



# Processing of chromatographic signals

how to separate the wheat from the chaff

PhD thesis • 2013

Lea Giørtz Johnsen

**Title**

Processing of chromatographic signals  
how to separate the wheat from the chaff

**Submission**

December 20<sup>th</sup>, 2012

**Defence**

March 4<sup>th</sup>, 2013

**Supervisors**

Professor Rasmus Bro  
Department of Food Science, Faculty of Science  
University of Copenhagen

Associate professor Thomas Skov  
Department of Food Science, Faculty of Science  
University of Copenhagen

Senior Research Scientist Asger Geppel  
Department of Assays  
Chr. Hansen A/S

**Opponents**

Associate professor Mikael Agerlin (Chairman)  
Department of Food Science, Quality and Technology  
University of Copenhagen

Professor Sarah Rutan  
Department of Chemistry  
Virginia Commonwealth University

Ph.D. Tore Vulpus  
Manager of MS Consult

Cover photo by Morten Tom-Petersen

PhD thesis • 2012 © Lea Giørtz Johnsen

ISBN 978-87-7611-540-1

Printed by SL grafik, Frederiksberg C, Denmark ([www.slgrafik.dk](http://www.slgrafik.dk))

*"Information is not knowledge"*

Albert Einstein



## Preface

---

This thesis has been made in collaboration between the Quality and Technology (Q&T) group at the department of Food Science, Faculty of Science, University of Copenhagen and the department of Assays at Chr. Hansen A/S. The project has been supervised by Rasmus Bro (Q&T), Thomas Skov (Q&T), and Asger Geppel (Chr. Hansen).

I am grateful to Rasmus for his enthusiasm, for the many inspiring discussions, and for always believing in me. Thank you, Thomas, for the many talks, both those concerning technical topics and those concerning all the other stuff. Asger, it was you who started this quest at Chr. Hansen. Thank you for that. Without your footwork this thesis would have taken a totally different path.

Of course a lot of other people have contributed to this work. All my colleagues at Q&T have made the years more enjoyable and you has always been ready to help whenever needed. Thank you for that. A special thanks to Maja for the, at time frustrating, but also enjoyable, times we have spent together.

I would also like to thank all my colleagues from the dept. of Assays. I am truly grateful for being a part of this department both in the social content, but also in the technical part. I would especially like to thank Morten for always supporting me and this project, and Ulf who always is inspiring and with whom I really enjoyed collaborating during the development of FastChrom.

And then there of course is Inger, thank you so much for proof reading all of this.

Finally, I would like to thank my family, Martin, Freja, and Magni, for taking my mind of all the nerdy stuff and for reminding me of what really matters.

*Lea G. Johnsen*

Copenhagen, December 2012

## Summary

---

The aim with this work is to make untargeted analysis of chromatographic data more accessible for those working with chromatography on a daily basis. This task has been fulfilled partly by developing some new tools, and partly by describing some existing methods with tutorials in this thesis.

In order to facilitate the choice of method a scheme has been presented which can guide the reader to select proper methods for *their* type of data. In this scheme, data is divided into two main categories; data originating from first-order instruments and data from higher-order instruments. These are then further divided into subcategories depending on the complexity of the chromatograms.

For low complexity chromatograms from first-order instruments, FastChrom has been developed. FastChrom is a graphical user interface, running in MATLAB, which removes baseline contributions and subsequently finds all peaks in the chromatograms and groups those across samples.

For high complexity chromatograms from first-order instruments, an already existing method is suggested and MATLAB code is provided, so users without MATLAB experience can take advantage of the method. The principle of the method is to remove artefacts like baseline and shifts before using multivariate data analysis on the full chromatograms.

For low complexity chromatograms obtained from higher-order instruments, PARAFAC2 is proposed as the preferred method. In order to make PARAFAC2 more accessible some new developments have been made, with automation of the evaluation of how many factors to include as the most important. With this, PARAFAC2 can be used to resolve overlapping peaks without time consuming manual evaluation which requires considerable chemometric and chromatographic knowledge. For intervals which are not well modelled with PARAFAC2, MCR is suggested and carefully described.

For high complexity chromatograms from higher-order instruments it is suggested to use the approach describe for the complex chromatograms from first-order instruments on e.g. the total ion chromatograms. A number of other methods are also listed, but not further described.

## Resume

---

Formålet med denne afhandling, er at gøre u-specifik analyse af kromatografisk data, tilgængelig for dem der arbejder med kromatografi i deres daglige arbejde. Dette er til dels blevet opnået ved at udvikle nye metoder, og til dels ved at beskrive eksisterende metoder med et antal brugervenlige tutorial i denne afhandling.

For at gøre det enklere at vælge metode, er der blevet udviklet et diagram, der guider brugeren til at vælge korrekte metoder til deres data. I dette diagram er data delt i to kategorier; data fra første-ordens instrumenter og data fra højere-ordens instrumenter. Disse er desuden yderligere inddelt i underkategorier afhængigt af kompleksiteten af kromatogrammerne.

FastChrom er blevet udviklet til kromatogrammer med lav kompleksitet. FastChrom er en grafisk brugergrænseflade der kører i MATLAB. FastChrom fjerner basislinje og finder efterfølgende alle toppe i kromatogrammerne og grupperer disse på tværs af prøver.

For kromatogrammer med høj kompleksitet fra første-ordens instrumenter, anbefales en eksisterende metode. Metoden er grundigt beskrevet, og der er vist MATLAB kode, så brugere uden erfaring i MATLAB kan udnytte denne metode. Princippet bag metoden er at fjerne basislinje og retentionstids skred inden analyse af de komplette kromatogrammer med multivariate data analyse.

For kromatogrammer med lav kompleksitet fra højere-ordens instrumenter er det anbefalet at bruge PARAFAC2. For at gøre PARAFAC2 lettere tilgængelig, er der blevet udviklet nogle nye redskaber, hvor automatisering af evalueringen af hvor mange faktorer der skal inkluderes, er den vigtigste. Med denne automatisering kan PARAFAC2 bruges uden tidskrævende manuel evaluering, hvilket desuden kræver kemometrisk og kromatografisk viden. Det anbefales at bruge MCR til intervaller der ikke kan modelleres tilfredsstillende med PARAFAC2.

For kromatogrammer med høj kompleksitet fra højere ordens instrumenter anbefales det at bruge samme metode som for komplekse kromatogrammer fra første-ordens instrumenter på f.eks. total ion chromatograms. En række andre metoder er også foreslået, men ikke gennemgået.



## List of publications

---

Paper I

**Lea G. Johnsen**, Thomas Skov, Ulf Houlberg, and Rasmus Bro.

“An automated method for baseline correction, peak finding and peak grouping in chromatographic data”

*Submitted for Analyst*

Paper II

Maja H. Kamstrup-Nielsen, **Lea G. Johnsen**, and Rasmus Bro.

“Core consistency diagnostic in PARAFAC2”

*Submitted for Journal of Chemometrics*

Paper III

Rasmus Bro, Riccardo Leardi, and **Lea G. Johnsen**.

“Solving the sign-indeterminacy for multi-way models”

*Accepted for publication in Journal of Chemometrics*

Paper IV

**Lea G. Johnsen**, José Manuel Amigo, Thomas Skov, and Rasmus Bro.

“Automated resolution of overlapping peaks in chromatographic data”

*Submitted for Analytical Chemistry*

## Other publications by the author

---

Carina Svendsen, **Lea G. Johnsen**, and Rasmus Bro.

“Exploring fermentation processes using gas chromatography-mass spectrometry and chemometrics”

In preparation

## List of abbreviations

---

COW	Correlation Optimized Warping
FID	Flame Ionisation Detector
GC	Gas Chromatography
GUI	Graphical user interface
H	Height
HPLC	High Performance Liquid Chromatography
<i>icoshift</i>	Interval correlation optimised algorithm
KI	Kovats Index
MCR	Multivariate Curve Resolution
MS	Mass Spectrometry
PARAFAC	PARAllel FACtor analysis
PARAFAC2	PARAllel FACtor analysis 2
PCA	Principal Component Analysis
PLS-DA	Partial Least Squares - Discriminant Analysis
rt	Retention time
S/N	Signal-to-Noise
SVD	Singular Value Decomposition
TIC	Total Ion Chromatogram
W	Width



## Table of contents

---

<b>PREFACE</b>	<b>I</b>
<b>SUMMARY</b>	<b>II</b>
<b>RESUME</b>	<b>III</b>
<b>LIST OF PUBLICATIONS</b>	<b>V</b>
<b>OTHER PUBLICATIONS BY THE AUTHOR</b>	<b>VI</b>
<b>LIST OF ABBREVIATIONS</b>	<b>VII</b>
<b>TABLE OF CONTENTS</b>	<b>IX</b>
<b>INTRODUCTION</b>	<b>1</b>
1.1 BACKGROUND	2
<b>INTRODUCTION TO CHROMATOGRAPHY</b>	<b>7</b>
<b>INTRODUCTION TO THE CHEMOMETRIC METHODS</b>	<b>13</b>
<b>3.1 NOMENCLATURE</b>	<b>13</b>
THE FIRST- AND SECOND- ORDER ADVANTAGES	14
UNIQUENESS	15
ALPHABETIC DEFINITION OF WORDS	16
<b>3.2 MULTIVARIATE CURVE RESOLUTION</b>	<b>17</b>
<b>3.3 PARAFAC</b>	<b>19</b>
<b>3.4 PARAFAC2</b>	<b>23</b>

---

---

<b>3.5 ALIGNMENT</b>	<b>25</b>
COW	25
ICOSHIFT	28
<b>WHICH METHODS TO CHOOSE</b>	<b>31</b>
4.1 CHOOSING THE RIGHT PRE-PROCESSING	31
4.2 AREA VS. HEIGHT	34
<b>SINGLE-CHANNEL DATA</b>	<b>37</b>
5.1 FASTCHROM	37
5.2 FULL CHROMATOGRAM ANALYSIS	42
<b>MULTI-CHANNEL DATA</b>	<b>49</b>
6.1 BASELINE SEPARATED INTERVALS - PARAFAC2	49
MANUAL EVALUATION OF PARAFAC2 MODELS	52
AUTOMATED EVALUATION OF PARAFAC2 MODELS	57
6.2 BASELINE SEPARATED INTERVALS - MCR	58
6.3 MORE COMPLEX DATA	64
<b>CONCLUSIONS AND PERSPECTIVES</b>	<b>65</b>
<b>REFERENCES</b>	<b>69</b>

---

## *Chapter 1*

# Introduction

---

This thesis is dealing with different aspects of processing of chromatographic data. The construction of the thesis is as follows:

The first chapter gives an overview of the problems sought solved with the work described in this thesis.

Chapter two gives a short introduction to chromatography; the purpose with this chapter is to give people, not usually working with chromatography, the tools to read the remaining part of the thesis.

In chapter three the theory of the chemometric methods used later on, is described. It is suggested that readers with no chemometric experience read at least the first section in this chapter. This section will introduce some basic nomenclature and principles which are useful to be familiar with, when reading the remaining part of the thesis. The last four sections of chapter three describe the theory of some chemometric methods which are useful when handling chromatographic data.

Chapter four describes some of the considerations one needs to make before starting processing the data.

Finally two sections are included (chapter five – single-channel data and chapter six – multi-channel data) where solutions to the problems outlined in the next section (1.1) are described. These two chapters are written as a number of tutorials, showing how different data should be treated. The chapters include new developments as well as well-established methods. In these chapters MATLAB code is provided. All code has been tested using MATLAB 2012a (Mathworks, Inc., Natick, Massachusetts, U.S.A.).

The examples in the thesis are data obtained from analysis with Gas Chromatographs (GC) coupled with either Flame Ionisation Detectors (FID) or Mass Spectrometer (MS) detectors, however many of the methods described here can easily be applied to other chromatographic methods.

## 1.1 Background

In the study of the metabolome of microbial, plants or any other system, metabolic profiling is often the preferred approach. The term “metabolic profiling” has been defined as,

*“Detection of a wide range of metabolites, [...] Relative changes in response (correlated to changes in metabolite concentration) are used to define metabolic differences. Metabolic profiling is normally applied in an inductive experimental strategy [...] where the metabolites of biological interest are not known a priori”*

(Dunn 2008)

In this definition there are two things which are very important to notice, namely that a wide range of metabolites are detected and that the metabolites of interest are not known. This has resulted in the development of a number of chromatographic methods with the aim of covering as many compounds as possible in one single analysis (Kanani, Chrysanthopoulos & Klapa 2008, Büscher et al. 2009). Since the focus in the analysis is shifting away from identification and quantification of a specific set of compounds, the approaches used for analysis of the chromatographic data also need to be changed. In the traditional methods for handling of chromatographic data a search for peaks at specific retention times, or peaks with specific m/z or UV spectra is conducted. However, with the increase in number of samples and data amounts, this approach becomes too time-consuming. Koek *et al.* (2011) have reported that it can easily take a full week to create such a target list for 20-40 samples, even for an experienced scientist. Besides the time consumption, the traditional approach may result in biased data evaluation with a high risk of missing valuable information since only appointed compounds are searched for. Therefore there is an urgent need for development of new untargeted methods which can extract information from data in a more efficient and unbiased way.

*“A man should look for what is, and not for what he think should be”*

Albert Einstein

The companies which develop the chromatographic instruments have developed some tools for extracting information from the obtained data. The traditional solution for handling data obtained from first-order instruments, offered by this commercial software, is that the user defines a number of elution time windows. Peaks eluting inside these windows will then be evaluated and their height or area reported. However, there are several disadvantages with this approach. The main drawback is that peaks eluting outside of the pre-defined windows

will not be taken into consideration even though they may represent important information. Another major disadvantage is that whenever shifts in retention time occur, the elution time windows must be re-calibrated. If the calibration of the elution time windows is not carefully checked, there is a high risk that peaks will shift outside the windows (situation A, Figure 1), or that a peak originating from a different compound shifts into the window (situation B, Figure 1).

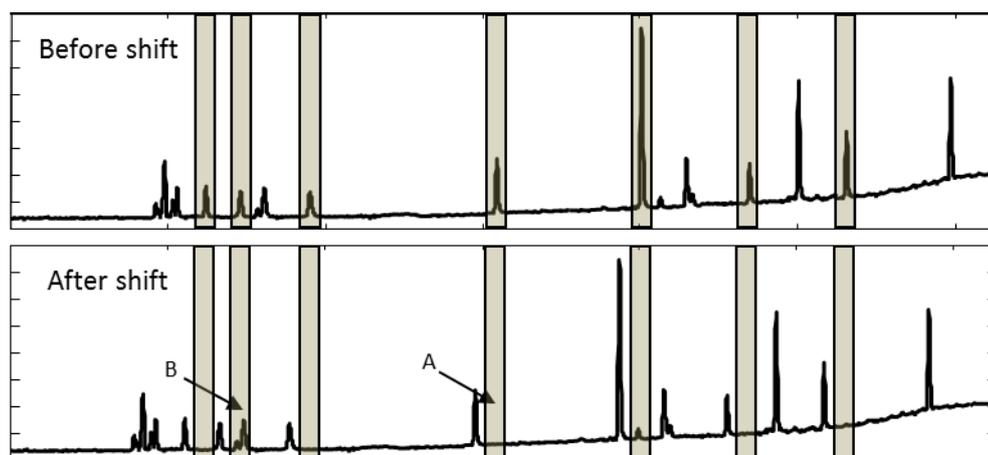


Figure 1. Illustration of the traditional approach of finding and identifying compounds analysed with GC-FID. Only peaks eluting inside the pre-set windows will be reported. If shifts occur, peaks might shift out of the window (A) or another compound may shift into the window and be wrongly identified (B).

In this thesis two different approaches are presented, which make all information contained in the chromatograms available. The two approaches are optimal for handling chromatograms with high and low complexity respectively. The method suggested for low complexity chromatograms is the newly developed method presented in paper I. The method for handling of high complexity chromatograms is the methodology described by Christensen *et al.* (Christensen *et al.* 2004, Christensen *et al.* 2005, Christensen, Tomasi 2007).

The evaluation of data obtained from multi-channel detectors in commercial software is, to some extent, based on the same principle as described for single channel detectors. However, the identity of the peaks eluting inside the pre-set windows will be confirmed by usage of the spectra representing that particular peak. This means that situation B in Figure 1, most likely, will be reported as the compound of interest being absent. However, this does not make the approach problem free, since there is still the problem with unreported peaks

outside the windows, as well as erroneous reports of compounds not being present in cases with shifts in the retention time.

An alternative approach, available in newer versions of commercial software, is to perform deconvolution (or mathematical resolution) of peaks. This approach is very useful when performing metabolic profiling. However, these deconvolution algorithms is not always performing in a satisfactory way (Skov, Bro 2008, Murphy et al. 2012). The main problem with the deconvolution algorithms is that they consider one sample at a time and do not use the information from the other samples to resolve overlapping peaks. This results in suboptimal models as illustrated in Figure 2. To the left in the figure the Total Ion Chromatograms (TIC) representing the raw data are shown. For one representative sample the full data is shown in the middle. Both here and in the TIC a small shift is seen, indicating that the datasets represent two overlapping peaks with a small difference in elution times. An example of deconvolution performed with an algorithm from commercial software is shown to the left in Figure 2. The deconvolution only describes a total of six peaks in all of the 45 samples. This means that a lot of information is lost. Furthermore it does not seem like the algorithm has been able to separate the two compounds, since a shoulder is present on some of the peaks obtained from the algorithm (circle in the rightmost plot in Figure 2). An additional problem is that a varying amount of baseline is included in the modelled peaks, and that the boundaries of the individual peaks do not seem to be optimally defined.

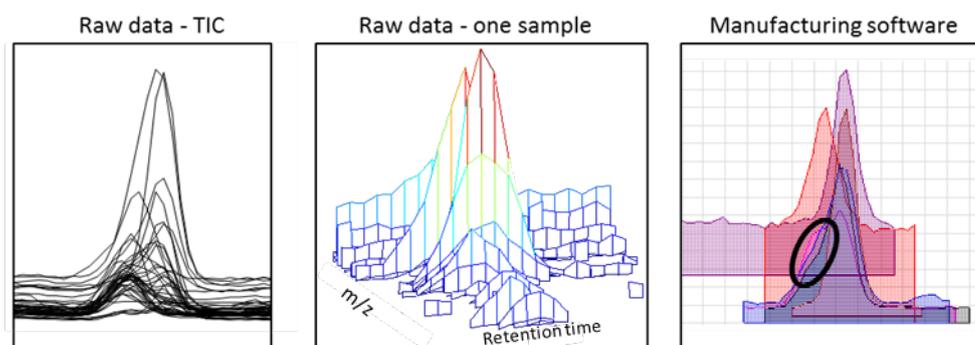


Figure 2. Left: Total Ion Chromatograms (TIC) from 45 samples. It seems like there are two groups of peaks indicating that the two different compounds are eluting in the interval. Middle: Plot of all mass channels from one sample. Here there also seems to be a small shift, which supports the theory of two different compounds are eluting in the interval. Right: The result from deconvolution with a commonly used commercial software, illustrated with elution time profiles. Only six peaks are found in total, meaning that a lot of the peaks in the raw data have not been properly modelled.

A common problem for the approaches in most commercial software is that the algorithms are “black boxes” meaning that the user has no knowledge about what the algorithm actually is doing, and therefore has a very limited possibility of evaluating the quality of the obtained result.

Curve resolution techniques offer an alternative to the deconvolution algorithms. These methods take the entire set of samples in the dataset into account. The curve resolution technique, PARAFAC2, has previously been shown to be a very useful tool for comprehensive analysis of chromatographic data (Bro, Andersson & Kiers 1999, Amigo et al. 2008, Khakimov et al. 2012). However, in order to evaluate how many factors to include in a PARAFAC2 model, a manual evaluation is required. This will inevitably result in models which are biased according to the users’ experience and personal opinion. Besides this it is also a very time consuming task and it requires a skilled chemometrician with considerable chromatographic knowledge. These obstacles may be why PARAFAC2 is not more widely used in routine analysis. In an attempt to make the use of PARAFAC2 more widespread, an automated approach for evaluation of PARAFAC2 models has been developed and presented in paper IV. This approach will enable non-chemometricians to use PARAFAC2 in comprehensive analysis of chromatographic data. In this thesis it is suggested to use this approach when dealing with data obtained from hyphenated chromatographic techniques, and tutorials are presented both for the automated and the manual approach. It is also described how data which is not well modelled with PARAFAC2 can be handled.



## Chapter 2

**Introduction to chromatography**

The word chromatography originates from the two Greek words “*chroma*” (colour) and “*graphein*” (to write). The technique was first developed by the Russian botanist M. S. Tswett (1872-1919) as described in his two papers from 1906 (Tswett 1906a, Tswett 1906b). During his work with plant pigments he discovered that they could be separated on columns packed with an adsorbent (he mainly used calcium carbonate). The separation was displayed on the column as separated bands in different colours representing the different pigments.

In modern chromatography, the word covers a wide range of methods, which all have in common that they separate compounds in the “mobile” phase based on their affinity to a “stationary” phase. The two most common chromatographic methods are Gas Chromatography (GC) and High Performance Liquid Chromatography (HPLC). Other methods do exist but will not be further described here.

In both HPLC and GC, the typical design will be as follows: The sample enters the system via an injection port; here it will be carried to the column by the mobile phase, which can be either gaseous (GC) or liquid (HPLC). The separation of the compounds will take place during the transport through the column. After passing through the column, the compounds will enter a detector, which monitors the separated compounds. A schematic overview of the design is shown in Figure 3.

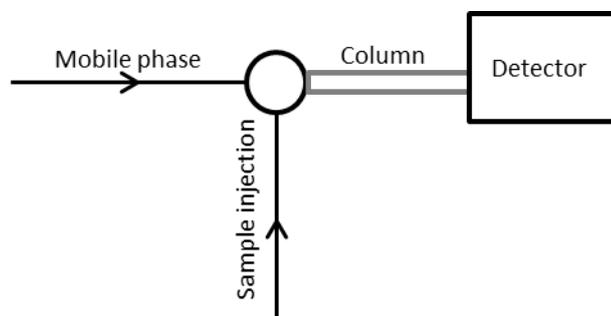


Figure 3. The sample enters the system via an injection port which is connected to the column. The column will typically be placed in a temperature controlled compartment.

In HPLC, the mobile phase is a liquid with low viscosity. The separation can be influenced by changing the composition of the mobile phase and the temperature of the column oven during the analysis. By changing the temperature also the affinity towards the stationary phase is changed, and hereby a better separation can be obtained. The mobile phase in GC is typically an inert gas. The composition of the mobile phase will not be changed during the analysis; separation is obtained by changing the pressure and temperature of the column.

Injection of the sample onto the column places all analytes in a narrow band in the beginning of the column. The mobile phase will then carry the compounds contained in the sample through the column. The analytes will, during the transport through the column, alternate between the stationary phase and the mobile phase. In the mobile phase the travelling speed will be the same for all compounds. The separation of different compounds therefore occurs due to differences in their affinity towards the stationary phase. The longer time a molecule is located in the stationary phase the longer time it will spend in the column. The time it takes for the compounds to travel through the column will therefore vary according to the chemical and physical properties of the individual compounds. During the transport through the column some band spreading of the compounds will also occur due to small differences in the time spent in the stationary phase. These differences result in a Gaussian distribution of the compound. The width ( $W$ ) of this distribution is dependent on the specific compound and the length of the column. The separation process is illustrated in Figure 4.

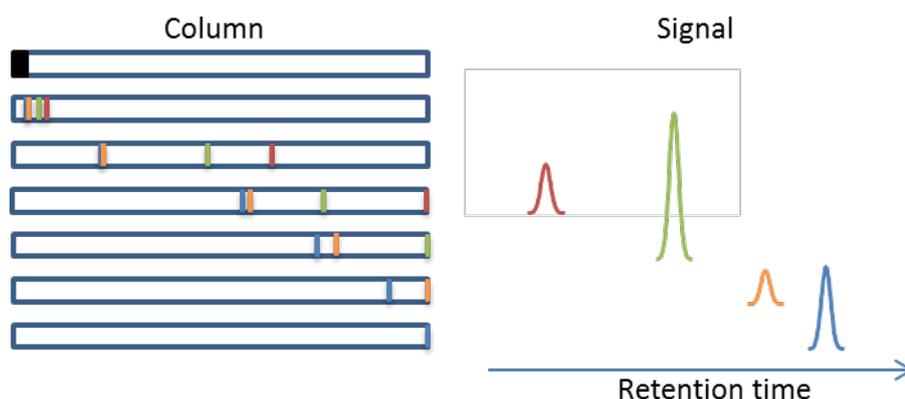


Figure 4. Separation in the column. All analytes are positioned in one narrow band on the column. During the transport through the column, they travel with different velocities and will therefore have different retention times. When an analyte reaches the detector a signal is recorded.

The total signal representing the individual compounds is normally proportional to concentration. This means that a doubling of concentration results in a doubling of the signal. The amount of signal can be found by summing the signal in the period where the peak of interest is eluting. In other words, the area under the peak will, under normal circumstances, be directly proportional to the concentration of the compound (within the linear range of the detector).

Ideally, the signal will only describe the compounds from the sample. However, in reality the obtained signal from a chromatographic analysis is consisting of three main contributions: desired chemical information, baseline, and noise, as illustrated in Figure 5. Noise is characterised by being high frequency variation. The desired chemical information is seen as a peak in the chromatogram, and is typically characterized by height (H), area, width (W) (determined at half height), and signal-to-noise (S/N) ratio (see Figure 5). The S/N ratio is used to state how certain the amount of analyte is determined. Low S/N values indicate that a big part of the signal is noise, and thus there will be a high degree of uncertainty. Traditionally an S/N limit of 10 has been used for quantification purposes and a limit of three for detection purposes.

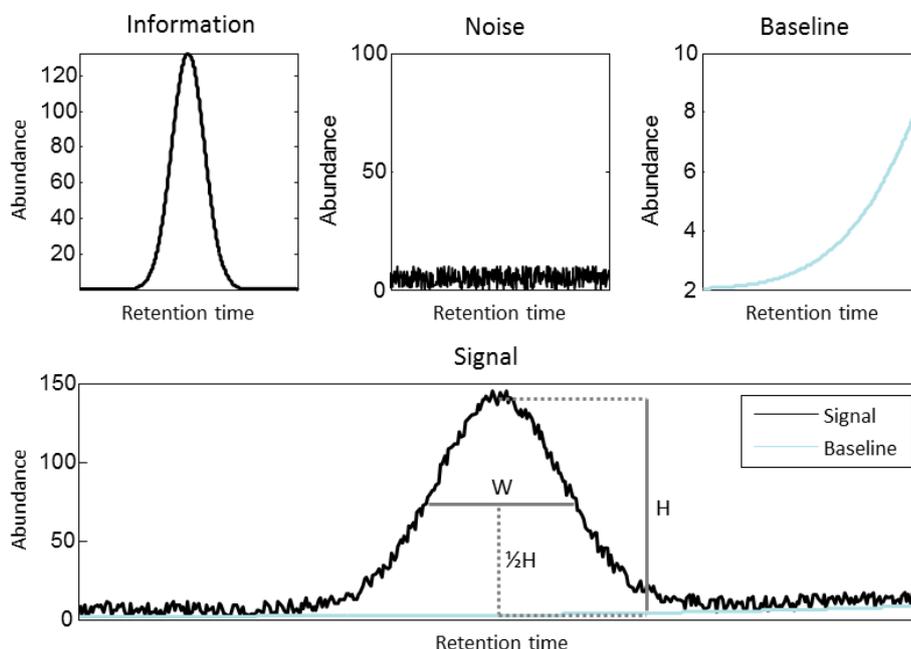


Figure 5. Composition of the chromatographic signal, and illustration of commonly used peak features. H: height of the peak. W: width of the peak.

The main objective of performing analytical chromatography is to be able to identify and quantify compounds. Traditionally, the peak area has been used for quantification purposes. However, as presented in section 4.2 heights are in most situations more robust in GC analysis. Regardless of whether height or area is chosen as representation of concentration, it is required that the information is separated from especially the baseline, but also to a certain degree the noise, so that these contributions are not included in the quantification. A high number of pre-processing methods, with the purpose of retrieving the pure information from the signal, has been developed during the years. The application of a selection of those, together with some new developments, will be described in chapters five and six.

The detectors used in HPLC and GC can be divided into two main categories: single-channel and multi-channel detectors. If the recorded signal is obtained from a single channel detector only one value is recorded for each time point. The multi-channel detector records several parallel signals for each time point resulting in a two dimensional dataset for each sample. A visualization of the difference between the appearance of the single-channel and the multi-channel signals is shown in Figure 6.

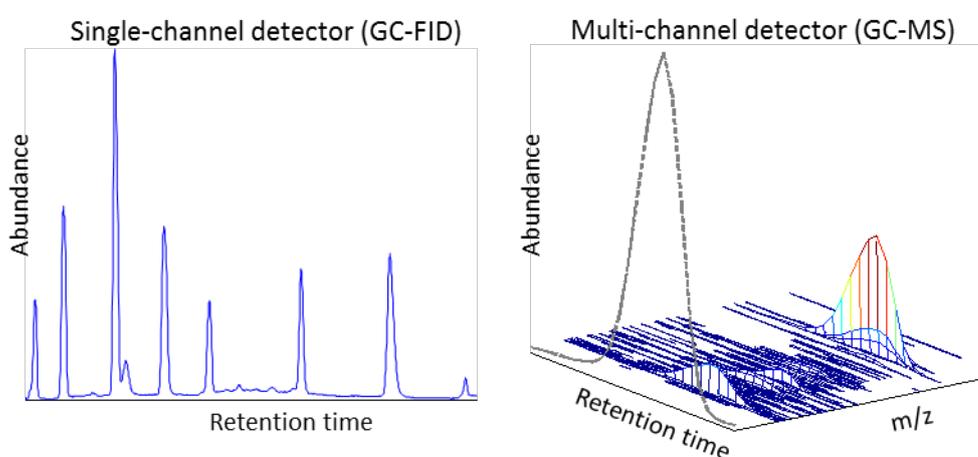


Figure 6. Illustration of the signal obtained from a single-channel detector vs. a multi-channel detector. The single-channel signal is from a GC coupled with a Flame Ionisation Detector (FID), and the multi-channel signal is from a GC coupled with a Mass Spectrometer (MS) detector. The dotted grey line in the multi-channel data is the Total Ion Chromatogram (TIC) which is obtained by summing over the  $m/z$  direction. The TIC is comparable with the obtained chromatogram from the single-channel detector.

Several types of both single- and multi-channel detectors with different properties exist. The Mass Spectrometer (MS) detector is widely used in both HPLC and GC. The MS measures a mass spectrum for each elution time point. This means that information of the size of the eluting compounds (and fragments hereof) is obtained, and this information can be used in identification. The spectra, obtained from GC-MS, are especially useful for identification purposes, since a given compound always results in the same spectrum. This spectrum, together with the obtained retention time, provides a unique fingerprint for that particular compound (with exception of isomers). Several libraries which can be used for identification of GC-MS spectra exist, and one of the most widely used libraries is the NIST mass spectral library.

Another type of multi-channel detector, which is used in HPLC, is the photo diode array. This detector measures a UV spectrum at each time point. Since many organic compounds have characteristic UV spectra, the obtained spectra can be used for identification purposes in the same way as the MS spectra.

Single-channel detectors do not provide any supplementary information besides the retention time (and to some degree peak width) which can be used for identification purposes. However, especially the Flame Ionisation Detector (FID), which can be used in combination with GC, has a high linear range and is a very robust detector.

The choice of method used to extract the information from a signal is, to a very high degree, dependent on whether the detector is a single- or multi-channel detector and the complexity of the signal. A guide on which method to choose is given in section 4.1.



## Chapter 3

# Introduction to the chemometric methods

---

### 3.1 Nomenclature

In this section some key-terms used in the remaining part of the thesis will be described. The section will be initiated with a description of how different types of data can be arranged. This will be followed by brief descriptions of what the terms “*uniqueness*” and “*first- and second-order advantage*” mean and why these terms are important when dealing with processing of chromatographic data. The section will be completed with a table with short definitions of key-words which are used throughout this thesis.

Data from detectors which measure only one value per sample (e.g. pH) will give scalars as the output (zero-order data). In cases with more than one sample, the resulting dataset will be a vector (one-way data). Data from single channel detectors like GC-FID can be arranged in a vector (first-order data). If more than one sample has been analysed, data can be arranged in a two-dimensional data matrix with the samples in the rows and the abundance at different time points in the columns (two-way data). For multi-channel detectors, like GC-MS, spectra containing a number of  $m/z$  values are measured for every time-point (second-order data). In cases with more than one sample, an array is a more suitable arrangement of data (three-way data). The different dimensionalities and arrangement of data are shown in Figure 7.

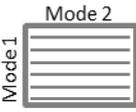
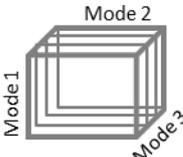
	One sample	More than one sample
One value per sample	Zero-order Scalar 	One-way Vector 
One vector per sample	First-order Vector 	Two-way Matrix 
One matrix per sample	Second-order Matrix 	Three-way Array 

Figure 7 Illustration of different dimensionalities of data and how they typically will be arranged.

The following nomenclature is used in the thesis (as suggested by Kiers (Kiers 2000)): Italics are indicating scalar numbers –  $F$  is reserved to indicate the number of factors included in a model and the letters  $I$ ,  $J$ , and  $K$  to the number of variables included in each mode. Bold lowercase is indicating vectors, bold uppercase is indicating two-way matrices and underlined, bold uppercase is indicating three-way arrays. The letter  $E$  is always indicating residuals and has the same dimensions as the raw data.

#### *The first- and second- order advantages*

The treatment of data is highly dependent on whether data is obtained from zero-, first- or second-order instruments.

If a signal, obtained from a zero-order instrument, is linearly correlated to the concentration of the related compound, the concentration can be estimated by measuring one standard sample with known concentration (or two if a background signal is present). By regression, the unknown concentration can then be estimated. However, if there, in a future sample, is a different background than in that of the standards, the estimated concentration will be wrong (Booksh, Kowalski 1994). This means that the signal from interferences must be constant, and since it is impossible to detect changes in this signal, substantial purification is necessary before quantification can be performed with signals obtained from zero-order instruments.

If the signal is obtained from a first-order instrument, it is possible to recognize a sample as outlier if unexpected components are present. Furthermore quantification can be performed despite of variations in background signal, as long as the same types of variations are present in the calibration samples. This is called the first-order advantage (Booksh, Kowalski 1994). Since the calibration set should cover as many types of variation in the background as possible, a high number of samples must be included in the calibration set. In order to utilise the first order advantage proper data handling methods must be applied (e.g. PLS).

For signals obtained from second-order instruments, quantification is possible, even in cases with interferences not included in the calibration sample. This is called the second-order advantage (Booksh, Kowalski 1994). The second order advantage means that the purification step, which traditionally has been necessary in quantification, no longer is required *and* quantification can be performed using just *one* standard. In order to take full advantage of the second-order advantage proper data handling techniques must of course be applied (e.g. PARAFAC2).

#### *Uniqueness*

Whenever models are used to describe data, it is not enough to create models which are mathematically correct, the models must also describe the real underlying chemistry; therefore it is problematic when some methods can result in several equally mathematically correct answers, since only one of these can describe the actual underlying chemistry. Methods which can give more than one mathematically correct solution to the same dataset is said to be non-unique, in contrast to unique methods which can only result in one mathematically correct model. Often results from non-unique methods are interpreted as if they were describing actual chemistry. However, this is not necessarily the case, since it might be one of the other possible solutions which are describing the chemistry. On the other hand the solution from a unique method *will* be describing the real chemical variations in data (assuming that the method is valid for the data at hand).

Uniqueness can be exemplified in the following way. Suppose that the amount of  $y$  is dependent on  $x$  in the following way:  $y = a + bx$ . Even if we are able to measure  $a$  and  $b$  it is not possible to determine the concentration of  $y$ . If, for instance,  $a$  is measured to 6 and  $b$  be to 4, both  $(x = -1, y = 1)$  and  $(x = 2, y = 13)$ , plus an infinite number of other combinations, are solutions to the equation. In this case we have a problem without a unique solution (the true concentration of  $y$  can only be *one* of the possible solutions). In bilinear models, like MCR, there exist a number of different combinations of components, which all result in a model with the same fit. This is called rotational freedom, and it is because of this, that the MCR

solution is not unique. In order to obtain a unique solution, constraints must be applied to the MCR algorithm. Examples of constraints could be non-negativity (which does not always provide uniqueness, but often only limits the solution space) or  $x = 4$  (which will result in a unique solution to the above-mentioned problem). In section 3.2 constraints often used in MCR will be described.

In trilinear models, like PARAFAC or PARAFAC2, uniqueness is obtained by combining information across observations. An example could be observation 1:  $y = 5 + 4x$  and observation 2:  $y = 9+3x$ . In this case only one solution is valid ( $x = 4, y = 21$ ) and hence the solution is unique.

#### *Alphabetic definition of words*

Compound	Is referring to real chemical analytes present in a sample.
Factor/component	Is referring to the factors in factor/component models, and not to be confused with the chemical analytes (compound) above.
First-order advantage	Samples with interference can be detected as outliers, and quantification can be achieved despite of changes in the background signal, as long as the same type of changes is included in the calibration samples. For a more thorough description see the paragraph above.
Low-rank	The rank of a matrix is the maximum linearly independent column (or row) vectors in the matrix. In a chromatographic dataset, the rank tells us how many compounds that are changing, independently of the others, across samples. Low-rank (in a chromatographic perspective) refers to the fact that only a limited number of compounds are varying.
Mode	Refers to the matrix or array holding the data. A three-way array has three modes (modes one, two and three), whereas an N-way array has N modes. See also Figure 7.
Multivariate	More than two variables per sample, the term is used as opposed to univariate or bivariate which refers to dataset with only one or two variables per sample, respectively.
Multi-way	Three-way (or higher).

Order	Refers to the way data is arranged. If data is arranged in a matrix, it is said to be second-order (see also Figure 7). The term is used interchangeably with the term way. The term order can also be used about an instrument. For instance will a GC-FID analysis result in first-order data, hence the instrument is said to be a first-order instrument (and not a one-way instrument).
Second-order advantage	Compounds can be quantified in presence of new interferences <i>and</i> quantification can be achieved using just <i>one</i> standard. For a more thorough description see the paragraph above.
Uniqueness	No rotational freedom, or in other words does a unique model only provide one possible mathematical solution, which therefore also must be the solution which describes the underlying chemistry. For a more thorough description see the paragraph above.
Way	Refers to the way data is arranged. If data is arranged in a matrix it is said to be two-way (see also Figure 7). The term is used interchangeably with the term order.

### 3.2 Multivariate Curve Resolution

The main goal with Multivariate Curve Resolution (MCR) is to extract the pure responses (e.g. spectra and concentrations) from the raw signal. MCR is often used to resolve overlapping peaks in individual samples, but MCR can be used to resolve all kinds of overlapping phenomena in two-way data. The resolution is achieved by decomposing the raw data matrix (**D**) into two smaller matrices holding pure concentration profiles (**C**) and pure response profiles (**S**), as illustrated in Figure 8.

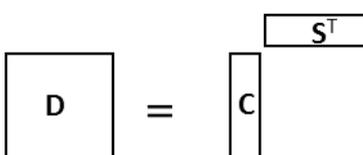
$$\text{MCR: } \mathbf{D} = \mathbf{C}\mathbf{S}^T$$


Figure 8. With MCR the matrix **D** ( $I \times J$ ), holding the raw data, is decomposed into **C** ( $I \times F$ ), holding the pure concentration profiles for the  $F$  components, and **S** ( $J \times F$ ), holding the pure responses for the  $F$  components.

The principle behind MCR is very similar to that of PCA; both methods are decomposing a matrix, holding the raw data, into two smaller matrices. However, there are fundamental differences between the two methods. In PCA the first component is describing the direction in the raw data with the highest variability. The second component is then, subsequently, found as the direction with the highest variability in the part of the raw data which was not described by the first component. This is continued until the desired number of components has been found. This means that the first components are unaffected by the total number of components included in the model. Furthermore the PCA solution is unique since there can only be one direction in data with highest variation.

In MCR all components are found simultaneously. If a two-factor model is calculated the two concentration profiles and two response profiles, which results in the best fit with the raw data, will be found simultaneously by iteratively calculating **C** and **S**. A consequence of all components being calculated simultaneously is that the appearance of the obtained profiles becomes dependent on how many components that are included in total. Another thing that is different from the PCA model is that the obtained model may not be unique, since different combinations of concentration and response profiles can result in the same fit. Since the MCR model is not unique, constraints must be applied to identify the model which describes the underlying chemistry. The most widely used constraint is non-negativity, this constraint is applied on both response and concentration profiles since these, under normal circumstances, will never be negative. Other types of commonly used constraints are unimodality (each profile can only describe one peak) and closure (the sum of the components must remain constant), but also information about where each of the peaks elutes (identify species), known spectra (equality constraint in spectra), and a number of other features can be applied as constraints. For a more thorough description of MCR the reader is referred to other literature (Tauler 1995, Gargallo et al. 1996, Jaumot et al. 2005).

If MCR is to be applied on three-way datasets, e.g. several samples obtained from second-order instruments, the three-way array must first be unfolded to a two dimensional matrix as illustrated in Figure 9. By unfolding sample wise, shifts in retention time is no longer a problem since the resulting matrix will not contain shift in any columns or rows. A side benefit of using such a dataset for MCR analysis, is that the problem with rotational freedom is decreased (or, if the right experimental design is used, solved) when calculating a model on several two-way datasets simultaneously (Tauler, Izquierdo-Ridorsa & Casassas 1993).

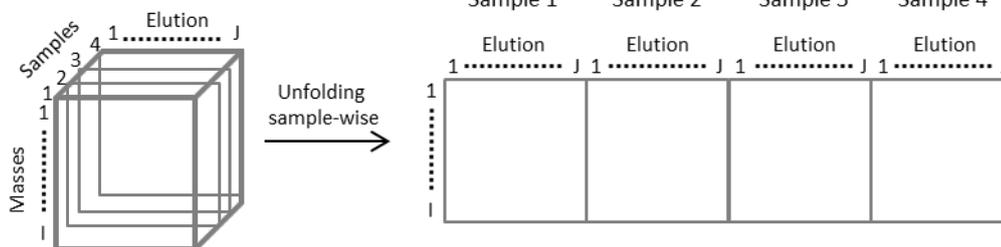


Figure 9. Sample-wise unfolding of the three-way array into a two-way matrix.

### 3.3 PARAFAC

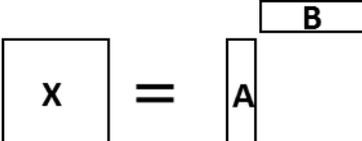
PARAFAC is a commonly used method for resolution of overlapping phenomena in three-way data (or higher). The PARAFAC algorithm is founded on Cattell's parallel proportional profiles, which are based on the assumption that, "... if a factor is one which corresponds to a true functional unity it will be increased or decreased as a whole." (Cattell 1944), p. 276). Cattell hypothesized that if such functional unities exist, a unique solution could be obtained by calculating the model on different occasions (or samples) simultaneously. Cattell was originally developing the parallel proportional profiles for general factor analysis, and the principle of a functional unity being increased, and decreased, as a whole is applicable to many areas. In Gas Chromatography coupled with Mass Spectrometry detectors (GC-MS), spectra libraries are routinely used to identify obtained spectra; this is only possible since changes in concentrations influence the spectra "as a whole".

The PARAFAC algorithm was first simultaneously developed by Harshman (1970) and Carroll and Chang (1970) (who named it canonical decomposition). One of the first to propose using the PARAFAC algorithm in chemometrics was Geladi who has made a thorough review of how to handle multi-way data (Geladi 1989). In the paper by Bro (1997) a review of the theory behind PARAFAC is given. In this section only the most important principles will be discussed with the purpose of introducing PARAFAC2 in the next section.

In principle PARAFAC is simply an extension of PCA to higher order data as illustrated in Figure 1. However, the PARAFAC algorithm has the major advantage that it is resolving the pure responses and concentrations. E.g. if PARAFAC is applied on a three-way array containing masses in mode one, elution time in mode two, and samples in mode three, the resulting model will consist of three matrices containing respectively; one mass spectrum for each component (**A**), one elution profile for each component (**B**), and relative concentrations of each of the modelled components in the individual samples (**C**).

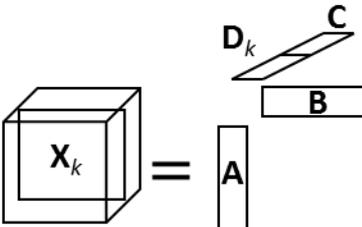
---

PCA:  $\mathbf{X} = \mathbf{A}\mathbf{B}^T$



---

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



---

Figure 10. Illustration of how similar PARAFAC and PCA are. For PCA,  $\mathbf{X}$  is a data matrix with dimensions  $I \times J$ ,  $\mathbf{A}$  ( $I \times F$ ) is the matrix holding the score vectors, and  $\mathbf{B}$  ( $J \times F$ ) the loading vectors.  $F$  is the number of factors in the model. For PARAFAC  $\mathbf{X}_k$  is the  $k^{\text{th}}$  slab of an  $I \times J \times K$  three-way data array.  $\mathbf{A}$  ( $I \times F$ ),  $\mathbf{B}$  ( $J \times F$ ) and  $\mathbf{C}$  ( $K \times F$ ) are all matrices holding loading vectors and  $\mathbf{D}_k$  is a diagonal matrix ( $F \times F$ ) with the  $k^{\text{th}}$  row of  $\mathbf{C}$  in its diagonal. The loading vectors describing the sample dimension are sometimes denoted score vectors.

One of the main advantages of PARAFAC, compared to many other curve resolution techniques, is that the obtained solution is unique. The uniqueness is obtained if the dataset contains at least two samples with independent variations in the concentration of the underlying phenomena (e.g. chemical compounds), assuming that these phenomena are characterised by unique patterns. And even in cases where one compound does not change across samples, unique and chemically meaningful solutions can be obtained as long as no other compound is constant. In cases where only some compounds are changing across samples, the solutions are only unique for these factors, while the solutions for the other factors are non-unique (Harshman 1970, Harshman, Lundy 1996). An additional requirement for obtaining a unique solution is that two (or more) compounds must not be totally overlapping (Manne 1995).

