \text{diag}(1(\sum_{\tau} (\hat{X})(C^{\phi} \square A)) \bullet B^{\tau})}{\sum_{\phi} \hat{X} (C^{\phi} \square A) + B^{\tau} \text{diag}(1(\sum_{\tau} (\hat{X})(C^{\phi} \square A)) \bullet B^{\tau})}$$

$$6. B_{j,r}^{\tau} = \frac{B_{j,r}^{\tau}}{\|B_r\|_F}, \hat{X} = \sum_{\tau} \sum_{\phi} C^{\phi} (B^{\tau} \square A)'$$

$$7. C^{\phi} \leftarrow C^{\phi} \bullet \frac{\sum_{\tau} (\hat{X})(B^{\tau} \square A)}{\sum_{\tau} (\hat{X})(B^{\tau} \square A) + \beta \frac{\partial C_{Sparse}(C)}{\partial C^{\phi}}}$$

8. Repeat from step 2 until convergence.

The unit Frobenius Norm constraint, included in steps 4 and 6 of the algorithm, was added to avoid the elements of the component matrix  $C$  to go to zero and so  $A$  and  $B$  to go to infinity, which can be an after-effect of the sparseness constraint.

The other algorithm proposed for this model, based on the KL divergence minimization

$$C_{SparseKL} = D\left(X_{(IxP)} \middle| \hat{X}_{(IxP)}\right) + C_{Sparse}(C) \quad (55)$$

with

$$D\left(X_{(IxP)} \middle| \hat{X}_{(IxP)}\right) = \sum_{i,p} X \log \frac{X}{\hat{X}} - X + \hat{X} \quad (56)$$

is shown hereafter:

1. Initialize  $A, B$  and  $C$  randomly

$$2. \quad \hat{X} = A \left( \sum_{\tau} \sum_{\phi} C^{\phi} \square B^{\tau} \right)'$$

$$3. \quad A \leftarrow A \cdot \frac{X}{AZ'} + \text{Adiag}\left(1 \cdot ((1Z) \bullet A)\right) \quad \text{where } Z = \left( \sum_{\tau} \sum_{\phi} B^{\tau} \square C^{\phi} \right)$$

$$1Z + \text{Adiag}\left(1 \cdot \left( \frac{X}{AZ'} Z \bullet A \right)\right)$$

$$4. \quad A_{i,r}^{\tau} = \frac{A_{i,r}^{\tau}}{\|A_r\|_F}, \quad \hat{X} = \sum_{\tau} \sum_{\phi} B^{\tau} (C^{\phi} \square A)'$$

$$5. \quad B^{\tau} \leftarrow B^{\tau} \cdot \frac{\sum_{\phi} \frac{\hat{X}}{\hat{X}} \left( C^{\phi} \square A \right) + B^{\tau} \text{diag} \left( 1 \cdot \sum_{\tau} \left( 1 \left( C^{\phi} \square A \right) \right) \bullet B^{\tau} \right)}{\sum_{\phi} 1 C^{\phi} + B^{\tau} \text{diag} \left( 1 \sum_{\tau} \left( \left( \frac{\hat{X}}{\hat{X}} \right) \right) \left( C^{\phi} \square A^{(i)} \right) \bullet B^{\tau} \right)}$$

$$6. \quad B_{j,r}^{\tau} = \frac{B_{j,r}^{\tau}}{\|B_r\|_F}, \quad \hat{X} = \sum_{\tau} \sum_{\phi} C^{\phi} (B^{\tau} \square A)'$$

$$7. \quad C^\phi \leftarrow C^\phi \bullet \frac{\sum_{\tau} \left( \frac{\uparrow_{\tau} X}{\hat{X}} \right)^{\downarrow \phi} (B^{\tau} \square A)}{\sum_{\tau} 1 \left( B^{\tau} \square A \right) + \beta \frac{\partial C_{Sparse}(C)}{\partial C^\phi}}$$

8. Repeat from step 2 until convergence.

No problems of divergence have been reported in the literature with these algorithms until now, in fact it has been shown that they were able to identify the components correctly, but although the NMF2D has been proven to converge, and the inclusion of sparsity hasn't made them diverge, there is neither a proof of their convergence, only well founded conjectures.

#### 4.1.4. PARATUCK2-ALS and PARALIND-ALS

Although PARATUCK2 was introduced by Harshman [22], it was not published an algorithm by that time, it was in [12] that was proposed an ALS algorithm for this model, which will be presented afterwards. The loss function for this algorithm comes also straight from the model:

$$\min_{A, D_k^A, H, D_k^B, B} \sum_{k=1}^K \left\| X_k - AD_k^A H D_k^B B \right\|_F^2 \quad (57)$$

So the algorithm will consist in the following steps:

1. Initialize  $A, D_k^A, H, D_k^B$  and  $B, k = 1, \dots, K$
2.  $F_k = D_k^A H D_k^B B', A = X(F^+)'$  where  $F = [F_1 \quad F_2 \quad \dots \quad F_K]$
3.  $F_k = B D_k^B H', \text{vec} X_k = (F_k \otimes A)' (C_{(k,:)}^A)'$
4.  $Z = (F_k \otimes A), C_{k,:}^A = (Z^+ \text{vec} X_k)'$
5. The same procedure follows to estimate  $B$  and  $C_{k,:}^B$  with the pertinent rearranges
6.  $Z = [B D_1^B \otimes A D_1^A \quad B D_2^B \otimes A D_2^A \quad \dots \quad B D_K^B \otimes A D_K^A]'$
7.  $\text{vec} H = Z^+ \text{vec} X$  where  $\text{vec} X = [\text{vec} X_1 \quad \text{vec} X_2 \quad \dots \quad \text{vec} X_k]'$
8. Repeat from step 2 until convergence.

This algorithm can be also extended for models of higher orders [12], but it has its complications. As for other algorithms, the initialization of the parameters can be done randomly, or with the loading vectors of an unfolded PCA in a way that is described in [12]. In the case of matrix, if  $R = S$  it can be defined as the identity matrix.

For fitting a PARALIND model an algorithm has been also proposed in the referenced papers [52], [53] but only for the models used in a specific type of data. This lack of generalization can be taken as a limitation, as for any change in the model the algorithm should be rewritten. Nevertheless, it can be seen that is much simpler than the PARATUCK2-ALS algorithm. The following algorithm is related specifically for FIA data analysis's PARALIND model, considering the linear dependency in only one mode, of left-dimension (column-dimension of  $A$ )  $R$  and right dimension (column dimension of  $B$  and  $C$ )  $S$ . This is very similar to the PARAFAC-ALS algorithm; except for the necessary step for updating the interaction matrix  $H$ . So the loss function would be as follows, using the matricized representation of the model:

$$\min_{A,H,B,C} \|X - AH(C \square B)\|_F^2 \quad (58)$$

and the algorithm:

1. Initialize  $A$ ,  $B$ ,  $C$  and  $H$  (if it is previously fixed)
2.  $vecH = [(B'B) * (C'C) \otimes (A'A)]^{-1} vec\left(\sum_{k=1}^K A' X_k B D_k\right)$  (if  $H$  is not fixed)
3.  $A = \left(\sum_{k=1}^K X_k B D_k H'\right) \left\{H [(B'B) * (C'C)] H'\right\}^{-1}$
4.  $B = \left(\sum_{k=1}^K X_k' A H D_k\right) \left\{(H'A'AH) * (C'C)\right\}^{-1}$
5.  $diagD_k = \{(B'B)(H'A'AH)\}^{-1} diag(AH'X_kB), k = 1, \dots, K$
6. Repeat from step 2 until convergence.

where the operator  $diag(X)$  is a column vector with the diagonal elements of  $X$  and the operator  $vec$  is the vector obtained by stringing out  $X$  column-wise to a column vector. This algorithm can be extended to higher order arrays and where there are linear dependencies in more than one mode, but not so easily. In [52], [53] some ways to integrate constraints in the algorithm can also be found.

