Data analysis and modeling

Part 2

Chemometrics

Andrzej Lasia
Université de Sherbrooke

2022

version 2.5 revised

"Torture the data long enough and they will confess to anything."[a]
Anonymous

"Statistics tell the biggest lies, everybody knows that! However, this is not necessarily true. It all depends on the user who is interpreting the results: If the statistical methods are applied appropriately, by somebody who understands their properties, excellent results are often reached."[a]

"Models are, for the most part caricatures of reality, but if they are good, then, like good caricatures, they portray, though perhaps in a distorted manner, some of the features of the real world."[b]

[a] Heikki Hyötyniemi, Multivariate Regression, Helsinki University of Technology, 2001.
[b] M. Kac, Science, 166 (1989) 695 .

Table of content

Table of Exercises

1 Multivariate data analysis

The purpose of data analysis in chemistry is to obtain information from the experimental data. Modern data acquisition systems provide a lot of data. In the preceding volume we have seen that data reduction and modeling may proceed through determination of the means or applications of linear, nonlinear, or multiple regressions. However, these methods often use a limited amount of acquired data.

1.1 Univariate approaches

In the preceding volume the **univariate** approach to the regression analysis was presented in which only one measurement was performed for one sample, $y_i = f(x_{1,i}, x_{2,i},\ldots)$. However, in practice, usually more than one measurement is carried out on each sample.

Let us look, for example, at the UV/VIS spectra of the mixtures of two chemical species. Modern spectrophotometers can acquire absorbance measurements every nm or less. However, in practical applications, usually multiple linear regression is used where absorbance at one selected wavelength is studied as function of concentrations. An example of spectra containing two components, registered every 1 nm (100 wavelength points, 9 spectra for different concentrations of components A and B), is displayed in Fig. 1.1. There are two overlapping peaks visible. Classical (univariate) **multiple linear regression** as shown in the previous book would use dependence of absorbance at one wavelength as a function of two concentrations:

$$A_i = b_A C_{A,i} + b_B C_{B,i} \tag{1.1}$$

which assumes additivity of absorbances and the Beer's law: $A = b\ C$, where $b = a\ l$, a is the specific absorptivity and l is the length of the optical path in the cell.

If the pure spectra of two individual component are not known the choice of the wavelength is a little bit arbitrary. In the presented example it looks like there are two peaks and the best choice would probably be somewhere in between, see the vertical line.

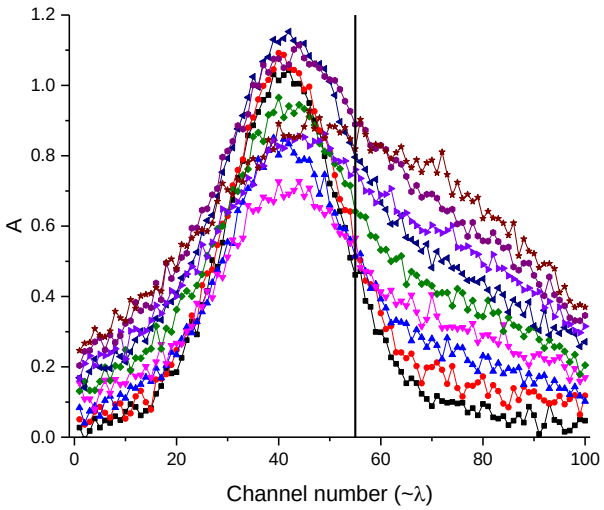


Fig. 1.1. Example of spectra of two compounds with overlapping peaks. Vertical line indicates a choice of the wavelength for a simple multiple regression.

The above model might be written as (previously it was written as **Y** = **Xb** + **ε)**:

$$\mathbf{A} = \mathbf{C}\mathbf{b} + \boldsymbol{\varepsilon} \tag{1.2}$$

with

$$\mathbf{A} = \begin{bmatrix} A_1 \\ A_2 \\ \ldots \\ A_N \end{bmatrix} \tag{1.3}$$

$$\mathbf{C} = \begin{bmatrix} C_{\mathrm{A},1} & C_{\mathrm{B},1} \\ C_{\mathrm{A},2} & C_{\mathrm{B},2} \\ \ldots & \ldots \\ C_{\mathrm{A},N} & C_{\mathrm{B,N}} \end{bmatrix} \tag{1.4}$$

and

$$\mathbf{b} = \begin{bmatrix} b_{\mathrm{A}} \\ b_{\mathrm{B}} \end{bmatrix} \tag{1.5}$$

are the absorptivity coefficients, for which the solution is:

$$\mathbf{b} = \left(\mathbf{C}'\mathbf{C}\right)^{-1}\mathbf{C}'\mathbf{A} \tag{1.6}$$

Coefficients **b** can be used to predict the spectra. However, from the new unknown spectrum one cannot obtain two concentrations if the measurements were made at one wavelength.