In order for an obtained PARAFAC model to be describing real chemical variation, some additional conditions must be met; there must not be shifts in data and the spectra (or pattern) of the compounds must not be too similar. If the present data fulfils these requirements, as well as the requirements listed above for uniqueness, chemically

meaningful and correct models can be obtained, assuming that the right number of factors is included in the model. Traditionally, the appropriate number of factors has been determined by manual evaluation of obtained models with different numbers of factors included. There exist a considerable number of parameters which can be used in this evaluation; among these are the appearances of the obtained loadings, the fit, the residuals, and core consistency (see the description below). In section 6.1 it is thoroughly described what to look for when determining the number of factors in chromatographic data, modelled with PARAFAC2, and to a great extent, the same approach can be used in evaluation of PARAFAC models. An alternative, automated, approach has been described by Furbo and Christensen (2012). However, this approach will not work on peaks with low S/N values. Another automated approach has been described by Hoggard and Synovec (2007). However, with this approach a target analyte is required in order to match the spectra, obtained from the model, with those from the analyte. To my knowledge there are no automated methods for untargeted analysis of low S/N peaks.

Core consistency is very useful for determination of cases where too many factors have been included in the model (Bro, Kiers 2003). It indicates how much of the variation, described by the model, that is really low-rank trilinear. Low-rank trilinear variation means that the compound described by one component is independent of the compounds described by the other components. Independent here does not mean uncorrelated. The components are in general correlated, but each such correlated component varies regardless of the other components and is therefore unaffected of changes in the concentration of other compounds. If a dataset includes four different phenomena (e.g. chemical compounds) and it is being modelled with five factors the four phenomena are forced to be divided over the five components somehow. The variation described by the five sets of loadings will no longer be independent of each other, and the variation described is therefore no longer low-rank trilinear. In these cases core consistency will be low.

Before evaluating and interpreting the obtained models, the problem with sign indeterminacy must be solved. Sign indeterminacy is a problem which occurs because one component vector can change sign, if one of the other component vectors also changes sign. Then, overall, there is no change in the component contribution to the model. This means that two models are identical, from a mathematical point of view, if two loadings of the same component have flipped. The sign indeterminacy can be illustrated in the following way; If  $\mathbf{S}$  is defined as diagonal matrices, with plus or minus one in their diagonal, the PARAFAC algorithm can be re-written as

$$\mathbf{X}_k = \mathbf{S}_1 \mathbf{A} \mathbf{S}_2 \mathbf{D}_k \mathbf{S}_3 \mathbf{B}^T,$$

where  $\mathbf{S}_1 \times \mathbf{S}_2 \times \mathbf{S}_3 = \mathbf{I}$ , and  $\mathbf{I}$  being the identity matrix with ones in the diagonal. As long as the overall sign of the model remains the same, the flip (or sign change) will not affect the model as such. However, the model can be very difficult to interpret, and evaluation of the numbers of factors also becomes much more difficult if the problem with sign indeterminacy is not solved. An example of such a flip is shown in Figure 11, where the elution profile and corresponding spectra profile of one compound have flipped in the model to the left. Since negative values in spectra and retention time profiles may indicate that too many factors have been included in the model, the flip might mistakenly be interpreted as a sign of over-fit. However, once the flip has been fixed (to the right in the figure) the model no longer seems over-fitted.

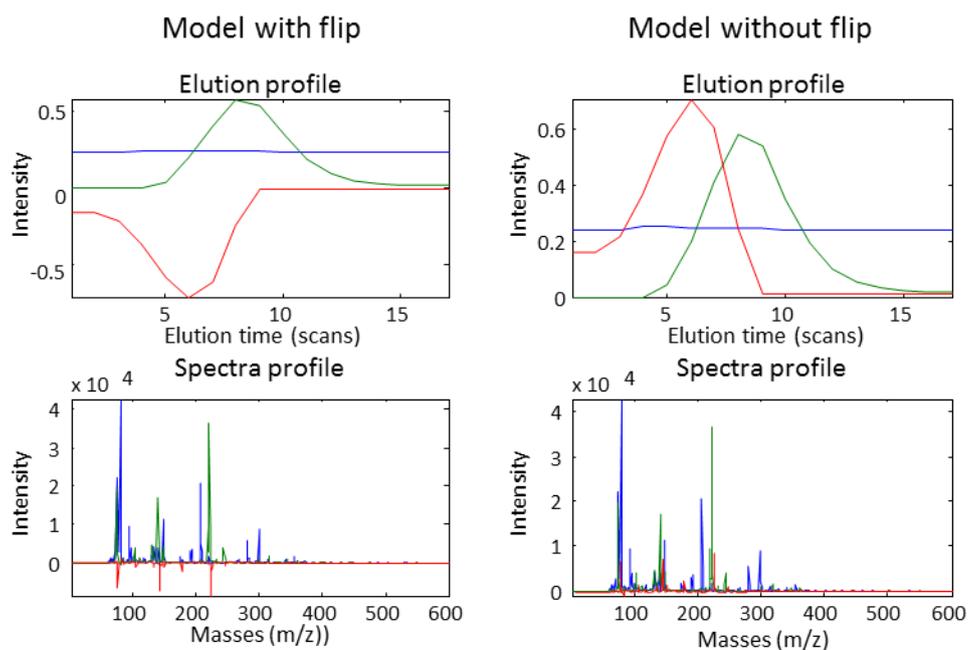


Figure 11. Illustration of sign indeterminacy. Left: One of the loadings in respectively the elution and spectra profiles has flipped. Right: No flip. From a mathematical point of view the two models are identical.

In paper III a solution to the problem with sign indeterminacy is suggested. The idea is that the data, in the model, should point in the same direction as the raw data. In a chromatographic dataset, the model should of course be positive, but in other kinds of data the natural sign is not always so obvious.

As mentioned above a limitation for PARAFAC is that in cases with shifts, it is not possible to find a model with one common elution profile for all samples. This means that for data with shifts in retention time (as almost always is the case in chromatographic analysis), data must either be aligned using methods like COW or *icoshift* (which will be further described in section 3.5), or alternatively other resolution techniques must be applied. These techniques could for instance be MCR (section 3.2) or PARAFAC2 (section 3.4). In cases with peak shape changes alignment will not help, but MCR or PARAFAC2 may still be able to provide chemically meaningful models.

### 3.4 PARAFAC2

In some cases (e.g. when shift in retention times occurs) PARAFAC becomes too restricted since it applies the same elution time profiles to all samples. These problems can sometimes be solved by using PARAFAC2 (Kiers, ten Berge & Bro 1999, Bro, Andersson & Kiers 1999).

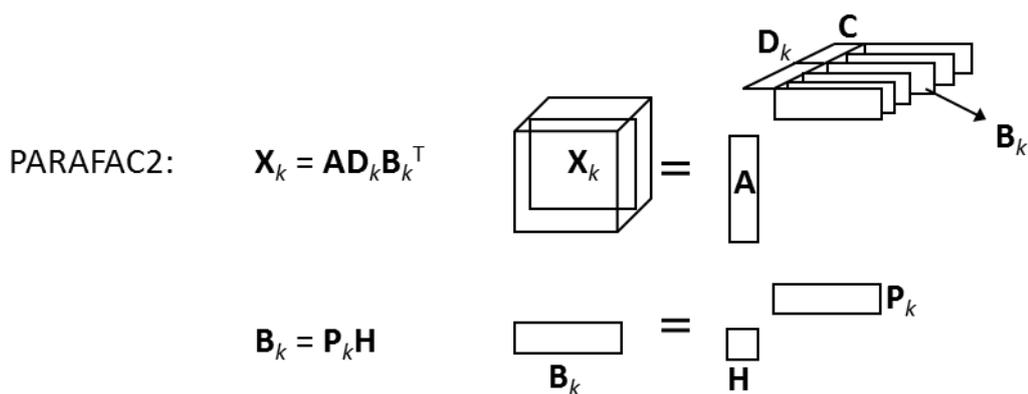


Figure 12. Illustration of the PARAFAC2 model. As in PARAFAC  $\mathbf{X}_k$  is the  $k^{\text{th}}$  slab of a  $I \times J \times K$  three-way data array. The matrices  $\mathbf{A}$  ( $I \times F$ ),  $\mathbf{B}_k$  ( $J \times F$ ), and  $\mathbf{C}$  ( $K \times F$ ) are all holding loading vectors and  $\mathbf{D}_k$  is a diagonal matrix ( $F \times F$ ) with the  $k^{\text{th}}$  row of  $\mathbf{C}$  in its diagonal. The loading vectors ( $\mathbf{c}$ ) which describe the sample dimension are sometimes denoted score vectors. The PARAFAC2 model deviates from PARAFAC by modelling unique elution time profiles for each sample ( $\mathbf{B}_k$ ). These profiles consist of a common part ( $\mathbf{H}$ ) and a unique part ( $\mathbf{P}_k$ ).

PARAFAC2 is very similar to PARAFAC; the main difference is that unique elution time profiles will be assigned for each of the samples. As the introduction of unique score factors destroy the uniqueness properties, an additional constraint is imposed in the PARAFAC2 algorithm, namely the cross-product of  $\mathbf{B}_k$  must be constant. This is obtained by dividing the matrix describing the dimension containing the shift ( $\mathbf{B}_k$ ) into an unique part ( $\mathbf{P}_k$ , with  $\mathbf{P}_k^T \mathbf{P}_k = \mathbf{I}$ ) and

a common part (**H**) as illustrated in Figure 12. Harsmann and Lundy (1996) and Berge and Kiers (1996) have shown, that with these constraints the PARAFAC2 solutions are unique under some mild assumptions. For a more detailed description of the algorithm the reader is referred to the publication by Kiers *et al.* (1999).

As for all other curve resolution methods, the PARAFAC2 model must be made with the right number of factors in order for the obtained model to be chemically meaningful. The evaluation of how many factors there should be included is basically performed in the same way as for PARAFAC; the profiles of the obtained loadings, residuals, fit, and number of iterations are also for PARAFAC2 important parameters to take into consideration.

In paper II it is shown that core consistency, also for PARAFAC2, is a useful parameter for evaluation of whether too many factors have been included in the model. Core consistency can be determined for PARAFAC2 models by rearranging the PARAFAC2 model into a structure similar to that of PARAFAC, as illustrated in Figure 13. Core consistency can then be calculated on the PARAFAC model “inside” PARAFAC2.

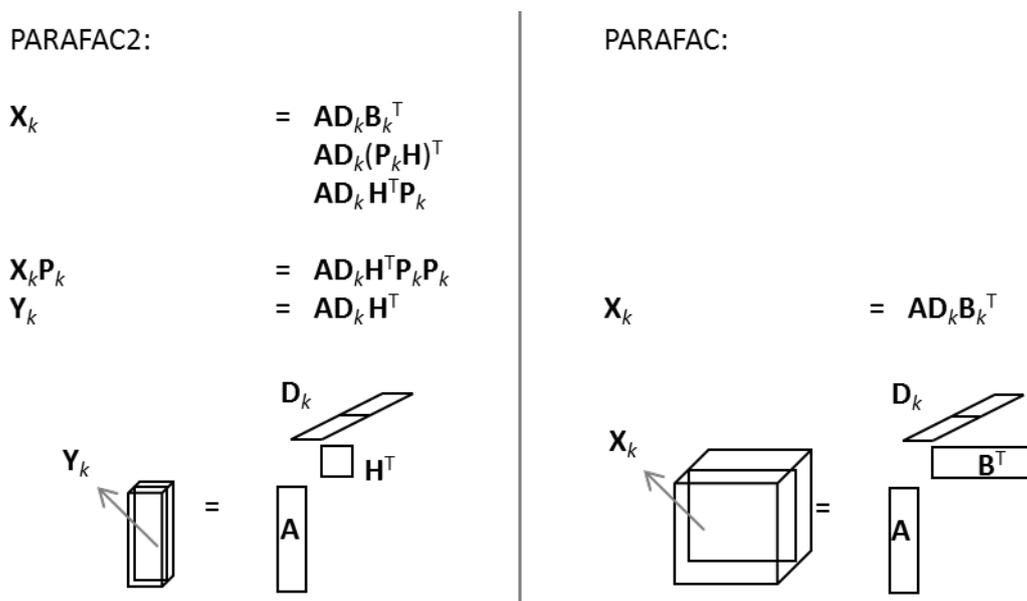


Figure 13. Illustration of how the PARAFAC2 model can be rearranged into a structure similar to a PARAFAC structure.

A more thorough description of how the right number of factors for a PARAFAC2 model is determined can be found in section 6.1. Here it will also be described how to use the automated evaluation (as proposed in paper IV). This method is based on classification of

when the model becomes over-fitted, with the use of a number of carefully selected diagnostics.

As for PARAFAC, sign indeterminacy is an issue for PARAFAC2 models. However, the issue is much more complex for PARAFAC2 since each sample has its own elution profile; this means that a single number in the  $K \times F$  matrix, **C**, can be flipped as long as the corresponding **p** vector (the sample specific part of the elution profile) is flipped accordingly. A proposal for a solution is described in paper III.

### 3.5 Alignment

In chromatographic analysis there will, inevitably, be some degree of shift of the obtained retention time between samples. Dependent on the chosen data processing and how severe the shifts are, it might be necessary to align the chromatograms prior to the extraction of information. Two very useful methods for correcting for such shifts are COW (Nielsen, Carstensen & Smedsgaard 1998) and *icoshift* (Tomasi, Savorani & Engelsen 2011). These methods will be shortly described in the following, but more detailed information can be found in the publications.

#### *COW*

The acronym “COW” means Correlated Optimized Warping. The principle is that all chromatograms in a batch are aligned towards a reference chromatogram, one chromatogram at a time. This reference must be carefully selected in order for the alignment to be successful. The reference can be selected automatically (Skov et al. 2006), but it is the user who must decide which type of reference to use. Different types of references could be the mean of the signals, the median of the signals, the bi-weighted mean of the signals, the maximum of the signals, or the maximum cumulative product of correlation coefficients. The first three reference types are very similar, they are all characterised by somewhat broader peaks than in the raw data. This is most pronounced in the mean signal, where the median signal is being a bit sharper. The bi-weighted mean, where the outliers are down weighted in the calculation of the mean, is an average between the two others, and results in a signal somewhere in between the mean signal and the median. The maximum signal is found by taking the maximum value at each data point (across all samples), this often results in a strange signal with very broad peaks, and is seldom the optimal reference signal. The maximum cumulative product of correlation coefficients is the only reference type, of those listed here, which results in a real chromatogram being used as reference, the reference is selected as the chromatogram which matches the others best.

After having chosen the reference, the alignment is performed by dividing the reference and the chromatograms, which are to be aligned, into a number of segments with flexible sizes. The degree of flexibility of each segment is controlled by the slack parameter, which indicates the maximum allowed change. After having tested all possible combinations of segment sizes, as allowed by the defined slack size, the combination of segment sizes which results in the best alignment is chosen. Segments which are of a different size than the corresponding segment in the reference are linearly interpolated to match the size in the reference. A consequence of this alignment procedure is that the flexibility is large in the centre of the chromatograms and lower at the end points. In cases with too low flexibility at the end points, segments of noise can be added before the alignment (Nielsen, Carstensen & Smedsgaard 1998).

In order to achieve the best possible alignment, it is necessary to optimize the two parameters, segment and slack, in such a way that there is enough flexibility for the algorithm to correct for the shifts, but not so much flexibility that misalignment occurs or the peak shapes changes. In the publication by Skov *et al.* (2006) a procedure is presented which, in an automated way, optimises these parameters. The optimisation algorithm tests combinations of slack and segment sizes, within a given range, by performing a simplex-based search for the optimal (or near optimal) combination. Two parameters are used to evaluate the performance of the alignments; simplicity and peak factor. The so-called simplicity value is the singular values taken to the fourth power, a high simplicity indicates that a lot of the variation is described by the first components, and hence the numerical rank of the system is low. The simplicity is used to evaluate how well aligned the chromatograms are. The peak factor is a measure for how much the area under the peaks is changed by the alignment, and is included to ensure that the peak shapes are preserved. For a more comprehensive description of the optimisation procedure the reader is referred to the original paper. In section 5.2 a more thorough description of how to use the automated procedure will be given. By using the optimisation procedure, the alignment of chromatograms can be performed with a very limited amount of manual work, even though the optimisation algorithm is requiring a significant amount of computation time.

Below an example of the performance of COW is given; this has been included so the reader can compare the procedure with *icoshift* which is described in the next section.

In Figure 14 it is shown how COW performs on a dataset with significant shifts in retention time. After alignment there are very little amounts of shifts in the majority of the chromatogram. However, in the area marked with a box in Figure 14B, only two compounds

are eluting, but three peak groups are created by COW. Since this misalignment is located in the beginning of the chromatograms, it can be solved by adding blocks of noise to each end of the unaligned chromatogram and subsequently aligning. In this way also this peak group is correctly aligned (area marked with box in Figure 14C).

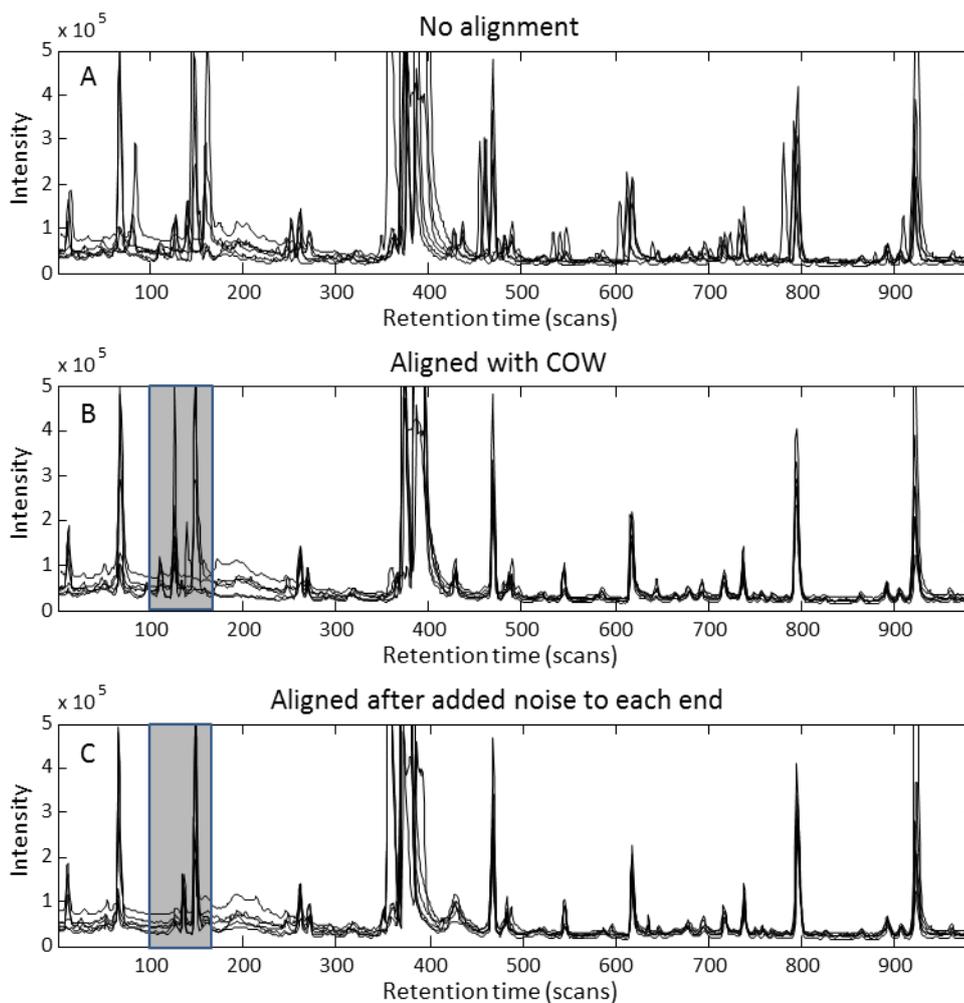


Figure 14. A: Raw data with varying degrees of shifts. B: After alignment with COW (segment length: 30, slack: 19), the automated procedure for determination of the optimal segment length and slack has been used. C: After alignment of the raw data with 50 data points of noise added to each end. Also here the segment length was 30 and the slack 19.

The main advantage of COW is that the procedure is completely automated, and that it can be used on chromatograms containing very few peaks, as well as in very complex cases. The

main disadvantages is the computational time, and more severe, the risk of changing the shape of the peaks if the slack size is set too high.

The COW algorithm, as well as algorithms for selection of references and determination of the optimal segment and slack sizes, is available from [www.models.life.ku.dk](http://www.models.life.ku.dk)

#### *icoshift*

*icoshift* means “interval correlation optimised shifting algorithm” and, as the name indicates, it is aligning the chromatograms piecewise, by aligning smaller intervals of the chromatograms towards a reference. The main difference from alignment with COW is that the intervals (or segments) are not stretched or squeezed. The alignment is simply achieved by moving the complete interval back or forth until optimal alignment is achieved relative to the reference. Therefore it is also for *icoshift* important to select a proper reference, but as opposed to COW, the intervals do not necessarily need to be aligned against the same reference chromatogram. This means that for each interval the reference chromatogram which is most optimal for that particular case can be chosen. The reference could be the mean of the signals, the median of the signals, or the specific chromatogram with the highest intensity in the individual intervals. In my experience, the latter, in most cases, gives the best result.

The optimal alignments are archived by initially aligning the total chromatogram, and then, subsequently, dividing the chromatograms into smaller intervals, which are then aligned individually. The intervals can be created by either defining a constant segment length, or by user defined intervals.

Below *icoshift* is used to align the same chromatograms as aligned in the section describing COW (above).

In Figure 15 the effect of both a full chromatogram alignment (B) and subsequently piecewise alignment on user defined intervals (C) are shown. When dividing the chromatograms into intervals, the boundaries must be selected in regions with baseline separation to ensure that no changes in peak shape occur. The example, in Figure 4, shows, how the alignment of the full chromatogram can ease the creation of the intervals. Especially the intervals marked with I and II could not have been created without the initial alignment, since the peaks of the two different regions were overlapping in the raw data.

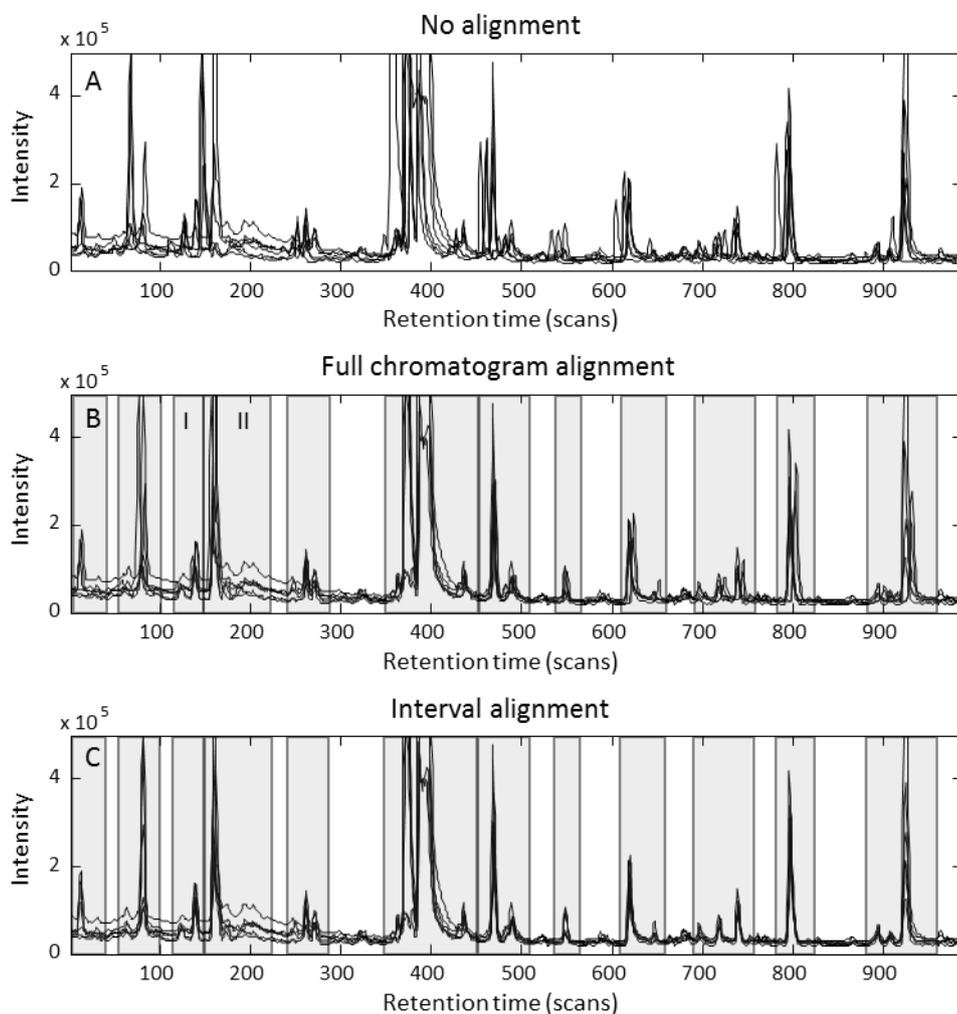


Figure 15. A: Raw data without any alignment. B: After alignment of the full chromatogram with *icoshift*. The gray areas indicate the interval which is used in the next step. C: After full *icoshift*, the chromatograms are now completely aligned.

The main advantage of *icoshift* is that it is a very fast algorithm, and that the peak shape is kept constant if the boundaries, of the intervals, are selected in regions with baseline. However, in cases with very complex chromatograms and no, or very few, regions with baseline, it can be very difficult to create the intervals in a reasonable way. In these cases *icoshift* may not be the right choice.

The *icoshift* algorithm for MATLAB is available from [www.models.life.ku.dk](http://www.models.life.ku.dk)



## Chapter 4

# Which methods to choose

---

Before a new set of data can be analysed in a proper way, the right techniques must be chosen. This choice is of course very dependent on the nature of data at hand, and for some it may seem difficult to select the right methods. This chapter will guide the reader to make the right decisions.

The first step in the data processing is to separate the signal representing the analytes of interest from the remaining part of the signal (typically baseline + noise). When deciding which pre-processing technique to use for this, it is of course important to take the structure of the data into account. As described in section 3.1 the second order advantage, obtained by using multi-channel detectors, will only be achieved if the right tools are applied. Furthermore, the complexity of the data is important to take into consideration, when it is chosen which methods to apply. A thorough guide on how to choose the right method for extraction of the pure signal is given in the first section (4.1) of this chapter.

Once the pure signal, representing the analyte, has been found, one needs to decide whether height or area should be used for quantification. Especially for single-channel data obtained from GC analysis this is an important discussion since height, in this case, is a very useful alternative to area; this issue is further discussed in section 4.2.

### 4.1 Choosing the right pre-processing

No matter what the aim with the analysis is, artefacts, like baseline and shifts, should be handled in one way or the other. If these artefacts are not dealt with, the final conclusions may be severely influenced by these non-informative variations. The actual choice of pre-processing techniques is to a high degree dependent on the purpose of the analysis and the nature of data. The pre-processing should at least be able to handle different baseline issues, but also some kind of alignment, or grouping of peaks across samples should be performed in order to be able to determine which of the peaks, in the different samples, that originate from the same compounds.

The diagram, in Figure 16, shows how different types of data are treated in the best way. Please be aware that the methods described in this thesis are only designed for cases where several samples are included in the dataset. In cases where data represents only one sample, the initial considerations will be different from those described below.

The first thing to consider is whether data is two-way or three-way (or higher). The treatment of two-way data will be described in chapter 5, whereas three-way data will be described in chapter 6. However, if data is three-way, but so complex that it is impossible to divide the chromatograms into smaller baseline separated intervals, data could be summed over masses (in case of MS data) and treated in the same way as very complex two-way data (as described in section 5.2). In section 6.3 more details about the different possibilities for very complex three-way data will be described.

For three-way data which can be divided into smaller baseline separated sections, it is recommended to try to resolve the individual parts with PARAFAC2, as described in section 6.1. In some cases PARAFAC2 will be unable to provide a decent solution. Such cases could be if data contains peaks which are totally overlapping, changes in peak shape, or other factors which cause the trilinearity of the data to be lost. In these cases MCR might be the right choice. However, MCR has other problems, e.g. that the right constraint must be applied in order for the solution to become unique. Therefore it requires a lot of knowledge about the samples (or data) in question to use MCR, and it makes the computation of the models much more demanding of hand held adjustments. Furthermore, there is a risk of over-fitting; this is not a problem with trilinear models like PARAFAC2. From this point of view, it is recommended to first try to use PARAFAC2, and if this does not provide a useful solution, then try to apply MCR. Modelling with MCR is further described in section 6.2.

For two-way data with frequent occasion of baseline it is recommended to use the methods incorporated in FastChrom, which is a graphical user interface running in MATLAB. This approach is carefully described in section 5.1.

Finally for the very complex two-way data with many overlapping peaks and rear occasions of baseline separation, it is recommended to use PCA or MCR on the complete chromatograms as described in section 5.2.

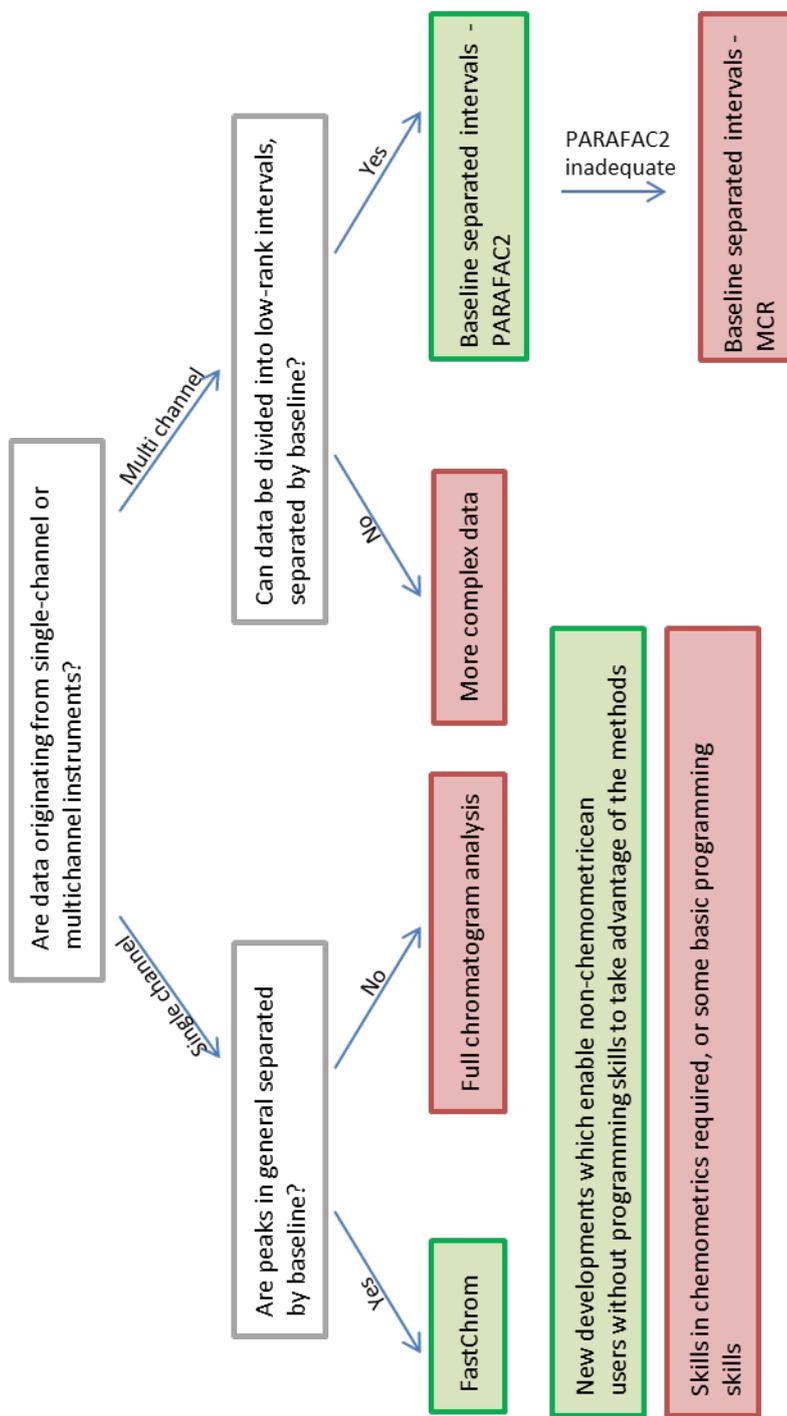


Figure 16 The diagram shows an overview of the different types of data and how to treat them, when several samples are included in the dataset. Cases with only one sample are not described in this thesis.

## 4.2 Area vs. height

In GC is the peak width dependent on the chemical compound and not concentration. This means that peak width is constant (under normal circumstances). Therefore, also the height can be used as a measure of the concentration. In cases where peaks are well separated, and the baseline is of limited influence, height and area are equally good measures of concentration. However, often peaks will be co-eluting (example A, B, and C in Figure 17), or the baseline will not have been determined correctly (example D, E, and F in Figure 17). In these case height will be a more robust measure than area, as shown by Bicking (Bicking 2006a, Bicking 2006b). In Figure 17 it is shown how different problems will influence respectively area and height (for a more thorough review on the topic, see the publications by Bicking).

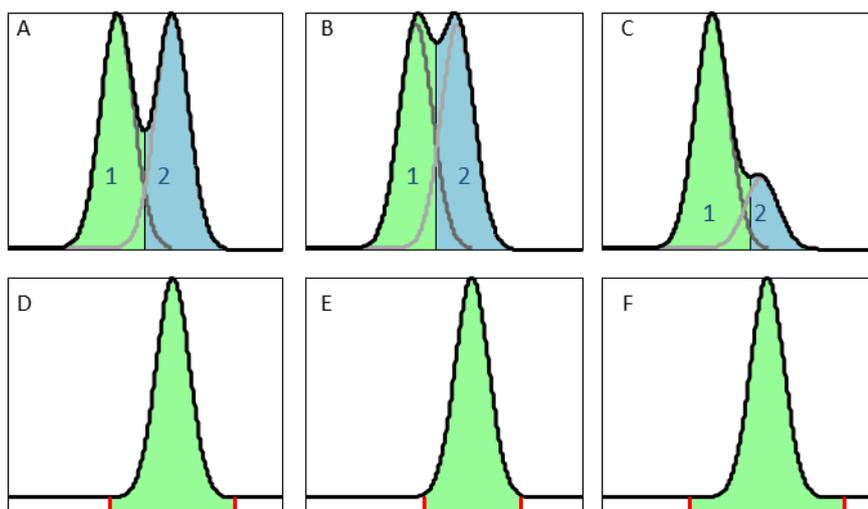


Figure 17 Illustration of different problems affecting the estimated height and area of the peaks. A-C illustrates different situations of fused peaks. D-F illustrates situations where baseline not has been successfully removed. The three examples also show how different determinations of the peak boundaries affect the areas. The filled areas indicate the peak area.

The influence of the different issues, illustrated in Figure 17, is shown in Table 1. The table shows that for situation A and B, where the two peaks has the same size, both area and height are unaffected by the fusion of the two peaks. However, in situation C, where one of the peaks is considerably smaller than the other, height is noticeably more robust than area. This is also the case in situation D to F, where a baseline is included in the signal. The baseline results in an overestimation of approximately 20% with regards to the area, whereas the

---

height is overestimated with less than 10 %. Furthermore, is the overestimation, of the area, depended on that the boundaries of the peak to be correctly estimated. In the example shown here changes of the boundaries results in a 10% change of the determined area. The estimation of peak boundaries is not affecting the determined heights.

A back draw for height is that if the column becomes overloaded it results in broadening of the peaks. In these cases will the height no longer be directly proportional with concentration.

Table 1. Illustration of the effect of different problems with peak estimations. The height and area are indicated as percent compared to the true value.

<i>Example</i>	<i>Peak</i>	<i>Height (%)</i>	<i>Area (%)</i>
A	1	100	100
	2	100	100
B	1	101	99
	2	101	101
C	1	100	99
	2	105	80
D		108	122
E		108	117
F		108	128



*Chapter 5***Single-channel data**

---

As described in chapter four, the choice of pre-processing technique dependent is on the nature of the data at hand. The present chapter is describing how to handle data obtained from first-order instruments. It is divided into two subsections which respectively describe how to pre-process chromatograms where the peaks are mainly well separated (section 5.1), and where they are not (section 5.2). There is of course a grey area in between these two categories. If in doubt about which of the methods to choose, it is recommended to first test the method for well behaving chromatograms (FastChrom, section 5.1), and subsequently, if the result is not satisfying, the method for analysis of full chromatograms (section 5.2).

**5.1 FastChrom**

The methods described in this section require that the chromatograms are “well behaving”, with this is meant that shifts within the experiments are minimal, the peak shapes are approximately Gaussian, and the peak width for the individual compounds relatively constant. If these requirements are met, the methods incorporated in FastChrom can be applied successfully.

FastChrom is a graphical user interface which runs in MATLAB. It consists of baseline estimation for each sample, peak identification and validation, and peak grouping across samples. In addition to these features it is possible to assign retention time indexes to the peaks. The use of indexes, instead of raw retention times, eases identification and comparison between different experiments. The different parts of FastChrom are described and critically evaluated in paper I. In the following a more thorough description of how to optimise the different parts of FastChrom will be presented.

The chromatograms illustrated in Figure 18 are used throughout this section. The chromatograms are relatively noisy and have a significant baseline which needs to be removed before quantification of any compounds is possible.

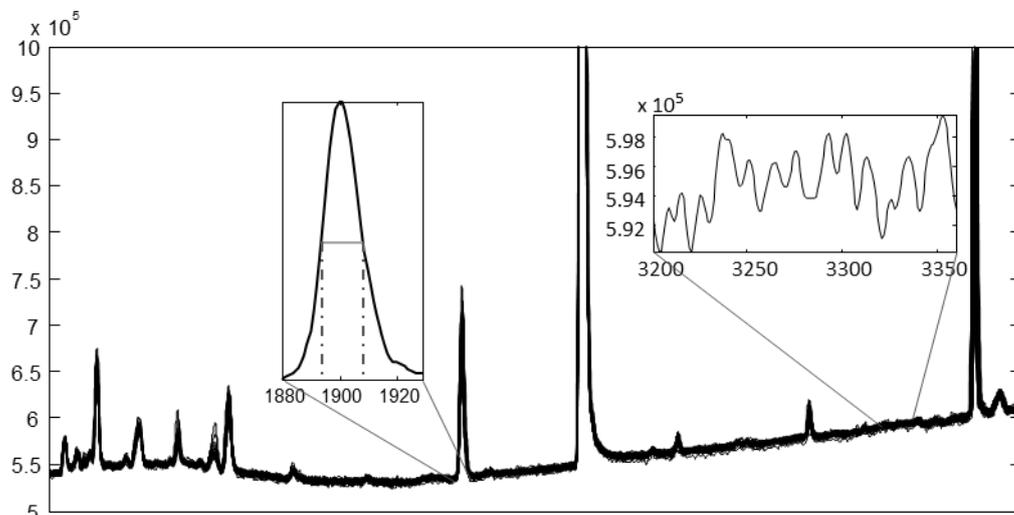


Figure 18. If the baseline estimation does not take the hump in the baseline into consideration, it will be of significant consequence for the quantification of the peaks eluting in that interval. The zoom to the left shows a representative peak from one of the samples, the peak width is 16 data points. In the zoom to the right an area without peaks is shown. It can be seen that the noise has a level of approximately  $7.5 \times 10^3$ .

Before FastChrom can be used an Excel-file must be created. The file must contain information about which of the samples that are control samples, Kovats Index samples (KI), and real samples, in a sheet named "Samples". If KI samples are included the Excel file must also contain information about the index of these samples in a sheet named "KI". The layout of these sheets is shown in Figure 19.

"Samples" sheet						"KI" sheet				
A	B	C	D	E	F	A	B	C	D	
1			Sample #/Name	Series	Classes	1	Compound	KI	RT	Window
2	Codes for the "series" column					2	Hexane	600	0.98	0.03
3	1	Control samples	KI		2	3	Heptane	700	1.0425	0.03
4	2	KI samples	1		3	4	Octane	800	1.1695	0.05
5	3	Exp	2		3	5	Nonane	900	1.4285	0.05
6			3		3	6	Decane	1000	1.956	0.05
7			4		3	7	Undecane	1100	2.7235	0.05
8			5		3	8	Dodecane	1200	3.3905	0.05
						9	Tridecane	1300	3.9485	0.05

Figure 19. The "Samples" and "KI" sheets which must be created before FastChrom can be used. The "Samples" sheet contains information about the names and nature of the samples. The "KI" sheet contains information about the index of the compounds in the KI standards, their retention time, and how much the retention time can vary. KI: Kovats Index.

In FastChrom there are a number of settings which must be defined. In Figure 20 the settings box is shown. The two settings which are unmarked indicate the part of the chromatogram which is being evaluated by FastChrom and the scan frequency in scans/min. It is only necessary to provide this last setting if the file format is RAX (which is the output format from Perkin Elmers GC-FID instruments). If the file format is CDF, MAT or XLS, the scan frequency is calculated by FastChrom.

Setting	Value
Thres	3300
Noise window	16
V1	10
V2	30
Scan frequency (scans/min)	750
Smoothing window	8
Shift	13
Min. peak height	22300
From	0 min
to	6.2 min

Figure 20. The “Settings” section in FastChrom. Red square: Settings affecting baseline. Green square: Settings affecting peak identification. Blue square: Settings affecting peak grouping

The first step in handling the chromatograms is to determine a baseline. The baseline estimation which is used in FastChrom is a new development and is thoroughly documented in paper I. It starts by identifying areas without peaks, in these areas the raw data are used directly as baseline. In areas with peaks the baseline is estimated in an iterative way with local linear regression.

FastChrom differentiates between areas with and without peaks by determination of the standard deviation in small segments. Areas with standard deviation below a user set threshold (“Thres” in Figure 20) are defined as regions without peaks. Besides the threshold the user needs to define the width of the windows used to calculate the standard deviation across the individual chromatograms (“Noise” in Figure 20).

The width of the window should be set first since it will affect the determination of the threshold. It should be approximately similar to the peak width (at half height) obtained in the chromatograms. By zooming in the plot of the raw data, one can simply count how many data points one peak spans in half height. In Figure 18 a zoom is shown with a peak with a

width of 16 data points. By inspecting a few other peaks, it is found that this is a representative peak width for this data set. The “Noise” setting is therefore set to 16.

After having performed an initial calculation with the “Noise” setting settled, the “Thres” can be determined by inspecting the standard deviation for a few chromatograms. When the “Thres” is determined it is important that it is the smallest peaks which are inspected. The inspection can be performed by clicking the “Thres” button in FastChrom.

The determination of these two settings (“Thres” and “Noise”) only needs to be done once for each application (this means that if the same type of samples are analysed, with the same GC method, on the same instruments, the same settings can be used).

The next step is to locate all peaks in the chromatograms. The settings affecting this are maximum and minimum peak width, minimum peak height, and the smoothing window (marked with green in Figure 20). The smoothing is only necessary in cases with a relatively high noise level, which is the case in the example used here. The width of the smoothing window is recommended to be set to one half of the peak width, this ensures that enough noise is removed so that the peaks become smooth, and it will not severely affect the peak height. If the smoothing window is chosen with this rule in mind, there is a very little likelihood that smoothing will have a negative effect, so if in doubt smoothing should simply be applied. If one does not want to apply smoothing, the smoothing window should be set to one.

The minimum and maximum peak widths and minimum height are included to ensure that noise is not identified as peaks. A recommendation for the minimum peak height could be three times the noise level, since this is commonly used as the detection limit. The level of noise can be determined by a closer inspection of regions with baseline in the raw data. In the example shown in Figure 18, the noise level is approximately  $7.5 \times 10^3$  and a recommendation for the minimum peak height, in this case, would therefore be  $22.3 \times 10^3$ . If it does not matter if noise is reported in the final peak list, the minimum peak height could be set to zero.

The minimum peak width is included to ensure that spikes are not identified as peaks. The minimum peak width is important since such a spike can be of considerable height but it is always narrower than an ordinary peak originating from a real chemical compound. This feature can only be determined based on knowledge of the data. The minimum peak width should be somewhat lower than the observed peak widths, and in the example used in this case a minimum peak width of 10 is reasonable. The maximum peak width is not a critical

parameter, and should just be set to be considerably higher than the normal peak width. In the example it has been set to 30.

Also these settings only need to be optimized once for each application.

Finally the peaks should be grouped across samples in such a way that peaks which are originating from the same chemical compound are clustered in the same group. This is done by defining the width of a grouping window (which must be odd). It is recommended to inspect the deviation of peaks in the dataset in order to determine the width of this grouping window. This could be done by performing a calculation with the settings already determined, and exporting the result to Excel. In the resulting Excel file the original retention

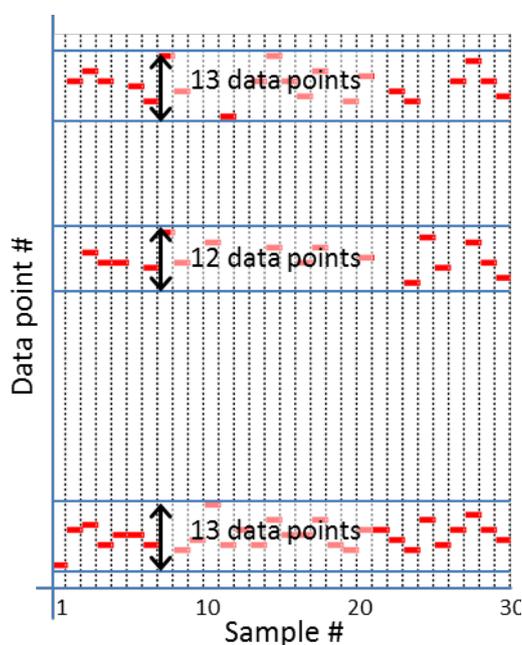


Figure 21. Distribution of peaks in the 30 samples and the two KI standards. The three peak clusters have internal shifts on respectively 13, 12, and 13 data points.

time of all peaks can be found in the sheet “Peaks”. The deviation of three representative peak groups is shown in Figure 21. In this case there are two peak groups with a deviation of 13 data points and one peak group with a deviation of 12 points. The “shift” parameter is therefore determined as 13. It is not always so clear what the optimal shift is, in such cases it is recommended to choose a width which is a bit too small rather than too wide, since it is easier to recognize peaks which should have been in the same group and manually merge these, than the other way around.