#### 4.1.5. TUCKALS3

As for PARAFAC model, various algorithms were proposed to fit the Tucker models, but the most important one is TUCKALS3, based on alternating least squares with orthogonal loading vectors [65], which will be shown hereafter based on the loss function:

$$\min_{A,B,C} \|(I - AA')X(C \otimes B)\|_F^2 \quad (59)$$

The steps of the algorithm taken from [Bro] are:

1. Initialize  $B$  and  $C$
2.  $A$  equals first  $P$  left singular vectors of  $X_{(I \times JK)}(C \otimes B)$
3.  $B$  equals first  $Q$  left singular vectors of  $X_{(J \times IK)}(C \otimes A)$
4.  $C$  equals first  $R$  left singular vectors of  $X_{(K \times IJ)}(B \otimes A)$
5. Repeat from step 2 until convergence
6.  $G = A' X (C \otimes B)$

To initialize  $B$  and  $C$  it is recommended to take the left singular vectors of the singular value decomposition of the correspondent matricized version of  $X$ . For the stopping criterion can be used the change in the fit between one iteration and another, as is expressed in the previous algorithm, and fit can be also obtained of the squared elements of  $G$ , so in the implementations step 5 and 6 can be exchanged without any problem.

The TUCKALS2 [89], [90] algorithm is similar to the one proposed for Tucker3, with the difference that the loading matrix of the mode that will not be compress is substitute by the identity matrix  $I$ . As the Tucker1 model matches with a bilinear model, the algorithms used for these models e.g. SVD can be also applied for this model.

#### 4.2. Tri-PLS algorithms

The N-PLS algorithm [18] is one of the multi-way models that can exceptionally be fitted sequentially. This is carried from the traditional PLS model and has the advantage that a model with higher number of components can be calculated from a previous solution with less components and backwards the job is already done. So, originating from the traditional PLS algorithm called bi-PLS, the three-way algorithm is named tri-PLS.

As the objective of the N-PLS model is to obtain the scores matrices such that the covariance between the independent and dependent variables is the maximum, the loss function will be based on this. In the following equation the optimization criterion will be expressed for a case where there is more than one dependent variable  $Y(I \times J)$ :

$$\max_{w^J, w^K} \left[ \text{cov}(t, Y) \mid t = X(w^K \otimes w^J); \|w^J\| = \|w^K\| = 1 \right] \quad (61)$$

We will present here the corresponding Tri-PLS2 [18] algorithm which is a combination of the tri-linear tri-PLS1 [3], [12], [18] (for one  $y$ ) and the traditional PLS2 algorithm, where all  $y_s$  can be modeled simultaneously:

1. Initialize  $u_1$  with a column in  $Y(I \times J)$  and  $f = 1$
2.  $z_{jk} = \sum_{i=1}^I u_f^i x^{ijk}$

3. Determine  $w_f^J$  and  $w_f^K$  as the first left and right singular vectors of  $Z$ .
4. Calculate  $t_f = X(w_f^K \otimes w_f^J)$
5.  $q_f = \frac{Y't_f}{\|Y't_f\|}$
6.  $u_f = Yq_f$
7. If  $u$  has converged, continue, else step1.
8.  $b_f = (T'T)^{-1}T'u_f$
9.  $X_i - t_{if}w_f^J(w_f^K)'$  and  $Y = Y - Tb_fq_f'$
10.  $f = f + 1$ . Continue from step 1 until proper description of  $Y$

This algorithm is equivalent to tri-PLS1 if only one dependent variable is available. Some improvements have been proposed for this algorithm in a way described in [thesis bro], but would lead to other complications as slowness. The N-PLS algorithm can be also extended to higher order arrays and also for higher order dependent variables arrays, in a similar way. More details about this can be found in [12].

### 4.3. Missing data

It is rare in any empirical study not to have at least a few data points for which the values are unknown. There can be a variety of causes for such missing data, such as accidents of some sort, flaws in the data collection, the impossibility to obtain measurements because of a too low signal, or by design. There may be no pattern to the missing data, so that they are scattered through the entire data array, or they may show specific patterns because certain data could not be collected, were no longer collected, or part of the data collection started later than that of the main body of the data.

The origin of the missing data can influence the way they are handled in an analysis, and the way the results of the analyses are interpreted. Missing data may also be produced on purpose, in order to perform analyses of data sets in which outlying data values have been identified or are suspected. In such situations the outliers can be designated as missing and analyses with and without these outliers can be performed to assess their influence on the solution. By treating the outliers as missing, it is possible to establish whether they are responsible for the three-way structure or interaction. If so, the complexity of the analysis may be reduced so that there may no longer be a need for a multi-way analysis of the data.

One of the interesting aspects of analyzing multi-way data in the presence of missing values is that due to the presence of more ways, the data are much richer than in the two-way case. The additional information makes the analyses easier and often more reliable, even when a considerable amount of data are missing. However, there must be sufficient information at the right places in the data set to allow for estimation of the missing data [5]. The treatment of missing data in any model should be very carefully, as this kind of information can't be compensated. There are a lot of people that sets up the missing values to



zero as the solution to this, but it can often give incorrect results, so this means that these values have no influence on fitting the model and this is not correct in a modeling situation.

Another approach is to impute the missing elements from neighboring elements in case that these elements are correlated. This approach can provide reasonable results if only a few elements are missing, but it does not work correctly otherwise.

In methods like two-way PCA and two-way and multi-way PLS regression fit through NIPALS-like algorithms, the approach is to simply skip the missing elements in the appropriate inner products of the algorithm [91]. This has the same problem as the last approach mentioned, and it can even be suboptimal in the cases that it may work [92].

There are some other approaches to treat missing data mentioned in [5], most of them applied to the PARAFAC model. In [3] two more approaches are also explained, imputation and weighted regression; this last can work for all models that have a well-defined overall optimization criterion, as PARAFAC and Tucker. Although Kiers has shown in [93] that both approaches give identical results, it is difficult to say which one is better, as it depends on the implementation, size of the data, and size of the computer memory as both can be computationally costly, though data imputation is easier to implement.