1.2 Multivariate approaches

In the preceding part we have considered that univariate vector of the measured parameter *A* is a function of one or more parameters *C*. **Multivariate analysis** uses **multiple responses** (e.g. absorbances at different wavelengths) to **multiple predictors** (e.g. concentrations).

Let us look at the analysis of absorbances (spectrophotometer responses),[1,2] for example at two different wavelengths roughly corresponding to two overlapping peaks, close to their suspected maxima, Fig. 1.2.

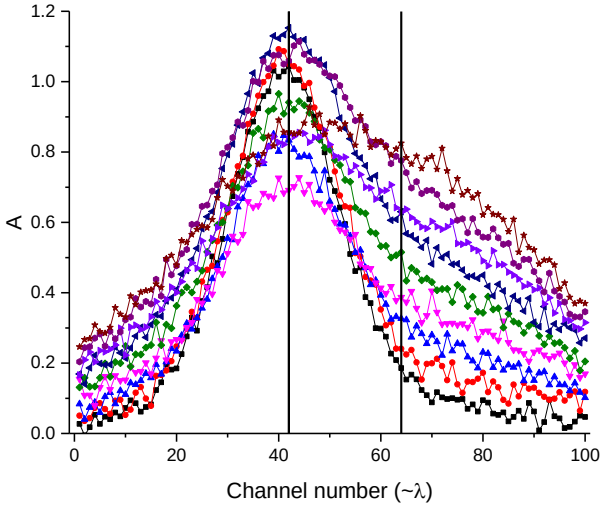


Fig. 1.2. Choice of two wavelengths to analyze spectra of two components with overlapping peaks.

In such a case one can write the problem with two columns of absorbances in Eq. (1.2), that is:

$$\mathbf{A} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \\ \ldots & .. \\ A_{N,1} & A_{N,2} \end{bmatrix} \tag{1.7}$$

where each column corresponds to one wavelength and each row to the a different sample, matrix of concentrations **C** is described by Eq. (1.4), and **b** by:

$$\mathbf{b} = \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix} = \begin{bmatrix} b_{\mathrm{A},1} & b_{\mathrm{A},2} \\ b_{\mathrm{B},1} & b_{\mathrm{B},2} \end{bmatrix} \tag{1.8}$$

where each row represents one compound and column corresponds to one wavelength. The predictions of concentrations $\hat{\mathbf{C}}$ are obtained from the solution of Eq. (1.2):

$$\hat{\mathbf{C}} = \mathbf{A}\mathbf{b}'(\mathbf{b}\mathbf{b}')^{-1} \tag{1.9}$$

Of course, it is better to use **more wavelengths** to have an averaging effect. However, only a small portion of all the experimental points is used in this analysis. One should notice that the number of wavelengths must be larger than the number of components to obtain the concentrations. The above equation may also include a **constant** term in **b** but **data centering with respect to the mean value removes that term**. However, this method is based on the Beer's law and the **complete composition**, i.e., concentration of each constituent must be known, and the baseline effects must be negligible or previously known.

If the **concentrations of all the compounds in the solution are not known** the above presented classical multiple linear regression may introduce a significant error in the determined absorptivity coefficients, **b**. The alternative approach is to use **inverse least squares method**. In

real samples often only concentrations of few components are known and are of interest in the analysis. In such cases the Beer's law might be rearranged to:

$$C = \frac{A}{b} = PA \tag{1.10}$$

where $P = 1/b$. It there are two components in the solution the equation the equation might be writes as:

$$\begin{aligned} C_{\mathrm{A}} &= A_1 P_{\mathrm{A},1} + A_2 P_{\mathrm{A},2} + E_{\mathrm{A}} \\ C_{\mathrm{B}} &= A_1 P_{\mathrm{B},1} + A_2 P_{\mathrm{B},2} + E_{\mathrm{B}} \end{aligned} \tag{1.11}$$

where indices 1 and 2 correspond to the wavelengths 1 and 2 and E are the errors. This equation may be written in the matrix form as:

$$\mathbf{C} = \mathbf{PA} + \mathbf{E} \tag{1.12}$$

with the solution for matrix **P**:

$$\mathbf{P} = \mathbf{C}\mathbf{A}'(\mathbf{A}\mathbf{A}')^{-1} \tag{1.13}$$

which allows determination of concentrations from Eq. (1.12):

$$\hat{\mathbf{C}} = \mathbf{PA} \tag{1.14}$$

It should be noticed that in this case the number of selected wavelengths cannot exceed the number of calibration (**training**) samples which is a significant limitation this method. This demands many training samples for good calibration.