The width of the grouping window should be checked by a quick inspection of the results for every new batch of samples.

## 5.2 Full chromatogram analysis

In cases with very complex data, the methods incorporated into FastChrom are insufficient. Indeed other methods could be used for baseline estimation and peak identification. However, when data are obtained from first-order instruments it is very difficult to resolve overlapping peaks in a robust and trustworthy way. My recommendation would therefore be to apply the approach suggested by Christensen *et al.* (Christensen, Tomasi & Hansen 2005, Christensen et al. 2005, Christensen, Tomasi 2007, Petersen et al. 2011). This method does not aim at being able to quantify the individual compounds, but rather to investigate which compounds that vary across samples, and to identify main characteristics of the samples. The idea is to remove baseline contributions and shift from the data, and subsequently use multivariate data analysis to identify relevant information. Below are the individual steps in the procedure presented, and in Box 1 the relevant MATLAB code is shown.

Throughout this section the chromatograms, shown in Figure 22, are used to illustrate the effect of the different steps. The chromatograms are TIC obtained from GC-MS analysis of beer. Even though the chromatograms represent data obtained from a second-order instrument, they will be treated here as if they were originating from a first-order instrument.

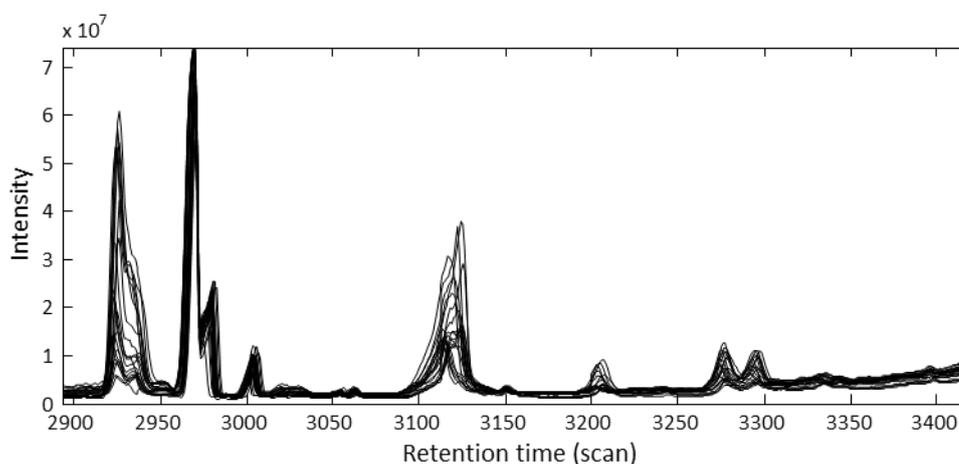


Figure 22. The figure shows a section of 26 TIC obtained from GC-MS analysis of beer.

The first step in the procedure is to remove the baseline. This step is carried out before the alignment since differences in baseline may affect the alignment in a negative way (Nielsen, Carstensen & Smedsgaard 1998). The removal of baseline contributions is performed by calculating the first derivative. However, this may increase the noise level in data. To

counteract this, smoothing can be applied e.g. with Savitzky-Golay (Savitzky, Golay 1964) to minimize the noise level in the derivatives. Smoothing with Savitzky-Golay is obtained by local polynomial regression to small segments of the original curve. The main advantage with this approach is that peak features such as minima, maxima and width are preserved to a high degree. The width of the windows and the order of the polynomial can be changed so it fits the noise level and complexity of the raw data, e.g., high noise levels require a wider window, while high complexity in data requires a higher order polynomial.

In Figure 23 the derivatives obtained from the beer chromatograms are shown. The derivatives have been calculated after smoothing with Savitsky-Golay (width 15 points, third order). The width of the window and the order of the polynomial were found by inspection of smoothed data with a number of different settings. The setting which resulted in the smoothest signal without removal of small peaks was chosen. The figure shows how the baseline, which appeared in the raw chromatograms, is now removed. Since the derivatives are equal to the slope of the original curve, a peak from raw data will appear as a double peak with a positive part (positive slope in the original peak) and a negative part (negative slope in the original peak). The intersection with zero is located at the same retention time as the apex of the original peak. In the zoom in Figure 23 the appearance of such a double peak is shown.

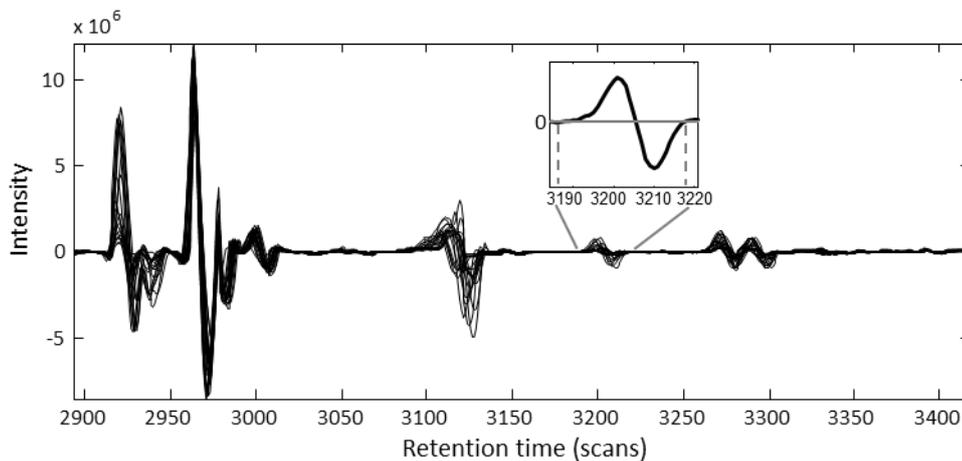


Figure 23. Illustration of the derivatives calculated on the beer chromatograms after smoothing with Savitsky-Golay (width 15 points, third order) has been applied. The zoom shows the appearance of a typical peak in the derivatives (only one chromatogram is shown in the zoom). The dotted lines in the zoom indicates the boundaries of the double peak, the peak width here is 29 points.

The next step is to align the chromatograms. In cases with severe shift throughout the chromatograms, it is recommended to initiate with an alignment of the entire chromatograms with *icoshift*, and subsequently align with COW. However, for the data used here, the shift is not so severe, and alignment is performed using only COW. In Figure 24 the appearance of five different types of references is shown. Type #4, maximum signal is characterised by having very broad peaks, and by not being able to describe the negative parts of the peaks appropriately. References #1, #2, and #3 (mean signal, median signal, and bi-weighted mean signal) are three different ways of finding the average signal, and perform somewhat better than criterion #4. However they are all characterised by having broad peaks (right in figure) and small peaks are insufficiently represented in these references (indicated with arrows in the left part of the figure). In this specific case criterion #5 (maximum cumulative product of correlation coefficient) is best suited as reference, and it is my experience that this method, in general, is the best choice. For a more detailed description about the different reference types shown here, see section 2.7.

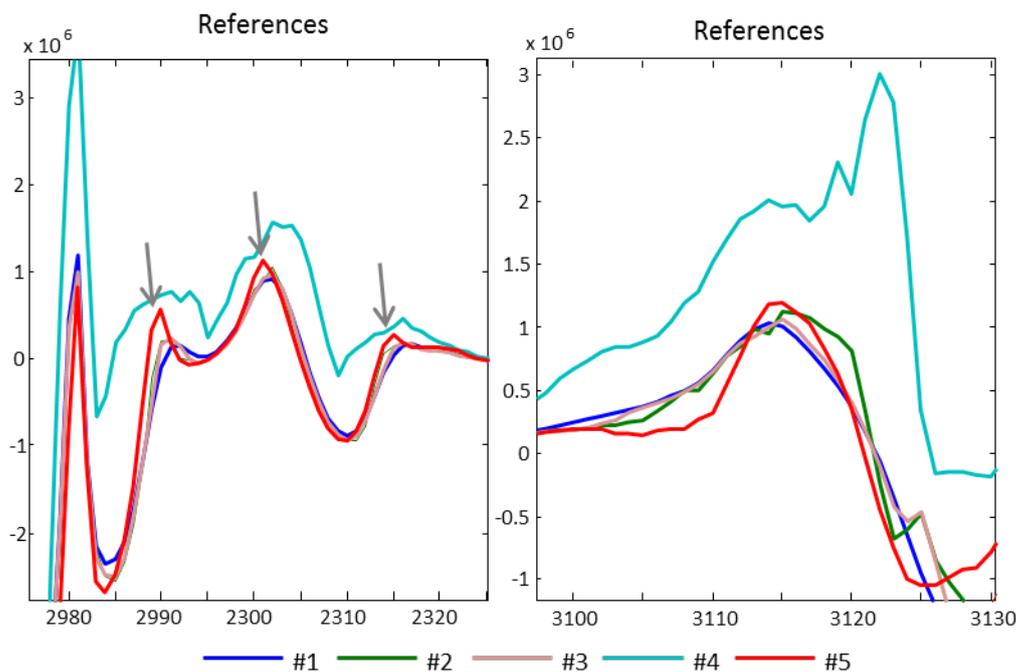


Figure 24. Illustrations of five different types of references; #1: mean signal, #2: median signal, #3: bi-weighted mean signal, #4: maximum signal, and #5: maximum cumulative product of correlation coefficient.

In section 2.7, it is mentioned that the optimisation of the segment length and slack size can be performed in an automated way. However, some boundaries for the search space must be defined. In the publication by Skov *et al.* (2006), which originally described the automated optimisation, it was recommended to set the boundaries of the segment length to the average peak width at baseline  $\pm \frac{1}{2}$  peak width. As shown in Figure 23 a typical peak in the beer data, has a width of 29 data points, the boundaries for the segment length were therefore set to 15 and 45. The boundaries for the slack size were, in the original publication, recommended to be set to 1 and 15, and then adjusted if the optimisation resulted in a slack size near 15. Therefore the boundaries for the slack size were set to 1 and 15. In the beer data, the shift was most severe in the middle part of the chromatogram, and pads of noise were therefore not added to the ends of the chromatograms before alignment. By using the automated optimization of segment length and slack size it was found that a segment length of 41 and a slack size of 13 were appropriate for the data at hand. In Figure 25 the result of the alignment is shown. It may seem like the middle part (around scan point 3100) has not been properly aligned. However, upon a closer inspection it can be seen that there are actually more than one peak in this area (zoom not shown).

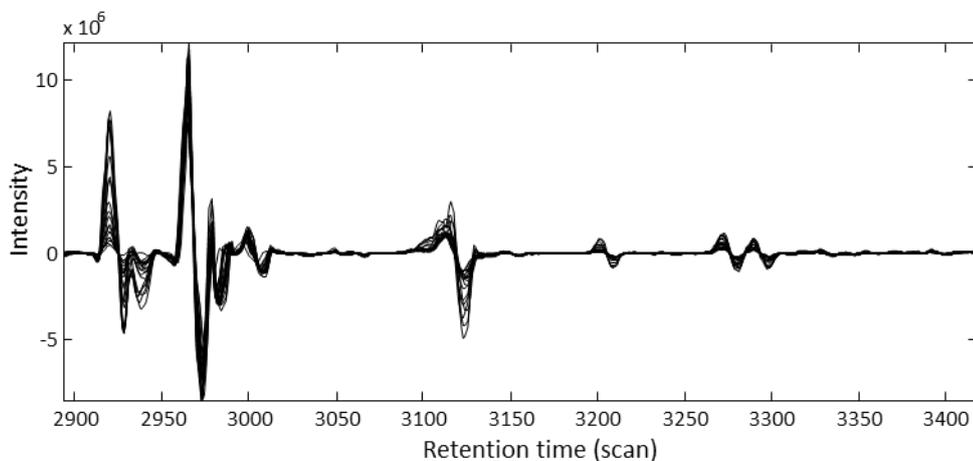


Figure 25. Derivatives aligned with COW. Segment length (41) and slack size (13) were found by the automated optimization procedure, the chromatogram with the highest correlation to the others was chosen as reference (reference type #5).

After having removed baseline contributions and solved the issues with shifts in retention time, some normalization should be applied before analysing with PCA. This thesis will not cover this subject, but in the publication by van den Berg *et al.* (2006) different types of normalization techniques are carefully described.

As an alternative to PCA, MCR can be applied. However, due to the non-negativity constraint in MCR, derivatives cannot be used; therefore other types of baseline removal should be applied. One alternative approach could be to fit a polynomial to the bottom of the chromatogram, or to regions specified as baseline regions.

Which one of the two methods (PCA or MCR) to choose, depends on the purpose with the experiment. It is out of the scope for this thesis to describe how these analyses are performed, but in the work done by Svendsen *et al.* (In preparation) it is described how the results can be interpreted, and it is shown that if the samples represent time series, e.g. samples are taken out at different time points during a fermentation course, then MCR is in general superior to PCA. If, on the other hand, one wishes to explore differences between groups of samples, a PCA analysis will most likely give the best result.

To ease interpretation of the PCA or MCR results, the cumulative sum of the loadings can be calculated to restore the chromatographic appearance of the peaks.

#### Box 1. Pre-processing of the full chromatograms

The code below shows how to import single-channel data into a matrix (x) with samples in mode 2 and retention time in mode 1. It is assumed that data is available in an Excel file, named "raw.xlsx" with retention time in the first column and the intensities from samples 1 to N in column 2 to N+1, in a sheet named "data".

```
x      = xlsread('raw.xlsx', 'data');  
rt     = x(:,1);  
x(:,1) = [];
```

TIC from GC-MS or LC-MS can be imported as shown in Box 2, if this is the case "x" should be replaced with "tic" in the code for smoothing below.

The code below shows how to smooth data with Savitsky-Golay with a window width of 15 and a third order polynomial. The effect of smoothing should be checked, therefore the code for plotting of the smoothed signal is included.

```
x_hat = savgol(x',15,3); % Replace x with tic here  
plot(x_hat')
```

Calculation of derivatives:

```
d = diff(y_hat');
```

Box 1, continued. Pre-processing of the full chromatograms

In cases with severe shift, full chromatogram alignment can be performed with the following code which applies *icoshift* with the maximal signal as reference:

```
[dCS,ints,ind,target] = icoshift('max',d', 'whole','b', [2]);
```

Alignment with COW can be performed with the following lines of code. After the first line of code has been evaluated, the user will be asked for which reference method the algorithm should use to select a proper reference. The method should be chosen by inspection of the created plots which illustrate the different references.

```
[ref,refs,N] = ref_select(dCS'); % Replace dCS with d in case icoshift  
has not been applied  
[par,OS,diag] = optim_cow(ds', [15 45 1 12], [1, ref]);  
[W,XW,Diagnos] = cow(ref,ds', par(1), par(2));  
plot(XW')
```



*Chapter 6***Multi-Channel data**

---

In this section it is described how to handle data originating from multi-channel detectors. The first two sections describe how to handle data where the obtained chromatograms can be divided into sub-sections with no more than 5-6 compounds eluting in each. The first section is focused on how to use PARAFAC2 for these kinds of data. This approach is recommended in general, while it, in some special cases, can be necessary to apply MCR as an alternative. This approach is described in section 6.2.

Throughout sections 6.1 and 6.2 small boxes with thorough guidelines on how to perform the individual steps in MATLAB are included. By simply copying those steps and, subsequently, pasting them into the MATLAB command window, it should be possible for those which are not familiar with MATLAB to take advantage of the methods described.

Section 6.3 gives some very brief suggestions of how to treat data where it is impossible to divide the chromatograms into smaller baseline separated intervals.

**6.1 Baseline separated intervals - PARAFAC2**

The first step is to make a visual inspection of the raw data. If this is done in MATLAB, the data must first be imported and plotted as described in Box 2. It is a prerequisite that the data is available in CDF format (also denoted netCDF). For a description of this format see the publication by Rew and Davis (Rew, Davis 1990).

After plotting the TIC, the intervals should be defined. This step can with some success be performed in an automated way by using the AI-Binning algorithm developed by De Meyer *et al.* (De Meyer *et al.* 2008). However, in my opinion, it is just as easy to do it manually since the intervals created by AI-Binning should be manually validated anyhow and then the time saved by the automation is quickly spent. In Figure 26 a section of 55 TICs is shown which could be divided into three intervals as indicated in the figure.

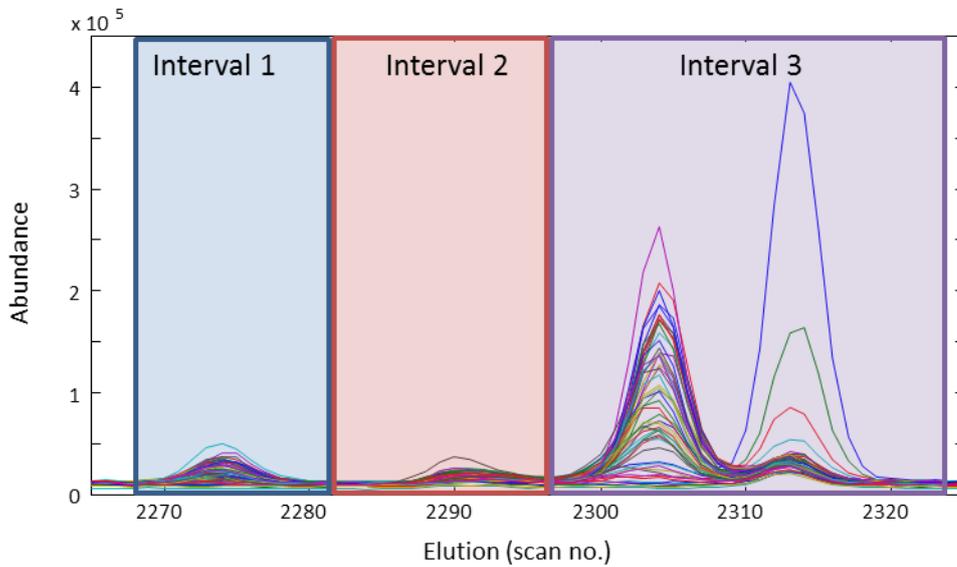


Figure 26. A small section of 55 total ion chromatograms. In this case three intervals could be created: 1) 2267 to 2283, 2) 2283 to 2296, and 3) 2297 to 2327

After dividing the chromatograms into low-rank-intervals PARAFAC2 should be applied and evaluated on each of these separately, either by manually evaluating the models, or by using the automated approach described in paper IV. Here both approaches will be described in the two sections below, using the intervals from Figure 26 as examples. In both approaches a modified version of the PARAFAC2 algorithm is used, the original algorithm is available from <http://www.models.life.ku.dk>. In Box 2 it is shown how to modify the PARAFAC2 algorithm.

Box 2. Import, initial visual inspection of raw data, and modification of the parafac2 algorithm.

Make sure that the “current folder” is the folder containing the raw data in CDF format.

The following lines of command will import data and create a three way array (M) containing the raw data (mass channels in first mode, elution time in second mode, and samples in third mode), a two way matrix (tic) containing the TIC of all samples (with elution time in first mode and samples in second mode), and two vectors (mz and rt) containing respectively the mass and elution time axes. After data have been imported the TICs are plotted.

The command requires that the iCDF function (available from <http://www.models.life.ku.dk>) is installed. The function only works on 32-bit computers; alternatively a function working on 64-bit computers can be obtained from the author upon request.

```
files = dir('*.CDF');

for i = 1:length(files)
    [m, t, rt, mz] = iCDF_load(files(i).name);
    M(:,1:size(m,2),i) = m;
    tic(:,i) = t;
end
M = permute(M, [2 1 3]);
keep M tic mz rt

plot(tic)
```

#### Modification of parafac2.m

The function must be present in either the MATLAB path or in the “current folder”. In the “command window” write:

```
edit parafac2.m
```

Now the function will appear in the “editor” with the first line as follows:

```
function [A,H,C,P,fit,AddiOutput] =
parafac2(X,F,Constraints,Options,A,H,C,P);
```

Insert “it” and save the modified function:

```
function [A,H,C,P,it,fit,AddiOutput] =
parafac2(X,F,Constraints,Options,A,H,C,P);
```

*Manual evaluation of PARAFAC2 models*

The principle behind the manual evaluation of how many compounds to include is to calculate and evaluate a number of models with increasing number of factors. This number should be increased until the model with one factor too much has been found.

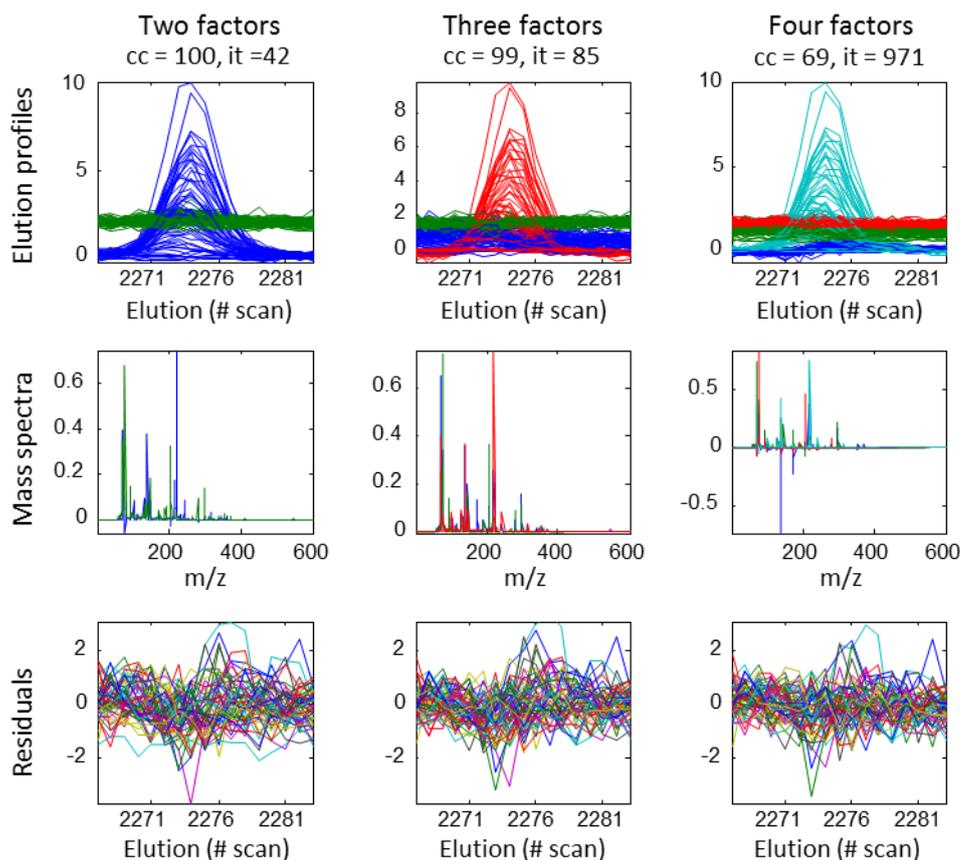


Figure 27. Elution profiles, mass spectra profiles and residuals from interval 1 modeled with PARAFAC2. The optimal model is the model with three factors. cc; Core consistency. it; iterations

The first step is to make an initial guess of how many factors to include. In interval one at least one factor should be included plus, probably, one to describe the baseline. A reasonable initial guess would therefore be to make a PARAFAC2 model with two factors. Upon inspection of this model (leftmost in Figure 27), there is nothing which really indicates that the model should not be the optimal. There are some small negative values in the obtained spectra, but these values are very low. In order to be sure that two factors result in the optimal model, a model with three factors is made. This model is illustrated in the middle in Figure 27. In this model the small amount of negative values in the spectra are absent, and

there is no indication that the model has too many factors included. So even though it does not seem like additional information is found (the extra component merely describes an additional baseline) the model seems more appropriate than the two-factor model. In order to control that no additional factors should be included, a four-factor model is created. This model shows clear signs of over-fit; core consistency is somewhat lowered, the number of iterations are considerably increased, and there are high negative values in the spectra profile (rightmost in Figure 27).

When all the different aspects of the three models are taken into account, it may be concluded that interval one is best described with three factors. However, the two-factor model could also be used, since spectra and concentration profiles obtained from the two-factor and three-factor models are very similar (not shown).

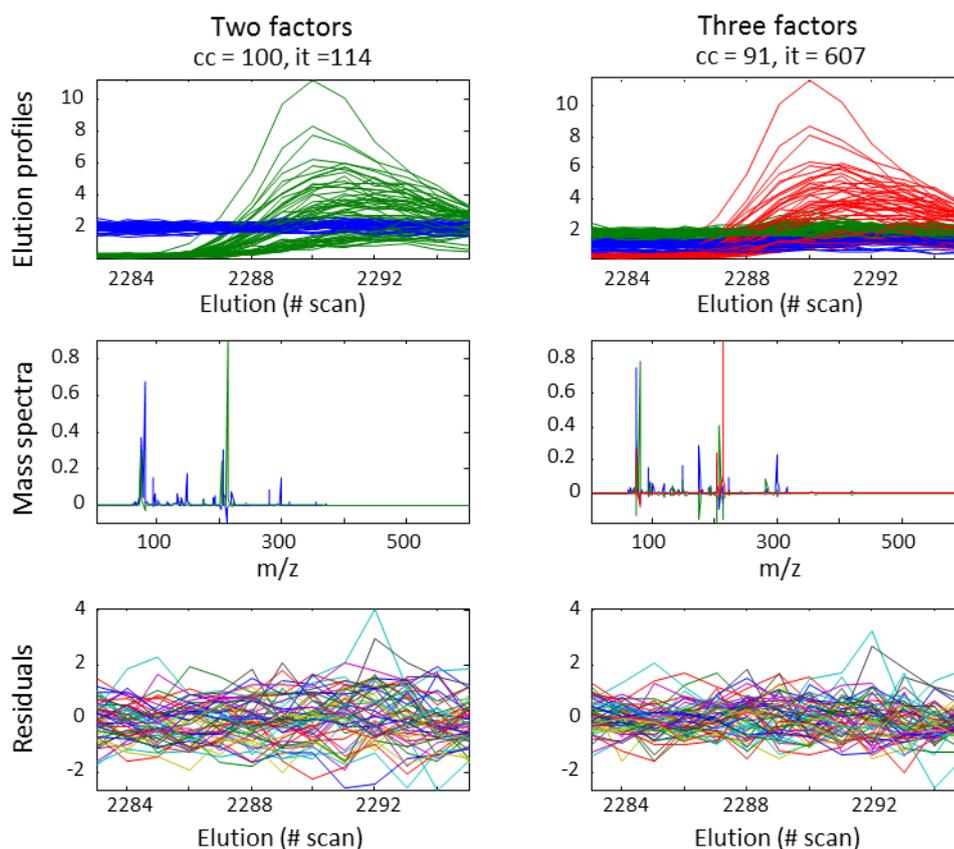


Figure 28. Elution profiles, mass spectra profiles and residuals from interval 2 modeled with PARAFAC2. The optimal model is the model with two factors. cc; Core consistency. it; iterations

By evaluating interval 2 in the same way as just described for interval 1, it can be concluded that this interval is best described by a model with two factors. This conclusion is based on the observation that the residuals in the two-factor model are very unsystematic, and that there, in the three-factor model, is an increase in negative values in the obtained spectra as well as a considerable increase in the iterations. The two- and three-factor models are shown in Figure 28.

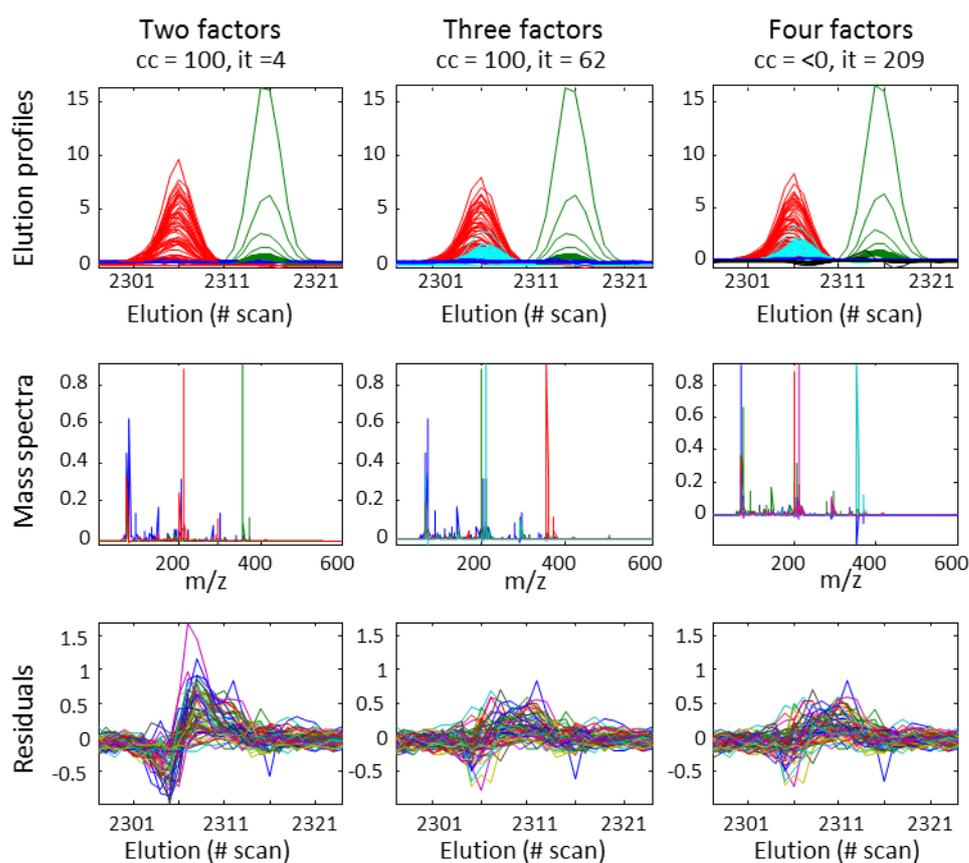


Figure 29. Elution profiles, mass spectra profiles and residuals from interval 3 modeled with PARAFAC2. The optimal model is the model with four factors. cc; Core consistency. it; iterations

In interval 3 two distinct peaks are eluting, therefore the initial model is made with three factors (one for baseline + two for the chemical compounds). Inspection of this model shows that there is some systematic behaviour in the residuals, besides this observation the model seems like a reasonable model (left in Figure 29). When one additional factor is included (middle plots in Figure 29) the systematic behaviour in residuals is reduced. However, there is also a noteworthy increase in the number of iterations which could indicate over-fitting,

but since there are no other indications that this model is over-fitted, the overall conclusions must be, that this is not an over-fitted model. By including four factors in the model (rightmost in Figure 29) the iterations increase even more, the core consistency becomes very low, and there is an increase in the negative values in the obtained spectra. All these observations indicate that this model is over-fitted. Based on these observations it may be concluded that this interval is optimally described by a three-factor model.

By applying PARAFAC2 on the three intervals, resolution of overlapping compounds and removal of baseline were achieved. However, as illustrated in the above, it is not trivial to find the optimal model, and it is both time consuming and requires expert evaluation. Therefore, it is suggested to start out with the automated approach, which is described in the next section.

After having found the optimal model, the spectra profiles can be exported with the algorithm `loads2chrom` (Murphy et al. 2012), and subsequently be identified with the open source software `OpenChrom` (Wenig, Odermatt 2010) and the Mass Spectral NIST library. The concentration profiles can be exported to Excel, and from here imported to other software for further interpretation (e.g. PCA analysis).

Box 3 shows the MATLAB commands, used for calculating a PARAFAC2 model on interval one. Also the commands necessary for creating the plots used in the evaluation, as well as commands used for creation of the MPL file, which can be used in `OpenChrom`, and exportation of the concentration profiles to Excel are shown.

Box 3. Manual calculation and inspection of PARAFAC2 models.

The following lines of commands calculate a PARAFAC2 model with 3 factors on the first of the three intervals shown in Figure 26 (scan 2267 to 2283).

The command requires that the `parafac2`, `corecondia`, `sign_flip` and `loads2chrom` files are located either in the MATLAB path or in the “current folder” (all these files are available from <http://www.models.life.ku.dk>).

```
% Creation of the interval and calculation of the PARAFAC2 model
F      = 3; % The number of factors to include
I1     = 2267; % First data point in the interval
I2     = 2283; % Last data point in the interval
X      = M(:,I1:I2,:);
max_it = 50000;
[A,H,C,P,it] = parafac2(X, F, [0 0], [0 max_it 0]);
% Calculation of core consistency
Y = zeros(size(X,1), F, size(P,2));
for i2 = 1:size(X,3)
    Y(:,:,i2) = X(:,:,i2)*P{i2};
end
cc = corcondia(Y,{A,H,C},[],0);
% The signs of the loadings are corrected
m.loads{2} = C;
m.loads{1}.H = H;
m.loads{3} = A;
m.loads{1}.P = P;
m.modeltype = 'parafac2';
x = permute(X, [2 1 3]);
[sgns,m] = sign_flip(m,x);
% The model is visualized with plots
Figure
name = horzcat(num2str(F), 'factors', '. Cc = ', num2str(cc));
set(gcf,'Name',name,'NumberTitle','off')
subplot(2,2,1)
plot(squeeze(sum(X,1)))
title('TIC from raw data'), xlabel('Scan no. '), axis tight
subplot(2,2,2)
d = zeros(size(X,3));
E = zeros(size(X));
for i = 1:length(P)
    for ii = 1:size(X,3)
        d(ii,ii) = m.loads{2}(i,ii);
    end
    E(:, :, i) = X(:, :, i) -
        (m.loads{3}*d*(m.loads{1}.P{1,i}*m.loads{1}.H));
    plot(m.loads{3}.P{i}* m.loads{3}.H*d), hold on
end
hold off
title('Weighted elution profiles'), xlabel('scan no. '), axis tight
subplot(2,2,3), plot(m.loads{3})
title('Mass spectra'), xlabel('m/z'), axis tight
subplot(2,2,4), plot(squeeze(sum(E,1)))
title('TIC from residuals'), xlabel('Scan no. '), axis tight
```

The following lines of commands creates a MPL file called “spec” and exports the concentration profiles to an Excel file named Conc.

```
loads2chrom('spec',model,rt(I1:I2),mz,'minutes');
xlswrite('Conc', C);
```

*Automated evaluation of PARAFAC2 models*

With the automated approach, models will be calculated for all intervals with one line of command. A suggestion for the optimal number of factors will also be made based on a classification model which includes some descriptive parameters for over-fit. The classification finds the model which is most likely to be the first over-fitted model, the model with one less factor is then suggested as the optimal model. The method has been validated in paper IV, where a more thorough description is also available.

After having calculated models for all the desired intervals, the models appointed as optimal should be briefly inspected in order to control that they look reasonable. In the inspection some of the same considerations, as described in the previous section, must be made.

The automated approach recommends using three factors for interval 1, three factors for interval 2, and four factors for interval 3. This is pretty much in agreement with the conclusions drawn from the manual assessment of the models. The only exception is interval two which, by the automated approach, is modelled with one additional factor (the models are illustrated in Figure 28). However the additional component, is merely describing a second baseline, and the peak modelled by the two-factor and three-factor models is described by similar mass spectra and concentration profiles as illustrated in Figure 30.

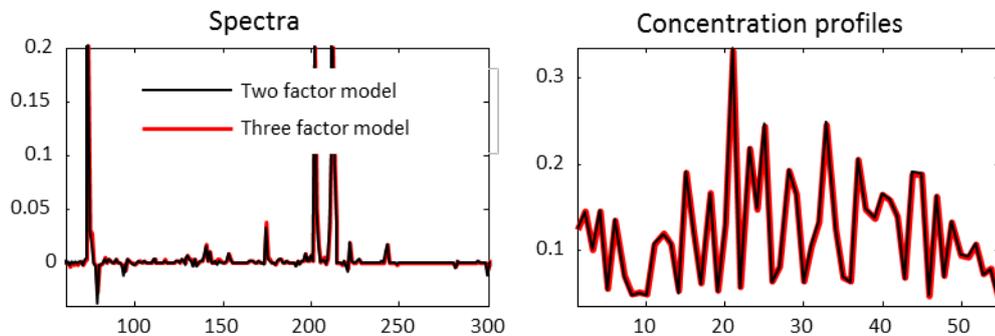


Figure 30. Comparison between obtained mass spectra and concentration profiles from the two-factor and three-factor model.

Box 4. Automated calculation and evaluation of PARAFAC2 models.

The following lines of commands calculate and evaluate PARAFAC2 models on the three intervals shown in Figure 26. Also the code necessary for creating MPL files is shown. These files will be named “spec1”, “spec2” and so forth until the number of intervals included in the calculations (in this case three). Finally the concentration profiles are exported to Excel.

The command requires that the `parafac2`, `auto_PF2`, and `sign_flip` functions to be located either in the MATLAB path or in the “current folder” (all the functions are available from <http://www.models.life.ku.dk/algorithms>). Furthermore PLS\_toolbox (Eigenvector) must be installed.

The first line defines the boundaries of the intervals separated by semicolon.

```
I = [2267 2283; 2283 2296; 2297 2327];
out = PF2auto(M, I);
view_models(out, M)

% Creation of the MPL files for identification and export of
% concentration profiles to Excel:

conc=[];
for i = 1:length(out.nfactors)
    m = out.models{i,out.nfactors(i)};
    T = horzcat('spec', num2str(i));
    loads2chrom(T,m,out.rt(out.int(i,1):out.int(i,2)),out.mz,'minutes');
    conc = [conc m.loads{2}];
end
xlswrite('Conc', conc)
```

## 6.2 Baseline separated intervals - MCR

Before any MCR models can be calculated data must be inspected and, as for PARAFAC2, the chromatograms must be divided into baseline separated intervals containing a limited number of compounds. MATLAB code for importing and plotting raw data can be found in Box 2.

In the examples in this section an interval containing glycine and isotopic labelled ( $N^{15}$  and  $C^{13}$ ) glycine is used to illustrate how to obtain an MCR model which describes the underlying chemistry. A total of five samples are included in the dataset with varying amounts of respectively the labelled and non-labelled glycine. The TIC obtained from the

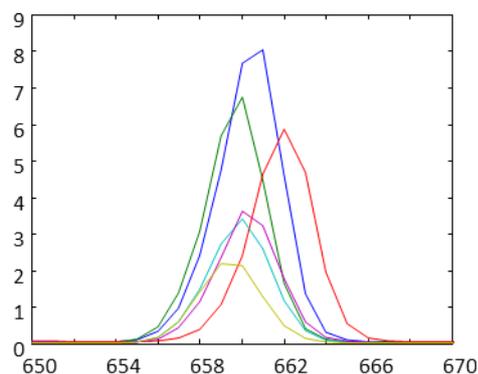


Figure 31. Raw data illustrated by the total ion chromatograms.

raw data is illustrated in Figure 31.

The main difference between MCR and PARAFAC2 is that MCR is a bilinear method, and hence is working on two-way matrices, in contradiction to the trilinear method PARAFAC2, which is working on three-way arrays. This means that data needs to be arranged in a matrix before any modelling can be performed. In Box 5 it is shown how a three-way array can be rearranged into a two-way matrix in MATLAB. After having rearranged the data MCR models can be calculated. There exist several different algorithms for this. The models described here have all been calculated using the MCR-ALS Graphic User Interface (GUI) (Jaumot et al. 2005) available from <http://www.mcrals.info>. Guidelines on how to use the GUI are found in the publication, and in Box 5, in the end of this section, it is described which settings that were used to create the models described throughout this section, also some MATLAB commands necessary for using the constraints described below are included in the box.

As for PARAFAC2 the MCR models must be made with the right number of factors included. However, it is even more complex to determine how many the “right number” is when using MCR. There are several reasons for this, one of the reasons is that there is no counterpart to core consistency for MCR models. Also the evaluation of increased negativity in models is useless in evaluation of MCR models, since non-negativity almost always is applied. In addition to this the use of different constraints will change the appearance of the models and this may make it even more difficult to choose how many factors to include.

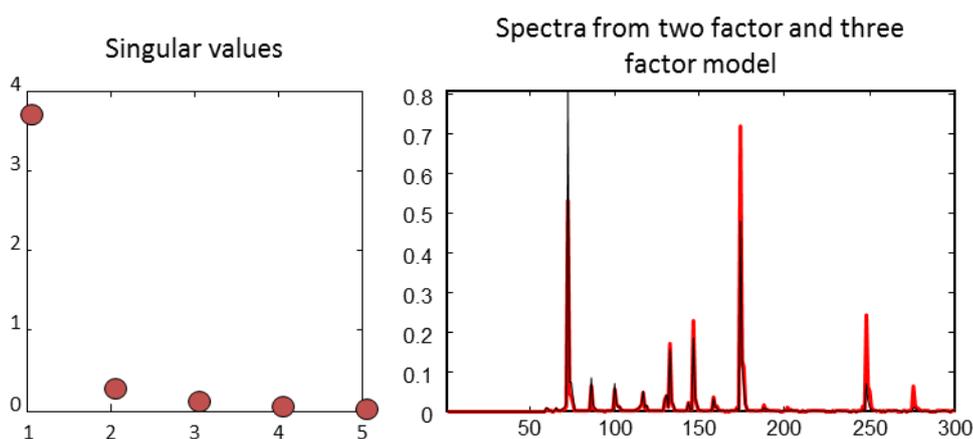


Figure 32. Left: Singular values for the data illustrated in Figure 31. Right: Comparison of the spectra representing two of the compounds in the three-factor model.

One approach is to use the appearances of obtained loadings together with Singular Value Decomposition (SVD) to determine the optimal number of factors. To the left in Figure 32

the singular values from SVD of the data illustrated in Figure 31 are shown. These suggest that the rank of data is two, since the singular values are changing very little by inclusion of additional components. By evaluating models with respectively two and three factors it is clear that the inclusion of the third factor is not resolving additional chemical compounds. As shown to the right in Figure 32 two of the three obtained spectra are very similar, this indicates that one chemical compound is described by two components. Since both the SVD and the evaluation of models indicate that two factors are appropriate for this dataset two-factor models are used in the remaining part of this section.

When creating MCR models constraints are used to ensure that the obtained models are unique and are describing the underlying chemistry in data. However, if the wrong constraints are used the model might describe some artefacts which are not present in data. It is therefore very important that the constraints used are based on knowledge about the present data. Furthermore it is recommended to use as few constraints as possible in order to let the data speak for itself. For this reason, an initial model is made with non-negativity in spectra and concentration profiles as the only constraint. At first glance it seems like the model is capable of separating the two compounds. However, if the known spectrum for the internal standard is compared with the spectrum obtained from the model, it becomes obvious that the model is not describing the real underlying chemistry (see Figure 33).

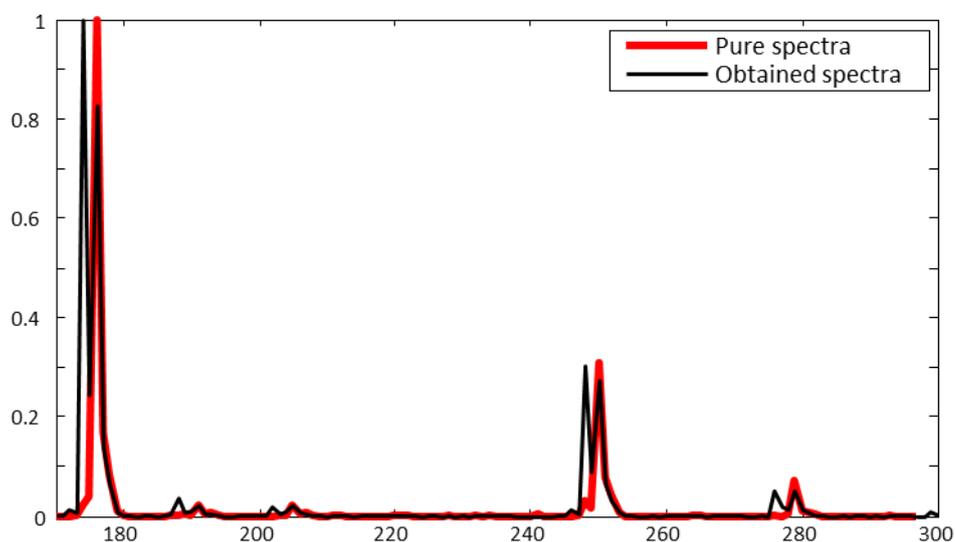


Figure 33. Comparison between the known spectra of the internal standard and the spectra obtained with an MCR model constraint with non-negativity alone.

Since the spectra of the two compounds are known, we can apply an **equality constraint in spectra**. This means that the algorithm is being constrained to match the known spectra. This is in practice done by creating a new matrix (xecs) holding the raw data plus the pure spectra from the two compounds. Furthermore a matrix with information about which of the rows that are containing these pure spectra must be created. In this way an “xces” matrix with the data, shown in Figure 31, plus the raw spectra of labelled and unlabelled glycine were created. An MCR model created in xces, constraint so that the rows with the pure spectra is forced to have no contribution from the other compound, will provide spectra profiles which completely match the profiles of the known compounds (not shown).

Another option is to apply an **equality constraint in concentration** profiles. This requires that the two eluting compounds are not totally overlapping. In that case a matrix can be created with information about where the compounds are *not* eluting. By constraining the model to follow this, the obtained solution will become unique (Manne 1995), but it requires a very thorough inspection of data, and in cases with completely overlapping peaks it is not a possibility. Therefore this constraint cannot be applied on the models created in this section.