#### 4.4. Outlier detection

The outliers are observations that differ from the rest of the other samples of the model and their presence in the data, in particular, outlier interactions among the modes of three-way array, affects the model most of the time (see [5] for types of unusual points). Consequently, this type of points must be detected, and removed or ignored when creating the models. There are two approaches to analyze them: their treatment before the analysis of the data by the analysis of residuals plot or the use of robust methods that could be able to estimate model's parameters correctly notwithstanding these points.

In the case of multi-way models, most of which are based on least squares procedures, carry its problem of being too sensitive to outliers. Another problem is that multi-way data are usually preprocessed before they are analyzed, so if there are outliers the analysis could be defective before it started. So there is also the need of robust methods for preprocessing.

There are some facts to be taken into account in the detection of unusual points in multi-way data, and it is that as it can be composed by combinations of subjects, variables, conditions, etc there can be outlying points in any of the modes. If one restricts oneself to subjects, there are two possibilities corresponding to the two common data arrangements: to detect outlying slices and to detect outlying fibers. In the first one the data are viewed according to the  $I$  by  $J \times K$  arrangement (a wide combination-mode matrix). In some robust procedures the robust estimates for the parameters are based on a subset of slices so as to eliminate the influence of outlying slices on the parameter estimates. The second arrangement would be in the way  $I \times K$  by  $J$  (a tall combination-mode matrix) with the robust estimates based on a subset of the  $I \times K$  fibers (i.e., rows). For the last case, there may be encountered some imbalance as some subjects have no observations in certain conditions. This can only be handled in a multi-way analysis by designating these fibers as missing in order to make the data set a "complete" multi-way array for analysis [5].

Some standard procedures for the analysis of residuals exist and some robust methods for multi-way analysis, including robust preprocessing methods. Most of these robust approaches are variants of strengthened principal component analysis extended to multi-way analysis (see

Robust statistical procedures. [94], A fast method for robust principal components with applications to chemometrics [95], ROBPCA: A new approach to robust principal component analysis [96], Robustness and outlier detection in chemometrics [97], A comparison of three procedures for robust PCA in high dimensions [98].

Among the works made on robust versions of multi-way component methods to treat outlying points are: Algorithms for robust PCA and applications for multi-way data [99] Robust PARAFAC for fluorescence data [100], A fully robust PARAFAC method for analyzing fluorescence data [101], A robust PARAFAC method [102], A robust version of the Tucker3 model [103].

We can then conclude that it is very important the detection of outliers because of their direct negative influence on models, making their results invalid. Although they can be analyzed by the residuals, it would be very important to work more on the development and improvement of multi-way robust methods for this purpose, as it is incipient yet.

## 5. Applications

After the main things of multi-way analysis have been explained, in this chapter we are planning to show how it is applied in many research areas. At the beginning of the history of multi-way analysis, the principal applications and contributions were made by psychometricians [1], [15], but as the years go by and the specialist earn new technologies that supplies them of complicated but useful amount of data, they have learned to model their data with this structure and take advantage of all the kindness and information that the developed methods offer.

Nowadays there are a lot of fields where multi-way analysis is used for exploratory analysis, classification and regression purposes. In the following subchapters we will try to explain how some of them make use of multi-way analysis.

### 5.1. Chemistry

One of the research areas where this technique is very popular is in chemometrics. As most of the new instrumental equipments are now combined, the output data has a natural three-way structure e.g. 1) Liquid chromatography with ultraviolet (UV) detection (HPLC-DAD) for different samples (*Samples x Time x Wavelengths*), 2) Gas Chromatography-Mass Spectrometry (GC-MS) for different samples (*Samples x Time x  $m/z$* ) and 3) Fluorescence spectra for different samples (Excitation-Emission wavelengths matrix per sample) (EEM) (*Samples x Exc Wavelengths x Emis Wavelengths*).

The objective of using the multi-way component methods in this data are mainly for exploratory analysis and calibration, to determine specific interrelations between the components, determination of the compounds found in each sample as well as the relative concentrations of compounds, etc. The application of PARAFAC to the Fluorescence data [104], [105], [106], [107], [108] has been one the most popular applications of multi-way analysis as is the one that shows best the modeling power and interpretation of factors of a PARAFAC model. The applications of multi-way in chemometrics can be also found for the treatment of environmental data, which has a lot more influence from outside, introducing noise and undesirable information into the data. In this data the presented models have also

shown very good results for exploratory analysis and calibration [109], [110], [111], [112], [113]. For classification purposes there have been also some applications in chemometrics. One example we bring here is with a GC-MS data of weathered petroleum oils [114], but in this case the component model that have give good results is the PARAFAC2 model, as this type of data shows the limitation of PARAFAC models with time shifts, which is very common in chromatography. In fluorescence data there was also a big study in food classification [115], showing once again the capability of the PARAFAC model for this type of data.

Batch processes are another important area where multi-way analysis has a wide application. In this analysis it is obtained an array of batches followed over time and measured by temperature or pressure sensors, etc., (*Batch x Time Wavelengths*). These processes sometimes are very difficult and time consuming, as to obtain the whole data set the specialist has to wait for the completion of the whole batch. Although this type of data can have problems with missing values, PARAFAC have shown it can beat this problem and model it with success [116]. These datasets can also have some problems with the reduction of the data in the time mode, so the data will have different dimensions. PARAFAC cannot deal with this, but a three-way analysis method called STATIS [117] can do it, as unlike the other three-way models, it explores each mode separately. Also in [118], a new approach for handling this problem with PARAFAC2 models was introduced. An extensive review of application of multi-way data is related in [119].

Details on how the multi-way data of these and other examples would be arranged can be found in [12], [3].

## 5.2. Sociology

Applications of multi-way analysis can be also found in sociology, to evaluate human behaviors. Some works have been done to find patterns in the use of time between the populations of different countries [120], having arranged the three-way data with the variables: (*countries x activities x category of populations*). There has been also a special interest in finding hidden structures in social networks e.g. patterns in relationships, communication. For this, investigations from chat rooms [121], have been done, being the three-way data of the form (*users x keywords x time*). Also for the investigations on communication has been used the information from the emails [122].

## 5.3. Neurology

Another of the most common applications of multi-way analysis has been in neurology, where the electroencephalographic data (EEG) has been modeled as a three-way array [123], [124] but with different arrangements in dependence of the investigation purpose.

There have been studies of epilepsy where the three-way data consist of wavelet coefficient at  $i$ th time sample,  $j$ th scale and  $k$ th electrode [125]. Another arrangement of multi-way data that have become very popular in neuroscience is in the following form: (*space x time x frequency*), which has been applied to solve important problems e.g. Brain computer interfacing (BCI) based on EEG activities which enable people suffering severe neurological disabilities but cognitively intact to operate computers by intention rather than

by physical contact. In this example PARAFAC has demonstrated its capability of successfully space-time-frequency decomposition of the EEG for BCI and annihilation of the brain background potentials [126]. Recently, for the analysis of brain activities, a toolbox for multi-channel time-frequency analysis using EEG and MEG data (ERPWAVELAB) was introduced by [127].