A last very useful constraint is to **identify species** which are absent in specific samples. This is in practice done by creating a samples-by-components matrix containing ones, except

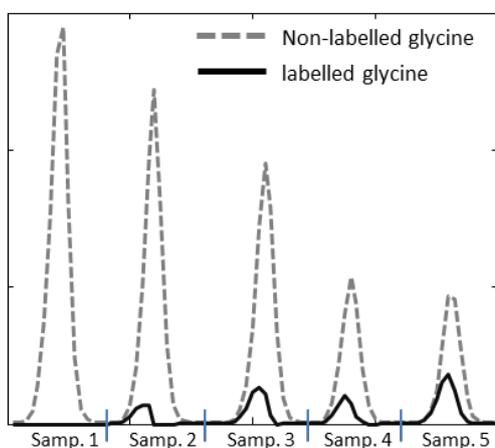


Figure 34. Elution profiles obtained when “identify species” constraint is applied.

positions representing samples where the compounds are known to be absent. If this constraint is applied, the concentration profiles are forced to zero in samples where the relevant compound is stated to be absent. No matter of the compound actually being present. By forcing the model to find a solution with no labelled glycine present in sample one, the elution profiles shown in Figure 34 are obtained. In accordance with the applied constraint, the concentration profile for labelled glycine is zero in sample one, even though labelled glycine was added

to all samples. With this in mind the conclusion must be that the model is not describing actual chemistry. An indication of this can be found if the obtained fit of this “identify species” constrained model (99.69) is compared with the obtained fit for the model without this constraint (99.87). The fact that the fit has decreased by the introduction of the

constraint shows that the solution has been forced away from the optimal solution space, and hereby also away from the chemical correct solution.

As exemplified above constraints should be used with caution, and the obtained model should be carefully validated. A key parameter in this validation is to check if the fit is decreased by applying the constraints; as explained previously, a decrease strongly indicate that the constraint has forced the solution away from the chemically correct solution. Another important aspect is to make sure that the constraints are supported by observations from raw data. An example could be to check if individual mass traces from raw data support the constraints.

When the model, which describes the underlying chemistry in the most accurate way, has been identified, spectra profiles can be exported and identified in the same way as for spectra profiles obtained with PARAFAC2. Also the concentrations can be exported to other software for further interpretation. In Box 5 it is shown how to export concentrations to XLS files and spectra to MPL files.

Box 5. Calculation and evaluation of MCR models.

A three-way array (I, with mass channels in first mode, elution time in second mode and samples in third mode) is created containing the interval from scan # 650 to 670. Subsequently the array is re-arranged into a two-way matrix (x, with elution time in first mode and mass channels in second mode), and the matrix is scaled so the highest intensity is equal to one:

```
S = 5; %The number of samples included
I = M(:, 650:670, :);
x = [];
for i = 1:S
    x = [x; I(:, :, i)];
end
x = x./max(max(x));
```

Now the MCR models can be calculated on the two-way matrix (x) with the MCR-ALS GUI. The GUI is initiated by writing `mcr_main` in the MATLAB command window (assuming that the function is located either in the MATLAB path or in the “current folder”).

By following the steps described below, the models described in this chapter can be created.

- 1) The variable containing the relevant data (x) is selected.
- 2) The number of components is selected using “SVD”.
- 3) Initial estimations are performed using “Pure”. The direction is chosen as “spectra” and the allowed noise is set to 5% (between 5 and 10% is generally recommended). After having performed the selection, select “Sort the list of purest variable in the output”. This ensures that the same chemical compound is modeled with the same component in subsequent models.
- 4) Unless the “identify species” constraint is to be applied perform the calculations with “nr. of matrices” set to one, otherwise set it to the number of samples included.

Box 5, continued. Calculation and evaluation of MCR models.

- 5) Select non-negativity both in “conc and spec”. Use “fnnls” and set the number of species to the number of components included in the model.
- 6) Select any additional constraints. (More about this below).
- 7) Initiate the optimization and select normalization to “spectra equal length”.

The code below exports concentrations to an Excel file named (Conc.xls), and spectra to an MPL file (spec.mpl).

```
S = 5; %The number of samples included
m.loads{1} = copt;
m.loads{2} = sopt;
m.modeltype = 'MCR';
loads2chrom('spec',m,rt,mz,'minutes');
C = [];
r = size(x,1)/S;
for i = 1:S
    C = [C; sum(copt(i*r-(r-1):i*r,:))];
end
xlswrite('Conc', C);
```

#### Equality constraints in spectra:

The following code create a matrix (ecs) which is to be used in step 6) above, as well as the matrix (xecs), which is holding the data and pure spectra, which should be used in the creation of the model. The code assumes the pure spectra of the species are available in an Excel file named “spectra” in sheets named “species1” and “species2”, respectively. It is important to ensure that “species1” is representing the compound modeled by the first component in the initial model (and the same for other species). When applying the constraint it should be selected as “lower or equal than”.

```
sp1 = xlsread('spectra','species1');
sp1 = sp1./max(max(sp1));
sp2 = xlsread('spectra','species2');
sp2 = sp2./max(max(sp2));

xecs = [x; sp1; sp2];

ecs = NaN(50,2);
ecs(end-1,2) = 1e-5;
ecs(end,1) = 1e-5;
```

#### Identify species constraint:

The following codes create a matrix (isc) which is to be used in step 6). The code is constraining species two to be absent in sample one. It is important to ensure that “species one” is modeled by the first component in the initial model (and the same for other species).

```
c = 2 %The number of components included
s = 5 %The number of samples included
isc = ones(s,c);
isc(1,2)=0;
```

Box 5, continued. Calculation and evaluation of MCR models.

#### Equality constraints in concentration:

The following codes create a matrix (ecc) which is to be used in step 6). In the example it is assumed that species one is *not* eluting in the intervals from scan # 1 to 3 and 40 to 65, while species two is *not* eluting in the intervals from scan # 1 to 10 and 62 to 65. It is important to ensure that “species one” is modeled by the first component in the initial model (and the same for other species). When applying the constraint it should be selected as “lower or equal than”.

```
c = 2 %The number of components included
ecc = NaN(size(x,1),c);
ecc([1:3 40:65],1) = 1e-5;
ecc([1:10 62 65],2) = 1e-5;
```

### 6.3 More complex data

For data where it is impossible to divide it into smaller, baseline separated intervals, containing only a few compounds, other methods than those described in sections 6.1 and 6.2 must be applied.

One very straight-forward approach would be to process the TIC or individual mass traces as described in section 5.2. This will give some indications of where the variation of interest occurs, as well as similarities and dissimilarities between samples. The drawback with this approach is that all the additional information, which was achieved by using the multi-channel detector, is disregarded during the analysis of the data.

Other approaches do exist, but they will not be further described in this thesis. Examples hereof could be the method described by Dixon *et al.* (2006), the freeware MetaboliteDetector (Hiller *et al.* 2009), or the R based open source platform XCMS (Smith *et al.* 2006). XCMS is mostly used for LC-MS data, but if combined with the TagFinder software (Luedemann *et al.* 2008) it is also useful for GC-MS data. Recently XCMS was launched as a web-based interface (Tautenhahn *et al.* 2012) enabling users, who are unfamiliar with R, to take advantage of the platform.

*Chapter 7***Conclusions and perspectives**

---

This thesis was created with the purpose of making untargeted analysis of chromatographic data more accessible for those who are working with chromatography on a daily basis. These people are experts within the field of chromatography, but may not have skills in chemometrics or be comfortable with command line based software like MATLAB. In this thesis, thorough descriptions are given on how different types of chromatographic data can be processed in an untargeted manner. In order for the data to be treated adequately a guide has been presented which helps the user to choose the right method. The first thing to consider is whether data is obtained from first-order or second-order instruments. Subsequently the complexity of the chromatograms should be evaluated. In the following the recommendations for the four categories of data, will be outlined, and the perspective of how these methods can become applicable in the daily work with chromatography will be discussed.

The MATLAB based graphical user interface (GUI), FastChrom, which is presented in paper I, is recommended for well behaving chromatograms without too much complexity obtained from first-order instruments. This method enables automatic baseline removal, peak detection, and peak grouping across samples. The method does not require any specific chemometrics skills; the only parameters the user needs to decide upon are related to knowledge about the chromatographic data. The method finds all peaks in the chromatogram and reports the height and possibly the retention index (if index samples are included), and since it is incorporated into a GUI it does not require that the user has any experience in using MATLAB.

For single-channel data with high complexity, it is recommended to use an approach where the chromatograms are pre-processed in a simple manner with the objective to use MCR or PCA to find the main sources of variability and to be able to identify characteristics of the samples. This approach requires some programming skills in some command line based software in order to be able to pre-process the chromatograms. Furthermore must the user have knowledge about how MCR and/or PCA models are interpreted.

For low complexity-data obtained from second-order instruments, PARAFAC2 is suggested to resolve overlapping peaks and to remove baseline contributions. It has previously been shown that PARAFAC2 is a suitable method for mathematical chromatography, meaning that it can separate co-eluting compounds mathematically. In order to ease interpretation of the obtained models, a solution to the problem with sign indeterminacies has been developed (paper III). Besides making the interpretation easier, it also helps in the evaluation of how many factors that should be included in the PARAFAC2 model. Another new development, which will make it easier to determine the appropriate number of factors, is core consistency for PARAFAC2 (paper II). Core consistency will be high (close to 100) for models which do not have too many factors included, and low for models with too many factors. In order to make PARAFAC2 more accessible for those which are not skilled in both chemometrics and chromatography, an automated procedure for estimation of how many factors a PARAFAC2 model should have included has been developed (paper IV). This procedure is based on a classification model which, with a number of carefully selected model diagnostics, is able to find the first over-fitted model. The model with one factor less is then suggested as the model which describes the data in the most appropriate way. The automated approach requires only few lines of MATLAB commands and can therefore be used by users without skills in MATLAB programming.

In cases where PARAFAC2 is unable to model the data in an adequate manner, it is recommended to use MCR. It has been demonstrated how constraints can be applied to MCR, and how these constraints can limit the solution space so the obtained solution is the one which describes the underlying chemistry. However, it has also been shown how the wrong constraints can force the model to give wrong solutions. The usage of constraints should therefore be based on knowledge about data and observations from raw data. The calculation of the MCR models can be performed in existing GUIs. However, in order to be able to apply the constraints it is necessary to be somewhat familiar with MATLAB.

For multi-channel data with high complexity it is recommended to use the method for high complexity single channel data on e.g. the TIC. A number of other methods suitable for this type of data are also suggested, but these methods are not further investigated.

With the automations developed in this thesis, one more step has been taken towards making comprehensive and untargeted analysis of chromatographic data more accessible for those working with chromatography. However, there is still some way to go before the average scientist, working with chromatography, can take full advantage of the methods already available, including the methods described in this work. As outlined above is the

usage, of a lot of the methods, still dependent on the user having some knowledge about how to use MATLAB (or other command line based software). In order for the methods to be really accessible at least GUIs must be developed which completely eliminates the need of using the command line to arrange or pre-process data. However, if we really want to reach the wider range of end-users, standalone software should be developed or, as the optimal solution, the methods should be incorporated into the software developed by the instrument vendors. With these improvements comprehensive untargeted analysis could be performed routinely, and the average analysis laboratory will be able to utilize the available methods.



## References

---

- Amigo, J.M., Skov, T., Coello, J., Maspoch, S. & Bro, R. **2008**, "Solving GC-MS problems with PARAFAC2", *Trends in Analytical Chemistry*, vol. 27, pp. 714-725.
- Berge, J., M.F. & Kiers, H., A.L. **1996**, "Some uniqueness results for PARAFAC2", *Psychometrika*, vol. 61, pp. 123-132.
- Bicking, M.K.L. **2006a**, "Integration Errors in Chromatographic Analysis, Part I: Peaks of Approximately Equal Size.", *LCGC North America*, vol. 24, pp. 402-414.
- Bicking, M.K.L. **2006b**, "Integration Errors in Chromatographic Analysis, Part II: Large Peak Size Ratios.", *LCGC North America*, vol. 24, pp. 604-616.
- Booksh, K.S. & Kowalski, B.R. **1994**, "Theory of Analytical Chemistry", *Analytical Chemistry*, vol. 66, pp. 782A-791A.
- Bro, R. **1997**, "PARAFAC. Tutorial and applications", *Chemometrics and Intelligent Laboratory Systems*, vol. 38, pp. 149-171.
- Bro, R., Andersson, C.A. & Kiers, H.A.L. **1999**, "PARAFAC2 - Part II. Modeling chromatographic data with retention time shifts", *Journal of Chemometrics*, vol. 13, pp. 295-309.
- Bro, R. & Kiers, H.A.L. **2003**, "A new efficient method for determining the number of components in PARAFAC models", *Journal of Chemometrics*, vol. 17, pp. 274-286.
- Büscher, J.M., Czernik, D., Ewald, J.C., Sauer, U. & Zamboni, N. **2009**, "Cross-platform comparison of methods for quantitative metabolomics of primary metabolism", *Analytical Chemistry*, vol. 81, pp. 2135-2143.
- Carroll, J.D. & Chang, J. **1970**, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition", *Psychometrika*, vol. 35, pp. 283-319.
- Cattell, R.B. **1944**, ""Parallel proportional profiles" and other principles for determining the choice of factors by rotation", *Psychometrika*, vol. 9, pp. 267-283.
- Christensen, J.H., Hansen, A.B., Karlson, U., Mortensen, J. & Andersen, O. **2005**, "Multivariate statistical methods for evaluating biodegradation of mineral oil", *Journal of Chromatography A*, vol. 1090, pp. 133-145.
- Christensen, J.H., Hansen, A.B., Tomasi, G., Mortensen, J. & Andersen, O. **2004**, "Integrated methodology for forensic oil spill identification", *Environmental science & technology*, vol. 38, pp. 2912-2918.

- 
- Christensen, J.H. & Tomasi, G. **2007**, "Practical aspects of chemometrics for oil spill fingerprinting", *Journal of chromatography.A*, vol. 1169, pp. 1-22.
- Christensen, J.H., Tomasi, G. & Hansen, A.B. **2005**, "Chemical fingerprinting of petroleum biomarkers using time warping and PCA", *Environmental science & technology*, vol. 39, pp. 255-260.
- De Meyer, T., Sinnaeve, D., Van Gasse, B., Tsiorkova, E., Rietzschel, E.R., De Buyzere, M.L., Gillebert, T.C., Bekaert, S., Martins, J.C. & Van Criekinge, W. **2008**, "NMR-based characterization of metabolic alterations in hypertension using an adaptive, intelligent binning algorithm", *Analytical Chemistry*, vol. 80, pp. 3783-3790.
- Dixon, S.J., Brereton, R.G., Soini, H.A., Novotny, M.V. & Penn, D.J. **2006**, "An automated method for peak detection and matching in large gas chromatography-mass spectrometry data sets", *Journal of Chemometrics*, vol. 20, pp. 325-340.
- Dunn, W.B. **2008**, "Current trends and future requirements for the mass spectrometric investigation of microbial, mammalian and plant metabolomes", *Physical biology*, vol. 5, pp. 011001.
- Furbo, S. & Christensen, J.H. **2012**, "Automated Peak Extraction and Quantification in Chromatography with Multichannel Detectors", *Analytical Chemistry*, vol. 84, pp. 2211-2218.
- Gargallo, R., Tauler, R., Cuesta-Sanchez, F. & Massart, D. **1996**, "Validation of alternating least-squares multivariate curve resolution for chromatographic resolution and quantitation", *Trends in Analytical Chemistry*, vol. 15, pp. 279-286.
- Geladi, P. **1989**, "Analysis of multi-way (multi-mode) data", *Chemometrics and Intelligent Laboratory Systems*, vol. 7, no. 1, pp. 11-30.
- Harshman, R.A. **1970**, "Foundations of the PARAFAC procedure: Models and conditions for an "exploratory" multimodal factor analysis", *AUCLA Working Papers in Phonetics*, vol. 16, pp. 1-84.
- Harshman, R., A. & Lundy, M., E. **1996**, "Uniqueness proof for a family of models sharing features of Tucker's three-mode factor analysis and PARAFAC/candecomp", *Psychometrika*, vol. 61, no. 1, pp. 133-154.
- Hiller, K., Hangebrauk, J., Jäger, C., Spura, J., Schreiber, K. & Schomburg, D. **2009**, "MetaboliteDetector: comprehensive analysis tool for targeted and nontargeted GC/MS based metabolome analysis", *Analytical Chemistry*, vol. 81, pp. 3429-3439.
- Hoggard, J.C. & Synovec, R.E. **2007**, "Parallel Factor Analysis (PARAFAC) of Target Analytes in GC × GC–TOFMS Data: Automated Selection of a Model with an Appropriate Number of Factors", *Analytical Chemistry*, vol. 79, pp. 1611-1619.
- Jaumot, J., Gargallo, R., de Juan, A. & Tauler, R. **2005**, "A graphical user-friendly interface for MCR-ALS: a new tool for multivariate curve resolution in MATLAB", *Chemometrics and Intelligent Laboratory Systems*, vol. 76, pp. 101-110.
- Kanani, H., Chrysanthopoulos, P.K. & Klapa, M.I. **2008**, "Standardizing GC–MS metabolomics", *Journal of Chromatography B*, vol. 871, pp. 191-201.

- Khakimov, B., Amigo, J.M., Bak, S. & Engelsens, S.B. **2012**, "Plant metabolomics: Resolution and quantification of elusive peaks in liquid chromatography-mass spectrometry profiles of complex plant extracts using multi-way decomposition methods", *Journal of Chromatography A*, vol. 1266, pp. 84-94
- Kiers, H.A.L. **2000**, "Towards a standardized notation and terminology in multiway analysis", *Journal of Chemometrics*, vol. 14, pp. 105-122.
- Kiers, H.A.L., ten Berge, J.M.F. & Bro, R. **1999**, "PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model", *Journal of Chemometrics*, vol. 13, pp. 275-294.
- Koek, M.M., Jellema, R.H., van der Greef, J., Tas, A.C. & Hankemeier, T. **2011**, "Quantitative metabolomics based on gas chromatography mass spectrometry: status and perspectives", *Metabolomics*, vol. 7, pp. 307-328.
- Luedemann, A., Strassburg, K., Erban, A. & Kopka, J. **2008**, "TagFinder for the quantitative analysis of gas chromatography—mass spectrometry (GC-MS)-based metabolite profiling experiments", *Bioinformatics*, vol. 24, pp. 732-737.
- Manne, R. **1995**, "On the resolution problem in hyphenated chromatography", *Chemometrics and Intelligent Laboratory Systems*, vol. 27, pp. 89-94.
- Murphy, K.R., Wenig, P., Parcsi, G., Skov, T. & Stuetz, R.M. **2012**, "Characterizing odorous emissions using new software for identifying peaks in chemometric models of gas chromatography—mass spectrometry datasets", *Chemometrics and Intelligent Laboratory Systems*, vol. 118, pp. 41-50.
- Nielsen, N.P.V., Carstensen, J.M. & Smedsgaard, J. **1998**, "Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping", *Journal of Chromatography A*, vol. 805, pp. 17-35.
- Petersen, I.L., Tomasi, G., Sørensen, H., Boll, E.S., Hansen, H.C.B. & Christensen, J.H. **2011**, "The use of environmental metabolomics to determine glyphosate level of exposure in rapeseed (*Brassica napus* L.) seedlings", *Environmental Pollution*, vol. 159, pp. 3071-3077.
- Rew, R. & Davis, G. **1990**, "NetCDF: an interface for scientific data access", *Computer Graphics and Applications, IEEE*, vol. 10, pp. 76-82.
- Savitzky, A. & Golay, M.J.E. **1964**, "Smoothing and differentiation of data by simplified least squares procedures.", *Analytical Chemistry*, vol. 36, pp. 1627-1639.
- Skov, T. & Bro, R. **2008**, "Solving fundamental problems in chromatographic analysis", *Analytical and Bioanalytical Chemistry*, vol. 390, pp. 281-285.
- Skov, T., Berg, F.v.d., Tomasi, G. & Bro, R. **2006**, "Automated alignment of chromatographic data", *Journal of Chemometrics*, vol. 20, pp. 484-497.
- Smith, C.A., Elizabeth, J., O'Maille, G., Abagyan, R. & Siuzdak, G. **2006**, "XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification", *Analytical Chemistry*, vol. 78, pp. 779-787.
- Svendsen, C., Johnsen, L.G. & Bro, R. **In preparation**, "Exploring fermentation processes using gas chromatography-mass spectrometry and chemometrics".

- 
- Tauler, R. **1995**, "Multivariate curve resolution applied to second order data", *Chemometrics and Intelligent Laboratory Systems*, vol. 30, pp. 133-146.
- Tauler, R., Izquierdo-Ridorsa, A. & Casassas, E. **1993**, "Simultaneous analysis of several spectroscopic titrations with self-modelling curve resolution", *Chemometrics and Intelligent Laboratory Systems*, vol. 18, pp. 293-300.
- Tautenhahn, R., Patti, G.J., Rinehart, D. & Siuzdak, G. **2012**, "XCMS Online: A Web-Based Platform to Process Untargeted Metabolomic Data", *Analytical Chemistry*, vol. 84, pp. 5035-5039.
- Tomasi, G., Savorani, F. & Engelsen, S.B. **2011**, "i coshift: An effective tool for the alignment of chromatographic data", *Journal of Chromatography A*, vol. 1218, pp. 7832-7840.
- Tswett, M.A. **1906a**, "Adsorptionsanalyse und chromatographische Methode. Anwendung auf die Chemie des Chlorophylls", *Berichte der Deutschen Botanischen Gesellschaft*, vol. 24, pp. 384-393.
- Tswett, M.A. **1906b**, "Physikalisch-chemische studien über das Chlorophyll. Die Adsorptionen", *Berichte der Deutschen Botanischen Gesellschaft*, vol. 24, pp. 316-329.
- Van Den Berg, R.A., Hoefsloot, H.C.J., Westerhuis, J.A., Smilde, A.K. & Van Der Werf, M.J. **2006**, "Centering, scaling, and transformations: improving the biological information content of metabolomics data", *BMC genomics*, vol. 7, pp. 142.
- Wenig, P. & Odermatt, J. **2010**, "OpenChrom: a cross-platform open source software for the mass spectrometric analysis of chromatographic data", *BMC Bioinformatics*, vol. 11

# Paper I

---

**Lea G. Johnsen**, Thomas Skov, Ulf Houlberg, and Rasmus Bro.

“An automated method for baseline correction, peak finding and  
peak grouping in chromatographic data”

*Submitted for Analyst*



Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

ARTICLE TYPE

## An automated method for baseline correction, peak finding and peak grouping in chromatographic data

Lea G. Johnsen<sup>\*ab</sup>, Thomas Skov<sup>a</sup>, Ulf Houlberg<sup>b</sup> and Rasmus Bro<sup>a</sup>*Received (in XXX, XXX) Xth XXXXXXXXXX 20XX, Accepted Xth XXXXXXXXXX 20XX*

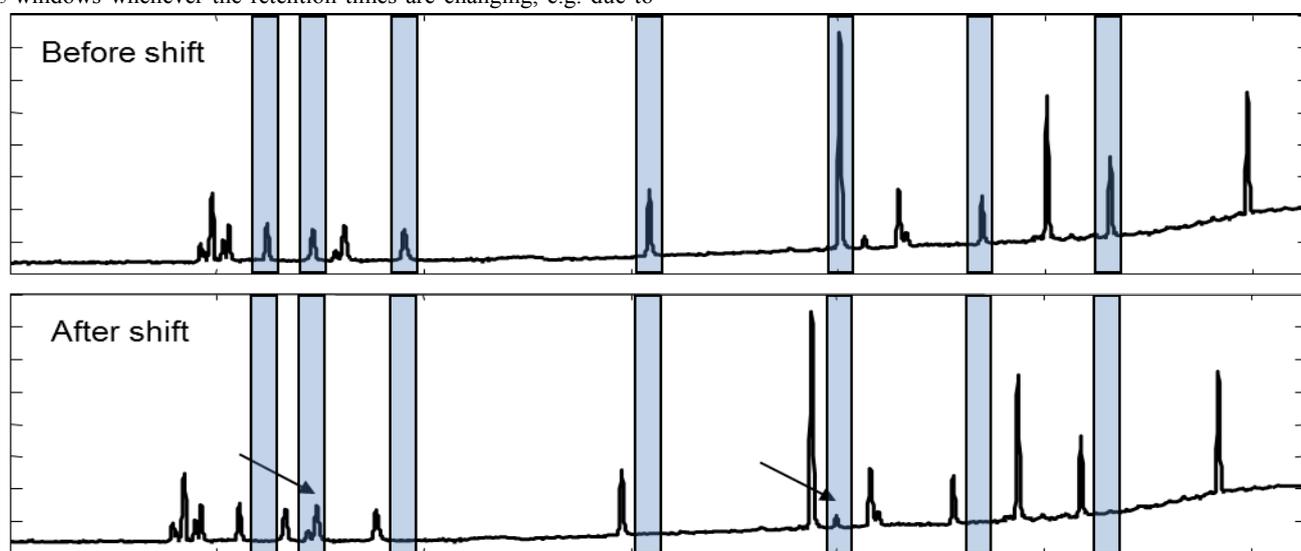
DOI: 10.1039/b000000x

An automated method (FastChrom) for baseline correction, peak detection and assignment (grouping) of similar peaks across samples has been developed. The method has been tested both on artificial data and a dataset obtained from gas chromatograph analysis of wine samples. As part of the automated approach, a new method for baseline estimation has been developed and compared with other methods. FastChrom has been shown to perform at least as well as conventional software. However, compared to other approaches, FastChrom finds more peaks in the chromatograms and not only those with retention times defined by the user. FastChrom is fast and easy to use and offers the possibility of applying a retention time index which eases identification of peaks and the comparison between experiments.

### Introduction

Most manufacturing software for handling of data obtained from a Gas Chromatograph coupled with a Flame Ionisation Detector (GC-FID) is designed to extract information of specific chemical compounds. This is normally done by defining a number of elution time windows followed by a peak search within each window. Only peaks which are eluting in these windows will then be detected. Such an approach requires knowledge about which compounds are important for a given question, and does not comply with an explorative and non-targeted approach. Furthermore, such an approach requires re-calibration of all windows whenever the retention times are changing, e.g. due to

column change or wear of columns. If the calibration is not thoroughly checked after each analysis, some peaks might shift out of their window and will therefore be assigned incorrectly or not be detected at all. Alternatively, a peak originating from the wrong compound might shift into the window and will then be identified as the compound of interest (see illustration in Figure 1). The re-calibration is a time consuming task, especially for experiments performed with worn columns – in such cases the windows sometimes need to be re-calibrated for every new experiment.



**Figure 1.** Position of windows to be evaluated before and after a shift in retention time has occurred. The arrows indicate places where peaks will be incorrectly assigned due to the shift

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

## ARTICLE TYPE

A method that could automate and improve the task of quantifying peaks would be desirable. Such an approach should contain automatic baseline estimation, peak finding, peak integration, and assignment of peaks in several chromatograms.

In addition to these requirements, it is preferred that the method can handle shifts in retention time from one experiment to another. The system should not be completely automated, since this would prevent a flexible use. However, it should make a skilled specialist in chromatography able to perform a comprehensive data analysis much simpler and efficient than is possible today.

Over the years, several publications have focused on the requirements mentioned above. Unfortunately most of them only focus on one of the elements and not on making a method covering all the aspects (e.g. baseline fitting<sup>1-3</sup> or peak detection<sup>4-8</sup>). Frenzel *et al.*<sup>9</sup> describe an automated system for handling of GC-FID data. However, this system has been designed for overall comparison of the general similarity of chromatograms and not for extraction of the areas of individual peaks. In the following a method (FastChrom) developed to handle all the above mentioned aspects is presented.

## Theory

The method established should be able to perform baseline fitting, peak finding, peak assignment across samples, and retention time indexing with as few parameters as possible to be set by the user. The parameters should be dependent of the GC-system, so that it is only necessary to set them once for every new application. It is assumed in the following that the chromatograms are originating from fairly similar samples, in the sense that the majority of the peaks appear in several of the obtained chromatograms. It is also assumed that any shift in retention time is minimal.

In the following, the different parts of the FastChrom method will be outlined. First the baseline fitting procedure will be described. This section will be followed by sections describing the peak detection, the peak assignment and the retention time index. In all of the sections it will be discussed, which parameters the user needs to adjust when using FastChrom, and how these adjustments will affect the result.

### Baseline

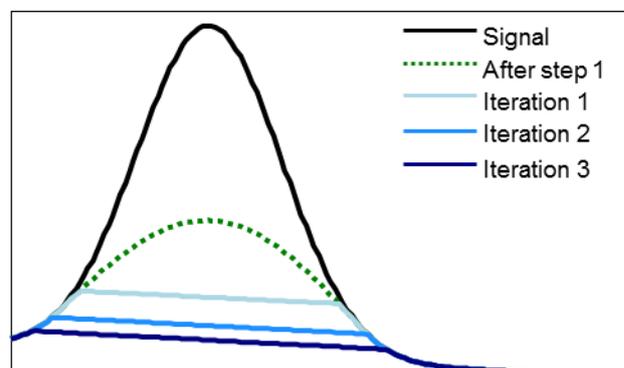
Baseline estimation models can in general be divided into non-parametric and parametric methods. The parametric methods have been claimed to outperform the non-parametric methods<sup>2</sup>. However, the majority of the parametric methods have a considerable number of parameters that must be optimised. This can be a time consuming and daunting task. Some of the most used parametric methods are based on the Whittaker smoother. The Whittaker smoother is an approach where the smoothing of the signal is controlled by the relationship between 1) the similarity to the original signal (or baseline) and 2) the noise

remaining in the signal. The importance of the two features is regulated by the given input parameters<sup>10</sup>. Two parametric methods, both based on the Whittaker smoother, which are often used, are adaptive iteratively reweighted Penalized Least Squares (airPLS)<sup>3</sup> and Asymmetric Least Squares (ALS)<sup>11</sup>, where ALS has one parameter less than airPLS, and is slightly easier to optimise.

Most of the non-parametric methods are based on polynomial fitting. However, polynomial fitting has been reported not to perform optimally in cases with low signal-to-noise ratios, and when having complex baselines<sup>12</sup>. Two recently reported non-parametric methods claim to perform at least as well as methods based on the Whittaker smoother: Automated Iterative Moving Average (AIMA)<sup>2</sup> and quantile regression<sup>1</sup>.

Quantile regression was introduced by Koenker and Bassett in 1978<sup>13</sup>, but it was not until 2011 that it was proposed for baseline fitting by Komsta<sup>1</sup>. Baseline estimation with quantile regression is based on polynomial fitting and fits the baseline to a small quantile of the signal (0.01 is proposed). In this way the polynomial is fitted to the lowest values and consequently the peaks will have little or no effect on the baseline estimated. An introduction to quantile regression is given by Koenker and Hallock<sup>14</sup> and a comprehensive description has been made by Koenker<sup>15</sup>.

Automated Iterative Moving Average (AIMA) works by identifying peaks using a combination of two tools. In step one a new signal is constructed by finding of local minima in the raw signal. Hereafter potential peak areas are identified by the original signal being higher than the new. Peak ends and starting points are identified by the signal and the local minimum being identical and between two such points, the local minimum signal is replaced by a straight line. The process is repeated until no changes occur. **Figure 2** illustrates the process.



**Figure 2.** Illustration of how the iterative removal of peaks in AIMA works.

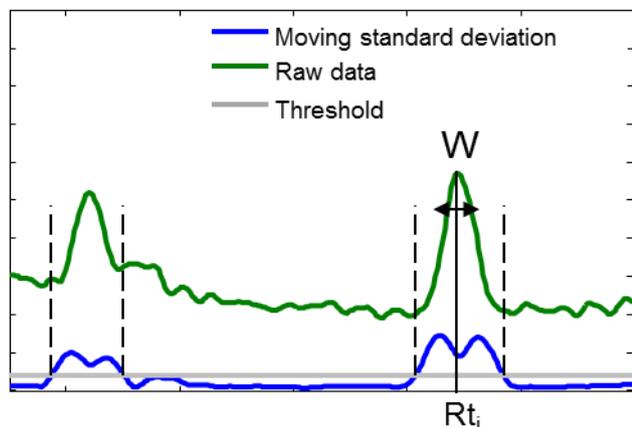
The optimal baseline fitting method should be either non-parametric or only be dependent on a few parameters which are easily optimised. In our experience, the non-parametric methods available are not performing in a satisfactory way. AIMA, for instance, has a tendency of estimating the baseline too high below

peaks and with quantile regression there is a risk of creating artificial peaks in the baseline corrected chromatogram (examples are shown in the result section). On the other hand, the parametric methods are often time consuming to optimise. Therefore we propose a new method for baseline estimation to handle these problems.

A comparison of the four methods (AIMA, ALS, quantile regression and FastChrom-baseline) will be given later.

The FastChrom-baseline fitting procedure uses information about which areas that do not contain any peaks. It is assumed that these areas are baseline and these are simply subtracted from the chromatogram. It is therefore only necessary to determine the actual baseline in the remaining areas. The estimation of baseline in "peak-regions" is based on the assumption, that the baseline is locally linear (see **Figure 4**). Extensions can easily be envisioned allowing for a more complex local baseline such as a polynomial baseline.

The first step in the estimation of the baseline is to identify areas with no peaks. Peak-regions are here defined as regions (or time section within the individual chromatograms) with a standard deviation above a certain threshold. Everything with a standard deviation below this threshold is considered noise/baseline and the remaining areas are considered as regions with possible peaks. The principle is illustrated in **Figure 3**. The width of the window used for determination of the standard deviation and the threshold can be changed by the user. It is usually sufficient to determine it once for each GC-application. If the analysis is performed on a different GC system or the settings of the GC-method are changed, the width should be re-determined. Further guidance on how to choose these parameters is given below.

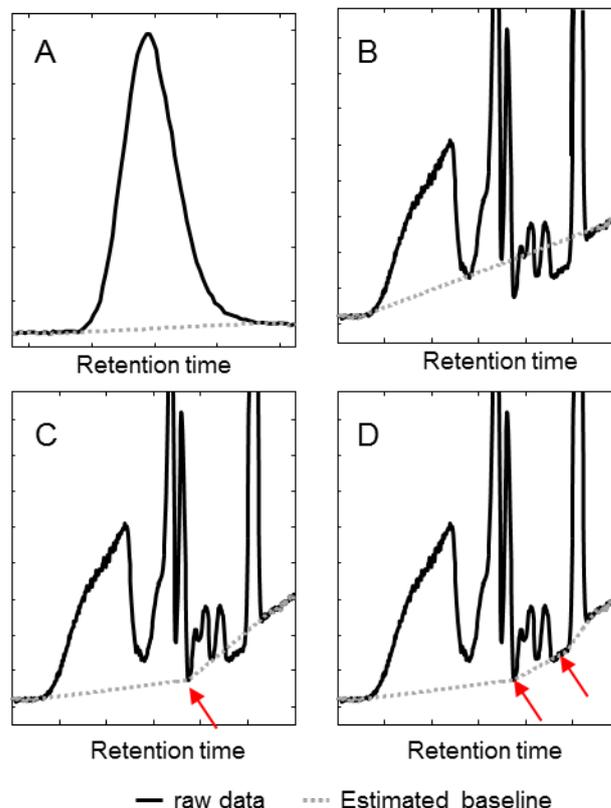


**Figure 3.** The standard deviation, which is used in determination of the peak-regions, is calculated as a moving standard deviation. This means that the standard deviation at  $Rt_i$ , is the standard deviation, across the chromatogram, inside a window with width  $W$ . Peak-regions are determined as regions where the standard deviation is higher than a user determined threshold (vertical line in figure). The dashed lines indicate the boundaries of the peak-regions.

The baseline in peak-regions is determined by linear regression between the points surrounding the region.

For resolved peaks and for peaks in regions with an approximately linear baseline this approach will give a good estimate of the baseline (**Figure 4A**). However, in some cases it will result in a poor estimated baseline (**Figure 4B**). In such

cases, the baseline will be estimated using an iterative approach. The algorithm will detect these cases by searching for areas where a number, equal to the minimum peak width at half height, of consecutive points in the raw data lies below the estimated baseline. If such areas exist, the new baseline will be forced through the point in the raw data which has the lowest value relative to the existing baseline (arrow **Figure 4C**). This procedure is repeated until no point lies below the estimated baseline. **Figure 4D** shows the final estimation of the baseline in this region.



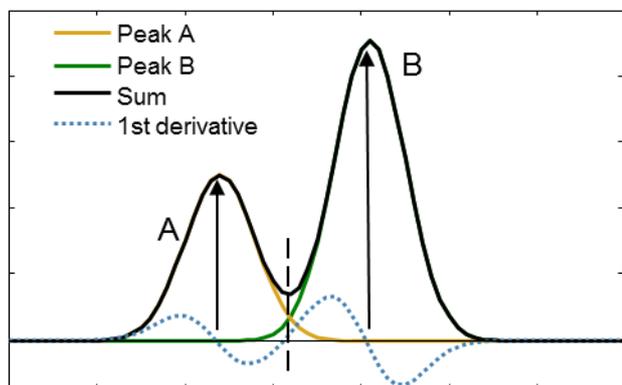
**Figure 4.** Illustration of the baseline estimation in peak-regions. A: An example of "well behaved" baseline. B: The initial estimation found by linear regression over the region. C: The linear regression is split into two. The arrow indicates the "anchor point" for the baseline. D: The final estimation of the baseline in the region with two anchor points indicated by arrows.

In the areas without peaks, the original data points are used as baseline. This means that after baseline correction these areas will have a zero value baseline.

Even though the proposed method for baseline fitting is not fully automated, it only has two parameters which need to be determined. The first is the width of the window used in determination of the standard deviation. This should not be set too narrowly since the apex of the peaks will then be recognised as non-peak regions. We recommend a size similar to the average peak width at half height. The second parameter is the threshold for non-peak versus peak-regions. This parameter can be determined by inspection of the standard deviation for a few chromatograms.

## Peak detection

Several methods have been published for peak detection. Most of these are based on derivatives of the signal<sup>4-6</sup>. By identifying where the first derivative intersects zero, the valleys between peaks and peak apex are identified as illustrated in Figure 5.



**Figure 5.** Illustration of how derivatives can be used in identifying peak apexes and valleys between peaks, by identification of where the derivative intersects with zero. The vertical line indicates the valley between peak A and peak B. The two arrows indicate peak apex.

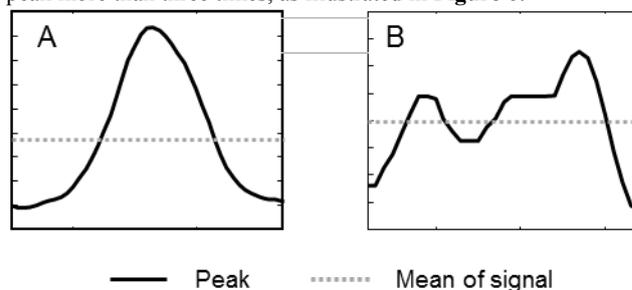
In cases with overlapping peaks it is not possible to identify the boundaries of the individual peaks with derivatives. This will in some cases cause the height or area determinations to be incorrect. However, peak height, which is used in the FastChrom peak detection, is less sensitive to these kinds of errors than peak area<sup>16, 17</sup>.

Other approaches do exist. Examples hereof are in the publications from Tian *et al.*<sup>8</sup> which use neural networks to identify and extract peaks from auditory brainstem responses, and Vivó-Truyols *et al.*<sup>7</sup> who use higher order derivatives to assess the number of peaks within a peak cluster. The method described by Vivó-Truyols *et al.* is mainly focused on peak detection with the aim of subsequently applying deconvolution.

Methods based on fitting Gaussian or Lorentzian functions to the chromatographic signal also exist (e.g. the publications by Jianwei<sup>18</sup> or Barboni and Chiaramonti<sup>19</sup>). These methods seek to solve the problem with overlapping peaks by resolving the signal into individual peaks. The drawbacks of these methods are that they often require prior knowledge of the number of compounds and that the solution may not be correct even though the right number of compounds is used<sup>20</sup>. Furthermore, peaks can be tailing or fronting and thus not be Gaussian or Lorentzian. If one wishes to resolve overlapping peaks it is in general advised to use methods with multi-channel detectors<sup>20</sup>.

The peak finding algorithm included in FastChrom is similar to the peak detection and peak validation described by Dixon *et al.*<sup>6</sup>. This approach is a simple and automated method with only a few parameters to adjust. Peaks are identified from the first derivative (*d*) of the smoothed chromatograms (corrected for baseline). The algorithm searches in *d* for points where the signal sign changes from zero or negative to positive. These points are used as start and/or end points of peaks. The maximum is found by searching for points where the first derivative intersects zero. The maximum values are grouped with the start/end points surrounding it. If the peak width and height do not meet the

requirements set by the user, the peak is discarded. The peak is also discarded if the raw signal crosses the mean intensity of the peak more than three times, as illustrated in Figure 6.



**Figure 6.** Two different scenarios are depicted: A) The mean of the signal is only crossed twice. B) The mean of the signal is crossed four times. This region is not detected as a peak

A total of three parameters, which the user has to adjust, are included in the peak finding algorithm. These are the size of smoothing window, minimum peak height and minimum peak width. Minimum peak height and width are found by inspection of the raw data. If they are set too high, peaks may not be recognised as peaks by FastChrom, and if they are set too low, noise may be included in the final peak list. Be aware that in cases with fused peaks, the peaks will become considerably wider. If in doubt, it is recommended to set them a bit too low. Hereby some noise may appear in the final peak list as peaks but these can, if desired, subsequently be manually removed. Smoothing is simply applied by calculating a moving average and it is, in our experience, only necessary to apply in cases with very low signal-to-noise levels. However, smoothing with a small smoothing window in cases where it is not strictly necessary will have very little effect on the final result. If the smoothing window is set too high, small peaks may be flattened and disregarded by FastChrom.

The peak finding algorithm was compared with three different versions of the peak finding algorithm incorporated in PLS\_toolbox (Eigenvector). All of the four peak finding algorithms gave similar results. As all the methods, like most other peak finding approaches, are using derivatives, this was to be expected, and the result points out that the choice of peak finding approach not is the critical part of FastChrom.

## Peak grouping

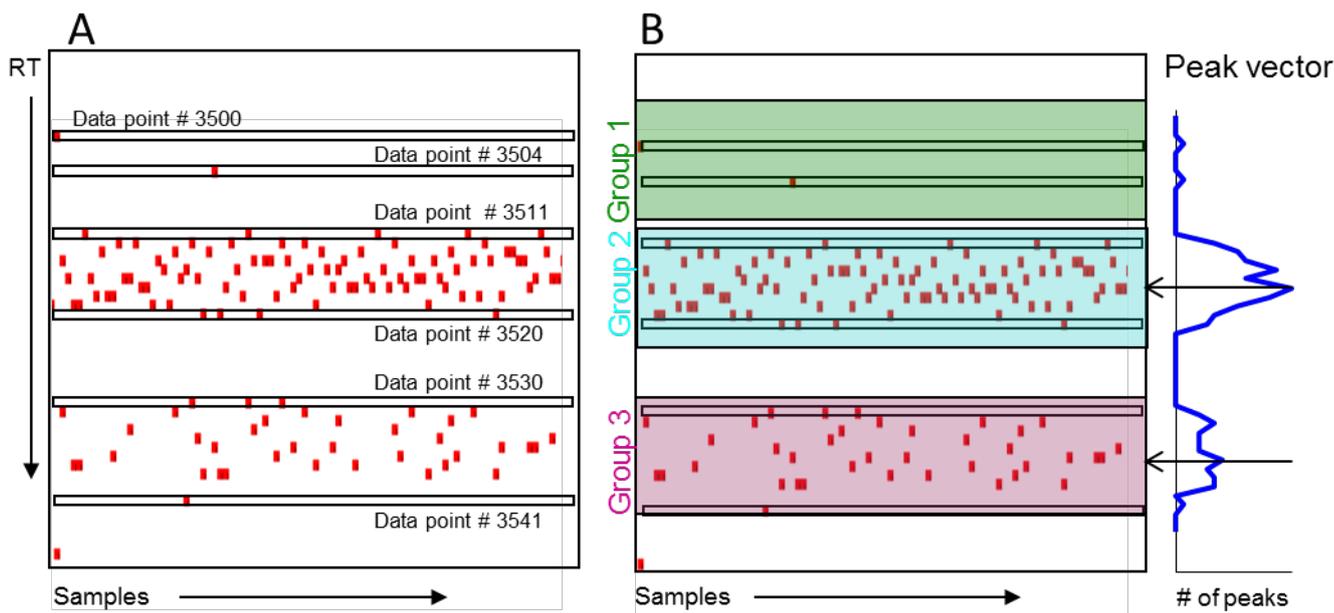
In order for two peaks, in different samples, to be recognised as originating from the same compound they must be identified as being the same. For data obtained from a single channel detector system the only way this can be determined is by using the similarity in retention time. It is assumed that peaks in two chromatograms having similar retention time are originating from the same chemical compound. Ideally such peaks would have apex in the same data point. However in practice, this is never the case. Therefore, one has to determine boundaries for how much the retention times for one specific compound vary across samples (sample to sample variation). To our knowledge there are no automated methods for this, but we propose the following approach.

When all peaks are detected, the sample to sample variation can be illustrated by colouring data points with peak apexes as shown

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

ARTICLE TYPE



**Figure 7.** A) Illustration of the variation in retention time of peaks across samples. Every red point illustrates a detected peak. In the example, at least two peak groups are represented. The variability of the two groups is at respectively 10 (from 3511 to 3520) and 12 (from 3530 to 3541) data points. The graph to the right shows the peak vector which shows the sum of peaks at each individual retention time. B) Shows how the grouping algorithm would group the illustrated peaks. The centres of the groups are controlled by the “density” of the peaks. The width of the window is set to 13.

in **Figure 7A**. The peak vector is defined as a vector indicating the sum of peak apexes across samples at each elution time (or data point). This peak vector is shown to the right in the figure and is used in the grouping.

In the peak grouping across samples, Peak Variability (PV) is the width of the grouping window. The parameter PV is defined as the maximum difference in retention time (in data points) for any given analyte. In the example shown in **Figure 7** a PV of 13 would be appropriate (the number must be odd). If the width of the window is set too narrow, peaks originating from the same compound will be placed in different groups. For example, e.g. group 2 in **Figure 7B** would be divided into two groups. On the other hand if the maximal shift is set too high, peaks would be wrongly grouped together. It is, in our experience, better to have the maximal shift a bit too low, since it is easier to recognise peak groups which have been divided, and manually merge these, than the other way around.

Whenever the peak grouping is performed on data from a new system, the PV should be determined in order to optimise the use of the method. The PV is found by manual inspection of the distribution of peaks, as illustrated in **Figure 7A**.