#### 5.4. Classification

The previous examples of applications of multi-way analysis methods are only a few of a wide variety that exist in a lot of fields. During our investigation we could notice that, most of the applications are for exploratory and regression purposes but for classification there are not so many. This is not because the problems are not there, classification problems are very common in any research area, it is maybe because there was not a classification tool yet and the exploratory tools sometimes are not enough.

Nowadays, the classification tasks for “profile data” design are being solved by unfolding the data and using traditional two-way classification methods, which has been shown here to have a lot of disadvantages, or by using PARAFAC or Tucker models and later analyzing the scores obtained in the decompositions [115], [128], which evidently have interesting characteristics that made them very popular among the researchers. This last is better than the unfolding alternative, but still carries the problems outlined in the previous sections about these models. Other times the classification task is only accomplished by the simple exploration with these methods [114].

In the last years a few models have been proposed for this purpose, not for the “profile data” type which concerns us, but for the “three-way binary data” type explained in [5]. This type of data  $X(I(\text{objects}) \times J(\text{variables}) \times K(\text{conditions}))$  has the same structure of the first one, but as its name indicates the values of the array are binary. So, the models defined for working with this data are analogous to PARAFAC and Tucker [129], [130], [131], [132], but based on Boolean algebra; the entries of the component matrices are also binary.

Recently, David Ballabio introduced the use of MOLMAP (Molecular Map of atom level-properties) approach [133] for the classification of multi-way data. This is an algorithm for calculating molecular descriptors for the study of molecule chemical information organized into three-way data structures. The MOLMAP algorithm is based on the use of Kohonen maps such that a representation of the objects of the original multi-way dataset can be obtained by projecting them onto the trained map, one at a time, and by mapping the  $J$  input vectors of each multi-way object, this way the similar vectors are mapped into the same or closely adjacent neurons. The final object is a score matrix  $M(I \times N * N)$ , where  $I$  represents the number of samples and  $N$  is the number of neurons that the square Kohonen map contains on each side. Finally, each sample is described by new variables that encode information both on the second and third mode of the original dataset [134]. Once this two-way matrix is obtained, any traditional two-way classifier should work. Although this technique is really new, it seems promising as good results are shown in the article, nevertheless it carries the MOLMAP disadvantages about the computational time that takes to optimize the Kohonen map.

## 6. Open problems

Multi-way data analysis is not a new technique, as can be noticed in this study and the reported literature. Nevertheless, as we were writing the document we tried to outline some limitations of what already exists in this field and might be seen as open problems that still have to be worked out:

1. Although a sequence of works have been published about the uniqueness conditions for PARAFAC models, they are not complete; most of them are for particular cases of multi-way arrays. So, many things can still be done in this issue.
2. The degeneracy problems of the PARAFAC-ALS algorithm are another important aspect should be treated. Some authors have been working on how to avoid this inconvenience, but some effort could be done to develop an algorithm with similar qualities to ALS's and fixing its degeneracy and maybe slowness problem.
3. The uniqueness in PARAFAC2 model has not been threshed as PARAFAC. It is a method that has very good qualities and has been proven to be sometimes even more convenient than PARAFAC, but only a few works were done on this issue.
4. As could be seen, other methods as S-PARAFAC and cPARAFAC has been developed to affront problems that can be found in specific types of multi-way data, but these are rather incipient methods yet. A lot of work can be done in refining them, as for example how to include multiple-shifting in S-PARAFAC models. The same applies for the Tucker versions of these models.
5. During our investigation, although we didn't deepen into all the existing methods of multi-way analysis for all types of multi-way data, we explored most of them and the ones are not explained here are in some way reported in the bibliography. We have noticed that, very poor work has been done for classification in multi-way data analysis, and is even worst for the profile data type, which is now the kind of multi-way data that interests us most. Besides the use of the existing exploratory analysis tools and other ways to affront this problem in some way, only one method was found [buscar molmap] specially for this purpose, so we think that of all the open problems we might encountered in multi-way analysis, this is the less attended until now by researchers and we already have some proposals for it.

### 6.1. Dissimilarity Representation

Recently for two-way data, a new type of representation was introduced by Pekalska et al. [135] as a new approach that links the structural and statistical approaches, and is known as Dissimilarity Representation (DR). This representation was mainly thought for classification; therefore it is based on the idea that, as classes are conformed by a set of similar objects, the proximity between them plays an important role in this type of problems. The DR consists basically in the representation of the objects by its dissimilarities with respect to the other objects, and the classifiers may be built on the dissimilarity space generated by a representation set, or in a feature space where the dissimilarity data is isometrically embedded. This type of representation can have several applications but the main researches have been done on classification [136], [137].

As there is not a general dissimilarity measure for all types of data, the dissimilarity measure selected allows emphasizing the information that would be very useful to classify a particular type of data, so the first thing here is to select a suitable dissimilarity measure for the problem at hand, a measure that is appropriate for comparing objects given the known data characteristics. Next, a representation set  $R(p_1, p_2, \dots, p_n)$  has to be selected, which is a set of representative objects from the classes, called prototypes.

The DR for a set of objects  $X$  is the matrix  $D(X, R)$  formed by the dissimilarities between each object  $x \in X$  and the objects of  $R$ . Each element of the matrix would be then a dissimilarity value between two objects,  $d(x_m, p_n)$ .  $R$  can be a subset of  $X$ , ( $R \subseteq X$ ) or  $X$  itself, being then  $D(X, X)$  a square dissimilarity matrix, or  $X$  and  $R$  can be completely different sets. There are many approaches to select the prototypes of the representation set [138].

In DR there are three main approaches. The first one addresses the given dissimilarities directly. There is also the one based on an approximate embedding of the dissimilarities into a (pseudo-)Euclidean space and at last, the so-called dissimilarity space approach which is the one proposed to be used here. The dissimilarity space  $\mathcal{D} \subseteq \mathbb{R}^n$  is generated by the column vectors of the dissimilarity matrix, where each dimension corresponds to the dissimilarity between the objects and a prototype  $d(\cdot, p_n)$ . As the dissimilarities are computed to the representation set, it constitutes already a dimensionality reduction and therefore it can be less computationally expensive.

The use of DR is especially advantageous with: 1) spectral data, 2) when vector representations of objects live in a high dimensional space and 3) when the number of examples is very small, and these last two elements turn out to be the major problems when working with spectroscopic chemical data. Some researches have been done about the proper dissimilarity measures for spectral data [139].

Another important advantage of this representation is that any traditional classifier that works on feature spaces can be also used on the dissimilarity space. This approach could be also extended to multi-way analysis if the appropriate way to represent the relations between the modes and characteristics of the data could be included in the dissimilarity measure.