The peak grouping is based on the assumption that the data point with the highest number of peaks, within the specific cluster, is placed in the middle. The centre of the grouping windows is therefore controlled by the peak vector and placed at the data point with the highest number of peaks, as shown in the right

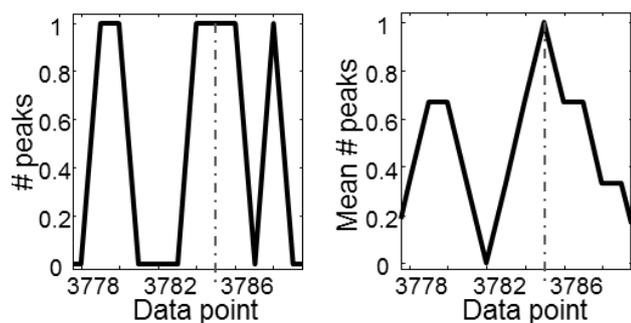
most plot (B) in **Figure 7**.

A so-called peak grouping algorithm has been developed. The algorithm searches for peaks from left to right in the peak vector. In the identification of peak groups, the parameter  $c$  is introduced which defines the data point where the previously found window stops. The parameter  $c$ , is used to prevent that one peak is placed in more than one group. The searching begins with setting  $c$  to zero (no prior window identified) and as the search continues  $c$  is changed so it reflects the location of the previous window (in this way  $c$  operates along the elution time axis).

The peak grouping algorithm consists of the following four steps. It searches along the elution time axis in the peak vector, which is created across samples. Consequently, peaks in all samples are considered in each search:

- 1 The first peak in the peak vector with a data point number ( $n$ ) higher than  $c$  is found.
- 2 The data point ( $n_{\max}$ ) with the highest number of peaks, in the range  $n$  to  $n + PV$ , is identified. If no data point in the interval is containing more than one peak,  $n_{\max}$  is set to the data point with the highest density of peaks by using a moving average (see **Figure 8**).
- 3 a) If  $n_{\max}$  is less than  $n + \frac{1}{2} PV$  the peaks in the interval  $n_{\max} - \frac{1}{2} PV$  to  $n_{\max} + \frac{1}{2} PV$  are grouped.  
b) If  $n_{\max}$  is higher than  $n + \frac{1}{2} PV$  the peaks in the interval  $n$  to  $n_{\max} - \frac{1}{2} PV$  are grouped.

4 c is set to  $n_{\max} + \frac{1}{2} PV$ , and the loop starts over until all peaks are grouped



5 **Figure 8.** Left: illustration of the peak vector in an interval with no more than one peak in each data point. Right: the moving average of the illustrated peak vector is shown. The moving average is used to locate the area with the highest density of peaks. In this example the centre of the grouping window is placed at data point number 3785 (dotted line).

#### 10 Retention index

Many of the traditional methods for evaluation of chromatograms require adjustment of the peak windows whenever the retention time shifts. This is not necessary with FastChrom since all peaks are ideally found. However, in order to ease identification, a retention index (RI) can be used instead of actual retention time. In cases where the retention time drifts the RI will remain relatively constant. The purpose with the introduction of RI in FastChrom is not to align within the sample batch, but to obtain a better identification of the peak groups, and to ease the comparison across experiments.

The Kovats index is widely used for adjusting for shifts in retention times between experiments. It is in general a reproducible and robust index, which takes advantage of the linear relationship between  $\log(\text{retention time})$  and the number of carbons in alkanes<sup>21</sup>. The Kovats index can be implemented using either internal or external standards. Internal standards are more accurate but introduce additional peaks and thereby elevate the risk of co-elution. With alignment by internal standards, additional correction of retention time (warping etc.) is in most cases not needed. The alternative approach, using external standards analysed before and after the samples, assumes a relatively constant retention time within the sequence. If the retention time is unstable within the experiments, retention time correction for the complete chromatograms (see the publication by Tomasi *et al.*<sup>22</sup> for details regarding COW and by Tomasi *et al.*<sup>23</sup> regarding *icoshift*) can be conducted before calculating the retention index.

We calculate the RI based on at least one RI standard containing a number of evenly distributed compounds with an established index number. The index number for the compounds in the sample will then be determined as relative retention times by

linear regression between the peaks in the RI standards. The composition of the RI standards used in this experiment consisted of 10 alkanes from 6C to 15C having indexes from 600 to 1500.

45 In addition, the standards also contain octanol and decanol with indexes at 1575 and 1809, respectively. The indexes for the two alcohols were determined with Total vaporization Head Space GC<sup>24</sup>, including hexadecane and octadecane in the sample.

#### Materials and methods

50 All programming has been performed in MATLAB 7.6.0 (R2008a) (Mathworks, Inc., Natick, Massachusetts, U.S.A.). Simulated data are obtained with the algorithm used by Komsta<sup>1</sup>. Real data are obtained from analysis of fermented milk analysed with a Perkin Elmer Autosystem XL GC coupled with a Perkin Elmer TurboMatrix110 Headspace sampler. The retention index has been validated with chemical standards mixed in water. Furthermore, we have used data obtained from analysis of wine samples originating from four different countries<sup>25,26</sup>, in validation of the entire method. In the original article 57 aroma compounds were extracted in the GC-MS manufacturer software, ChemStation (Agilent, Santa Clara, California, U.S.A.). PCA (Principal Component Analysis) models are calculated using PLS\_Toolbox 6.5.2 (Eigenvector Research, Inc., Wenatchee, Washington, U.S.A.).

#### 65 Results/Discussion

##### Baseline

##### Simulated data

Baseline fitting was performed on 5000 simulated chromatograms in order to evaluate the overall performance of the four tested methods. The performance was evaluated by calculating how much the signal obtained after baseline correction deviates from the known signal (in %) as well as how much the heights of the peaks in the baseline corrected signal, deviates from the heights of the peaks in the original signal (without baseline). This last comparison was also evaluated as % deviation.

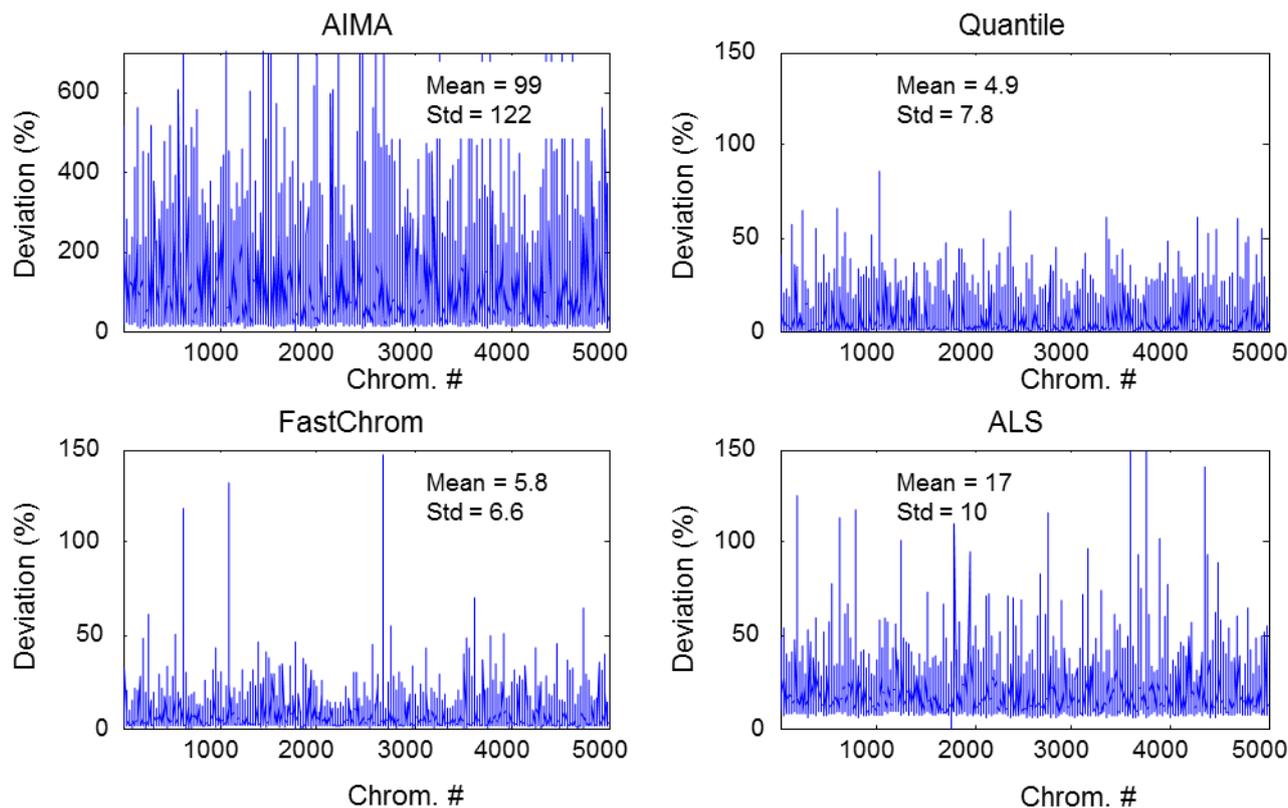
The deviations between known and obtained signal, shown in **Figure 9**, indicate that AIMA does not perform well on many of the simulated chromatograms. Quantile regression and FastChrom performed almost equally well with mean deviations on respectively 4.9 and 5.8 % and standard deviations on 7.8 and 6.6. ALS performed very similarly on all the simulated chromatograms, but with a higher mean deviation than FastChrom and quantile regression, indicating that the result will always be deviating from the true baseline. The comparison of peak heights shows more or less the same pattern, with the only exception that ALS and FastChrom are performing very similar.

Detailed results from the comparison of peak heights are shown in the supplementary material.

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

ARTICLE TYPE

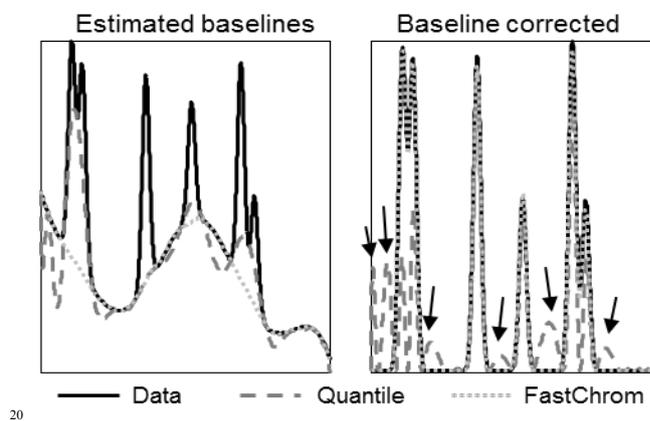


**Figure 9.** Deviation between obtained signal and known signal for four different baseline fitting methods. Top, left: performance of AIMA. Top, right: performance of quantile regression. Bottom, left: performance of FastChrom-baseline method. Bottom, right: performance of ALS.

5 In an attempt to find the weak spots of the methods, visual inspection of the chromatograms resulting in the worst baseline estimations was performed. This showed that the baseline fit is wrong throughout the entire chromatogram when quantile regression fails (**Figure 10**). In addition to this the fluctuation in the baseline estimated by quantile regression results in a number of pseudo-peaks when the chromatograms are corrected for baseline (arrows, **Figure 10**). Furthermore, the inspection showed that the cases where FastChrom failed were cases with a very steep increase in baseline followed by an equally steep decrease and a peak placed at the top of the “hill”.

10 The estimations performed by ALS are generally worse than those obtained with FastChrom and quantile regression. In addition to this, ALS has the major drawback that the determination of parameter values is not a trivial task.

15

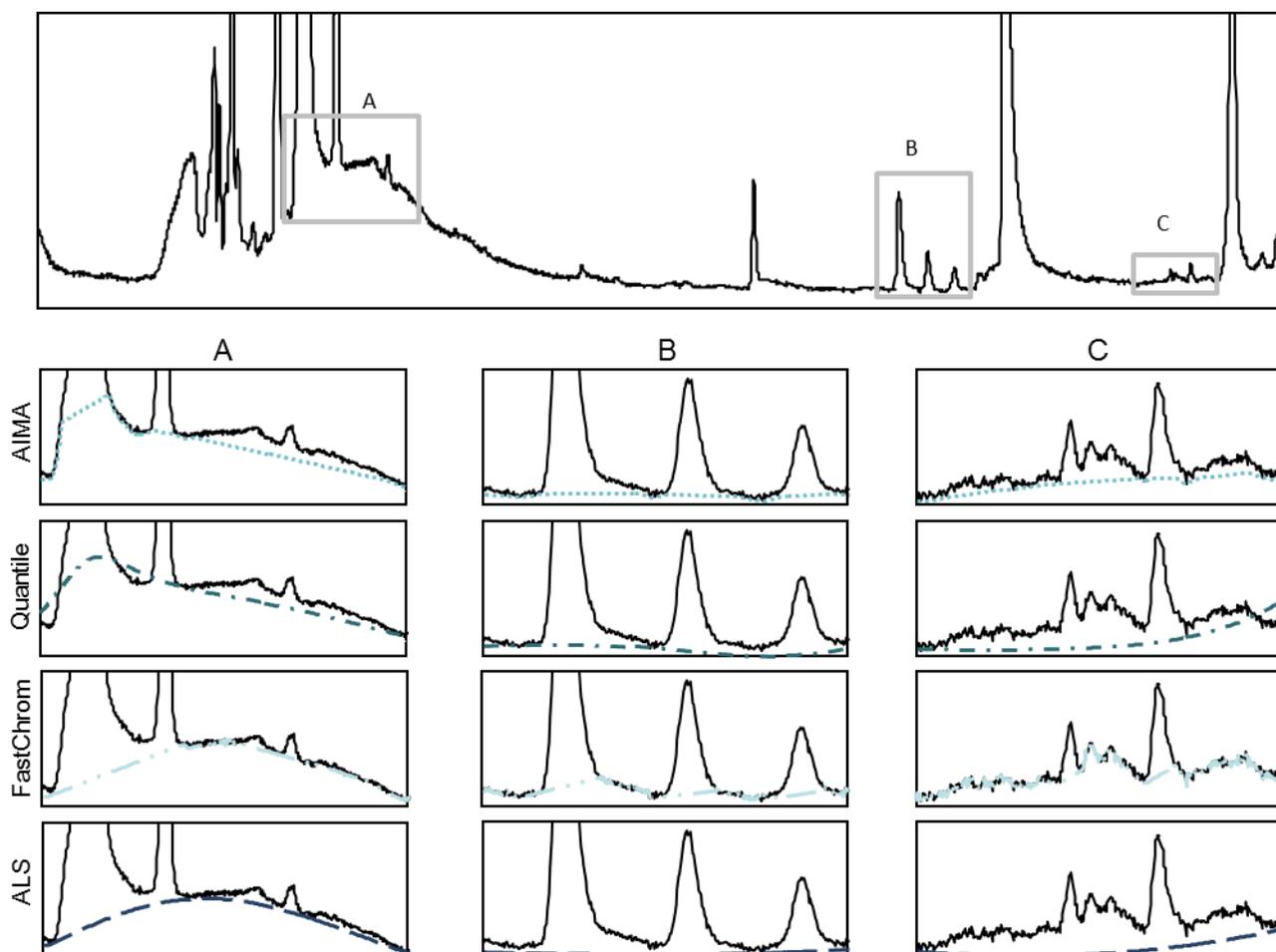


**Figure 10.** Examples of poor performance of quantile regression. The fluctuation in the baseline estimated with quantile regression results in several artificial peaks and affects the entire chromatogram

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

ARTICLE TYPE



**Figure 11.** A real chromatogram with a very diverging baseline. Region A illustrates a very complex baseline region, region B illustrates a nice behaving baseline, and region C illustrates a region with very low signal-to-noise ratio.

#### Real data

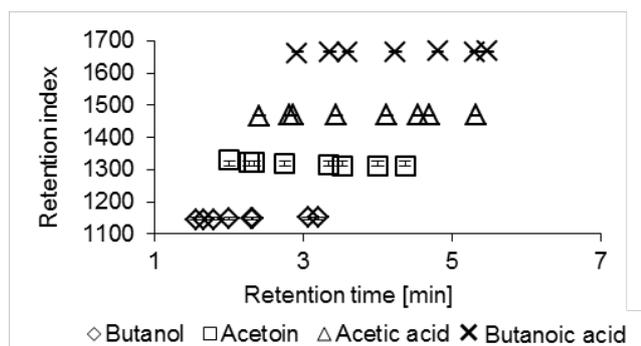
All four baseline methods were also tested on real data. As an example it is shown how the methods perform on a chromatogram with a relatively high noise level and with an unusual baseline. This example is chosen since all four methods perform well on well behaving chromatograms.

**Figure 11** (top) shows a complete chromatogram obtained from analysis of cheese. Three areas with very different baseline characteristics have been chosen. Example A illustrates an area with a very unusual baseline. Example B is an example of an almost ideal situation and example C is an example with very low signal-to-noise levels. In the lower part of the figure it is shown how the four methods are performing in the three different cases. As can be seen all of the methods are performing well in the example with the “well behaving” chromatogram (B). Example A illustrates how AIMA has a tendency to estimate the baseline too high under peaks. This effect is, to our experience, a general feature for the baselines fitted by AIMA, and sometimes also

occurs in well behaving chromatograms. It also illustrates how quantile regression fails to estimate a reasonable baseline due to a too high degree of flexibility. Both FastChrom and quantile regression perform well in this situation. However, example C is illustrating how both quantile regression and ALS are failing due to a too high degree of flexibility. Both FastChrom and AIMA are performing well in this case.

#### Retention Index

Validation of the robustness of the retention index was performed by a number of GC-FID analyses of the same sample, using different temperature profiles. The changes in the temperature profiles resulted in more than 50% changes in the retention times of the later eluting peaks. At each of the different temperature profiles, a RI standard was analysed before the sample. The RI standard was used to calculate the index for each of the peaks in the sample.



**Figure 12.** Stability of the RI across analysis with shifting retention times. The latest peak in the chromatogram is shifted by more than two minutes due to changes in the temperature program, but the index remains stable.

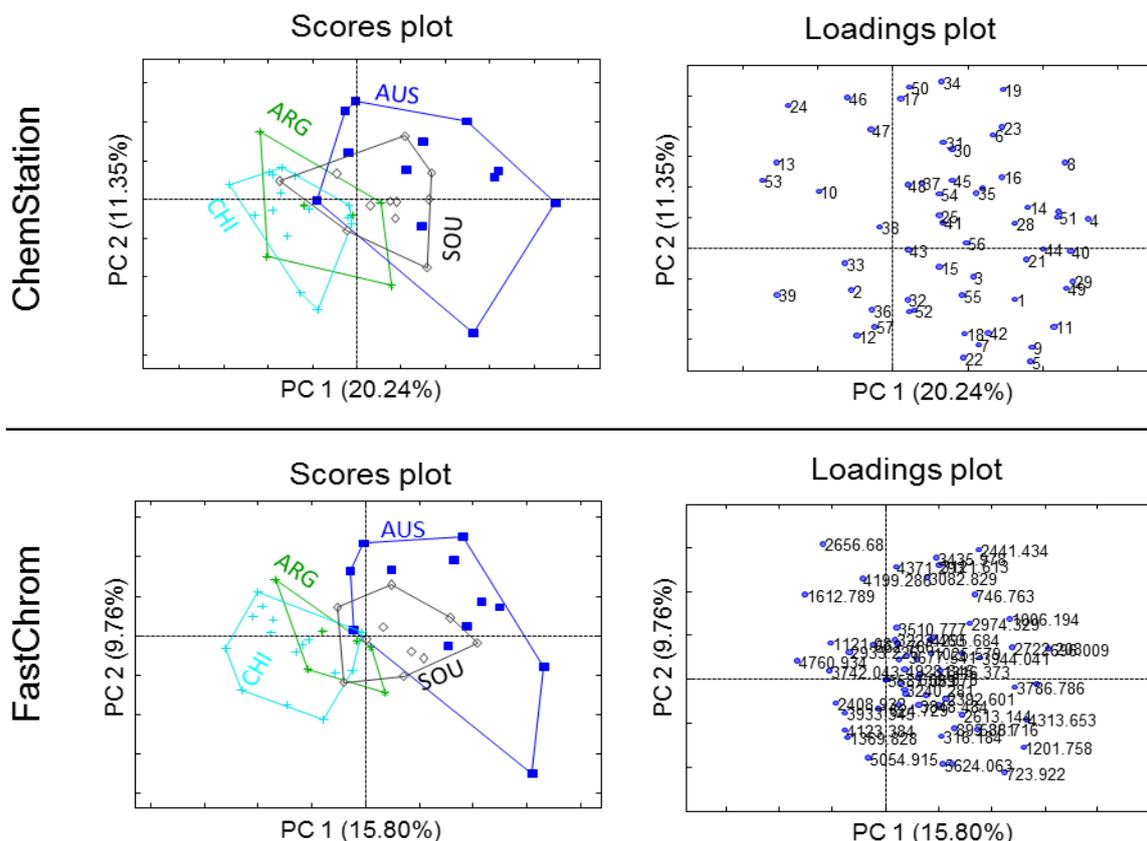
The variation in retention times and index is shown in **Figure 12**. Even though the retention time is changed with more than two minutes for the last peak, the use of index seems to normalise these differences to a very high degree.

### The complete FastChrom method

In order to validate the performance of the complete FastChrom method, it has been applied to TIC data from 44 different wine samples analysed with GC-MS<sup>25,26</sup>. When the data were

processed with the graphical user interface, a total of 85 compounds were extracted, while 57 compounds were found by ChemStation. In order to evaluate and indicate the relevance of the extracted compounds, Principal Component Analysis (PCA) was performed, respectively, on the 57 compounds extracted with ChemStation and the 85 compounds extracted with FastChrom (**Figure 13**).

The score plots from the two models only show small differences in the separation between the four countries. In order to be able to compare the two models, the within class variation and between class variation was calculated. The within class variation was calculated as the area of the convex hull covered by the individual classes. The results showed that the within class variation was lower in the PCA calculated on data from FastChrom (with a mean area of 24 compared to a mean area of 37 in the ChemStation model). The between class variation was evaluated as the number of samples not overlapped by other classes than its own (23 in the model based on data from FastChrom vs. 15 in the model based on data from ChemStation). These numbers show that also the between class variation is better in the model based on data from FastChrom. The PCA indicates that FastChrom performs as well as the manufacturing software at least in terms of providing information that can separate samples originating from different countries.



**Figure 13.** Principal Component Analysis models obtained from the compounds extracted respectively with ChemStation and FastChrom. AUS: Australia, ARG: Argentina, SOU: South Africa, CHI: Chile. Labels in loadings for the ChemStation samples are peak numbers while labels in the FastChrom loadings are referring to the data point numbers.

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

## ARTICLE TYPE

Besides the PCA, the performance of FastChrom has been validated by comparison of signal-to-noise ratios as determined by FastChrom and as determined by Totalchrom Navigator (Perkin Elmer, Inc., Waltham, Massachusetts, U.S.A.) on GC-FID data (Figure 14). This validation shows that the performance of these two methods is comparable

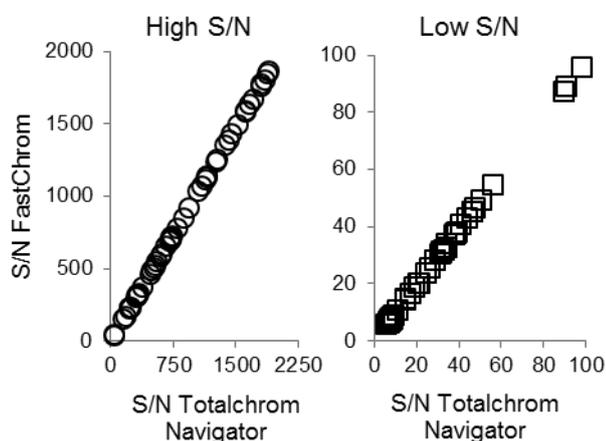


Figure 14. Comparison of Signal-to-Noise (S/N) found by Totalchrom Navigator with S/N found by FastChrom. It is clear that there is a nice linear relationship between the two, indicating that the performance of FastChrom is comparable with the performance of the manufacturing software.

## Conclusion

A novel automated method FastChrom for processing of GC-FID data has been proposed. The method consists of a baseline estimation, peak detection, peak grouping across samples and assignment of a retention time index. The method for baseline estimation is a new method, and it has been shown that the performance is better than the non-parametric methods AIMA and quantile regression, as well as the parametric method ALS.

FastChrom has been compared with peak extraction performed in traditional manufacturing software, and it has been shown that FastChrom performs at least as well as the traditional software. However, our method finds more peaks in the chromatogram and is easier and faster to apply.

FastChrom has been integrated in a Graphical User Interface (GUI) allowing users without MATLAB competences to use it. The GUI can import data in four different formats: RAX, CDF, MAT and XLS and can export height, width, retention time and retention index to an Excel spread sheet, or to PLS\_Toolbox where PCA or other multivariate methods can easily be applied. In addition, also the original positions of the un-grouped peaks are exported to excel. The user interface can be found at www.models.life.ku.dk/algorithms.

## Notes and references

- <sup>a</sup> Dept. Food Science, University of Copenhagen, Denmark. Tel: 0045 3533 3222; E-mail: rb@life.ku.dk  
<sup>b</sup> Chr. Hansen AS, Bøge Alle 10-12, 2970 Hørsholm, Denmark. Tel: 0045 4574 7474; E-mail: dklgj@chr-hansen.com
1. L. Komsta, *Chromatographia*, 2011, **73**, 721-731.
  2. B. D. Prakash and Y. C. Wei, *Analyst*, 2011, **136**, 3130-3135.
  3. Z. M. Zhang, S. Chen and Y. Z. Liang, *Analyst*, 2010, **135**, 1138-1146.
  4. J. Excoffier and G. Guiochon, *Chromatographia*, 1982, **15**, 543-545.
  5. K. H. Jarman, D. S. Daly, K. K. Anderson and K. L. Wahl, *Chemom. Intell. Lab. Sys.*, 2003, **69**, 61-76.
  6. S. J. Dixon, R. G. Brereton, H. A. Soini, M. V. Novotny and D. J. Penn, *J. Chemom.*, 2006, **20**, 325-340.
  7. G. Vivó-Truyols, J. R. Torres-Lapasió, A. M. van Nederkassel, Y. Vander Heyden and D. L. Massart, *J. Chromatogr. A*, 2005, **1096**, 133-145.
  8. J. L. Tian, M. Juhola and T. Gronfors, *Artificial Intelligence in Medicine*, 1997, **10**, 115-128.
  9. T. Frenzel, A. Müller and K. H. Engel, *Eur. Food Res. Technol.*, 2003, **216**, 335-342.
  10. P. H. C. Eilers, *Anal. Chem.*, 2003, **75**, 3631-3636.
  11. P. H. C. Eilers and H. F. M. Boelens, *Leiden University Medical Centre report*, 2005.
  12. J. Zhao, H. Lui, D. I. McLean and H. Zeng, *Applied Spectroscopy*, 2007, **61**, 1225-1232.
  13. R. Koenker and Jr., G. Basset, *Econometrica*, 1978, **46**, 33-50.
  14. R. Koenker and K. F. Hallock, *Journal of Economic Perspectives*, 2001, **15**, 143-156.
  15. R. Koenker, *Quantile regression*, Cambridge Univ. Press, Cambridge 2005.
  16. M. K. L. Bicking, *LC-GC North America*, 2006, **24**, 402-414.
  17. M. K. L. Bicking, *LC-GC North America*, 2006, **24**, 604-616.
  18. L. Jianwei, *J. Chromatogr. A*, 2002, **952**, 63-70.
  19. T. Barboni and N. Chiaramonti, *Chromatographia*, 2006, **63**, 445-448.
  20. J. M. Amigo, T. Skov and R. Bro, *Chem Rev.*, 2010, **11**, 4582-4605.
  21. K. Héberger, *J. Chromatogr. A*, 2007, **1158**, 273-305.
  22. G. Tomasi, F. v. d. Berg and C. Andersson, *J. Chemom.*, 2004, **18**, 231-241.
  23. G. Tomasi, F. Savorani and S. B. Engelsen, *J. Chromatogr. A*, 2011, **1218**, 7832-7840.
  24. B. Kolb and L. S. Ettre, *Static Headspace-Gas Chromatography: Theory and Practice*, 2nd Edition, Wiley-VCH., 2006.
  25. D. Ballabio, T. Skov, R. Leardi and R. Bro, *J. Chemom.*, 2008, **22**, 457-463.
  26. T. Skov, D. Ballabio and R. Bro, *Anal. Chim. Acta*, 2008, **615**, 18-29.

## Paper II

---

Maja H. Kamstrup-Nielsen, **Lea G. Johnsen**, and Rasmus Bro.

“Core consistency diagnostic in PARAFAC2”

*Accepted for publication in Journal of Chemometrics*



## Core consistency diagnostic in PARAFAC2

Maja H. Kamstrup-Nielsen<sup>a</sup>, Lea G. Johnsen<sup>a,b</sup>, and Rasmus Bro<sup>\*a</sup>

<sup>a</sup> Department of Food Science,  
University of Copenhagen, Denmark

<sup>b</sup> Chr. Hansen A/S, Denmark

\* Corresponding author

PARAFAC2 is applied in multiple research areas, e.g. where data containing shifts are analysed, but it is a challenge to determine the appropriate number of components in the model. In this paper it is hypothesized that the core consistency diagnostic, which is currently applied in e.g. PARAFAC1, can be used to determine model complexity in PARAFAC2. Theoretically a PARAFAC1 model is fitted 'inside' the PARAFAC2 algorithm and it should therefore be possible to apply the core consistency diagnostic from PARAFAC1 in PARAFAC2. To support this hypothesis three different datasets, as well as simulated datasets, have been evaluated by means of PARAFAC2 and the core consistencies have been investigated. There is a general trend that if the core consistency is low the model is over-fitted as in PARAFAC1. Also, core consistency captures the true variation in the data whereas small peaks are easily overlooked by visual inspection of noisy models. However, for determining the number of components in a PARAFAC2 model we suggest usage of the core consistency in combination with other model parameters such as residuals, loadings, and split-half analysis.

Key words: Core consistency; PARAFAC2; Number of components; Model complexity

## 1. Introduction

PARAllel FACtor analysis 2 (PARAFAC2) [1;2] has been applied in many different areas [3-5] and has for example proven to be useful for mathematical separation of overlapping chromatograms and to overcome issues in batch data with different temporal duration and dynamics. The main reason for applying PARAFAC2 is that it can sometimes model data containing shifts and related shape changes, e.g. chromatograms with shifts in retention time.

PARAFAC2 is closely related to PARAllel FACtor analysis (PARAFAC). In this paper the PARAFAC model will be denoted PARAFAC1, in order to distinguish PARAFAC2 from PARAFAC1 [1]. PARAFAC1 decomposes three-way data with low-rank trilinear structure into loading matrices which provide mostly unique estimates of the underlying variations in data. In PARAFAC2, data do not have to be low-rank trilinear – one of the directions in the data array can deviate in certain ways and still be meaningfully modelled by PARAFAC2 [2;6]. Despite the deviation from low-rank trilinearity, PARAFAC2 still provides unique estimates of the underlying latent variables under fairly mild conditions [7].

What remains a challenge in using the PARAFAC2 model is to determine the appropriate number of components. Harshman and De Sarbo [8] have proposed to use split-half analysis to determine the right number of factors. Split-half analysis can be considered as a type of resampling approach where PARAFAC2 is applied on different subsets of data. If the right number of factors is used, the result should be similar for all subsets. However, there are a number of drawbacks for split-half analysis. First of all, the subsets must be carefully selected. For instance all compounds must be present in all subsets in order for the resulting models to be similar. Another inconvenience is that the computation time increases when using split-half analysis.

For the PARAFAC1 model, the core consistency diagnostic is useful when determining the number of components. The core consistency diagnostic has been described by Bro and Kiers [9]. So far research

has dealt with the determination of model complexity in PARAFAC1 by means of core consistency, but the core consistency has never been incorporated in a PARAFAC2 setting and no similar alternative approaches to determination of model complexity have been suggested.

The objective of this paper is to develop an approach for calculating a model diagnostic similar to core consistency but for PARAFAC2 models. We will show that with some manipulations, we can define a core consistency value for a PARAFAC2 model. We will also investigate if this diagnostic can be applied to determine the number of components in PARAFAC2 models. First, the theory behind the structure of PARAFAC1 and PARAFAC2 will be outlined. Second, the theory behind and the relevance of the core consistency will be presented. Three examples on different data sets are given where core consistency is used to evaluate the model complexity. In addition the use of core consistency in PARAFAC2 is validated using simulated data.

## 2. Theory

PARAFAC1 is a multi-way method used to handle three-way (or multi-way in general) arrays and the principle is outlined e.g. by Harshman [10] and Bro [11].

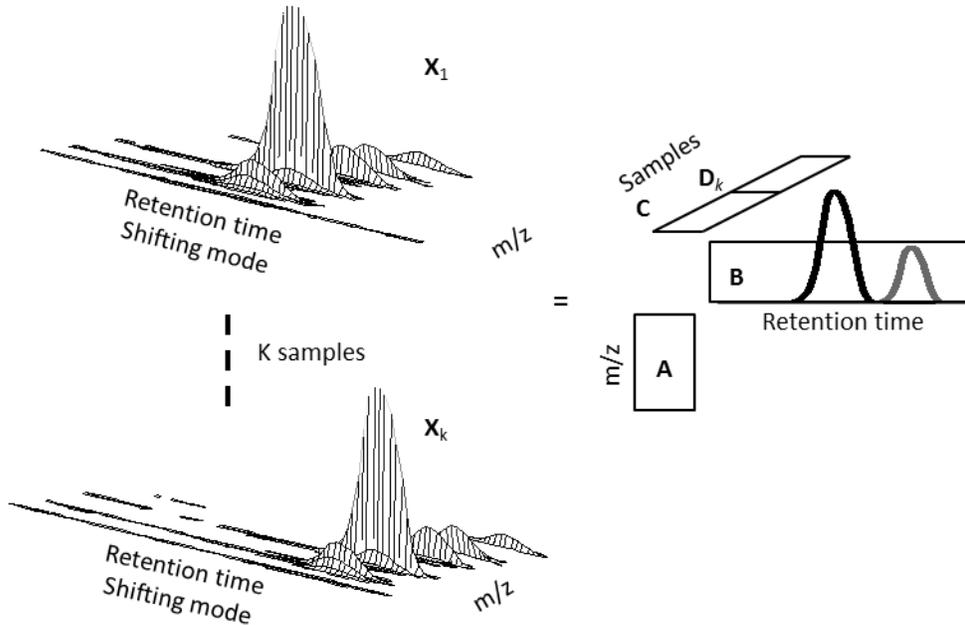
Let  $\mathbf{X}_k$  be an  $I \times J$  matrix with  $k = 1, \dots, K$  as the  $k^{\text{th}}$  slab of an  $I \times J \times K$  three-way array  $\underline{\mathbf{X}}$ .  $I$  is the number of observations (samples) in the first mode,  $J$  the number of variables in the second mode and  $K$  the number of variables in the third mode [2]. Using this terminology and disregarding noise for simplicity, the PARAFAC1 model has the following structure

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

Here **A** typically denotes the score matrix and **B** is the loading matrix for the second mode, which can be considered to correspond to the loading matrix in PCA. The extension from PCA then lies in the  $\mathbf{D}_k$  matrix which is a diagonal matrix of dimension  $R \times R$ , where  $R$  is the number of components. This matrix contains parameters from the loadings from the third mode. The loading matrix of the third mode is usually termed **C** ( $K \times R$ ) and  $\mathbf{D}_k$  holds the  $k^{\text{th}}$  row of **C** on its diagonal. In multi-way data analysis, the component matrices **A**, **B**, and **C** are oftentimes all called loading matrices. The term score matrix can then be introduced specifically for the loadings in the sample mode.

Unlike a bilinear model, PARAFAC1 provides unique estimates of its parameters **A**, **B**, and  $\mathbf{D}_1, \dots, \mathbf{D}_K$  under certain conditions without additional abstract constraints such as orthogonality which is used in PCA. The bilinear representation  $\mathbf{AB}^T$  has rotational freedom and PCA is only uniquely identified because of the additional constraints that are imposed on the parameters.

In order to set the stage for PARAFAC2, the PARAFAC1 model is illustrated in the following by means of a small part of GC-MS chromatographic data from Amigo et al. [3]. Instead of having samples in the first mode, as is common, these will be in the third mode for convenience of introducing PARAFAC2 subsequently.  $K$  chromatographic samples with  $I$  mass channels and  $J$  retention times have been modelled using a PARAFAC1 model. In this example, there is only one analyte present in the  $K$  samples, which is illustrated as the single peak in the second mode in Figure 1. However, in the  $k^{\text{th}}$  sample the retention time for this analyte is different from that of the first sample, which can also be seen in the second mode ( $J^{\text{th}}$  direction) in the figure. The second mode loading matrix, **B**, for a two-component PARAFAC1 model is supposed to contain estimates of the retention time profiles. Two components seem appropriate for this model, since each component in the second mode estimates the retention time profile in each sample. Hence, two components are necessary in order to extract the shifting information of the samples and thereby reveal the retention times of the analyte.



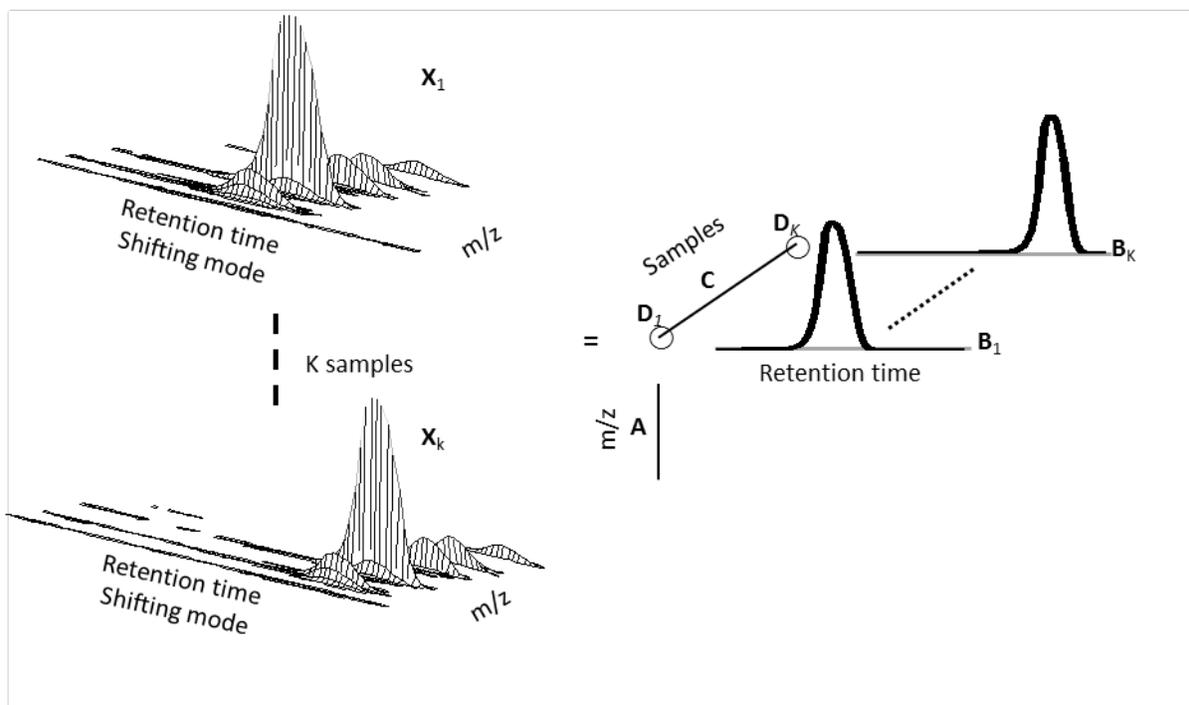
**Figure 1.** Chromatographic example to illustrate PARAFAC1. The data matrix  $\mathbf{X}_k$  is decomposed into estimates of the parameters  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{D}_k$  using two components.

In PARAFAC1 it is assumed that the loading matrix  $\mathbf{B}$  is representative of the underlying variation in all frontal slabs, i.e. that all slabs,  $\mathbf{X}_k$ , can be described in the row-space using the same  $\mathbf{B}$  ( $\mathbf{A}\mathbf{D}_k\mathbf{B}^T$ ). This means that for chromatographic data, the underlying retention time profiles of each analyte have to have identical shapes for each sample. This is not the case in the present example, where the samples as mentioned have shifting retention times, as illustrated in the second mode in the figure. Using PARAFAC1 on such data will typically lead to including more components than underlying chemical variations as seen in the example. These subsequent components can be difficult or impossible to interpret. Using the PARAFAC2 model is one way to circumvent such problems. The PARAFAC2 model can be written:

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{B}_k^T, k = 1, \dots, K \quad (2)$$

The parameters are almost identical to those of PARAFAC1. The only difference between equations (1) and (2) is the second mode loading matrix  $\mathbf{B}$ . In PARAFAC2  $\mathbf{B}_k$  is specific for every slab,  $k$ , in the third mode whereas  $\mathbf{B}$  is equal for all slabs in PARAFAC1. Note that residuals are not included in equation (2) for simplicity.

In Figure 2 it is illustrated how the use of a sample-specific  $\mathbf{B}_k$  matrix can help providing a more meaningful model of the shifting chromatographic data in Figure 1.



**Figure 2.** Same chromatographic example as illustrated in Figure 1. Here a PARAFAC2 model is fitted to data. Only one component is necessary.

In a PARAFAC2 model of these data, each sample will have its own retention time loading matrix  $\mathbf{B}_k$  and a one-component model is then sufficient to estimate the underlying retention time profile for the analyte present in the two samples regardless of the shift in retention time. All the information

concerning the shift is extracted by this component and the shift is modelled by the  $K$  different  $\mathbf{B}$  loadings.

However, the parameter estimates in PARAFAC2 would not immediately be unique if the model was only defined through equation (2). An additional constraint is also part of the model. The cross-product of  $\mathbf{B}_k$  ( $\mathbf{B}_k^T \mathbf{B}_k$ ) is required to be constant across  $k$  and it can be shown that this constraint leads to uniqueness of the model under mild conditions [7]. Constant cross-product across  $k$  is obtained by defining  $\mathbf{B}_k$  as

$$\mathbf{B}_k = \mathbf{P}_k \mathbf{H}, k = 1, \dots, K \quad (3)$$

where  $\mathbf{P}_k^T \mathbf{P}_k = \mathbf{I}$ , hence  $\mathbf{P}_k$  is orthogonal. The matrix  $\mathbf{P}_k$  handles what is unique for each sample in the shifting mode and  $\mathbf{H}$  handles what is related between samples [12]. With this definition the cross-product for  $\mathbf{B}_k$  will be constant because with an orthogonal  $\mathbf{P}_k$  it holds that,

$$\mathbf{B}_k^T \mathbf{B}_k = \mathbf{H}^T \mathbf{P}_k^T \mathbf{P}_k \mathbf{H} = \mathbf{H}^T \mathbf{H} \quad (4)$$

If we substitute  $\mathbf{B}_k$  in equation (2) with equation (3) we can rearrange the PARAFAC2 model in the following way

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k (\mathbf{P}_k \mathbf{H})^T \quad \Leftrightarrow$$

$$\mathbf{X}_k \mathbf{P}_k = \mathbf{A} \mathbf{D}_k \mathbf{H}^T \mathbf{P}_k^T \mathbf{P}_k \quad \Leftrightarrow$$

$$\mathbf{Y}_k = \mathbf{A} \mathbf{D}_k \mathbf{H}^T, k = 1, \dots, K \quad (5)$$

Equation (5) points to an interesting approach for understanding PARAFAC2. When the orthogonal  $\mathbf{P}_k$  matrices are known, we can rephrase the PARAFAC2 model as a PARAFAC1 model in terms of frontal slabs of data ‘compressed’ with their own specific  $\mathbf{P}_k$  matrix; hence, a PARAFAC1 model can be fitted on

a data array of  $\mathbf{Y}_k$  slabs. This is interesting in understanding how PARAFAC2 handles changes such as retention time shifts in the second mode and it is also useful for the purpose of this paper in developing a core consistency measure for PARAFAC2 models.

The number of components to use in a PARAFAC1 model can be estimated by means of the core consistency diagnostic [9]. PARAFAC1 can be considered as a constrained Tucker3 model [13] but where the core array has been fixed to a super-diagonal array of ones. The idea behind the core consistency diagnostic is to estimate what the core would actually have been if it was not constrained. This is estimated using the PARAFAC1 loadings as fixed loadings in a Tucker3 model; hence only estimating the core array. If this estimated core array is close to a super-diagonal of ones, we say that the core consistency is high and that the variation described by the PARAFAC1 model is indeed low-rank trilinear. If the core is very different, e.g. has high off-diagonal elements, then the core consistency is low and this indicates that the PARAFAC1 model, which presumably should be modelling low-rank trilinear variation, is really modelling other things as well. This indicates that this particular model is not suitable.

As mentioned previously, the PARAFAC2 model can be considered a PARAFAC1 model on ‘de-shifted’ data with slabs  $\mathbf{Y}_k$ . We hypothesize that the number of components can be equally well assessed from this PARAFAC1 model and that we can therefore use the straightforward core consistency of the PARAFAC1 model ‘inside’ PARAFAC2 as a tool for determining model complexity. In order to investigate the hypothesis, obtained core consistencies have been evaluated for the three different datasets.

### **3. Materials and methods**

All models and calculations were performed in Matlab 2012a (Mathworks, Inc., Natick, Massachusetts, U.S.A.). PARAFAC2 models were calculated with the algorithm from the N-way toolbox (available from [www.models.life.ku.dk](http://www.models.life.ku.dk), July 2012).

### **Fluorescence amino acid data**

The first dataset consists of five samples, each containing tyrosine, tryptophan, and phenylalanine in different amounts. Each sample has been measured on a PE LS50B spectrofluorometer (excitation 240-300 nm, emission 250-450 nm). The dimensions of the dataset are 5 (samples) × 201 (emission) × 61 (excitation).