## 7. Conclusions

In this work we have presented a general panoramic of multi-way data analysis, making emphasis in some essential issues to be taken into account. Although multi-way analysis is not a new technique, as most of the methods were developed some years ago, it has had its big deploy in applications in the last years. We did not analyze here all the multi-way methods, because as reported in Kroonenberg's book, there is a wide relation of multi-way methods for different types of multi-way data, though, there are not reported methods for classification. We centered, among other important aspects of multi-way analysis, in explaining the models and algorithms of the most popular multi-way methods along the years: PARAFAC, Tucker and NPLS, outlining their advantages and disadvantages. During this work we have noticed that researchers have focused their attention on improving the

existing multi-way methods for special problems, which are mainly for exploratory analysis and regression tasks, but have left behind other important issues as classification problems.

We can conclude that multi-way analysis is having more applications to a sped up step and thus, has been forcing the researchers to improve and create more methods to analyze them, but it is still not enough. Methods for resolving classification problems with multi-way data, which are very important and frequent in any research area, have not been minded, and then only a few methods have been proposed. So, referring to this aspect and other important issues of multi-way analysis that have been outlined in this work, tell us that still much to do in what multi-way analysis refers.

## References

1. Cattell, R.B.: The three basic factor-analytic research designs- their interrelations and derivatives. *Psychological Bulletin* 49 (1952) 499–452
2. Tucker, L.: The extension of factor analysis to three-dimensional matrices. Holt, Rinehart & Winston, New York (1964)
3. Smilde, A.K., Bro, R., Geladi, P.: *Multi-way Analysis. Applications in the chemical sciences.* John Wiley & Sons Ltd, West Sussex, England (2004)
4. Acar, E., Yener, B.: Unsupervised Multiway Data Analysis: A Literature Survey. *Knowledge and Data Engineering, IEEE Transactions on* 21 (2009) 6-20
5. Kroonenberg, P.M.: *Applied multiway data analysis.* John Wiley & Sons, Inc, Hoboken, New Jersey (2008)
6. Kiers, H.A.L.: Towards a standardized notation and terminology in multiway analysis. *J. Chemometrics* 14 (2000) 105-122
7. Harshman, R.A., Lundy, M.E.: Data Preprocessing and the extended PARAFAC model. In: HG Law, CW Snyder, J.J.H., & RP McDonald (ed.): *Research methods for multimode data analysis*, Vol. 6, New York (1984) 216-284
8. Bro, R., Smilde, A.K.: Centering and scaling in component analysis. *J.Chemometrics* 17 (2003) 16-33
9. ten-Berge, J.: Convergence of PARAFAC preprocessing procedures and the Deming-Stephan method of iterative proportional fitting. In: Coppi R. , B.S. (ed.): *Multiway Data Analysis.* Elsevier, Amsterdam (1989) 53-63
10. Golub, G.H., Loan, C.F.V.: *Matrix Computations.* The Johns Hopkins University Press, Baltimore,MD (1996)
11. Golub, G.H., Loan, C.F.V.: *Matrix Computations.* The Johns Hopkins University Press, Baltimore,MD (1989)
12. Bro, R.: *Multi-way analysis in the food industry. Models, algorithms, and applications.* Department of Dairy and Food Science, Vol. PhD Thesis. Royal Veterinary and Agricultural University, Denmark (1998)
13. Cattell, R.B.: Parallel proportional profiles and other principles for determining the choice of factors by rotation. *Psychometrika* 9 (1944) 267-283
14. Tucker, L.: Implications of factor analysis of three-way matrices for measurement of change. In: Harris, C.W. (ed.): *Problems in measuring change.* University of Wisconsin Press, Madison (1963) 122-137
15. Tucker, L.: Some mathematical notes in three-factor analysis. *Psychometrika* 31 (1966) 279-311

16. Harshman, R.A.: Foundations of the Parafac procedure: models and conditions for an explanation multi-modal factor analysis. UCLA Working Papers in Phonetics, Vol. 16, Los Angeles, CA (1970) 1-84
17. Carroll, J., Chang, J.J.: Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika* 35 (1970)
18. Bro, R.: Multiway calibration. Multi-linear PLS. *J.Chemometrics* 10 (1996) 47-61
19. Carroll, J., Chang, J.J.: IDIOSCAL: A generalization of INDSCAL allowing IDIOSyncratic reference system as well as an analytic approximation to INDSCAL. Paper presented at the Spring Meeting of the Classification Society of North America, Princeton, NJ (1972)
20. Bro, R.: PARAFAC. Tutorial and Applications. *Chemometrics and Intelligent Laboratory Systems* 38 (1997) 149-171
21. Harshman, R.A., Lundy, M.E.: The PARAFAC model for three-way factor analysis and multidimensional scaling. In: H. G. Law, C.W.S., Jr., J. Hattie, & R. P. McDonald (ed.): *Research Methods for Multimode Data Analysis*. Praeger, New York (1984) 216-284
22. Harshman, R.A., Lundy, M.E.: PARAFAC: Parallel factor analysis. *Computational Statistics & Data Analysis* 18 (1994) 39-72
23. Kruskal, J.: Rank, decomposition, and uniqueness for 3-way and N-way arrays. In: Coppi R. , B.S. (ed.): *Multiway Data Analysis*. Elsevier, Amsterdam (1989) 8-18
24. Rao, C., Mitra, S.: *Generalized Inverse Inference and its Applications*. John Wiley & Sons, Inc, New York (1971)
25. Schott, J.: *Matrix analysis for statistics*. John Wiley & Sons, Inc., New York (1997)
26. Harshman, R.A.: Determination and Proof of Minimum Uniqueness Conditions for PARAFAC1 UCLA Working Papers in Phonetics, Vol. 22, Los Angeles, CA (1972) 11-117
27. Kruskal, J.: Three-way arrays: rank and uniqueness of trilinear decompositions with applications to arithmetic complexity and statistics. *Linear Algebra and its applications* 18 (1977) 95-138
28. Leurgans, S., Ross, R., Abel, R.: A decomposition for three-way arrays. *Siam Journal on matrix Analysis and applications* 14 (1993) 1064-1083
29. Kruskal, J.: Ranks of N-way arrays and the Geometry of  $2 \times 2 \times 2$  arrays. Technical Memorandum. AT&T Bell Laboratories, Murray Hill NJ (1985)
30. Sidiropoulos, N.D., Bro, R.: On the uniqueness of multilinear decomposition of N-way arrays. *J. Chemometrics* 14 (2000) 229-239
31. Stegeman, A., Sidiropoulos, N.D.: On Kruskal's uniqueness condition for the Candecomp/Parafac decomposition. *Linear Algebra and its Applications* 420 (2007) 540-552
32. Kruskal, J.: More factors than subjects, tests and treatments: an indeterminacy theorem for canonical decomposition and individual differences scaling. *Psychometrika* 41 (1976) 281-293
33. ten-Berge, J.: On uniqueness in CANDECOMP/PARAFAC. *Psychometrika* 67 (2002)
34. Stegeman, A., ten-Berge, J.: Kruskal's condition for uniqueness in Candecomp/Parafac when ranks and k-ranks coincide. *Computational Statistics & Data Analysis* 50 (2006) 210-220
35. ten-Berge, J.: Partial uniqueness in CANDECOMP/PARAFAC. *J.Chemometrics* 18 (2004) 12-16
36. Mitchell, B., Burdick, D.: Slowly convergence PARAFAC sequences: Swamps and two-factor degeneracies. *J.Chemometrics* 8 (1994) 155-168
37. Krignen, W.P., Kroonenberg, P.M.: *Degeneracy and Parafac* Department of Psychology, University of Groningen (2000)
38. Paatero, P.: Construction and analysis of degenerate PARAFAC models. *J. Chemometrics* 14 (2000) 285-299
39. Stegeman, A.: Degeneracy in Candecomp/Parafac explained for  $p \times p \times 2$  arrays of rank  $p + 1$  or higher. *Psychometrika* 71 (2006) 483-501



40. Stegeman, A.: Degeneracy in Candecomp/Parafac and Indscal explained for several three-sliced arrays with a two-valued typical rank. *Psychometrika* 72 (2007) 601-619
41. Krijnen, W.P., Dijkstra, T.K., Stegeman, A.: On the non-existence of optimal solutions and the occurrence of "degeneracy" in the Candecomp/Parafac model. *Psychometrika* 73 (2008)
42. Stegeman, A., Lathauwer, L.D.: A method to avoid diverging components in the Candecomp/Parafac model for generic  $I \times J \times 2$  arrays. *SIAM Journal on Matrix Analysis and Applications* 30 (2009) 1614-1638
43. Harshman, R.A.: PARAFAC2: Mathematical and technical notes. *UCLA Working Papers in Phonetics*, Vol. 22, Michigan (1972) 30-44
44. Kiers, H.A.L.: PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model. *J.Chemometrics* 13 (1999) 275-294
45. Harshman, R.A., Hong, S., Lundy, M.E.: Shifted factor analysis-Part I: Models and properties. *J.Chemometrics* 17 (2003) 363-378
46. Hong, S., Harshman, R.A.: Shifted factor analysis-Part III: N-way generalization and application. *J.Chemometrics* 17 (2003) 389-399
47. Morup, M., Schmidt, M.N.: Sparse Non-Negative Tensor Factor Double Deconvolution (SNTF2D) for multi-channel time frequency analysis. (2006)
48. Morup, M., Schmidt, M.N.: Sparse Non-Negative Matrix Factor 2-D Deconvolution Technical Report, Insitute for Mathematical Modelling. Technical Univerisity of Denmark (2006)
49. Morup, M., Hansen, L.K., Parnas, J., Arnfred, S.M.: Decomposing the time frequency representation of EEG using non-negative matrix and multi-way factorization. Technical Report (2006)
50. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research* 4 (2004)
51. Harshman, R.A., Lundy, M.E.: Uniqueness proof for a family of models sharing features of Tucker's three-mode factor analysis and PARAFAC/CANDECOMP. *Psychometrika* 61 (1996) 133-154
52. Bro, R., Harshman, R.A., Sidiropoulos, N.D.: Rank-deficient models for multi-way data. Part I: Theory. *J. Chemometrics* (2001)
53. Bro, R., Harshman, R.A., Sidiropoulos, N.D.: Modeling multi-way data with linearly dependent loadings. *Chemometrics Group, Department of Food Science, Royal Veterinary & Agricultural University Denmark* (2005)
54. Reis, M.M., Ferreira, M.M.C.: PARAFAC with splines: a case study. *J. Chemometrics* 16 (2002) 444-450
55. Jansen, J.J., Bro, R., Hoefsloot, H.C.J., Berg, F.W.J.v.d., Westerhuis, J.A., Smilde, A.K.: PARAFASCA: ASCA combined with PARAFAC for the analysis of metabolic fingerprinting data. *J. Chemometrics* 22 (2008) 114-121
56. ten-Berge, J., Smilde, A.K.: Non-triviality and identification of a constrained Tucker3 analysis. *J.Chemometrics* 16 (2002) 609-612
57. ten-Berge, J.: Simplicity and typical rank of three-way arrays, with applications to Tucker-3 analysis simple cores. *J.Chemometrics* 18 (2004) 17-21
58. Morup, M., Hansen, L.K., Arnfred, S.M.: Algorithms for sparse non-negative Tucker. (2007)
59. Smilde, A.K., Kiers, H.A.L.: Multiway covariates regression models. *J.Chemometrics* 13 (1999) 31-48
60. Bro, R., Smilde, A.K., Jong, S.d.: On the difference between low-rank and subspace approximation: improved model for multi-linear PLS regression. *Chemometrics and Intelligent Laboratory Systems* 58 (2001) 3-13
61. Timmerman, M.E., Kiers, H.A.L.: Three mode principal components analysis: Choosing the numbers of components and sensitivity to local optima. *British Journal of Mathematical and Statistical Psychology* 53 (2000) 1-16