### **Chromatographic wine and apple data**

The second dataset consists of 36 apples ripened for respectively five, eight and 15 days and the samples are analysed using HS-GC-MS. The details concerning the analysis can be found in [14]. The dataset has the dimensions 154 (masses) × 5033 (retention times) × 36 (samples).

The last dataset consists of 24 samples of red wine. The aroma profiles of the samples were measured using dynamic headspace gas chromatography coupled to a mass spectrometer (HS-GC-MS). Details concerning the measurements can be found in the original papers [3;15]. The dimensions of the dataset are 200 (masses) × 6000 (retention times) × 69 (samples).

## **4. Results**

The use of core consistency in PARAFAC2 has been tested using three different data sets: fluorescence amino acid data [11], chromatographic apple data [14], and finally chromatographic wine data [15]. In addition the core consistency has been tested on simulated data.

### **4.1. Fluorescence**

In this dataset there are no shifts, so the result from PARAFAC2 should be similar to that of PARAFAC1. This enables us to compare the core consistencies obtained from the two methods.

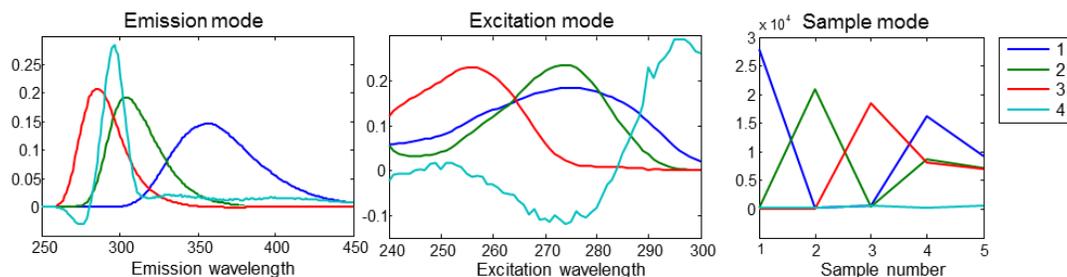
## Results and discussion

The models have been calculated without any constraints and with the samples in the last mode. The core consistencies and the explained variances of the models are seen in Table I.

**Table I.** Overview of the core consistencies and the explained variances for PARAFAC1 and PARAFAC2 models with one to five factors in the fluorescence amino acids data without shifts.

Model	No. of factors	Core consistency	% Fit
PARAFAC1	1	100	64.39
	2	100	86.77
	3	99.87	99.94
	4	92.49	99.95
	5	< 0	99.96
PARAFAC2	1	100	67.03
	2	100	92.94
	3	100	99.96
	4	< 0	99.97
	5	< 0	99.98

The core consistencies for the models from the PARAFAC1 algorithm indicate that four factors are appropriate for the dataset. Since the data are obtained from simple samples only containing three different amino acids, it would be expected that three factors would be appropriate. Visual inspection of the model (Figure 3) shows that the emission and excitation profiles for the fourth factor have some negative values. In addition the profile for this compound seems rather noisy also indicating that this model is over-fitted. It is likely that the fourth component is related to the small amount of Rayleigh scattering that is present in the data. In any case, for both the three- and the four-component models, the three main components come out similarly. The fourth extra component is of such a small magnitude that it does not affect the modelling of the three main components.



**Figure 3.** Illustration of the obtained PARAFAC1 model with four factors. Both the emission and excitation loadings for the fourth factor are rather noisy and have negative values, indicating that the model is over-fitted.

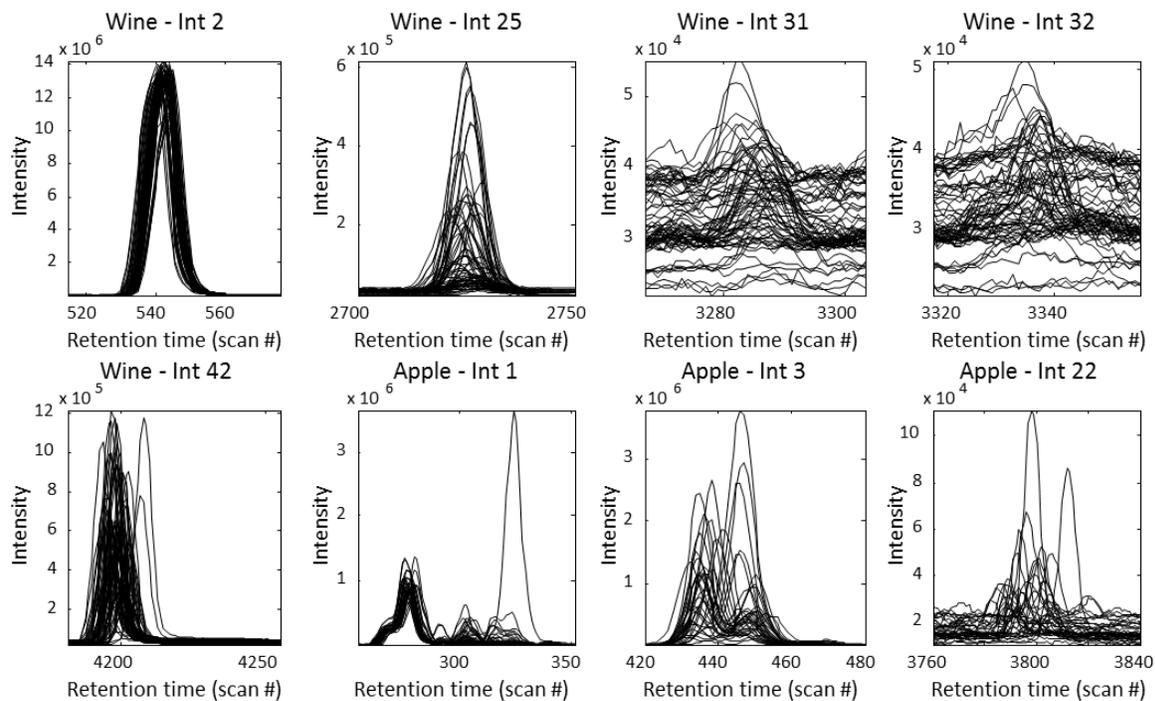
The core consistencies for the PARAFAC2 models indicate that three factors are appropriate for this dataset and the visual appearance of this three-component model is also appropriate (not shown). Hence, core consistency seems to be useful for assessing the number of components for this data set. The fact that normal PARAFAC1 and PARAFAC2 do not have the same behaviour with respect to the small and somewhat spurious fourth component is not surprising. The Rayleigh scattering that leads to the fourth PARAFAC1 component is not low-rank trilinear and hence is not expected to affect a PARAFAC1 and a PARAFAC2 model in a similar fashion.

Be aware that models, which have not converged or have converged in a local minimum, can result in a core consistency which is artificially low, and it is therefore very important to make sure that the model has converged and has reached the global minimum when core consistency is used in evaluation of model quality. A simple ad hoc approach to this is to repeat the PARAFAC2 algorithm a number of times and make sure that the best-fitting model is obtained several times.

## 4.2. Chromatography

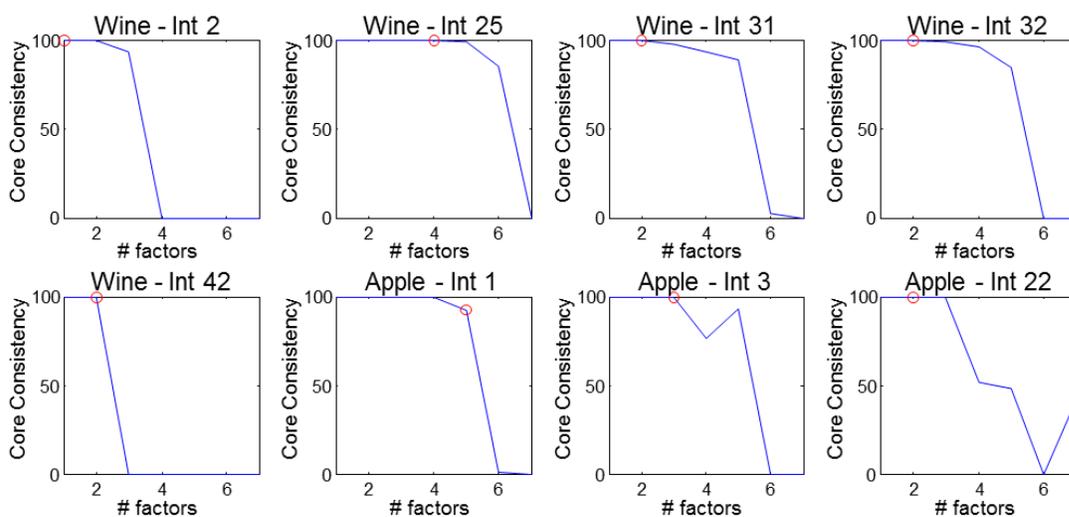
### Results and discussion

The apple and wine datasets are very large and consist of several peak regions. Each dataset is divided into smaller parts and PARAFAC2 models are fitted on these subsets individually. The apple data are divided manually into 26 intervals and the wine data into 50 intervals. The intervals chosen reflect a wide range of different features; overloaded peaks (e.g. wine interval 2), low signal-to-noise levels (e.g. wine intervals 31 and 32), minimal shifts in retention time (e.g. wine intervals 25 and 31), severe shifts in retention time (e.g. wine interval 42 and apple interval 3), and very complex intervals including several peaks (e.g. apple intervals 1 and 22). Intervals representing the different features are shown in Figure 4. To illustrate the features of core consistency, intervals 31 and 32 from the wine data and interval 1 from the apple data are illustrated in some detail below.



**Figure 4.** Examples showing a selection of the 76 intervals. The intervals cover peaks with both low and high signal-to-noise ratios, different degrees of shift, and different degrees of complexity.

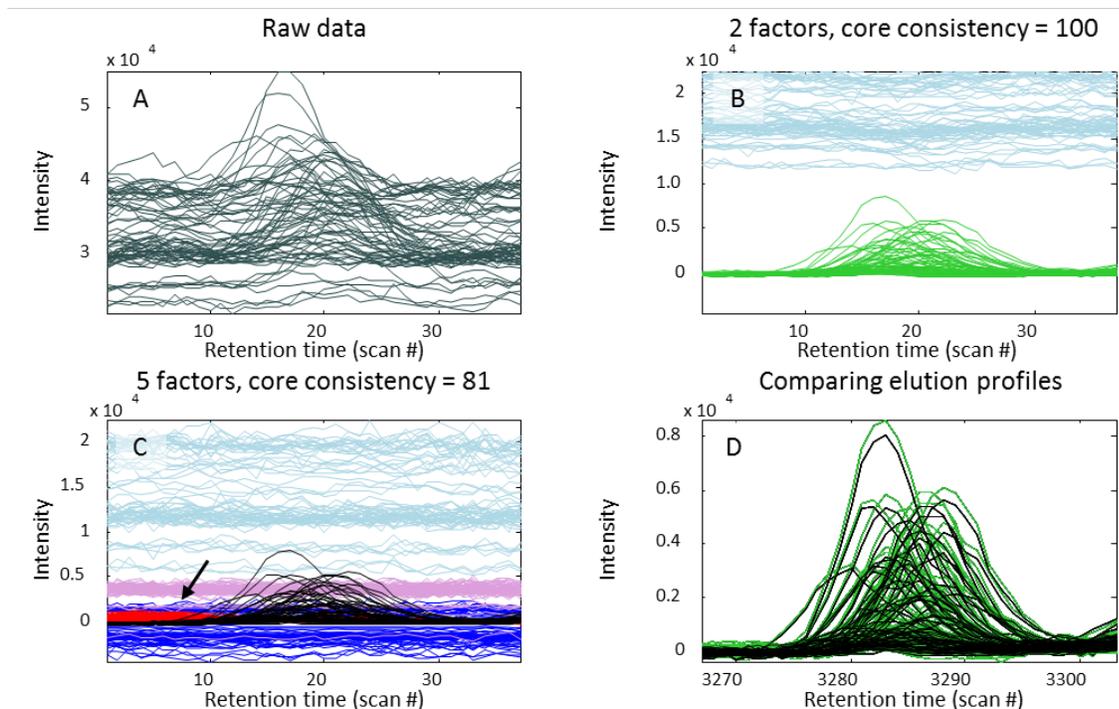
Core consistency was calculated for models with one to seven factors for all the 76 intervals, and all of the intervals were manually inspected in order to find the models with the optimal number of factors. Parts of the obtained core consistencies are shown in Figure 5. Models were evaluated based on residual analysis, as well as inspection of elution profiles and spectra obtained from the models.



**Figure 5.** Examples showing a selection of the obtained core consistencies, the remaining can be found in the supplementary material. The circles are indicating models with the optimal number of factors as initially decided by the authors. The line indicates the core consistency of each interval with the number of factors included in the model going from one to seven.

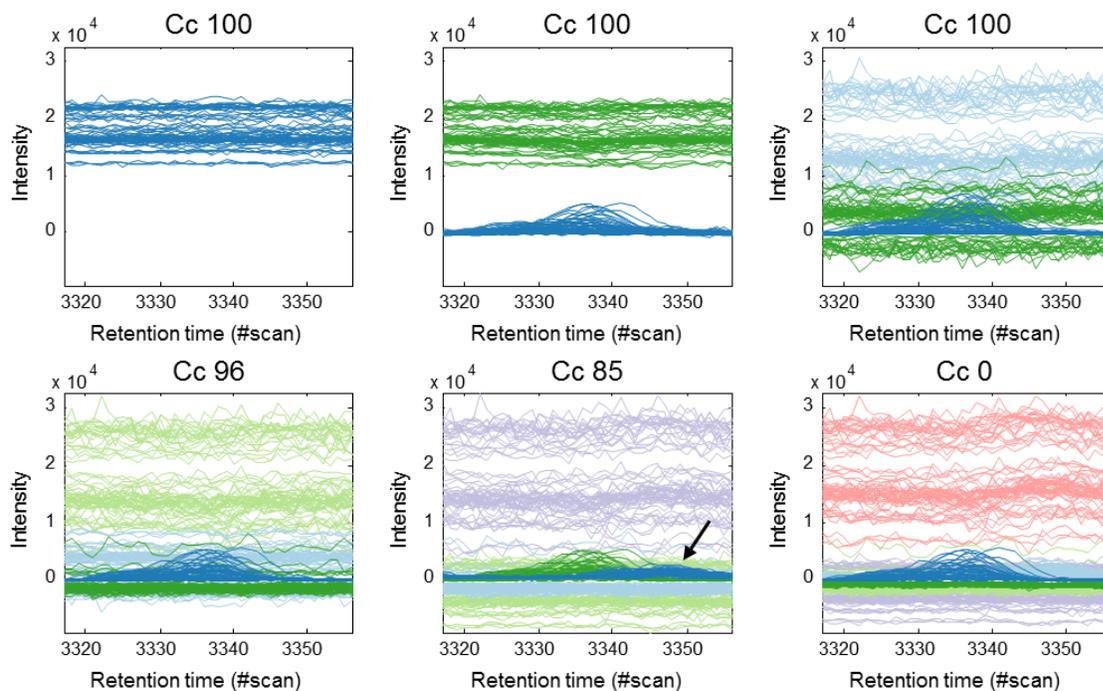
In agreement with the publication by Amigo et al. [14], we find that interval 1 in the apple dataset is best described with five factors (elution profiles not shown). As shown in Figure 5 the core consistencies are high for the models with one to five factors and low for the models with six and seven factors. So for this interval it seems like the core consistency is a useful tool in the determination of the model complexity.

Manual inspection of the models calculated on interval 31 from the wine dataset, suggests that a PARAFAC2 model with two factors is optimal (see elution profiles in Figure 6B). However, core consistency is indicating that five factors are optimal even though the five factor model is apparently over-fitted (see elution profiles in Figure 6C). Please note that the component which does not describe a peak in the two factor model is not indicating over-fit, but is merely describing the baseline, which in this case is rather high compared to the height of the peak.



**Figure 6.** A) Raw data from wine, interval 31. Elution profiles obtained with a two factor (B) and a five factor (C) PARAFAC2 model. The arrow indicates a compound which only appears in the five factor model. D) Illustration of the similarity between the main peak described by the models with two and five factors.

Figure 6D shows the estimated main peak from the two models illustrated in Figure 6. Clearly the two factor model and the five factor model capture the same elution profile. The spectral profiles as well as the concentration profiles (plots not shown) support that it is the same chemical variation which is described by the two models. This tendency is also seen for other seemingly over-fitted models. In the five factor model describing interval 31, the three 'additional' components simply describe baseline. The last component seems to describe a small peak which is only detected in the five factor model. The same behaviour with high core consistency is observed in the models calculated on interval 32 from the wine data. The elution profiles from the models of this interval with one to six factors are shown in Figure 7.



**Figure 7.** Interval 32, wine: elution profiles of models with one to six factors. The core consistencies (Cc) are high with exception of the last model with six factors. Notice that in the model with five factors an additional small peak appears (indicated with the arrow).

The inspection of the seemingly over-fitted models from intervals 31 and 32 with high core consistencies shows that in both models an additional factor actually appears, but it is very small and therefore difficult to locate (Figure 6C, arrow and Figure 7, arrow). In these cases it seems like the data contain noise and artefacts which contribute more to the variation than the lastly described small peaks. The presence of these additional compounds is supported when the mass channels in the raw data are inspected (not shown). Nothing indicates that these peaks are not chemical compounds present in the samples and therefore it would be appropriate to use five factors in both intervals.

The results support that core consistency actually captures the true variation in the data, whereas a visual inspection might put too much emphasis on the noise. Thereby small but potentially important

peaks may be overlooked. When analysing all the intervals with low signal-to-noise ratios, the same conclusion can be made; hence more factors than initially determined need to be included if all chemical variation is to be captured as suggested by core consistency.

### **4.3. Simulated data**

When calculating models on real data it can be difficult to determine the true rank of the data. Therefore we have included results from PARAFAC2 models of simulated data as well.

In the original paper concerning core consistency in PARAFAC1 [9], calculations on simulated data were also included. The authors showed that core consistency does not work very well on perfect data; meaning data that follow the PARAFAC1 model and only has additional random identically distributed Gaussian noise. It was argued that this problem was of limited consequence as 1) perfect data are simple to model in any case and 2) it is very rare that such data are met in practice. This was also supported by the fact, that the problems observed with ideal data were not observed for any of the quite diverse example data sets.

In order to assess core consistency in the original publication, a certain amount of model error was introduced in the simulated data to more adequately simulate real data. The model error introduces variance resulting in data which are not truly trilinear.

A similar approach is adopted here. Data with different ranks (three and five) and different congruence values [16] (0, 0.20, 0.50, and 0.90) were generated, in order to cover varying types of data, by creating a  $\underline{Y}$  array according to equation (5). The components in these data were drawn from a Gaussian distribution and in addition i.i.d. noise was added to the  $\underline{Y}$  array in “low” and high levels (15% and 40%, respectively). Then three levels of model errors were introduced (5%, 10%, and 15%) to affect the

trilinearity of the data. Subsequently each slab of  $\underline{Y}$  was multiplied by an orthogonal  $\mathbf{P}_k$  matrix to simulate PARAFAC2 data. This resulted in data arrays of size  $10 \times 15 \times 30$ . One hundred datasets were created for each combination of rank, Gaussian noise, model error, and congruence values. For the rank three data, PARAFAC2 models with one to five factors were calculated and for each model, the core consistency was determined. Similarly for the rank five data, PARAFAC2 models with one to seven factors were calculated.

Upon inspection of the obtained models, it was found that models with congruence values of 0, 0.20, and 0.50 in general fit the raw data quite accurately. However, this was, in most cases, not the case for models calculated on data with high congruence (0.90) – this was also reflected in the core consistencies. For these models, the core consistencies are in general very low for models with too few factors included (some core consistencies are below zero), regardless of the different model errors and noise levels introduced. Real data are oftentimes correlated, but a congruence value of 0.90 is quite high and the problem has not been observed when calculating the core consistency in the three real data sets. The high congruence data is not considered further here, but may point to a limited usefulness of core consistency with highly correlated data.

The models based on the remaining data (congruence values of 0, 0.20, and 0.50) are summarized according to core consistency in Table II. Since 100 models have been calculated for each combination of rank, congruence, noise, and model error, the core consistencies are presented as averages calculated on core consistencies where all negative values are set to zero. Otherwise core consistencies with very high negative values would dominate the obtained mean value. Positive core consistencies are included as is.

The averaged core consistencies in the table show that there is a significant drop in core consistency when the number of factors exceeds the rank of the raw data, suggesting that core consistency indeed

can be used as an indication of over-fit. However, the core consistencies rarely approach zero, and in some cases the starting point is quite low for the core consistency, i.e. for rank five data with high noise and high model error. Nevertheless, there is still a drop in the average core consistency when the number of factors included exceeds the true rank of the data.

**Table II.** Summary of averaged core consistencies from simulated models with different properties. The gray areas mark over-fitted models

				Noise level: Low			Noise level: High			
				Model error:			Model error:			
				Low	Med.	High	Low	Med.	High	
Rank 3	Congruence	0	Factors	2	100	100	100	100	100	100
			3	100	100	100	100	100	100	
			4	73	36	64	80	71	78	
	0.20	Factors	2	100	100	100	98	100	96	
		3	100	100	100	100	100	100		
		4	43	36	40	70	64	66		
	0.50	Factors	2	79	97	76	70	100	69	
		3	78	91	74	78	96	75		
		4	35	13	31	65	43	61		
Rank 5	Congruence	0	Factors	4	100	100	100	97	98	97
			5	100	100	100	99	99	99	
			6	72	71	64	72	76	78	
	0.20	Factors	4	99	99	100	94	93	90	
		5	99	99	99	72	96	96		
		6	9	7	6	43	47	47		
	0.50	Factors	4	51	50	61	41	44	42	
		5	58	48	58	41	49	55		
		6	12	13	7	22	27	19		

The observations mentioned above indicate that core consistency can be used to find the true rank of data with high and low signal-to-noise ratios and different levels of correlations within the data. When

compared to the simulation results in the original publication, the results also indicate that core consistency under certain circumstances may be less effective when used for model selection with PARAFAC2 than with PARAFAC1.

## 5. Conclusion

After evaluating the suggested core consistency diagnostic on several PARAFAC2 models from different real as well as simulated datasets, we conclude that core consistency is a helpful parameter in the evaluation of PARAFAC2 models. In some cases, usage of core consistency provides a better estimation of the underlying features than solely visual inspection. However, core consistency should not be used as the only measure of model complexity. It should be combined with additional measures or parameters such as investigation of residuals and loadings.

### Reference List

1. Harshman RA. PARAFAC2: Mathematical and Technical Notes. *UCLA Working Papers in Phonetics* 1972; **22**: 30-44.
2. Kiers HAL, Ten Berge JMF, Bro R. PARAFAC2 - Part I. A direct fitting algorithm for the PARAFAC2 model. *J.Chemom.* 1999; **13**: 275-294.
3. Amigo JM, Skov T, Coello J, MasPOCH S, Bro R. Solving GC-MS problems with PARAFAC2. *Trends in Analytical Chemistry* 2008; **27**: 714-725.
4. Matero S, Poutiainen S, Leskinen J *et al.* Monitoring the wetting phase of fluidized bed granulation process using multi-way methods: The separation of successful from unsuccessful batches. *Chemometrics and Intelligent Laboratory Systems* 2009; **96**: 88-93.
5. Robeyst N, Grosse CU, De Belie N. Monitoring fresh concrete by ultrasonic transmission measurements: Exploratory multi-way analysis of the spectral information. *Chemometrics and Intelligent Laboratory Systems* 2009; **95**: 64-73.
6. de Juan A, Tauler R. Comparison of three-way resolution methods for non-trilinear chemical data sets. *J.Chemom.* 2001; **15**: 749-772.

7. ten Berge JMF, Kiers HAL. Some Uniqueness Results for PARAFAC2. *Psychometrika* 1996; **61**: 123-132.
8. Harshman RA, De Sarbo WS. An Application of PARAFAC to a Small Sample Problem, Demonstrating Preprocessing, Orthogonality Constraints, and Split-Half Diagnostic Techniques. In: *Research Methods for Multimode Data Analysis* eds Law HG, Snyder CW, Hattie JA, McDonald RP, New York: Praeger, 1984: 602-642.
9. Bro R, Kiers HAL. A new efficient method for determining the number of components in PARAFAC models. *J.Chemom.* 2003; **17**: 274-286.
10. Harshman RA. Foundations of the PARAFAC procedure: Models and Conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics* 1970; **16**: 1-84.
11. Bro R. PARAFAC. Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems* 1997; **38**: 149-171.
12. Bro R, Andersson CA, Kiers HAL. PARAFAC2 - Part II. Modeling chromatographic data with retention time shifts. *J.Chemom.* 1999; **13**: 295-309.
13. Tucker LR. Some Mathematical Notes on 3-Mode Factor Analysis. *Psychometrika* 1966; **31**: 279.
14. Amigo JM, Popielarz MJ, Callejon RM *et al.* Comprehensive analysis of chromatographic data by using PARAFAC2 and principal components analysis. *Journal of Chromatography A* 2010; **1217**: 4422-4429.
15. Skov T, Ballabio D, Bro R. Multiblock variance partitioning: A new approach for comparing variation in multiple data blocks. *Anal.Chim.Acta* 2008; **615**: 18-29.
16. Lorenzo-Seva U, ten Berge JMF. Tucker's Congruence Coefficient as a Meaningful Index of Factor Similarity. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences* 2006; **2**: 57-64.

## Supporting information

### Core consistency diagnostic in PARAFAC2

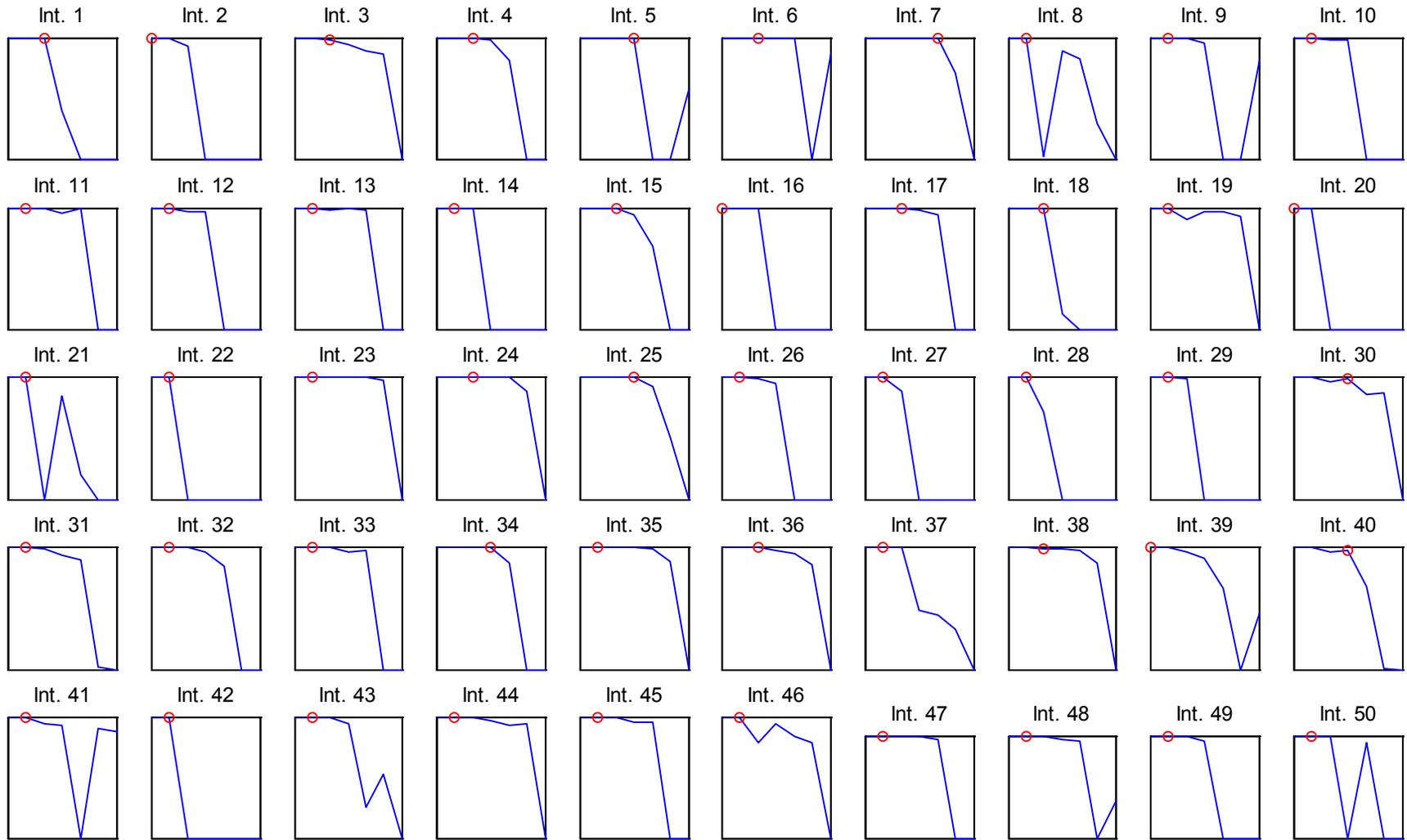
Maja H. Kamstrup-Nielsen<sup>a</sup>, Lea G. Johnsen<sup>a,b</sup>, and Rasmus Bro<sup>\*a</sup>

<sup>a</sup> Department of Food Science,  
University of Copenhagen, Denmark

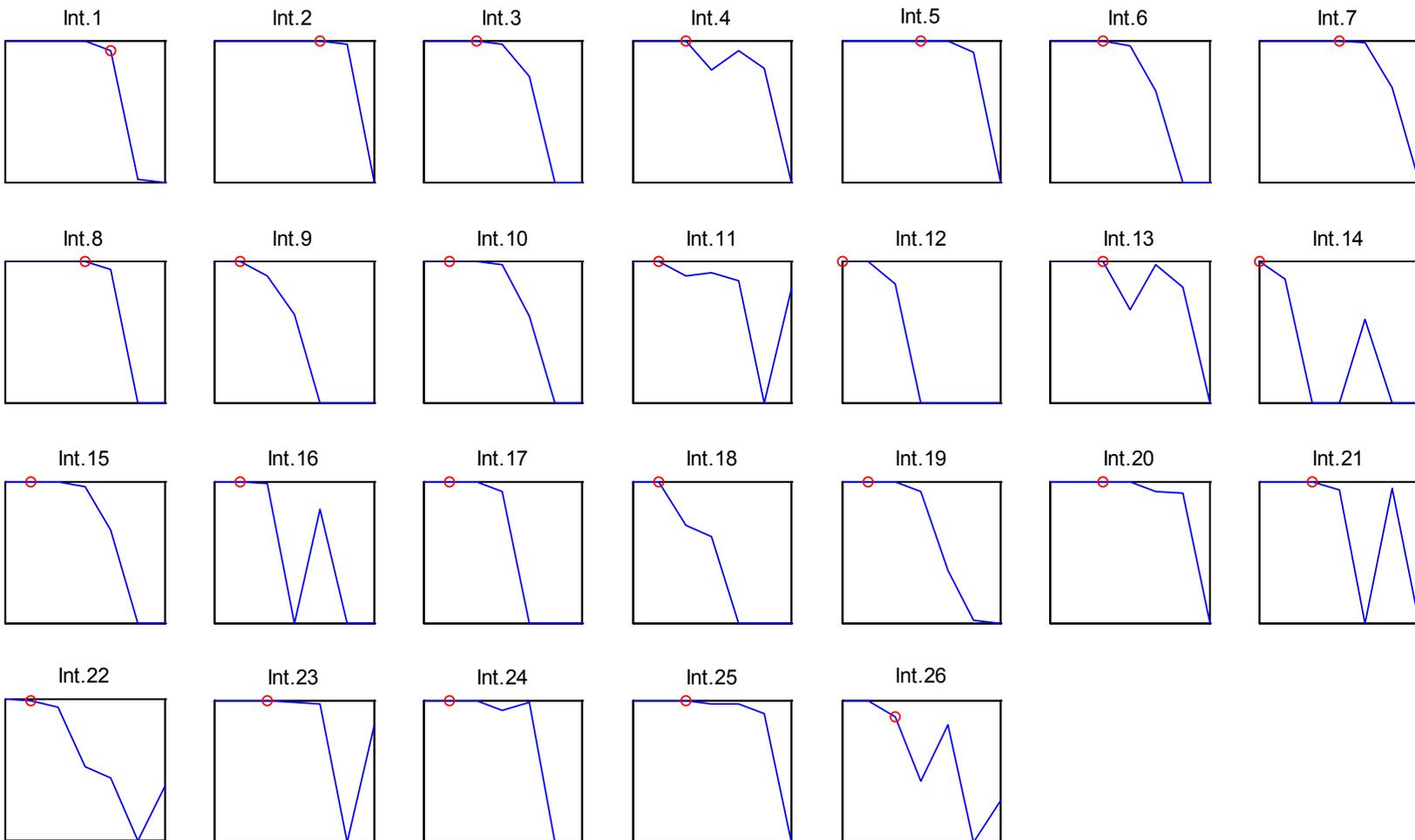
<sup>b</sup> Chr. Hansen A/S, Denmark

The supporting information contains figures with the obtained core consistencies for all intervals from the apple and wine data.

# Wine data



# Apple data



## Paper III

---

Rasmus Bro, Riccardo Leardi, and **Lea G. Johnsen.**

“Solving the sign-indeterminacy for multi-way models”

*Accepted for publication in Journal of Chemometrics*



# Solving the sign-indeterminacy for multi-way models

---

Rasmus Bro\*<sup>1</sup>,  
Riccardo Leardi<sup>2</sup>,  
Lea Giørtz Johnsen<sup>1,3</sup>

<sup>1</sup>Dept. Food Science, University of Copenhagen, Denmark.

<sup>2</sup>Department of Pharmacy, University of Genoa, Italy.

<sup>3</sup>Dept. Assays, Chr. Hansen A/S, Denmark.

## Abstract

Bi- and multilinear models such as PCA and PARAFAC have intrinsic sign indeterminacies. For example, any loading vector can be multiplied by minus one if another vector of that particular component is also multiplied by minus one without affecting the loss function values. This sometimes causes problems, e.g., with respect to interpretation. In this paper, a method is developed to fix the sign indeterminacy for the PARAFAC, Tucker3 and PARAFAC2 models.

## Introduction

Latent variable models such as PCA [1-3], PARAFAC [4,5] and Tucker [6,7] have intrinsic sign indeterminacies. E.g., in a PCA model it holds that the scores ( $\mathbf{T}$ ) and loadings ( $\mathbf{P}$ ) are found to minimize the least squares loss function  $\|\mathbf{X}-\mathbf{TP}^T\|_F^2$ . This loss function is identical to  $\|\mathbf{X}-(-\mathbf{T})(-\mathbf{P}^T)\|_F^2$  and hence, the score matrix can be exchanged with  $-\mathbf{T}$  as long as the loading matrix is replaced with  $-\mathbf{P}$ . Mathematically, there is no way to distinguish the two solutions.

In a two-way model, changing the sign of e.g. the score vector of component number three is explicitly countered by having to change the sign of the corresponding third loading vector. Samples that have high

positive scores on this original component have high values on the variables with high loading elements. This interpretation is unchanged, even when signs are switched, so the sign indeterminacy is of moderate consequence for a two-way model. In Bro et al. [8] a method was developed to assign meaningful signs to scores and loadings in PCA models. In this paper, this method is further developed to allow a similar sign correction of common multi-way models.

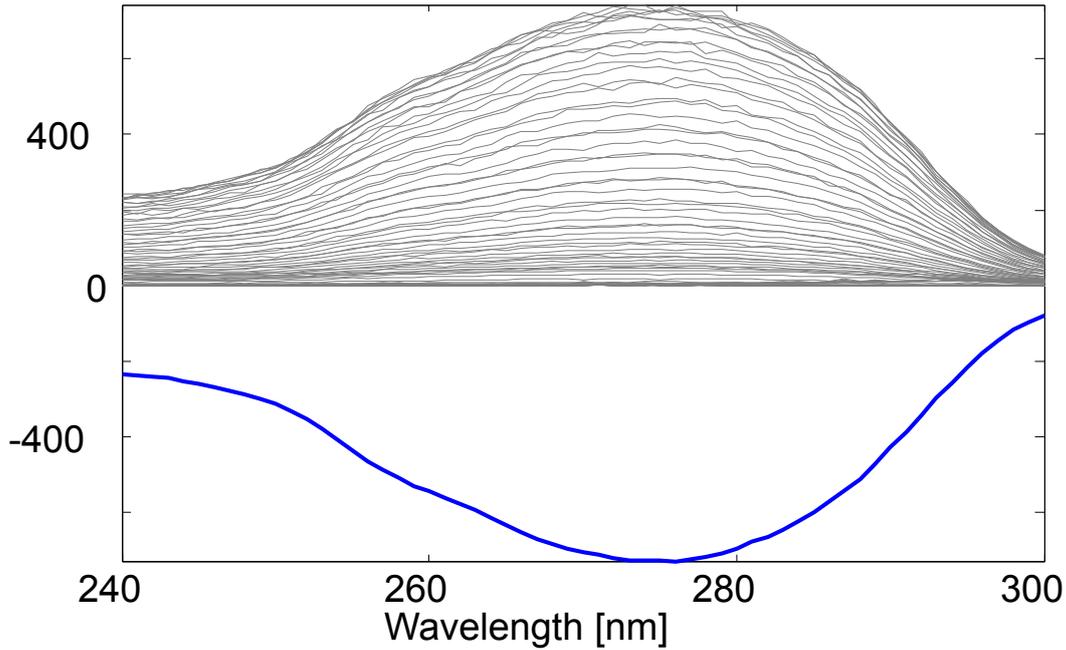
In a PARAFAC model, it is also possible to change the sign of say the first column of the first mode component matrix  $\mathbf{A}$  ( $\mathbf{a}_1$ ), and it must be countered by either changing the sign of the corresponding second mode loading vector,  $\mathbf{b}_1$ , or third mode loading vector,  $\mathbf{c}_1$ . Hence, for a one-component PARAFAC model it holds that this component can consist of one of the vectors

$(\mathbf{a}_1, \mathbf{b}_1, \mathbf{c}_1)$ ,  $(-\mathbf{a}_1, -\mathbf{b}_1, \mathbf{c}_1)$ ,  $(-\mathbf{a}_1, \mathbf{b}_1, -\mathbf{c}_1)$ , or  $(\mathbf{a}_1, -\mathbf{b}_1, -\mathbf{c}_1)$ .

Any of these representations of the component will have the same loss function value and are hence equally valid from a mathematical point of view.

For data such as many kinds of spectroscopy, the signs are easy to deduce from the appearance of the components, because underlying spectra, concentrations, or time profiles are positive. In other situations though, there is no intrinsic convention that can help guide the appropriate choice of signs.

In 2008, the sign problem for PCA was suggested resolved using an assumption that the ‘natural’ sign is the one that leads to a component that points in the direction where the majority of the data is pointing [8]. The basic premise of this approach is hinted at in *Figure 11*.



*Figure 1. A set of spectra (thin lines) modeled by a one-component PCA model. The first loading vector (estimated using the built-in function SVD in MATLAB R2011b) is shown with a thick line. It is apparent that the loading has a direction opposite to the majority of the data. Switching the sign of the loading (and the corresponding score) will give a model that is in better accordance with the data.*

In this paper, similar approaches will be developed for PARAFAC, Tucker3 and PARAFAC2 models. In the following we will use standard notation as given by Kiers [9]. Furthermore, residuals are excluded in all equations throughout, as the residuals are immaterial for the points made here.

## **Theory**

### *PARAFAC*

In PARAFAC, sign indeterminacies arise in the low-rank trilinear model because

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{B}_k^T = \mathbf{A}\mathbf{S}_1\mathbf{S}_3\mathbf{D}_k\mathbf{S}_2\mathbf{B}_k^T, \text{ for } k = 1, \dots, K$$

where  $\mathbf{S}_1$ ,  $\mathbf{S}_2$  and  $\mathbf{S}_3$  are diagonal matrices with plus or minus one on the diagonal. Together, they fulfill that  $\mathbf{S}_1\mathbf{S}_2\mathbf{S}_3 = \mathbf{I}$ . Hence, the model given by  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$  can be replaced by a model given by  $\mathbf{AS}_1$ ,  $\mathbf{BS}_2$  and  $\mathbf{CS}_3$  without changing the loss function. This extends without problems to PARAFAC models of higher order than three.

In order to determine the appropriate sign, it is sufficient to consider one component at a time. The contribution from other components can be removed from the data before assessing the sign of any given component [8].

When assessing the appropriate sign for e.g. the first mode component, the PARAFAC model is re-expressed as a bilinear model

$$\mathbf{X}_{\text{unfold}} = \mathbf{a}\mathbf{z}^T$$

Where  $\mathbf{a}$  is the component (column of  $\mathbf{A}$ ) currently considered and  $\mathbf{z}$  is the Khatri-Rao product of the corresponding columns of  $\mathbf{B}$  and  $\mathbf{C}$ .  $\mathbf{X}_{\text{unfold}}$  is the three-way array unfolded/matricized appropriately. The vector  $\mathbf{a}$  is normalized. For each column of  $\mathbf{X}_{\text{unfold}}$ , the inner product with  $\mathbf{a}$  is calculated, squared, and multiplied with the sign of the inner product as was suggested in the original PCA sign correction approach [8]. If a vector is in the same direction as  $\mathbf{a}$  (corrected for the size), then this number is large and positive and if it points in the opposite direction, then the number is negative. The sum of all numbers indicates how strongly  $\mathbf{a}$  is in the same or opposite direction as the majority of the data:

$$s_{\mathbf{a}} = \sum_{j=1}^J \text{sign}(\mathbf{a}^T \mathbf{x}_j) (\mathbf{a}^T \mathbf{x}_j)^2 \quad \text{Eq. 1}$$

where  $\mathbf{x}_j$  is the  $j$ th column of the matricized array. The same procedure is repeated in each mode giving a preferred sign for the component in each mode as well as a magnitude of *how* preferred the sign is.

If the number of negative signs is even, then the signs of each mode  $s_1$ ,  $s_2$ , and  $s_3$  will have a combined product of one, and the signs of component vectors can hence be changed accordingly without changing the loss function. For example, if both  $\mathbf{a}_1$  and  $\mathbf{b}_1$  has a negative  $s$  value, but  $\mathbf{c}_1$  does not. Then  $\mathbf{a}_1$  is replaced with  $-\mathbf{a}_1$  in the model and likewise for  $\mathbf{b}_1$ . As the

product of  $-\mathbf{a}_1$ ,  $-\mathbf{b}_1$  and  $\mathbf{c}_1$  remains the same as of  $\mathbf{a}_1$ ,  $\mathbf{b}_1$  and  $\mathbf{c}_1$ , the model is unchanged.

If the number of negative signs is odd, the magnitudes are used to decide which one of the signs should be disregarded. The vector which has the sign with the lowest associated magnitude is modified opposite to what the sign suggests, thus making the product of signs equal to one. Conflicting numbers of negative signs occur e.g. when data are centered, because then the direction in the centered mode can become arbitrary (yet of small magnitude). For more information on the basic procedure please consult Bro et al. [8].

### *Tucker3*

In essence, the above defines how to assign proper signs for the PARAFAC model. Next, the Tucker3 model is considered. The Tucker3 model is a complicated model to explore and visualize because of the core array. Essentially all vectors in one mode interact with all the vectors in all other modes. This makes it impossible to visualize all modes of a Tucker3 model simultaneously as also described by Kroonenberg in his work on so-called joint plots [10]. It also makes it impossible to rigorously define a preferred overall direction/sign of a vector because any one component vector can have different preferred directions depending on interactions in the other modes. Hence, a generic sign convention for Tucker3 will have to be somewhat ad hoc. One possible and feasible solution can be to focus on models that have approximately superdiagonal cores. Such rotated models can be simpler to interpret *if* the rotated model does indeed end up having an approximately superdiagonal core [11,12]. In such cases, we will advocate that the model be interpreted as a PARAFAC model disregarding the off-superdiagonal core elements when defining appropriate signs of components. That way, the signs of the components are switched solely reflecting the interactions of vectors in different modes with similar component number.

A slightly more general solution is also developed. This approach can be used whenever the model is not interpreted as a PARAFAC model; that is when the user also pays consideration to off-superdiagonal elements of the core in the interpretation. This method is also the one we have implemented

in the software associated with this paper, but the ‘PARAFAC-approach’ is of course still viable. The general procedure proceeds as follows. Assuming at first, that the core array has no preferred sign, the sign of loading vectors in each mode can be assigned independent of all the other modes because any sign switch in one component matrix can be countered by a sign switch of the core.

The data to use for calculating the sign for a given vector is found by subtracting the remaining components. This means using all components in the other modes and using all but the given component in the mode of interest and the corresponding core array. This can be exemplified as follows. Assume that the sign of the  $f$ th component in mode one is sought. The Tucker3 model of the three-way array is given by the components in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and the core array  $\mathbf{G}$ . To remove the influence of remaining components, the model is subtracted as

$$\mathbf{X}_{\text{res}} = \mathbf{X} - \mathbf{A}_f \mathbf{G}_f (\mathbf{C} \otimes \mathbf{B})^T. \quad \text{Eq. 2}$$

Here  $\mathbf{X}$  is the matricized three-way array and  $\mathbf{A}_f$  is the first mode component matrix with column  $f$  excluded. The matrix  $\mathbf{G}_f$  is the matricized core array where the  $f$ th horizontal slab has been excluded. The residual matrix  $\mathbf{X}_{\text{res}}$  now contains the part of the original data which the  $f$ th column of  $\mathbf{A}$  is modeling. Hence, the part of the Tucker3 model pertaining to that column can be written

$$\mathbf{X}_{\text{res}} = \mathbf{a}_f \mathbf{z}_f^T + \mathbf{E} \quad \text{Eq. 3}$$

where  $\mathbf{a}_f$  is the  $f$ th column of  $\mathbf{A}$  and  $\mathbf{E}$  is the original residual array of the Tucker3 model. The vector  $\mathbf{z}_f$  is defined by

$$\mathbf{z}_f = \mathbf{g}_f (\mathbf{C} \otimes \mathbf{B})^T. \quad \text{Eq. 4}$$

where the vector  $\mathbf{g}_f$  is the vectorized  $f$ th horizontal slab of the core array.