62. Kiers, H.A.L., Kindersen, A.D.: A fast method for choosing the numbers of components in Tucker3 analysis. *British Journal of Mathematical and Statistical Psychology* 56 (2003) 119-125
63. Zeng-Ping, C., Zhuo, L., Yu-Zhen, C., Ru-Qin, Y.: Efficient way to estimate the optimum number of factors for trilinear decomposition. *Analytica Chimica Acta* 444 (2001) 295-307
64. Bro, R., Kiers, H.A.L.: A new efficient method for determining the number of components in PARAFAC models. *J.Chemometrics* 17 (2003) 274-286
65. Kroonenberg, P.M., Leeuw, J.d.: Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika* 45 (1980) 69-97
66. Jiang, J., Hu, H.L., Li, Y., Yu, R.Q.: Three-way data resolution by alternating slice-wise diagonalization (ASD) method. *J.Chemometrics* 14 (2000) 15-36
67. Wu, H.L., Shibaukawa, M., Orguna, K.: An alternating trilinear decomposition algorithm with application to calibration of HPLC-DAD for simultaneous determination of overlapped chlorinated aromatic hydrocarbons. *J.Chemometrics* 12 (1998) 1-26
68. Chen, Z.P., Wu, H.L., Jiang, J.H., Li, Y., Yu, R.Q.: A novel trilinear decomposition algorithm for second-order linear calibration. *Chemometrics and Intelligent Laboratory Systems* 52 (2000) 75-86
69. Chen, Z.P., Li, Y., Yu, R.Q.: Pseudo alternating least squares algorithm for trilinear decomposition. *J.Chemometrics* 15 (2001) 149-167
70. Jiang, J.H., Wu, H.L., Li, Y., Yu, R.Q.: Alternating coupled vectors resolution (ACOVER) method for trilinear analysis of three-way data. *J.Chemometrics* 13 (1999) 557-578
71. Li, Y., Jiang, J.H., Wu, H.L., Chen, Z.P., Yu, R.Q.: Alternating coupled matrices resolution method for three-way arrays analysis. *Chemometrics and Intelligent Laboratory Systems* 52 (2000) 33-43
72. Paatero, P.: A weighted non-negative least squares algorithm for three-way 'PARAFAC' factor analysis. *Chemometrics and Intelligent Laboratory Systems* 38 (1997) 223-242
73. Tomassi, G., Bro, R.: PARAFAC and missing values. *Chemometrics and Intelligent Laboratory Systems* 75 (2005) 163-180
74. Sanchez, E., Kowalski, B.R.: Generalized rank annihilation factor-analysis. *Anal. Chem.* 58 (1986) 496-499
75. Sanchez, E., Kowalski, B.R.: Tensorial resolution: a direct trilinear decomposition. *J.Chemometrics* 4 (1990)
76. Chen, Z.P., Wu, H.L., Li, Y., Yu, R.Q.: Novel constrained PARAFAC algorithm for second-order linear calibration. *Analytica Chimica Acta* 423 (2000) 187-196
77. Faber, N.M., Bro, R., Hopke, P.k.: Recent developments in CANDECOMP/PARAFAC algorithms: a critical review. *Chemometrics and Intelligent Laboratory Systems* 65 (2003) 119-137
78. Tomasi, G., Bro, R.: A comparison of algorithms for fitting the PARAFAC model. *Computational Statistics & Data Analysis* 50 (2006) 1700-1734
79. Cao, Y.Z., Chen, Z.P., Mo, C.Y., Wu, H.L., Yu, R.Q.: A PARAFAC algorithm using penalty diagonalization error (PDE) for three-way data array resolution. *Analyst* 125 (2000) 2303-2310
80. Wetzell, P., Andrews, D., Hamilton, D., Faber, K., Kowalski, B.: Maximum likelihood principal component analysis. *J.Chemometrics* 11 (1997) 339-366
81. Vega-Montoto, L., Wentzell, P.: Maximum likelihood parallel factor analysis (MLPARAFAC). *J.Chemometrics* 17 (2003) 237-253
82. Vega-Montoto, L., Gu, H., Wentzell, P.: Mathematical improvements to maximum likelihood parallel factor analysis: theory and simulations. *J.Chemometrics* 19 (2005) 216-235
83. Vega-Montoto, L., Wentzell, P.: Mathematical improvements to maximum likelihood parallel factor analysis: experimental studies. *J.Chemometrics* 19 (2005) 236-252

84. Vega-Montoto, L.: Maximum likelihood methods for three-way analysis in chemistry. Department of Chemistry. Dalhousie University, Halifax, Nova Scotia (2005)
85. Cramer, J., Yoon-Chang, K., Vogt, F., Booksh, K.: Accelerating the analyses of 3-way and 4-way PARAFAC models utilizing multi-dimensional wavelet compression. *J.Chemometrics* 19 (2006) 593-606
86. Kiers, H.A.L.: An alternating least squares algorithm for PARAFAC2 and three-way DEDICOM. *Computational Statistics & Data Analysis* 16 (1993)
87. Smaragdis, P.: Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. *International Symposium on Independent Component Analysis and Blind Source Separation (ICA)* (2004)
88. Egert, J., Korner, E.: Sparse coding and NMF. *Neural Networks, Vol. 4* (2004) 2529-2533
89. Kroonenberg, P.M., Leeuw, J.d.: TUCKALS2. A principal component analysis of three-mode data. *Research Bulletin. Dept. Data Theory, Leiden University* (1977)
90. Kroonenberg, P.M.: The TUCKALS line. A suite of programs for three-way data analysis. *Computational Statistics & Data Analysis* 18 (1994) 73-96
91. Martens, H., Naes, T.: *Multivariate calibration*. John Wiley & Sons, Ltd, Chichester (1989)
92. Grung, B., Manne, R.: Missing values in principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 42 (1998) 125-139
93. Kiers, H.A.L.: Weighted least squares fitting using ordinary least squares algorithms. *Psychometrika* 62 (1997) 251-266
94. Huber, P.J.: *Robust statistical procedures*. PA: SIAM, Philadelphia (1996)
95. Hubert, M., Rousseeuw, P.J., Verboven, S.: A fast method for robust principal components with applications to chemometrics. *Chemometrics and Intelligent Laboratory Systems* 60 (2002) 101-111
96. Hubert, M.: ROBPCA: A new approach to robust principal component analysis. *Technometrics* 47 (2005) 64-79
97. Debruyne, M., Engelen, S., Hubert, M., Rousseeuw, J.: Robustness and outlier detection in chemometrics. *Critical Reviews in Analytical Chemistry* 36 (2006) 221-242
98. Engelen, S., Hubert, M., Branden, K.V.: A comparison of three procedures for robust PCA in high dimensions. *Austrian Journal of Statistics* 34 (2005) 117-126
99. Engelen, S.: Algorithms for robust PCA and applications for multi-way data. Vol. PhD. Department of Mathematics, Katholieke Universiteit Leuven, Leuven, Belgium (2006)
100. Engelen, S., Moller, S.F., Hubert, M.: Robust PARAFAC for fluorescence data. (2006)
101. Engelen, S., Moller, S.F., Munk, B.: A fully robust PARAFAC method for analyzing fluorescence data. *J.Chemometrics* 23 (2009) 124-131
102. Engelen, S., Hubert, M.: A robust PARAFAC method. (2005)
103. Pravdova, V., Estienne, F., Walczak, B., Massart, D.L.: A robust version of the Tucker3 model. *Chemometrics and Intelligent Laboratory Systems* 59 (2001) 75-88
104. Andersen, C.M., Bro, R.: Practical aspects of PARAFAC modeling of fluorescence excitation-emission data. *J.Chemometrics* 17 (2003) 200-215
105. Nahorniak, M.L., Cooper, G.A., Yoon-Chang, K., Booksh, K.S.: Three- and four-way parallel factor (PARAFAC) analysis of photochemically induced excitation-emission kinetic fluorescence spectra *Analyst* 130 (2005) 85-93
106. Olivieri, A.C., Arancibia, J.A., Peña, A.M.d.l., Durán-Merás, I., Mansilla, A.E.: Second-Order Advantage Achieved with Four-Way Fluorescence Excitation-Emission-Kinetic Data Processed by Parallel Factor Analysis and Trilinear Least-Squares. Determination of Methotrexate and Leucovorin in Human Urine. *Anal. Chem.* 76 (2004)
107. Christensen, J., Povlsen, V.T., Sørensen, J.: Application of Fluorescence Spectroscopy and Chemometrics in the Evaluation of Processed Cheese During Storage. *J. Dairy Sci.* 86 (2003) 1101-1107

108. Nahorniak, M.L., Booksh, K.S.: Optimizing the implementation of the PARAFAC method for near-real time calibration of excitation–emission fluorescence analysis. *J.Chemometrics* 17 (2003) 608-617
109. Goicoechea, H.C., Yu, S., Olivieri, A.C., Campiglia, A.D.: Four-Way Data Coupled to Parallel Factor Model Applied to Environmental Analysis: Determination of 2,3,7,8-Tetrachloro-dibenzo-para-dioxin in Highly Contaminated Waters by Solid-Liquid Extraction Laser-Excited Time-Resolved Shpol'skii. Spectroscopy. *Anal. Chem.* 77 (2005)
110. Stedmon, C.A., Markager, S., Bro, R.: Tracing dissolved organic matter in aquatic environments using a new approach to fluorescence spectroscopy. *Marine Chemistry* 82 (2003) 239-254
111. Obeidat, S.M., Glasser, T., Serge Y. Landau, Anderson, D.M., Rayson, G.D.: Application of multi-way data analysis on excitation–emission spectra for plant identification. *Talanta* 72 (2007) 682-690
112. Reis, M.M., Gurden, S.P., Smilde, A.K., Ferreira, M.M.C.: Calibration and detailed analysis of second-order flow injection analysis data with rank overlap. *Analytica Chimica Acta* 422 (2000) 21-36
113. Peré-Trepat, E., Ginebreda, A., Tauler, R.: Comparison of different multiway methods for the analysis of geographical metal distributions in fish, sediments and river waters in Catalonia. *Chemometrics and Intelligent Laboratory Systems* 88 (2007) 69-83
114. Ebrahimi, D., Li, J., Hibbert, D.B.: Classification of weathered petroleum oils by multi-way analysis of gas chromatography-mass spectrometry data using PARAFAC2 parallel factor analysis. *Journal of Chromatography A* 1166 (2007) 163-170
115. Sádecká, J., Tóthová, J.: Fluorescence Spectroscopy and Chemometrics in the Food Classification - a Review. *Czech J. Food Sci.* 25 (2007) 159-173
116. Bro, R., Qannari, E.M., Kiers, H.A.L., Næs, T., Frøst, M.B.: Multi-way models for sensory profiling data. *J.Chemometrics* 22 (2008) 36-45
117. Stanimirova, I., Walczak, B., Massart, D.L., Simeonov, V., Saby, C.A., Crescenzo, E.D.: Statis, a three-way method for data analysis. Application to environmental data. *Chemometrics and Intelligent Laboratory Systems* 73 (2004) 219-233
118. Skov, T., Bro, R.: A new approach for modelling sensor based data. *Sensors and Actuators B* 106 (2005) 719-729
119. Bro, R.: Review on Multiway Analysis in Chemistry—2000–2005. *Critical Reviews in Analytical Chemistry* 36 (2006) 279-293
120. Fraire, M.: Multiway data analysis for comparing time use in different countries - Application to timebudgets at different stages of life in six European countries. *electronic International Journal of Time Use Research*, Vol. 3 (2006) 88-109
121. Acar, E., Camtepe, S.A., Krishnamoorthy, M., Yener, B.: Modeling and multiway analysis of chatroom tensors. *IEEE International Conference on Intelligence and Security Informatics*. Springer (2005) 256-268
122. Bader, B.W., Harshman, R.A., Kolda, T.G.: Temporal analysis of social networks using three-way dedicom. In: SAND2006-2161, T.R. (ed.). Sandia National Laboratories (2006)
123. Estienne, F., Matthijs, N., Massart, D.L., Ricoux, P., Leibovici, D.: Multi-way modelling of high-dimensionality electroencephalographic data. *Chemometrics and Intelligent Laboratory Systems* 58 (2001) 59-72
124. Martínez-Montes, E., Valdés-Sosa, P.A., Miwakeichi, F., Goldman, R.I., Cohen, M.S.: Concurrent EEG/fMRI analysis by multiway Partial Least Squares. *NeuroImage* 22 (2004) 1023-1034
125. Acar1, E., Aykut-Bingol, C., Bingol, H., Bro, R., Yener, B.: Multiway analysis of epilepsy tensors. *Bioinformatics* 23 (2007) i10-i18
126. Nazarpour, K., Sanei, S., Shoker, L., Chambers, J.A.: Parallel space-time-frequency decomposition of EEG signals for brain computer interfacing.

127. Morup, M., Hansen, L.K., Arnfred, S.M.: ERPWAVELAB a toolbox for multi-channel analysis of time-frequency transformed event related potentials. (2006)
128. Singha, K.P., Malika, A., Basant, N., Singha, V.K., Basant, A.: Multi-way data modeling of heavy metal fractionation in sediments from Gomti River (India) *Chemometrics and Intelligent Laboratory Systems* 87 (2007) 185-193
129. Leenen, I., Mechelen, I.V., Boeck, P.D., S. Rosenberg: INDCLAS: A three-way hierarchical classes mode. *Psychometrika* 60 (1999) 9-24
130. Ceulemans, E., Mechelen, I.V., Leenen, I.: Tucker3 hierarchical classes analysis. *Psychometrika* 68 (2003) 413-433
131. Ceulemans, E., Mechelen, I.V.: Tucker2 hierarchical classes analysis. *Psychometrika* 69 (2004) 375-399
132. Ceulemans, E., Mechelen, I.V.: Hierarchical classes models for three-way threemode binary data: Interrelations and model selection *Psychometrika* 70 (2005) 461-480
133. Zhang, Q.-Y., Aires-de-Sousa, J.: Structure-based classification of chemical reactions without assignment of reaction centers. *J. Chem. Inf. Model* 45 (2005) 1775-1783
134. Ballabio, D., Consonnia, V., Todeschini, R.: Classification of multiway analytical data based on MOLMAP approach *Analytica Chimica Acta* 605 (2007) 134-146
135. Pekalska, E., Duin, R.P.W.: The Dissimilarity Representation For Pattern Recognition. *Foundations and Applications, Vol. 64* (2005)
136. Pekalska, E., Duin, R.P.W.: Classifiers for dissimilarity-based pattern recognition. *International Conference on Pattern Recognition, Vol. 2, Barcelona, Spain* (2000) 12-16
137. Pekalska, E., Duin, R.P.W.: Dissimilarity representations allow for building good classifiers. *Pattern Recognition Letters* 23 (2002) 943-956
138. Pekalska, E., Duin, R.P.W.: Prototype selection for finding efficient representations of dissimilarity data. In: R. Kasturi, D.L., and C. Suen (ed.): *International Conference on Pattern Recognition, Vol. 3, Quebec, Canada* (2002) 37-40
139. Paclik, P., Duin, R.P.W.: Dissimilarity-based classification of spectra: computational issues. *Real Time Imaging* 9 (2003) 237-244

RT\_014, noviembre 2009

Aprobado por el Consejo Científico CENATAV

Derechos Reservados © CENATAV 2009

**Editor:** Lic. Lucía González Bayona

**Diseño de Portada:** DCG Matilde Galindo Sánchez

RNPS No. 2142

ISSN 2072-6287

**Indicaciones para los Autores:**

Seguir la plantilla que aparece en [www.cenatav.co.cu](http://www.cenatav.co.cu)

C E N A T A V

7ma. No. 21812 e/218 y 222, Rpto. Siboney, Playa;

Ciudad de La Habana. Cuba. C.P. 12200

*Impreso en Cuba*