Using the model representation in Eq. 3, the appropriate sign of the  $f$ th column of  $\mathbf{A}$  can be determined from Eq. 1.

If the sign of column  $f$  is switched in  $\mathbf{A}$ , then correspondingly, the  $f$ th horizontal slab of the core array is multiplied by minus one. With this approach, all vectors in all component matrices point in a preferred

direction all other things being equal. It may happen, though, that some core elements are negative and we argue that in interpreting a Tucker3 model it is most natural that core elements are positive. This can be compared to having negative singular values in a singular value decomposition. While mathematically feasible, a positive value is more natural. It is not necessarily possible to transform any Tucker3 model to have all core elements positive. Rather than attempting this, only the largest core elements are investigated. Starting with the largest (negative) core element, the sign of this is switched by looking at the three vectors in each mode that it reflects. Assume that the core element is element  $(i,j,k)$ , then the magnitude of  $s$  for the corresponding vectors  $\mathbf{a}_i$ ,  $\mathbf{b}_j$ , and  $\mathbf{c}_k$  are assessed according to Equation 1. Each of these three vectors has been sign corrected as outlined above but the corresponding (large) core element is negative. It is therefore suggested to switch the sign of the one of these three vectors that has the smallest value of  $s$ . This way, the largest combinations of core elements will end up having a ‘natural’ core sign. This defines the sign convention for Tucker3.

## *PARAFAC2*

Finally, the PARAFAC2 model is considered. This is, by far, a more complicated model to deal with. The PARAFAC2 model can be written

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{H}^T\mathbf{P}_k^T = (\mathbf{A})(\mathbf{D}_k\mathbf{H}^T\mathbf{P}_k^T) = \mathbf{A}\mathbf{G}_k^T, \text{ for } k = 1, \dots, K$$

which implies that the concatenated frontal slabs can be written

$$[\mathbf{X}_1 \ \mathbf{X}_2 \ \dots \ \mathbf{X}_K] = \mathbf{A}[\mathbf{G}_1 \ \mathbf{G}_2 \ \dots \ \mathbf{G}_K]^T$$

This is a bilinear model and the sign ambiguity within the product of  $\mathbf{D}_k$  and  $\mathbf{P}_k$  is essentially eliminated in this representation because only their product appears (inside  $\mathbf{G}_k$ ). From this bilinear model, the overall sign of  $\mathbf{A}$  and the concatenated matrix can be determined using the two-way sign-correction approach described in Eq. 1 [8]. This also extends to higher-order PARAFAC2 models, where instead of  $\mathbf{A}$ , the Khatri-Rao product of all but the two ‘special’ modes would take the position of  $\mathbf{A}$ . Because the sign of  $\mathbf{A}$  is then fixed, each slab,  $\mathbf{X}_k$ , can now be assessed using that

$$\mathbf{X}_k = (\mathbf{A}\mathbf{D}_k)(\mathbf{H}^T\mathbf{P}_k^T) = (\mathbf{A}\mathbf{D}_k)\mathbf{S}_k\mathbf{S}_k(\mathbf{H}^T\mathbf{P}_k^T), \text{ for } k = 1, \dots, K$$

where  $\mathbf{S}_k$  is a diagonal matrix with 1 or -1 on the diagonal. We can further develop this as

$$\mathbf{X}_k = (\mathbf{A}\mathbf{D}_k)\mathbf{S}_k\mathbf{S}_k(\mathbf{H}^T\mathbf{P}_k^T) = (\mathbf{A}\mathbf{D}_k\mathbf{S}_k)(\mathbf{S}_k\mathbf{H}^T\mathbf{P}_k^T). \text{ for } k = 1, \dots, K$$

Because the matrix  $\mathbf{S}_k$  is specific to  $k$  we cannot apply  $\mathbf{S}_k$  to  $\mathbf{H}$ . This would change  $\mathbf{H}$  and hence invalidate the model of other slabs. However, it can be shown that

$$\mathbf{S}_k\mathbf{H}^T\mathbf{P}_k^T = \mathbf{H}^T\mathbf{M}_k\mathbf{P}_k^T$$

where  $\mathbf{M}_k = \mathbf{H}^T\mathbf{S}_k\mathbf{H}$  because it follows that  $\mathbf{H}^T\mathbf{M}_k = \mathbf{H}^T(\mathbf{H}^T\mathbf{S}_k\mathbf{H}) = \mathbf{S}_k\mathbf{H}^T$  because  $\mathbf{H}$  is a square and full rank matrix per definition. Hence, we can sign correct  $\mathbf{P}_k$  using  $\mathbf{M}_k$  instead of  $\mathbf{S}_k$ . With this, the preferred sign for each pair of  $\mathbf{P}_k$  and  $\mathbf{D}_k$  can be determined and corrected. Note, that the PARAFAC2 model is quite special in that each element of the diagonal of  $\mathbf{D}_k$  can switch sign independently because of  $\mathbf{P}_k$ . This means that each and every element of the loading matrix  $\mathbf{C}$  can change sign independently of all others. Also note that we do need to obtain the appropriate signs of the columns of  $\mathbf{P}_k$  (and  $\mathbf{H}$ ). Even though only the product of the two appear in the actual PARAFAC2 model, the correct signs of  $\mathbf{H}$  are needed to find the appropriate signs of the remaining parameters (see below).

Having fixed the sign of  $\mathbf{C}$  there is still a potential sign indeterminacy within  $\mathbf{P}_k\mathbf{H}$  because  $\mathbf{P}_k\mathbf{H} = \mathbf{P}_k\mathbf{S}\mathbf{S}\mathbf{H}$ . Notice, that  $\mathbf{S}$  is common to all slabs. We take a pragmatic approach and determine the appropriate sign for each slab as

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{H}\mathbf{S}_k\mathbf{S}_k\mathbf{P}_k^T, \text{ for } k = 1, \dots, K$$

Subsequently, the most abundant sign is chosen by using the sign of the sum of all  $\mathbf{S}_k$ .

Thus, having fixed the sign ambiguity of the  $\mathbf{P}_k$  matrices, the model is now corrected. For higher order models, it may be necessary to express the model as a PARAFAC model given the fixed  $\mathbf{P}_k$  matrices. Assuming a higher order model where components of several modes are held as a

Khatri-Rao product matrix ([13]) in  $\mathbf{A}$ , a model of the slabs  $\mathbf{X}_k\mathbf{P}_k$  can be expressed as a PARAFAC model as

$$\mathbf{Y}_k = \mathbf{X}_k\mathbf{P}_k = \mathbf{A}\mathbf{D}_k\mathbf{H}^T\mathbf{P}_k^T\mathbf{P}_k = \mathbf{A}\mathbf{D}_k\mathbf{H}^T. \text{ for } k = 1, \dots, K$$

From this, the signs within the several modes in  $\mathbf{A}$  (see above) can be determined by using the sign fix approach of an ordinary PARAFAC model. Hence, all signs are thereby fixed.

## Results

In order to verify that the sign correction is meaningful, a few examples are given. One example on a Tucker model is given and two examples focusing on the PARAFAC2 model. The PARAFAC corrections are more straightforward extensions of the original sign correction of [8], so these are not further discussed here.

For exemplifying Tucker sign corrections, a data set is analyzed which describes the average daily amount of pollen for 40 weeks (first mode) for 16 plant families (second mode) during five years (third mode) in an area close to Tortona, Piedmont, Northern Italy [14]. The weeks taken into account go from week six (mid-February) to week 45 (beginning of November).

The 16 families are the following: Betulaceae, Corylaceae, Cupressaceae-Taxaceae, Fagaceae, Oleaceae, Pinaceae, Salicaceae, Chenopodiaceae-Amarantaceae, Compositae, Graminaceae, Plantaginaceae, Polygonaceae, Urticaceae, Alternaria, Cladosporium, Others + non identified.

The years covered are 2006-2010. For the final Tucker3 model, the number of components chosen was two in the first mode, two in the second mode and one in the third mode. The model is rotated to be superdiagonal in the  $2 \times 2$  plane of the core and this is perfectly achievable when the last mode has only one component. Hence, components can be compared across modes. That is, score one in mode one is only related to score one in mode two and likewise for score two. After sign correction, the loading plots are as shown in Figure 22.

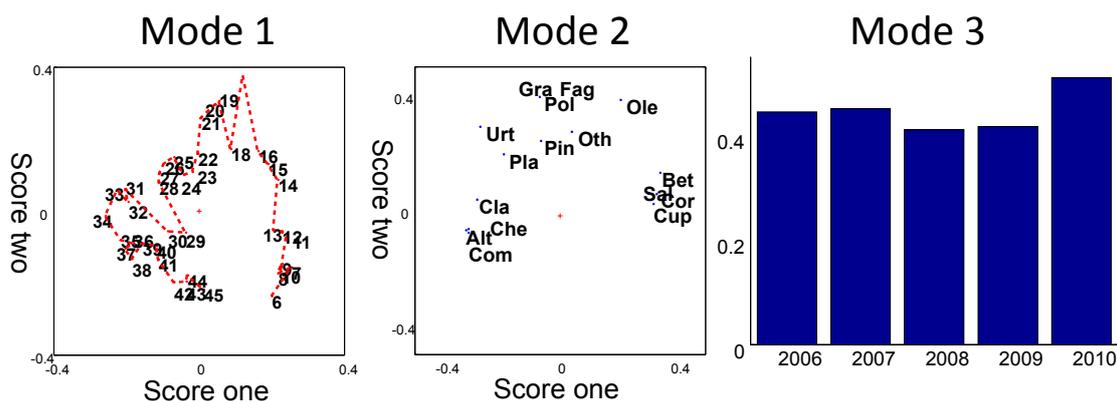


Figure 2. Sign-corrected scores and loadings of a Tucker3 model of pollen data with a diagonal core array.

As opposed to the “original” orientation in the model, the sign correction allows to have a direct joint interpretation of the loading plots. The period of pollination of each family can be easily seen (e.g., Salicaceae, Cupressaceae, Betulaceae and Corylaceae in spring, Fagaceae, Graminaceae and Polygonaceae in summer, Compositae, Cladiorporium and Alternaria in autumn) and e.g. that in 2010, the spring pollination was slightly larger than in 2008 and 2009.

For an example of sign correcting a PARAFAC2 model, a data set of 44 red wine samples is used. The volatiles of the samples were collected from ten mL of each wine on a Tenax-TA trap. The trapped volatiles were desorbed using an automatic thermal desorption unit and transferred to a gas chromatography system (HP 6890 GC). The GC was equipped with a mass spectrometric detector operating in the electron ionization mode at 70 eV. More experimental details can be found in [15].

An example of a typical PARAFAC2 model from a part of the elution time is shown in Figure 33. A three-component unconstrained PARAFAC2 model seems to be appropriate but several loadings are turned upside down. This is readily seen in the mass spectral mode, where two loading vectors are exclusively negative. Also note, that in the sample mode, the sign indeterminacy means that every element in a given loading vector can change sign independent of the others. It is very easy to see in the right-

most sign corrected version that the sign correction not only is meaningful, but also greatly helps in discerning more subtle details of the model.

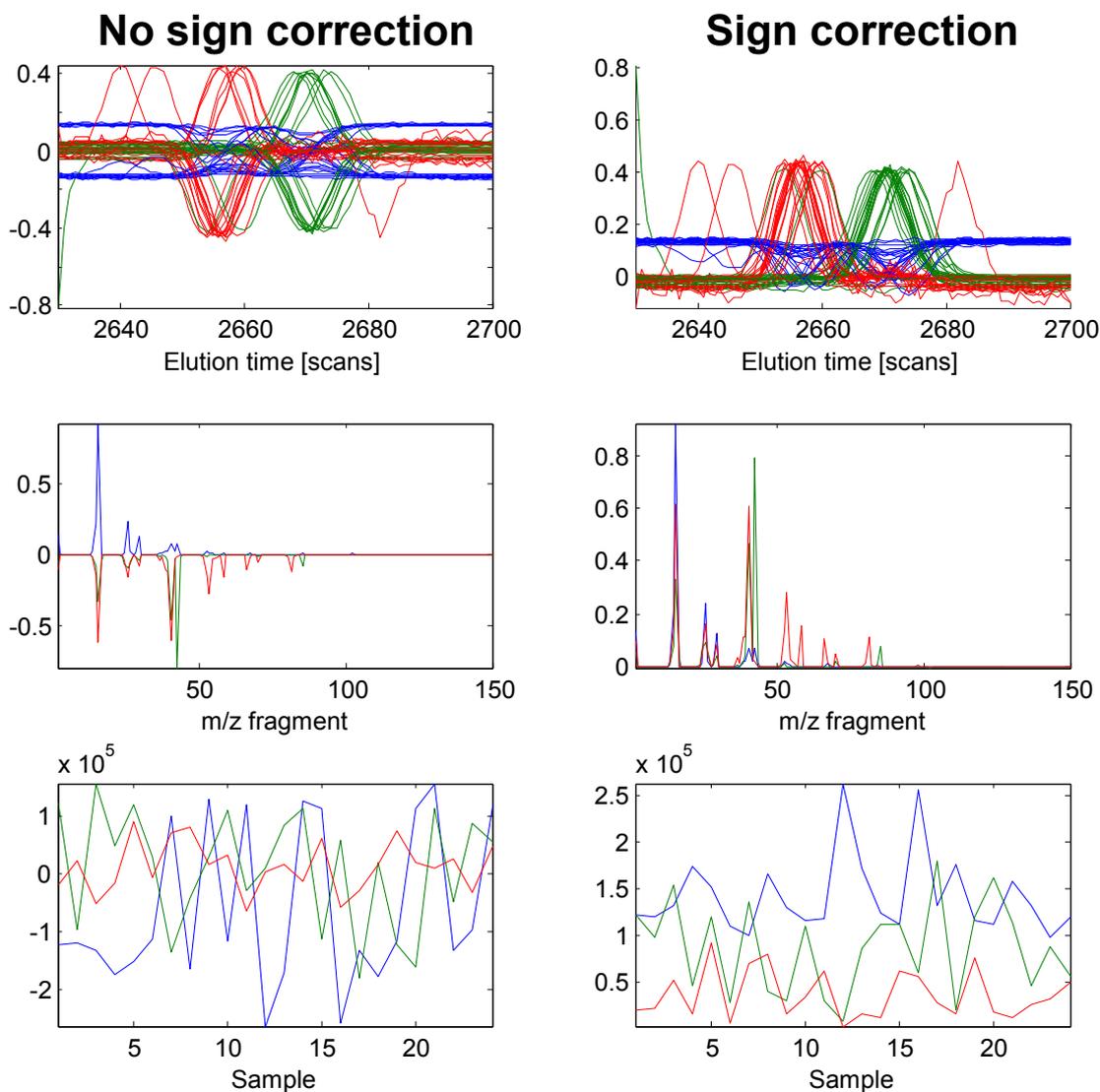


Figure 3. Left is the result of three-component PARAFAC2 model of a chromatographic data set (top – elution mode loadings ( $\mathbf{B}_k$ ), middle – mass spectral loadings ( $\mathbf{A}$ ), bottom – sample mode loadings ( $\mathbf{C}$ )). To the right is the same model upon sign correction.

Another example can be seen in Figure 44. This data comes from GC-MS analysis of cheese after the samples have been oxidized with

methoxyamine (20mg/mL in pyridine) followed by derivatization with MSTFA (as suggested by Kanani et al [16]). The samples were analysed on an Agilent Technologies 7890A GC-system coupled with a 5975C inert XCMSD detector. In the un-corrected model, it is clear that the loadings for one of the components are turned upside down in the elution time mode as well as in the mass spectral mode. After sign correction the model appears as chemically meaningful with respect to the signs.

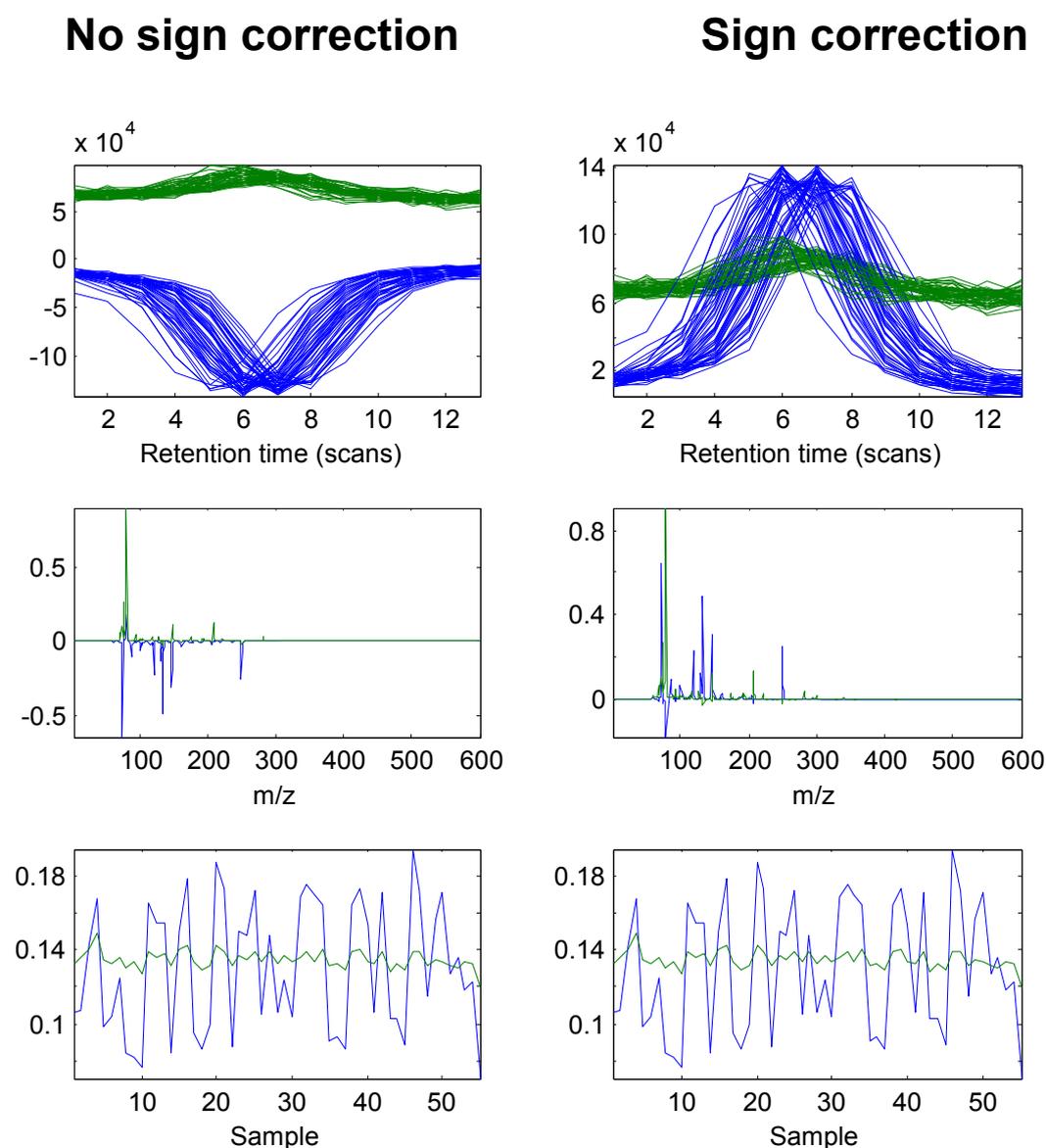


Figure 4. Left is the result of a two-factor PARAFAC2 model of a chromatographic data set (top – elution mode loadings ( $B_k$ ), middle – mass

spectral loadings (A), bottom – sample mode loadings (C)). To the right is the same model upon sign correction. In this example one of the spectral profiles has been flipped as well as the corresponding elution profiles.

## Conclusion

A formal approach has been developed for correcting for sign indeterminacies in various multi-way models. Some illustrative examples have been given to show that the correction indeed makes sense from an interpretational point of view.

The sign-correcting function is available at <http://www.models.life.ku.dk> as a MATLAB routine.

## Reference List

1. Pearson K, On lines and planes of closest fit to points in space, *Philosophical Magazine*, 1901, **2**, 559-572.
2. Hotelling H, Analysis of a complex of statistical variables into principal components, *Journal Of Educational Psychology*, 1933, **24**, 417-441.
3. Jackson JE, Principal components and factor analysis: part I - principal components, *Journal of Quality Technology*, 1980, **12**, 201-213.
4. Bro R, PARAFAC. Tutorial and applications, *Chemom Intell Lab Syst*, 1997, **38**, 149-171.
5. Harshman RA, Foundations of the PARAFAC procedure: Models and conditions for an 'explanatory' multi-modal factor analysis, *UCLA working papers in phonetics*, 1970, **16**, 1-84.
6. Tucker LR, The extension of factor analysis to three-dimensional matrices, *Contributions to Mathematical Psychology*, (Eds. Frederiksen,N and Gulliksen,H), Holt, Rinehart & Winston, New York, 1964, 110-182.
7. Kroonenberg PM, *Three-mode Principal Component Analysis. Theory and Applications*. DSWO Press, Leiden, 1983.

8. Bro R, Acar E, Kolda TG, Resolving the sign ambiguity in the singular value decomposition, *J Chemom*, 2008, **22**, 135-140.
9. Kiers HAL, Towards a standardized notation and terminology in multiway analysis, *J Chemom*, 2000, **14**, 105-122.
10. Kroonenberg PM, The TUCKALS line. A suite of programs for three-way data analysis, *Computational Statistics and Data Analysis*, 1994, **18**, 73-96.
11. Kroonenberg PM, de Leeuw J, Principal component analysis of three-mode data by means of alternating least squares algorithms, *Psychometrika*, 1980, **45**, 69-97.
12. Leardi R, Armanino C, Lanteri S, Alberotanza L, Three-mode principal component analysis of monitoring data from Venice lagoon, *J Chemom*, 2000, **14**, 187-195.
13. Bro R, Multi-way Analysis in the Food Industry. Models, Algorithms, and Applications. Ph.D. thesis, University of Amsterdam (NL), <http://www.models.life.ku.dk/research/theses/>, 1998.
14. Criscenzo E, Analisi multivariata su dati relativi a concentrazioni polliniche del Tortonese. Ph.D. thesis, University of Genoa, Italy, 2012.
15. Ballabio D, Skov T, Leardi R, Bro R, Classification of GC-MS measurements of wines by combining data dimension reduction and variable selection techniques, *J Chemom*, 2008, **22**, 457-463.
16. Kanani H, Chrysanthopoulos P, Klapa M, Standardizing GC-MS metabolomics, *Journal of Chromatography, B*, 2008, **871**, 191-201.

## Paper IV

---

**Lea G. Johnsen**, José Manuel Amigo, Thomas Skov, and Rasmus Bro.

“Automated resolution of overlapping peaks in chromatographic data”

*Submitted for Analytical Chemistry*



# Automated resolution of overlapping peaks in chromatographic data

*Lea G. Johnsen\*†, José Manuel Amigo, Thomas Skov, Rasmus Bro*

Quality & Technology, Department of Food Science, Faculty of Science,

University of Copenhagen, Denmark

KEYWORDS Chromatography; PARAFAC2; automated; GC-MS;

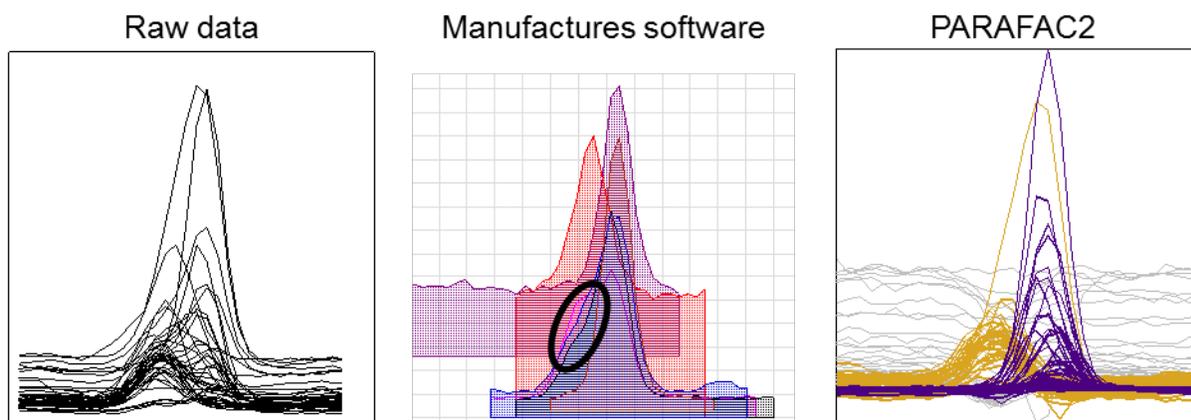
ABSTRACT PARAllel FACtor analysis 2 (PARAFAC2) has been shown to be a powerful tool for resolution of complex overlapping peaks in chromatographic analyses. It is particularly useful due to its ability to handle shifts in the elution time mode and peak shape changes. Like all curve resolution techniques, PARAFAC2 will only find chemically meaningful parameters (elution time profiles and mass spectra) if the correct number of factors are determined. So far, the only way to determine an appropriate number of factors is to calculate models with different number of factors, and then inspect the models manually. This approach is time consuming and the result may be biased due to the manual assessment of the model quality, making PARAFAC2 inaccessible for analytical chemists in general.

Here we develop a method which can determine an appropriate number of factors in an automated way. The automation is based on a number of model diagnostics (quality criteria) collected from models with different numbers of factors. Combining these diagnostics it is possible to assess what the appropriate number of components is. In this work only GC-MS data is considered. However, it will most likely be fairly straight forward to expand the work to also cover LC data. Automating the model quality evaluation of the PARAFAC2 model enables both the inexperienced and trained user to perform comprehensive and advanced analysis of chromatographic data with a minimum of manual work.

## Introduction

Nowadays, the most widespread approach for chromatographic data analysis is using the software provided by the manufacturer of the instrument. However, it has been shown that the algorithms implemented in many commercial packages can result in suboptimal utilization of the information in the data, compared to what can be provided by curve resolution techniques like multivariate curve resolution (MCR) or parallel factor analysis (PARAFAC or PARAFAC2)<sup>1, 2</sup>. A quite typical example is shown in Figure 1, where the deconvolution procedure from commonly used manufactures software has been applied to the data from the 45 samples. The peaks in the TIC (left in the figure) seem to be divided into two groups indicating that the data represents two different chemical compounds. If the individual mass traces (not shown) are inspected it is clearly seen that there are indeed two chemical compounds. The result from the manufactures software is shown in the middle plot in Figure 1. Several problems can be observed with this result. First of all, the software only finds a total of six peaks in all samples; this means that a lot of the peaks from the raw data are not described by this model. Furthermore, several of the peaks contain a shoulder (circle in Figure 1) indicating that the resolution of the two co-

eluting peaks has not been successful. A third problem is that none of the peaks in the model have been separated from the baseline, and that the level of the included baseline is fluctuating. On the other hand, by using PARAFAC2 on the dataset, these problems are seemingly solved. The PARAFAC2 model is able to split the peak into three main contributions arising from the two chemical compounds and signal from the baseline (right-most plot in Figure 1).



**Figure 1.** To the left, TIC of 45 elution profiles. The example shows the performance of respectively manufacturing software deconvolution algorithm (middle) and PARAFAC2 (right) on the raw data (illustrated with the TIC leftmost in the figure). The manufactures software finds a total of six compounds in all the 45 samples (illustrated with six elution profiles in the figure). The PARAFAC2 model finds three components in each sample (illustrated with estimated elution profiles to the right) one of these are describing baseline (grey) and two are describing two different chemical compounds (orange and purple).

PARAFAC2 is a method able to resolve many chromatographic artefacts (e.g., baseline drift, overlapping, elution time shifts, etc.)<sup>3</sup>. PARAFAC2 allows separating each source of variability in the data by using the spectral information gathered for each elution time. The resulting model provides three important sets of parameters: an estimated elution time profile for each compound in each sample, an estimated pure spectrum for each compound, and relative concentrations for each of these chemical compounds<sup>4</sup>.

Previous publications<sup>5-7</sup> have shown that the use of PARAFAC2 enables a comprehensive analysis of chromatographic data, including resolution of overlapping peaks. However, a common practical issue in all curve resolution techniques is that the right number of factors must be determined in order to obtain chemically meaningful profiles<sup>8</sup>. Unfortunately, it is not straight forward to determine how many factors to include. This may be one reason why curve resolution methods are not usually seen in routine chemical analysis. Therefore, an automated selection of the appropriate PARAFAC2 model would provide a significant improvement in terms of allowing non-chemometrically skilled chemists to take advantage of modern data analysis solutions.

When a PARAFAC2 model is calculated, several statistical and empirical diagnostics can be used to evaluate the reliability of the model. Traditionally, only explained variance, residuals, and mere observation of the obtained elution time and spectral profiles have been used when determining the appropriate number of factors in PARAFAC2. Additionally, it has recently been suggested to use core consistency in the evaluation of PARAFAC2 models<sup>9</sup>. Nevertheless, there exist a number of additional diagnostics which can be used to evaluate the obtained PARAFAC2 model. In this manuscript, we suggest 102 different diagnostics which all aim to describe some aspect of the quality of a PARAFAC2 model.

From these initial statistical and empirical diagnostics, we determine a classification model, which, in an automated way, can find an appropriate number of factors to include in the PARAFAC2 model. In order to obtain a good classification model, we determine which of the initial descriptive quality criteria are most important for the classification, and only those will be used in the final classification model. The proposed method is tested on four different GC-MS

datasets originating from different chromatographic instruments and from different sample matrices (apples, wine, aroma standards, and cheese).

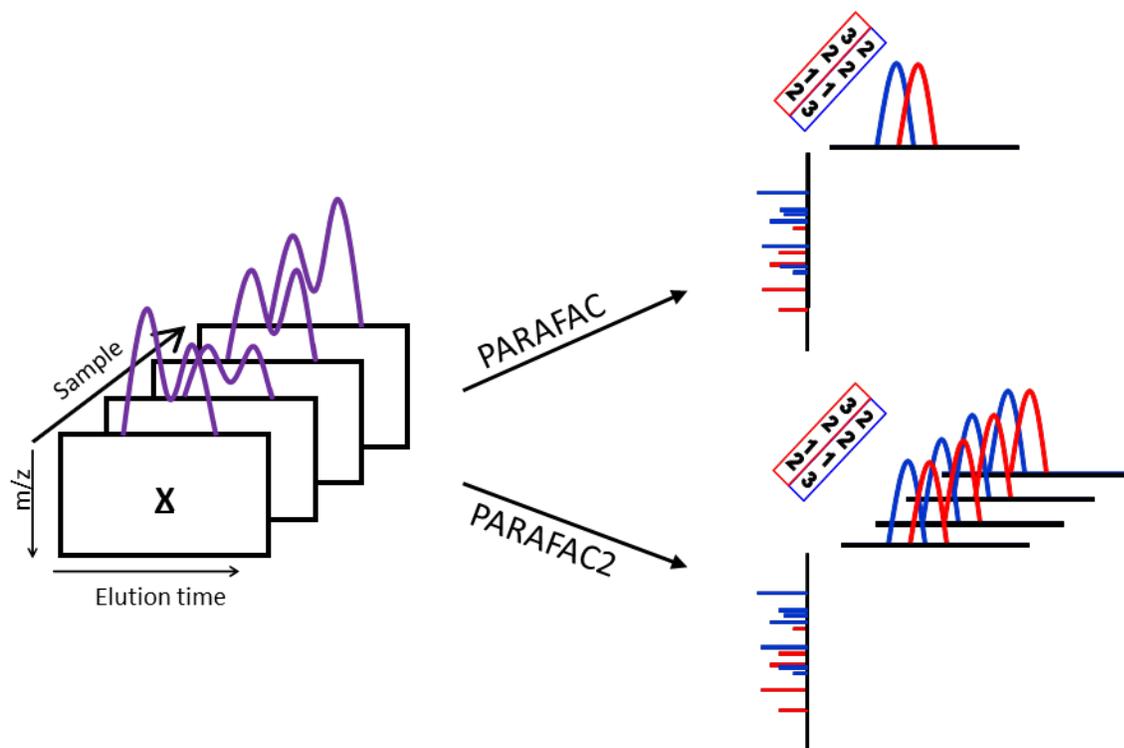
The manuscript will be initiated by a section describing the theory behind PARAFAC2 and a brief description of the diagnostics developed for assessing PARAFAC2 model complexity. The theory behind the classification will not be described in this paper, but can be found described elsewhere in the literature<sup>10, 11</sup>. The description of the diagnostics will be followed by a section which describes how the classification model is optimized and a validation of the final classification model.

Throughout this paper the word compound refers to the chemical compounds contained in a sample, while the words factor and component refers to the outcomes of the model.

## Theory

PARAFAC was introduced simultaneously by Harshman<sup>8</sup> and Carroll and Chang<sup>12</sup> (who named it canonical decomposition). Harshman based his PARAFAC model on the idea of parallel proportional profiles by Cattell<sup>13</sup>. The idea behind parallel proportional profiles is that if the relative amounts of overlapping phenomena are changing across samples, then it is possible to resolve the unique patterns for each of these phenomena.

The PARAFAC solution is uniquely identified (up to scaling and permutations) under mild conditions. This indirectly implies that a correctly specified PARAFAC model, applied to e.g. GC-MS data, can provide estimates of the pure mass spectra, concentration profiles, and pure elution time profiles when there are not elution time shifts (as illustrated in Figure 2).



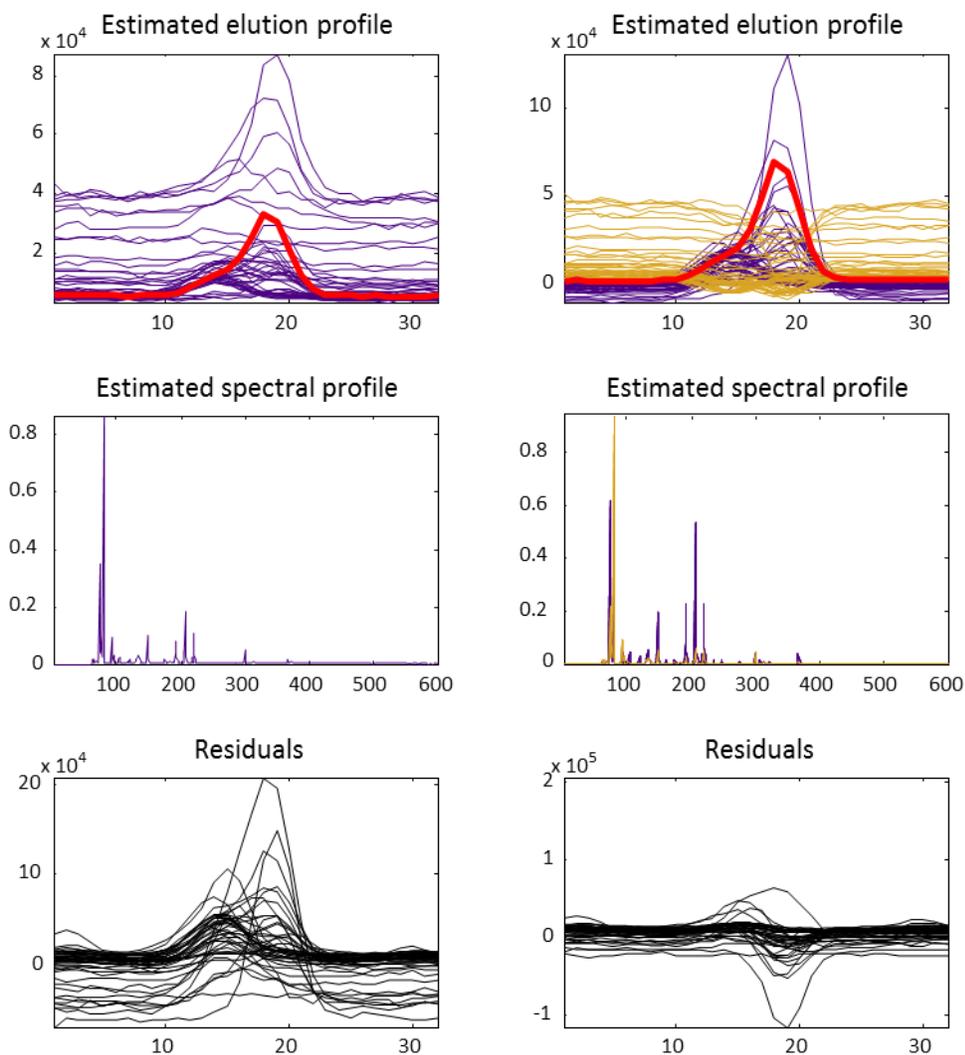
**Figure 2.** Illustration of the differences between PARAFAC and PARAFAC2. In PARAFAC one common elution profile is used to describe each compound in all samples, whereas in PARAFAC2 each compound in each sample is modelled with a distinct elution profile. Adapted with permission from Amigo *et al.*<sup>3</sup>. Copyright © 2010 American Chemical Society.

Chromatographic data often contains shift in the elution time dimension and PARAFAC is unable to handle shifted data efficiently without the data being aligned prior to modeling, since only one common elution time profile is estimated for each compound. Hence, PARAFAC assumes that the *same* elution profile can model the compound in all samples. However, it has been shown that PARAFAC2 can be used to solve the problem with shifts in retention time<sup>4, 14</sup>. In PARAFAC2, an elution time profile is found for each compound in each sample as illustrated in Figure 2. As for PARAFAC, the PARAFAC2 solutions are also unique. A thorough description of PARAFAC2 has been made by Bro *et al.* elsewhere<sup>4, 14</sup>.

For both PARAFAC and PARAFAC2, a model with too many factors will describe variation which is not chemically meaningful. Sometimes, one or a few extra components do not disturb the model. For example, for PARAFAC it has been shown that in some cases more than one model can be chemically meaningful and provide estimates of the underlying patterns even though the models have different numbers of factors<sup>8</sup>. This implies that in some cases, there is not *one* optimal model but rather a range of appropriate models. In our experience the same goes for PARAFAC2.

Figure 3 shows two examples of under-fitted models obtained from the data presented in Figure 1. In these cases both residuals and elution profiles indicate that the models do not have enough factors included. The residuals are, especially in the one-factor model, behaving in a very systematic way. This indicates that there is still chemical information in the data which is not explained by the model. In the elution profiles, the under-fitting is indicated by the modelled peak-shapes which suggest that two (or more) co-eluting compounds are described by the same elution profile.

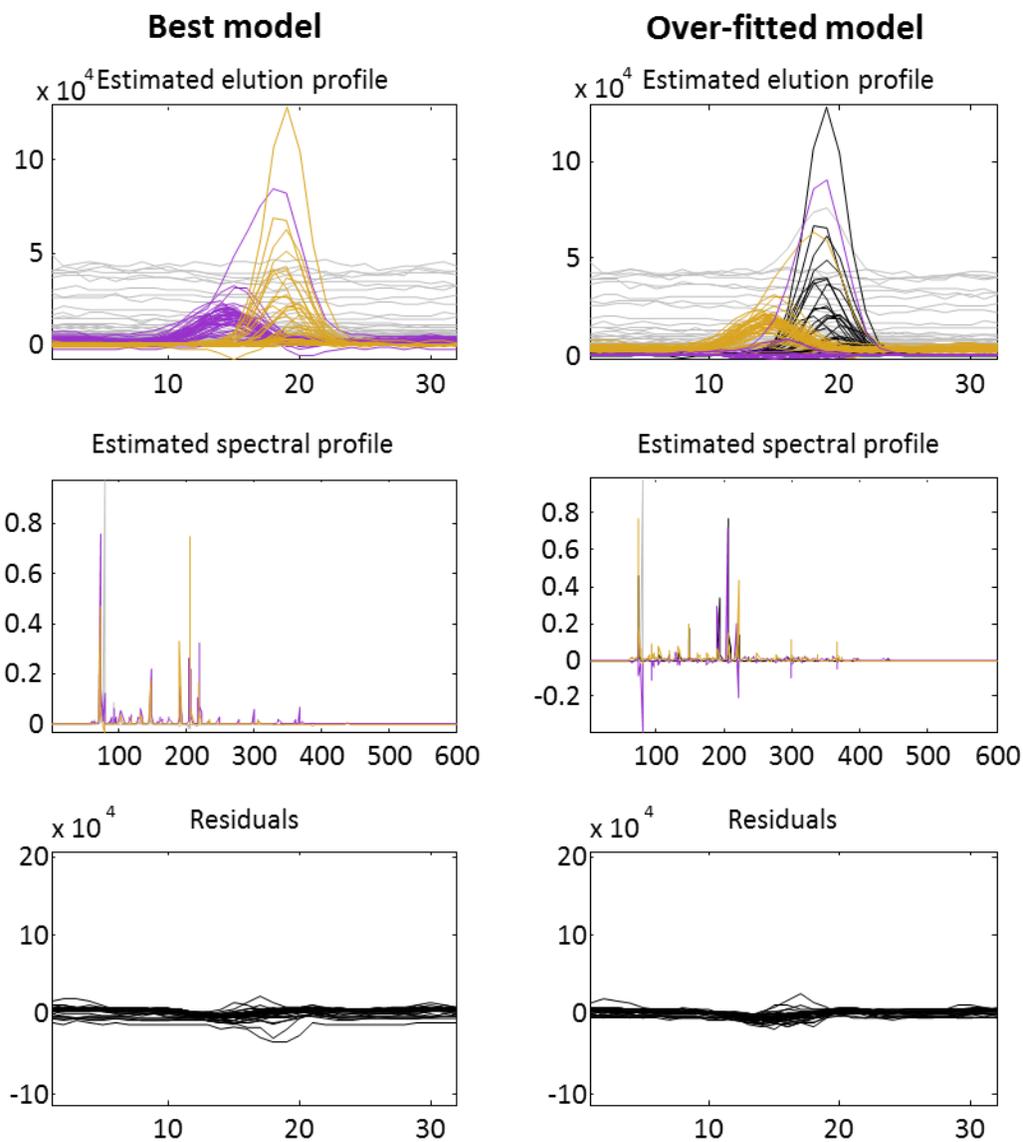
## Under-fitted models



**Figure 3.** PARAFAC2 models with too few factors applied to the data presented in Figure 1. Left: one factor. Core consistency: 100, iterations: 2. Right: two factors. Core consistency: 99, iterations 2. Core consistency and iterations are used in the evaluation of the models.

In Figure 4, a good model is shown to the left and an over-fitted model to the right. Both models show how the residuals become highly unsystematic when a sufficient number of factors are included. The over-fitted model, in this example, is characterized by having low core consistency (below zero) as well as an increase in negative values in the obtained mass spectra,

compared to the model with one less factor. Also the number of iterations, which has increased considerably, compared to the model with one less factor, indirectly indicates that this model is over-fitted.



**Figure 4.** Best model: core consistency: 97, iterations: 30. Over-fitted model: core consistency: <0, iterations: 383. Core consistency and iterations are used in the evaluation of the models.

One disadvantage in the above evaluation of obtained models is that all of the diagnostics are subjective, such evaluations will inevitably be biased by different personal opinions on which of the diagnostics one thinks is more important in the evaluation. Another disadvantage is that the evaluation is a time consuming task. To counter these problems, an automated quality control is proposed herein by investigating the diagnostics, listed in **Table 1**, as descriptors for quality of the PARAFAC2 model. All the diagnostics are calculated on models where the problem with sign-indeterminacy has been solved with the method proposed by Bro *et al.*<sup>15</sup>.

**Table 1.** Main features of 34 diagnostics which are used to describe the PARAFAC2 models.

#	Diagnostic	Brief description
1	Time	Over-fitted models often require longer calculation times.
2	Iterations	As #1 but measured in number of iterations.
3	Log(iterations)	Differs from #2 by leveling out smaller changes caused by different starting points or difficult datasets.
4	Smoothness*	Over-fitted models can result in noisy elution profiles.
5	Number of peaks*	Models with too few factors will sometimes result in estimated elution profiles with more than one peak. The average number of peaks in the elution profiles can tell something about if the model is under-fitted.
6	Non-one-peakness*	Addresses the same feature as #5, but measured as how much the profile deviates from having only one peak.
7	SSQ residuals	The residuals for models with too few factors are often relatively high. The sum of the squared residuals is weighted against noise level in raw data in order to make them independent of differences between dataset.
8	Explained variance	Explained variances are often low for models with too few factors.
9	Negative area in elution*	In models with too many factors included there will often be considerable negative values in the estimated elution profile.
10	Abs(Spec area)/(Spec area)*	As for elution profiles, negative values in the estimated spectral profile can indicate that the model is over-fitted. Here assessed as the relation between the absolute area and the area.
11	(Pos spec area)/(Neg spec area)*	As #10, but assessed as the relation between positive and negative area in the spectral profile.
12	SSQ(Epca) mz	PCA can be used to investigate the true rank of the data. Here assessed as the sum of the squared residuals from a PCA model of data unfolded in the mass direction.

13	SSQ(Epca) rt	As #12, but data is unfolded in the elution time direction.
14	SSQ(Epca) samp	As #12, but data are unfolded sample wise.
15	SSQ(Epca)/SSQ(Epf2) mz	The relation between #12 and #7.
16	SSQ(Epca)/SSQ(Epf2) rt	The relation between #13 and #7.
17	SSQ(Epca)/SSQ(Epf2) samp	The relation between #14 and #7.
18	Poss. Corr. Spectra *	Very similar spectra within a model may indicate that the model is over-fitted. This is assessed as the maximal correlation between spectra within the model.
19	Neg. Corr. Spectra *	Two-factor-degeneracy <sup>16</sup> may indicate over-fit. This is assessed as the maximal negative correlation between estimated spectra within the model.
20	Core consistency *	Low (or negative) core consistency indicates over-fitted models.
21	Split half*	Solutions obtained from models with many factors are not chemically unique and therefore the resulting models for subsets (sample wise) of data should only be identical if the right number of factors is used.
22	Baseline found*	If no factors are describing the systematic baseline this may indicate that more factors should be included in the model.
23	Epca rt/mz	The relation between #13 and #12.
24	Epca rt/samp	The relation between #13 and #14.
25	Epca mz/samp	The relation between #12 and #14.
26	TIC corr.	Data, reconstructed from models with too few factors, are not likely to be very similar to raw data, whereas reconstructed data from models with the right number or too many factors most likely are very similar to raw data. This is assessed as the correlation between the TIC from raw data and the model.
27	Positive congruence *	As #18, but assessed as congruence instead of correlation.
28	Negative congruence *	As #19, but assessed as congruence instead of correlation.
29	Max Durbin Watson (TIC)*	High amounts of systematic behaviour in the residuals indicate that more factors should be included in the model. This is assessed as the maximal Durbin Watson criteria <sup>17</sup> on the TIC of the residuals.
30	Median Durbin Watson (TIC)*	As #29, assessed as the median.
31	Max Durbin Watson (summed over time)*	As #29, but determined on residuals summed over time.
32	Median Durbin Watson (summed over time)*	As #30, but determined on residuals summed over time.
33	Simplicity (TIC)*	As #29 but assessed using summed squared eigenvalues from SVD on residuals (as TIC).
34	Simplicity (summed over time)*	As #33, but on residuals summed over time.

The diagnostics indicated with \* are described more thoroughly in the supplementary material.

For all of the above 34 diagnostics also the difference (“diff”) between the diagnostic value of the present model and the model with one factor less was included (diagnostics #35-68). This approach was inspired by the DIFFIT approach described by Timmerman and Kiers<sup>18</sup>. The “diff” value for a model with  $k$  factors is defined as the value at  $k$  factors minus the value at  $k-1$  factors. Also the relative change is determined for all 34 diagnostics (diagnostics #69-102). The relative change is defined as the “diff” divided by the original value. This gives a total of 102 diagnostics.

## Materials and methods

### Datasets

A total of four different GC-MS datasets have been included in this study: One from analysis of apples<sup>6</sup>, one from analysis of wine<sup>19</sup>, one from analysis of standards with different additions of aroma compounds<sup>1</sup>, and one from analysis of cheese derivatized with methoxamine (20mg/mL in pyridine) followed by derivatization with N-methyl-N-(trimethylsilyl) trifluoroacetamide as described by Kanani *et al.*<sup>20</sup>.

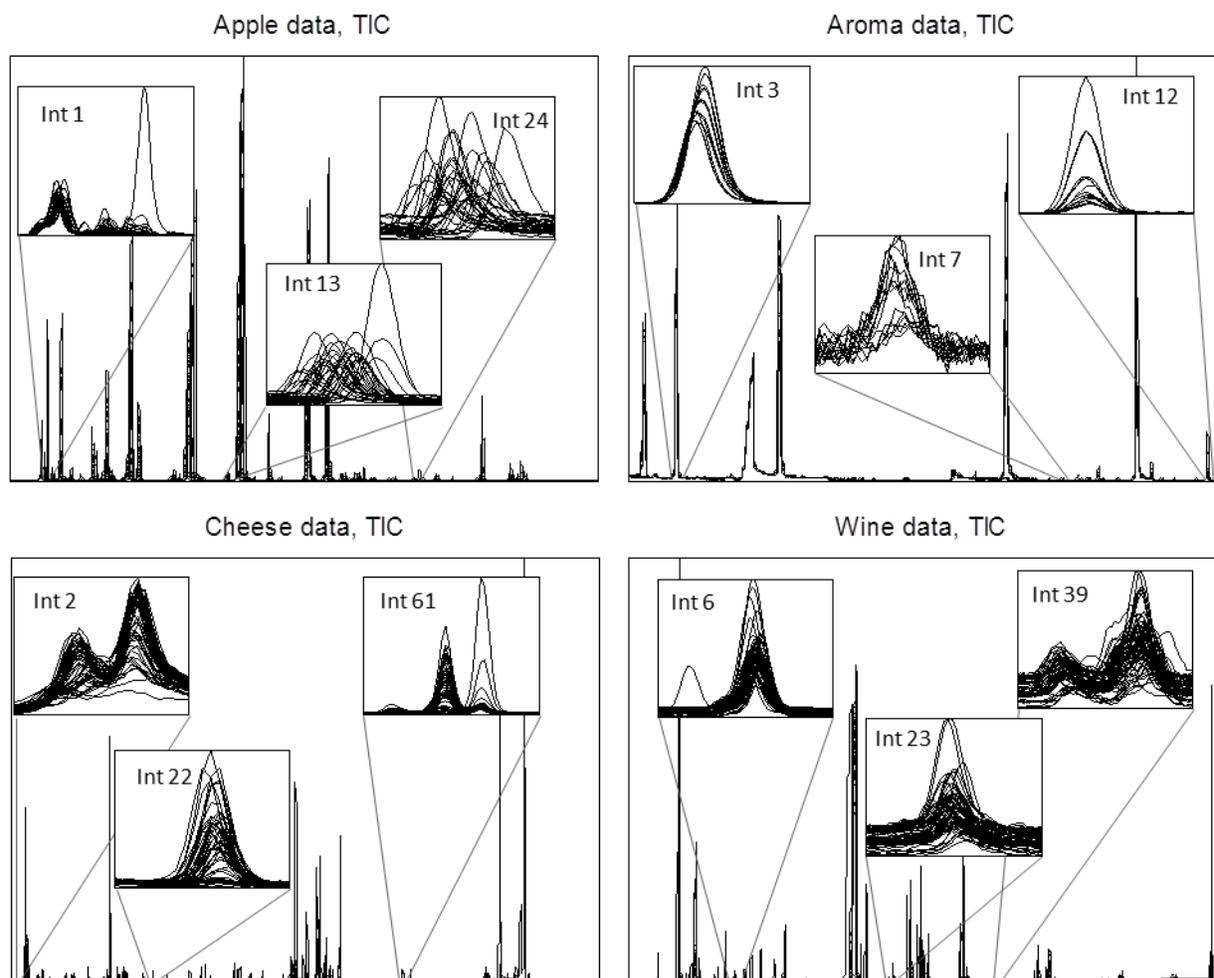


Figure 5. Overview of the raw data, the zooms are showing examples of different intervals.

The four datasets were divided into smaller intervals with an estimated maximum of five compounds in each interval. The TICs from the four datasets are shown in Figure 5, with zooms showing examples of these intervals. In total 155 intervals were created. PARAFAC2 models with one to seven factors were calculated for each of these intervals and the diagnostics described in **Table 1** were determined for each of these 1085 models. By working on baseline-separated intervals, the problems are typically much easier to analyse and more unambiguous

results are obtained. PARAFAC2 models were calculated without any constraints (such as non-negativity) since these constraints might disguise indications of over-fit.

The ‘correct’ number of factors for each model was determined by having a skilled chromatographic chemometrician evaluating the core consistencies, number of iteration used, obtained elution and spectral profile, as well as residuals, considering the chemical information that the interval contains (as described in the section above).

The classification model was constructed with PLS-DA as follows: The 155 intervals were randomly divided into a calibration (75 intervals) and a validation (80 intervals) set. Some of the intervals were not modelled well by PARAFAC2. The reason for that could be that two compounds were totally overlapping (embedded), the peak shapes were severely changed, or that the spectra of different compounds were too similar. These intervals were removed from the calibration set before any variable selection was performed in order for them not to influence the selection.

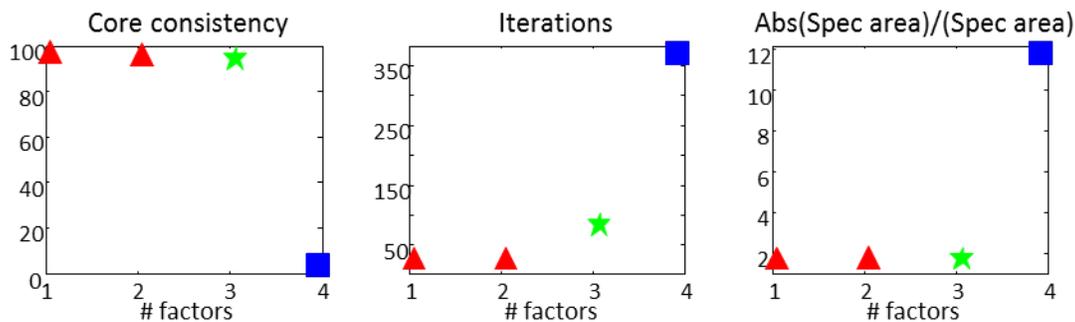
## **Software**

All algorithms and models have been developed using MATLAB R2012a (Mathworks, Inc., Natick, Massachusetts, U.S.A.). PLS\_Toolbox (Eigenvector Research, Inc., Washington, Wenatchee) has been used for principal component analysis (PCA) and partial least squares-discriminant analysis (PLS-DA) models. PARAFAC2 models have been calculated using the algorithm available from [www.models.life.ku.dk](http://www.models.life.ku.dk) (Dec. 2012).

## **Results/Discussion**

By visual inspection of the raw diagnostics it became obvious that the majority was following an overall pattern. In Figure 6, the values of core consistency (diagnostic 20), the number of iterations (diagnostic 2), and the relation between the absolute area and the area of the spectral

profile (diagnostic 10) are shown. The values are obtained from the PARAFAC2 models illustrated in Figure 3 and Figure 4.



**Figure 6.** Development of a typical diagnostic with increasing number of factors included. Here illustrated with three of the 34 diagnostics listed in Table 1. The diagnostics are obtained from the PARAFAC2 models illustrated in Figure 3 and Figure 4. Red triangles: Under-fitted models. Green stars: Best model. Blue squares: Over-fitted model.

Small changes in the diagnostic value are observed as more and more factors are included until the model becomes over-fitted. At this point there is a significant change after which the diagnostic value only changes slowly again. Due to the way we have defined the “diff” diagnostic, the value obtained from the first over-fitted model will describe this jump in the raw diagnostic values. Therefore, the first over-fitted model will be distinguishable from the ones with fewer components. This approach is similar to what was described by Hoggard and Synovec<sup>21</sup> in their paper concerning automated determination of the number of factors to include in PARAFAC models.

In order to determine which of the 102 selected diagnostics were useful in classification of the first over-fitted model, variable selection was conducted. By using a combination of different variable selection techniques<sup>22</sup>, the most important diagnostics were selected.

The following seven diagnostics represents a good compromise between having a good classification power without too many variables:

- Core consistency (20)
- Change in the negative area in the elution profile (43)
- Change in negative correlation between spectra (53)
- Logarithm of the number of iterations (3)
- The positive correlation between spectra (18)
- The relative change in how systematic the residuals are (indicated with Durbin Watson) (97)
- The relative change in the correlation between the TIC from raw data and the TIC from the obtained model (94)

The specificity (0.97 for over-fit and 0.95 for not over-fit) obtained with a model including these seven diagnostics is very similar to the model with highest specificity but including five additional diagnostics. None of these seven final diagnostics could be removed without a significant loss of classification power.

The regression vector obtained in the PLS-DA model (Figure 7) indicated that the core consistency, changes in the negative area in the elution profile, and changes in negative correlation between spectra are negatively correlated to over-fit. On the other hand, the following diagnostics were positively correlated with the first over-fitted model: the logarithm of the number of iterations, the positive correlation between spectra, the relative change in how systematic the residuals are (indicated with Durbin Watson), and the relative change in the correlation between the TIC from raw data and the reconstructed TIC from the obtained model.

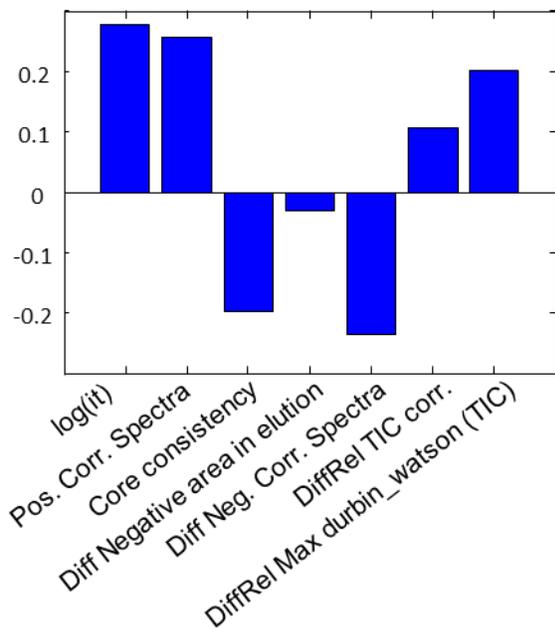
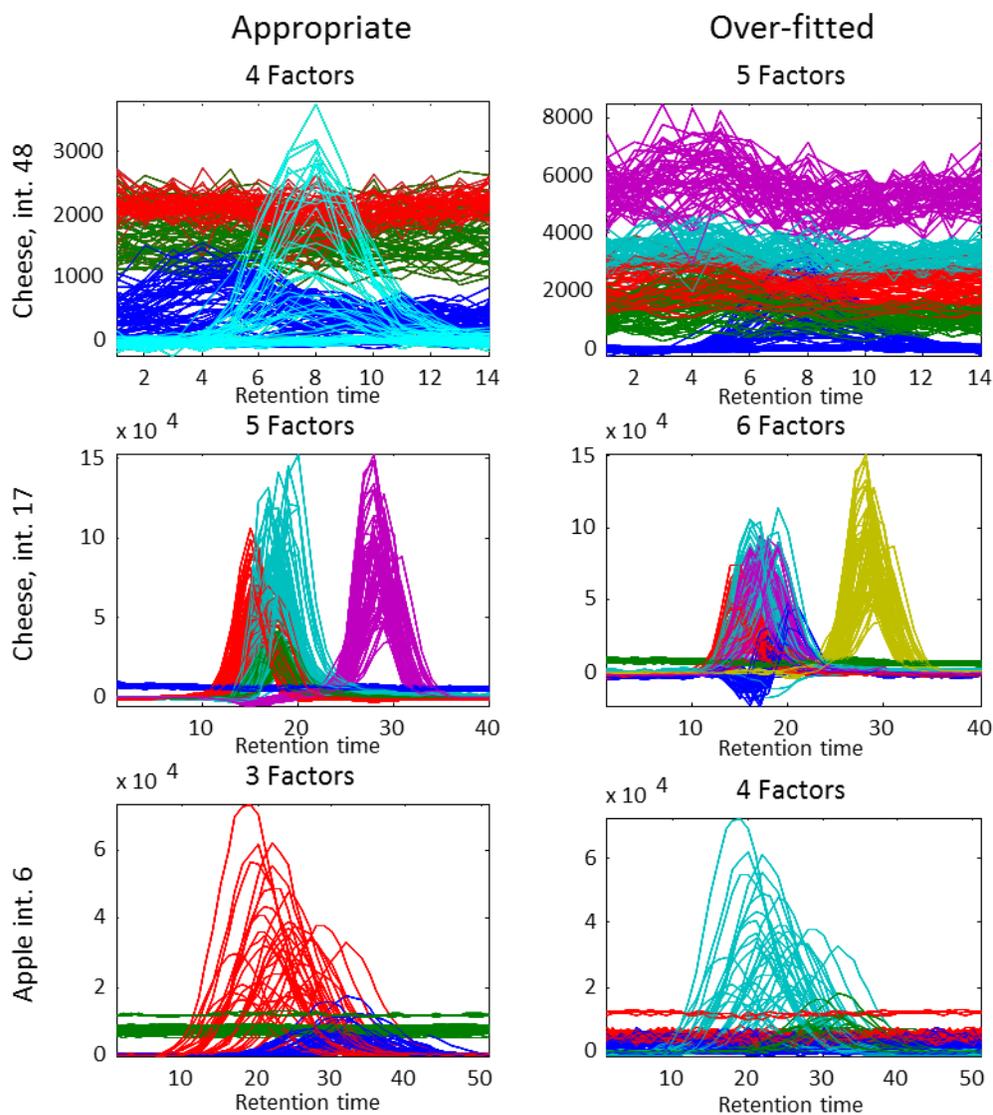


Figure 7. Regression vector for the PLS-DA model.

The PLS-DA model based on these seven diagnostics was used to classify the models in the test set (a total of 80 intervals). Most of these intervals were correctly classified according to the manual evaluation. Furthermore, it was found that the majority of the apparently misclassified models were either; models which may have been misclassified by the manual evaluation, or cases where PARAFAC2 where unable to describe the data in an appropriate way.

Rightmost in Figure 8, the elution profiles from three of over-fitted models, with very different characteristics are shown. These were all correctly classified as being the first over-fitted models, leading to the models with one factor less, shown to the left in the figure, to be determined as being appropriate models. The diversity in these models indicates that the classification is performing well for many different types of GC-MS data.



**Figure 8.** Right: Examples of models correctly classified by the PLS-DA model as being the over-fitted model. Left: The models with one factor less, concluded to be the appropriate models.

In Table 2 the apparently misclassified models are listed. Misclassifications can be divided into three categories; really misclassified (nine models), intervals not well described by any PARAFAC2 model (nine models), and models where the classification might be assessed as correct (nine models). In the table it can be seen which of these categories the individual models

falls into. In the following, a thorough inspection of the three types of misclassifications will be given.

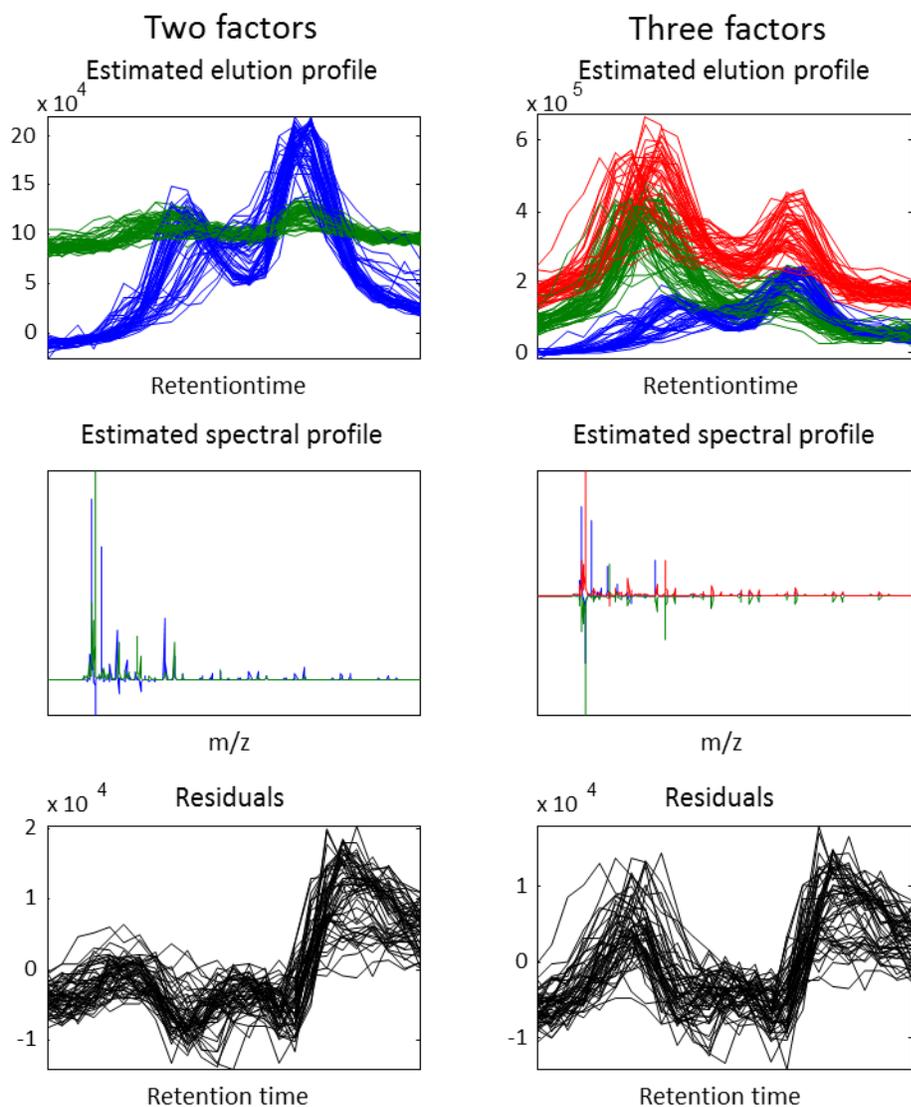
Table 2. Models that may be incorrectly classified from the test-set with a total of 80 intervals.

	Dataset	Interval	# factors	Comments	
Misclassified as not over-fit	Apple	19	3	May be correct	
		22	4	PARAFAC2 inappropriate	
		24	3	May be correct	
	Cheese	2	2	PARAFAC2 inappropriate	
		22	3	May be correct	
		38	3	PARAFAC2 inappropriate	
		31	3	May be correct	
		46	3	May be correct	
	Aroma	1	2	May be correct	
		5	2	PARAFAC2 inappropriate	
		7	2	May be correct	
		8	2	May be correct	
		11	3	PARAFAC2 inappropriate	
	Misclassified as over-fit	Apple	1	5	Misclassified
			13	3	Misclassified
22			4-5	PARAFAC2 inappropriate	
Cheese		5	4	Misclassified	
		10	3	Misclassified	
		12	4-6	Misclassified	
		14	4	PARAFAC2 inappropriate	
		16	2-3	PARAFAC2 inappropriate	
		20	3	Misclassified	
		58	3	PARAFAC2 inappropriate	
		65	2	Misclassified	
Wine		30	4	Misclassified	
		38	3	May be correct	
		41	2-4	PARAFAC2 inappropriate	

### Intervals not well described by PARAFAC2

These are cases where it is not possible to get a model which looks meaningful from a chemical point of view. An example of this is interval 2 from the cheese data (Figure 9). In the model with two factors (to the left), the two peaks have not been separated. Furthermore, the

factor which describes the baseline contains a number of small peaks. This indicates that this component is actually describing baseline plus at least one chemical compound. These observations indicate that additional factors should be included. In the model with three factors (to the right) there is still one elution profile which describes both of the peaks. Additionally, the spectra from component two (green profiles) are exclusively negative, which indicates that the model is over-fitted. In models with more than three factors included, the signs of over-fit become even more obvious. When inspecting the mass spectra of the two peaks (not shown) it can be seen that the two compounds have many fragments in common, but that they also have some differences. In this case we must therefore conclude that this interval cannot be well described with PARAFAC2. Imposing non-negativity in the estimated spectral profile and concentrations did not improve the models.

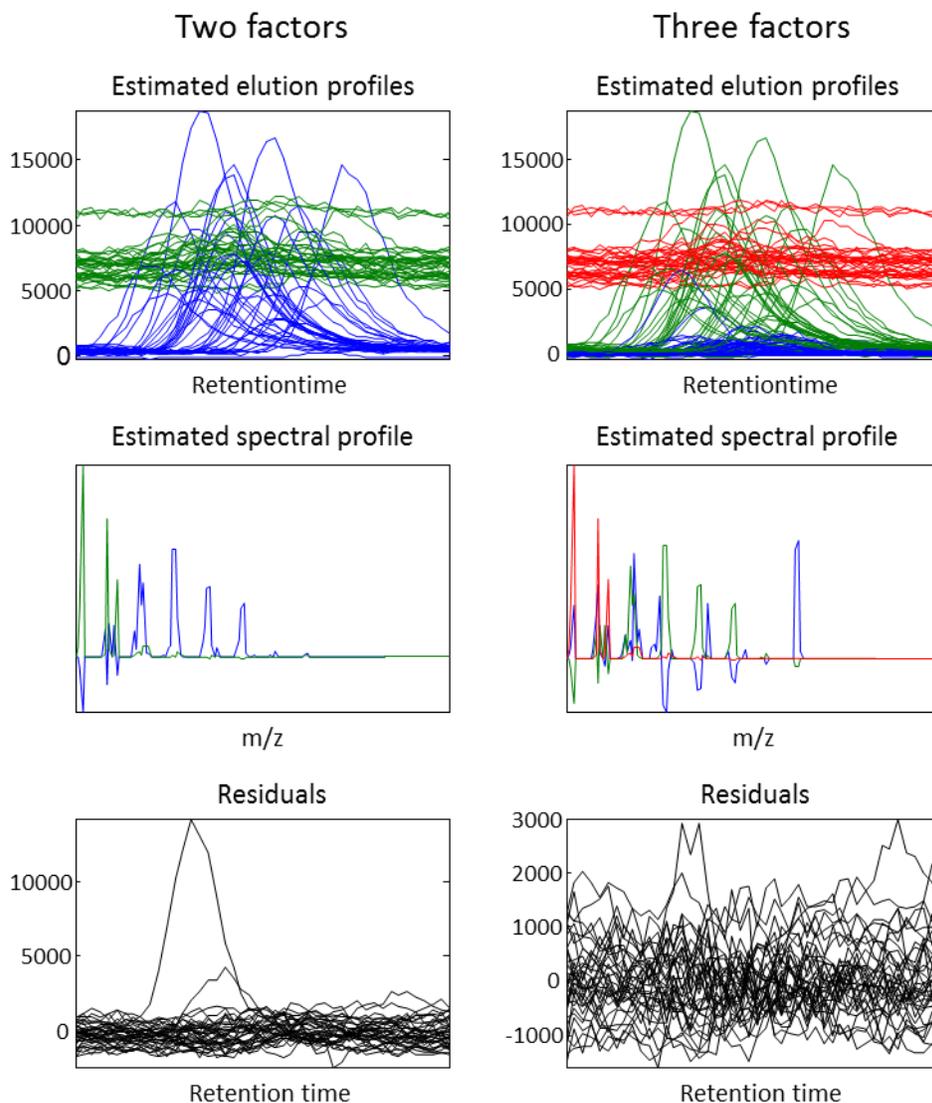


**Figure 9.** Interval 2 from the cheese data modelled with respectively two and three factors. The model with two factors seems to have too few factors, whereas the model with three seems to have too many factors included.

It might be possible to identify these kinds of models by including the right diagnostics. The development of such a method is out beyond the scope of this paper, but is definitely something which should be investigated further in future work.

### **May be correct**

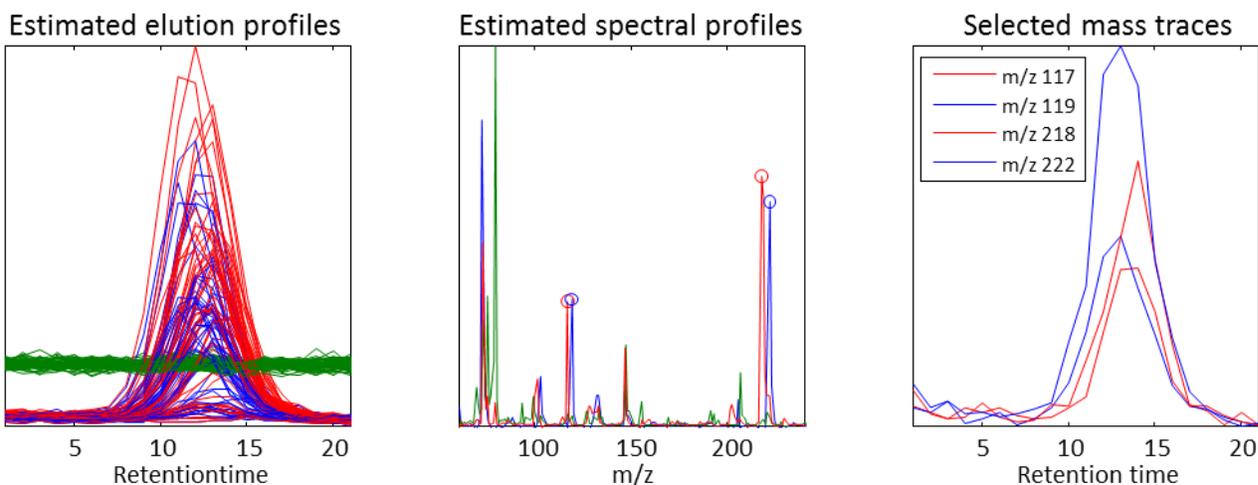
In cases assessed here as the classification might be right, there exists more than one number of factors for which it could be argued that the model is appropriate. An example of this could be interval 24 from the apple data. In Figure 10, the PARAFAC2 models with two and three factors, respectively, are illustrated. In the model with two factors (to the left) there is still a small amount of systematic behaviour in the residuals indicating that an additional factor should be included in the model. However, if an additional factor is included (to the right) there is an increase in negative values in the spectral profile indicating that too many factors have been included. By constraining the model with non-negativity in spectral profiles and concentration the model with three factors no longer have indications of being over-fitted. This indicates that the model with three factors indeed is not over-fitted as also suggested by the classification model.



**Figure 10.** Interval 24 from the apple data modelled with respectively two and three factors. The model with two factors seems to have too few factors since there still is some systematic behaviour in the residuals, but otherwise it seems like a nice model. The model with three factors seems to have too many factors included, due to the increase in negative values in the spectra profile; otherwise it is a nice model.

Another example of a model which might not be misclassified after all is from cheese, interval 22. Upon thorough inspection of the model and raw data, there are indications that this three factor model may have been correctly classified as not over-fitted and that it is the initial manual assessment which is wrong. The spectral profile obtained by the model (middle plot, Figure 11)

shows that there are some masses that separate the two peaks. If these masses are inspected in the raw data (leftmost plot in Figure 11), it can be seen that there actually seems to be a shift, indicating that two chemical compounds are eluting. However, the two compounds are poorly resolved (as also shown by the modelled elution time profiles (leftmost in Figure 11) and this might be the reason why PARAFAC2 are having some difficulties in modelling them.



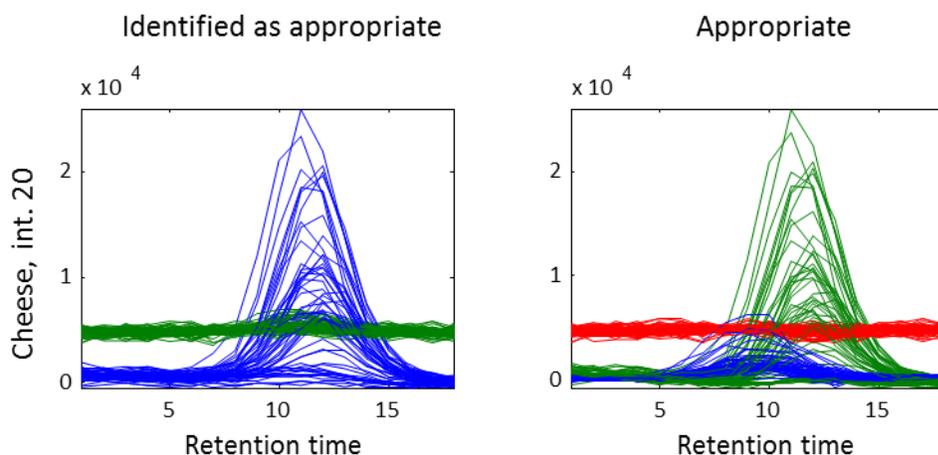
**Figure 11.** Illustration of the elution time profiles (left) and mass spectral profiles (middle) obtained from Cheese, interval 22, modelled with three factors. The leftmost plot shows the mass traces of the masses which separates the two peaks (indicated with circles in the middle plot).

## Misclassified

These models can be further divided into two sub-categories: misclassified as not over-fitted and misclassified as over-fit. However, there are no models which fall into the first category, indicating that the actual specificity of the classification model is very high. In the second category there are eight models.

In the cheese interval 5 there is a small amount of tailing which is not caught in the model identified as the correct one. However, neither the obtained spectral profile nor the concentration profiles (not shown) are affected by this. So from all practical aspects this will not affect any

further interpretation of data. In the apple data interval 13, a small peak is missing in the model identified as being appropriate. However, this peak is of such small magnitude that it does not affect the obtained concentration profiles and mass spectra, which are very similar in the two models. The same situation is observed for cheese, int. 20. In this case the missing peak is having a higher magnitude, but nevertheless the compounds are modeled meaningfully, and very similar to those in the appropriate model (illustrated in Figure 12). Furthermore both spectra and concentration profiles (not shown) of the main peak (which is present in both models) are practically unaffected. This means that further interpretation not will be affected besides the fact that a compound is missed.



**Figure 12.** Illustration of the difference in elution time profiles between the model appointed as being appropriate and the actually appropriate model (apple data, Interval 20).

The misclassifications of the remaining five intervals (apple, int. 1; cheese, int. 10, 12, and 65; wine, int. 30) result in models which deviate more severely from the appropriate model.

Summing up, these observations actually show that over 90% of the models are identified as correct with the suggested approach, are describing real underlying chemical information

(assuming that such a model can be found using PARAFAC2). This is to our best belief a significant improvement compared to the alternatives offered by the manufactures software.

## Conclusion

By the usage of different variable selection techniques we have found that a PLS-DA model based on seven quality criteria can be used to automate selection of the number of components in PARAFAC2. The seven quality criteria which turned out to be of most importance for the classification are:

- 1) Core consistency
- 2) Change in the negative area in the elution profile
- 3) Change in negative correlation between spectra
- 4) Logarithm to the number of iterations
- 5) The positive correlation between spectra
- 6) The relative change in how systematic the residuals are (indicated with Durbin Watson)
- 7) The relative change in the correlation between the TIC from raw data and the TIC from the obtained model.

The obtained PLS-DA model was evaluated on an independent test set. This validation showed that over 90% of the models which were determined to be appropriate by the automated procedure were describing the underlying chemistry. This makes the approach very useful for handling huge amounts of data, without the need of a skilled chemometrician to manually evaluate every model.

A MATLAB routine has been written which calculates PARAFAC2 models and finds valid models using the approach described in this paper. The function, which is called AutoChrome, is available at [www.models.life.ku.dk](http://www.models.life.ku.dk) (Dec. 2012). If the prediction of the number of compounds is

combined with additional tools<sup>2</sup>, identification can be accomplished using the open source software OpenChrom<sup>23</sup> combined with the NIST mass spectral library.

### Supporting Information

Thorough explanation of appointed diagnostics and illustration of estimated elutions profiles for cheese, interval 5 and apple, interval 13 are available as supplementary material. This material is available free of charge via the Internet at <http://pubs.acs.org>.

### Corresponding Author

\*E-mail: [dklgj@chr-hansen.com](mailto:dklgj@chr-hansen.com)

### Present Addresses

† Chr. Hansen A/S, Bøge Alle 10-12, 2970 Hørsholm, Denmark. Tel: 0045 4574 7474.

### Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

### References

- (1) Skov, T.; Bro, R. *Anal. Bioanal. Chem.* **2008**, *390*, 281-285.
- (2) Murphy, K. R.; Wenig, P.; Parcsi, G.; Skov, T.; Stuetz, R. M. *Chemometrics Intellig. Lab. Syst.* **2012**, *118*, 41-50.
- (3) Amigo, J. M.; Skov, T.; Bro, R. *Chem. Rev.* **2010**, *110*, 4582-4605.
- (4) Bro, R.; Andersson, C. A.; Kiers, H. A. L. *J. Chemom.* **1999**, *13*, 295-309.
- (5) Amigo, J. M.; Skov, T.; Bro, R.; Coello, J.; MasPOCH, S. *Trends Anal. Chem.* **2008**, *27*, 714-725.
- (6) Amigo, J. M.; Popielarz, M. J.; Callejón, R. M.; Morales, M. L.; Troncoso, A. M.; Petersen, M. A.; Toldam-Andersen, T. B. *J. Chromatogr., A* **2010**, *1217*, 4422-4429.
- (7) Khakimov, B.; Amigo, J. M.; Bak, S.; Engelsen, S. B. *J. Chromatogr., A* **2012**.
- (8) Harshman, R. A. *AUCLA Working Papers in Phonetics* **1970**, *16*, 1-84.

- (9) Kamstrup-Nielsen, M. H.; Johnsen, L. G.; Bro, R. *J. Chemom.* **Submitted 2012.**
- (10) Ballabio, D.; Todeschini, R. In *Infrared spectroscopy for food quality analysis and control*; Academic Press: 2009; pp 83-102.
- (11) Ståhle, L.; Wold, S. *J. Chemom.* **1987**, *1*, 185-196.
- (12) Carroll, J. D.; Chang, J. *Psychometrika* **1970**, *35*, 283-319.
- (13) Cattell, R. B. *Psychometrika* **1944**, *9*, 267-283.
- (14) Kiers, H. A. L.; ten Berge, J. M. F.; Bro, R. *J. Chemom.* **1999**, *13*, 275-294.
- (15) Bro, R.; Leardi, R.; Johnsen, L. G. *J. Chemom.* **Submitted 2012.**
- (16) Kruskal, J.; Harshman, R.; Lundy, M. *Multiway data analysis* **1989**, 115-122.
- (17) Draper, N. R.; Smith, H. *Applied regression analysis*; Wiley New York: 1981; .
- (18) Timmerman, M. E.; Kiers, H. A. L. *Br. J. Math. Stat. Psychol.* **2000**, *53*, 1-16.
- (19) Ballabio, D.; Skov, T.; Leardi, R.; Bro, R. *J. Chemom.* **2008**, *22*, 457-463.
- (20) Kanani, H.; Chrysanthopoulos, P. K.; Klapa, M. I. *J. Chromatogr., B* **2008**, *871*, 191-201.
- (21) Hoggard, J. C.; Synovec, R. E. *Anal. Chem.* **2007**, *79*, 1611-1619.
- (22) Andersen, C. M.; Bro, R. *J. Chemom.* **2010**, *24*, 728-737.
- (23) Wenig, P.; Odermatt, J. *BMC Bioinformatics* **2010**, *11*.
- (24) Savitzky, A.; Golay, M. J. E. *Anal. Chem.* **1964**, *36*, 1627-1639.
- (25) Lorenzo-Seva, U.; Ten Berge, J. M. F. *Methodology* **2006**, *2*, 57-64.
- (26) Rayens, W. S.; Mitchell, B. C. *Chemometrics Intellig. Lab. Syst.* **1997**, *38*, 173-181.
- (27) Bro, R.; Kiers, H. A. L. *J. Chemom.* **2003**, *17*, 274-286.
- (28) Furbo, S.; Christensen, J. H. *Anal. Chem.* **2012**, *84*, 2211-2218.
- (29) Skov, T.; Berg, F. v. d.; Tomasi, G.; Bro, R. *J. Chemom.* **2006**, *20*, 484-497.

# Supporting information

## Automated resolution of overlapping peaks in chromatographic data

*Lea G. Johnsen, José Manuel Amigo, Thomas Skov, Rasmus Bro*

Quality & Technology, Department of Food Science, Faculty of Science,

University of Copenhagen, Denmark

The supporting information contains an additional explanation of some of the diagnostics used in the main part. Furthermore a figure is included illustrating the estimated elution profiles for cheese, interval 5 and apple, interval 13.

## **Additional descriptions of selected parameters**

### **Smoothness (#4)**

To our experience, models with too many factors tend to have increased noise content. The smoothness is assessed as the mean of the sum of squared differences between the elution profile obtained from the model, and the smoothed profile. The smoothing was performed using Savitzky-Golay smoothing<sup>1</sup> with a width of 7 and a second order derivative.

### **Number of peaks/non-one-peakness (#5 and 6)**

For models with two few factors one elution profile may have more than one peak. The number of peaks is assessed by the peakfind algorithm available in the PLS\_Toolbox (Eigenvector Research). Non-one-peakness is defined as how much the elution profiles deviates from only containing one peak. It is determined as the difference between the modelled elution profile and the elution profile with unimodality constrains applied.

### **Negativity (#9-11)**

When PARAFAC2 models without non-negativity constraints are used, the model describing real components from a GC-MS dataset should not contain any negative values. However, when the model becomes over-fitted, it no longer describes real compounds and the negativity starts to increase. This is assessed both for elution profiles and spectral profiles. In both modes the negativity is assessed as the ratio between the sums of negative and positive values across the factors. In spectral mode, negativity is also assessed as the ratio between the sum of absolute values and raw values as:

$$P = \frac{\sum |x_{ij}|}{\sum x_{ij}}$$

where  $x_{ij}$  is the loading values for the  $i^{\text{th}}$  variable and the  $j^{\text{th}}$  factor.

### **Positive and negative correlation and congruence (#18-19 and 27-28)**

When a model with too many factors is applied on a dataset, two factors may be describing the same compound and hence spectra might be very similar. Lorenzo-Seva and ten Berge<sup>2</sup> has shown that Tucker's congruence coefficient is a useful index for the similarity of factors. The Tucker congruence coefficient is calculated as

$$\phi(x, y) = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2}}$$

where  $x_i$  and  $y_i$  are the loading values for the  $i^{\text{th}}$  variable in factor  $x$  and  $y$  respectively. Tucker's congruence coefficient differs from the more widely used correlation coefficient in that it is sensitive to additive constants, meaning that not only the shape of the factors but also the level is taken into account<sup>2</sup>.

The correlation and congruence are both assessed as the maximal positive and negative coefficient between any two spectra from that particular model. High negative coefficients could indicate two factor degeneracy<sup>3</sup> which also in some cases is an indication of over-fit.

### **Core consistency (#20)**

It has been shown that core consistency is a good indicator for when a model is over-fitted both in PARAFAC<sup>4</sup> and PARAFAC2<sup>5</sup>. For PARAFAC, core consistency is a measure of whether the modelled variation is low-rank trilinear or if other kinds of variation also are included in the model. In cases where the variation described by the model is truly low-rank trilinear core consistency is 100 and it decreases with more and more non-trilinear variation included in the model. The core consistency algorithm from PLS\_Toolbox (Eigenvector Research) is used to calculate the core consistency.

### **Split half (#21)**

In split half analysis the dataset is split into a number of subsets and a model is found for each of these. Solutions obtained from models with too many factors are not describing the true underlying chemistry and therefore the resulting models will not be identical

### **Baseline (#22)**

Furbo *et al.*<sup>6</sup> have shown that for PARAFAC, the appearance of a baseline is a useful indicator for proper models. Elution profiles describing baseline is defined as profiles where the peakfind available in the PLS\_Toolbox (Eigenvector Research) finds no peaks.

### **Systematic residuals (#29-32)**

Models with too few factors will often be characterised by having systematic residuals. The Durbin Watson<sup>7</sup> criteria can be used as a measure for continuity; it calculates the ratio between the sum of the first derivate of a vector and the sum of the raw vector. High values indicate randomness whereas low values indicate correlation. The most intuitive approach is to sum the residuals across the spectral mode, and look for continuity in the “TIC” from the residuals. However, in addition we also calculated the Durbin Watson criteria for data summed over the retention time mode since we suspect that shift in retention time may cause problems. The parameter for each model is then assessed as both the maximal and median Durbin Watson criteria across samples.

### **Simplicity (#33 and 34)**

Simplicity<sup>8</sup> is using the summed squared eigenvalues from SVD to assess how systematic the variation of the residuals is. Simplicity is assessed both on residuals summed over time and masses.

## Illustration of cheese, int 5 and apple, int 13

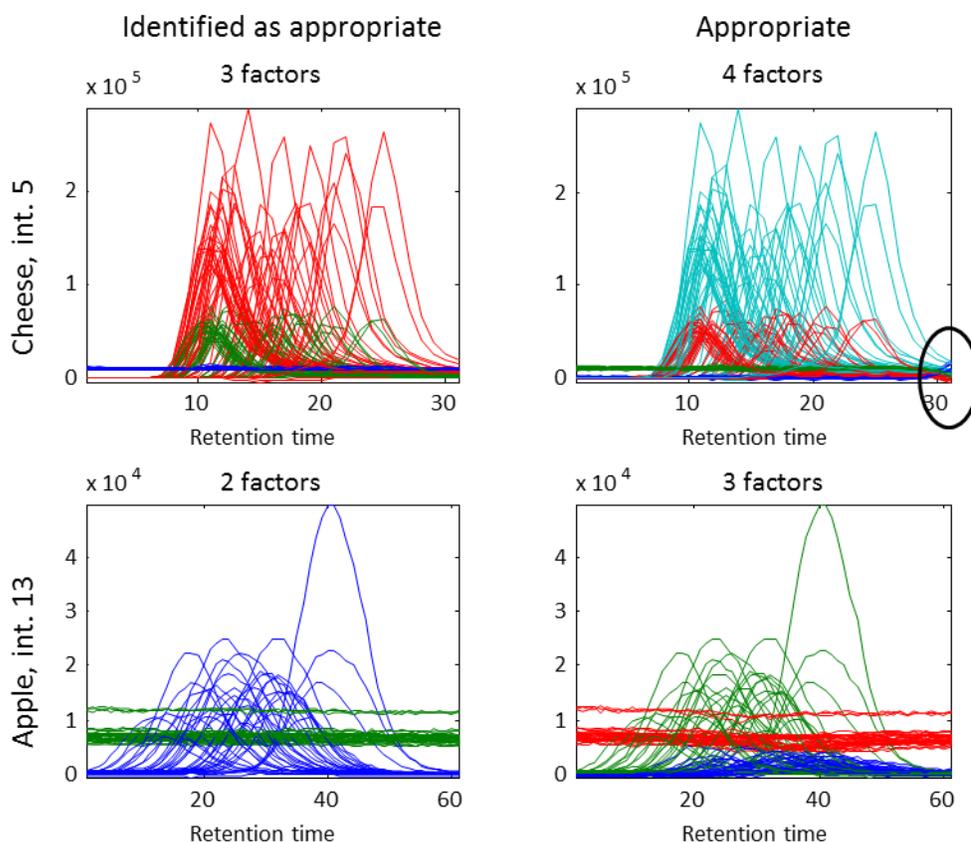


Figure S-1. Illustration of the difference in estimated elution time profiles between the model appointed as being appropriate and the appropriate models.

## Literature

- (1) Savitzky, A.; Golay, M. J. E. *Anal. Chem.* 1964, 36, 1627-1639.
- (2) Lorenzo-Seva, U.; Ten Berge, J. M. F. *Methodology* 2006, 2, 57-64.
- (3) Rayens, W. S.; Mitchell, B. C. *Chemometrics Intellig. Lab. Syst.* 1997, 38, 173-181.
- (4) Bro, R.; Kiers, H. A. L. *J. Chemom.* 2003, 17, 274-286.
- (5) Kamstrup-Nielsen, M. H.; Johnsen, L. G.; Bro, R. *J. Chemom.* Submitted 2012.
- (6) Furbo, S.; Christensen, J. H. *Anal. Chem.* 2012, 84, 2211-2218.
- (7) Draper, N. R.; Smith, H. *Applied regression analysis*; Wiley New York: 1981; .
- (8) Skov, T.; Berg, F. v. d.; Tomasi, G.; Bro, R. *J. Chemom* 2006, 20, 484-497.