Approaches described above are based on Beer's law and require the complete knowledge of composition of every component in the mixture and are susceptible to baseline effects since equations used assume that the response at each wavelength is due entirely to the components studied.

The problems with the above approaches are that they cannot be used in complex mixture samples where the individual constituents have overlapping spectral peaks and the band selection can be difficult if the spectra of individual components are not known. In such cases large prediction errors will appear.

1.3 Chemometrics multivariate approach (latent variables method)

The above method uses only **a few wavelengths** which increases the estimation error. Moreover, in the mixture of **more** components, when the absorption peaks overlap, the concentration determination is even more difficult. These classical methods will not be considered here in detail as they are less important and are described in the literature.[3]

The methods developed by **chemometrics** deal much better with such problems. They might use **all the spectra** containing hundreds of experimental wavelengths and they work even

- when the total composition of the mixture is not known,
- in the presence of the baseline,
- in the presence of many compounds,
- even if the linear relation between the measured signal and concentration is not observed.

Moreover, from the measured spectra, the spectra of each pure component might be found. Such an analysis might be carried out in other applications, e.g., in instrumental analytical chemistry: spectroscopies IR UV/visible, atomic spectroscopy, chromatography (HPLC, gas-mass GC-MS), NMR, optical spectroscopy and pyrolysis, electroanalysis, etc. It has been applied

to pharmaceutical and food chemistry, manufacturing industry, process chemistry, biological and medical chemistry, forensics (determination of origin of samples), image analysis, materials chemistry (including thermal analysis), physical chemistry of equilibria, reactions, process analytics, etc.[9]

There are different **definitions of chemometrics**, for example:

"Chemometrics is the branch of chemistry concerned with the analysis of chemical data (extracting information from data) and ensuring that experimental data contain maximum information (the design of experiments)" or "How to get chemically relevant information out of measured chemical data, how to represent and display this information, and how to get such information into data."[4]

"Chemometrics is the discipline of analytical chemistry concerned with the application of statistics, mathematics, and other methods of formal logic to the generation and analysis of chemical data."[5]

"Chemometrics is the chemical discipline that uses mathematical, statistical and other methods employing formal logic (i) to design or select optimal measurement procedures and experiments, and (ii) to provide maximum relevant chemical information by analyzing chemical data."[11]

These definitions stress dealing with analysis of **large amount of experimental data**. Multivariate analysis uses power of abstract matrix analysis. In contrast with the univariate analysis the authors are using **different notation** in chemometrics. The matrix containing the measurements (e.g. spectra) is called $\mathbf{X}(I \times J)$, in which I rows contain, for example, I spectra, each for a different sample, each sample measured at J wavelengths. Each spectrum (sample) is for a given concentration composition and there are I samples. The concentration matrix is $\mathbf{C}(I \times K)$ where K is the number of chemical species. In order to have a sensible model, the number of compounds must be less than or equal than the smaller of the number of experiments or number of variables. Methods described in this chapter are based on the **Principal Component Analysis**, **PCA**.

As mentioned above, multiple linear regression methods have the disadvantage that all significant chemical components of the mixture must be known. **PCA based methods do not require details about the spectra or concentrations of all the compounds** in a mixture, although it is important to make an estimation of how many significant components characterize our mixture, but it is not necessary to know their characteristics (e.g. spectra).

Principal components analysis is based on an abstract mathematical matrix operations and another representation of the matrices **X** and **C**. Details will be presented in the following chapters, after introducing some notions of the matrix algebra which is used in the analysis.

1.4 History of chemometrics

Mathematical foundations for the multivariate analysis were proposed by Karl Pearson in 1901.[6] Statisticians first developed these techniques in different areas: psychometrics (1930'), econometrics, and biometrics. The term chemometrics was introduced to the literature by Swedish chemist S. Wold in 1971[7]. In 1974 the American analytical chemist B.R. Kowalski founded the International Chemometrics Society. The major difference between chemometrics and other domains is in data collection. In social/economic sciences collected data are unique and must be analyzed with a rather limited possibility of repetition. In chemistry data acquired

using different analytical methods (most often spectra: UV/VIS, NIR, chromatograms: GC, HPLC, GC-MS, LC-MS electrochemistry, etc.) can be repeatedly acquired and number of points in each spectrum increased. There is also a different approach to outliers which in the social sciences might be very important but in physical sciences can be checked by repeating the measurement. Besides, functional i.e. linear or nonlinear relations between parameters are expected in chemistry, therefore, data analysis in chemistry is often different from that used in social/biological sciences and it is more rigorous.

There is now rich literature on chemometrics[1-3,6-23] which may be consulted.

1.5 Problems with learning chemometrics

Brereton, a leading chemist working in chemometrics stated several problems with learning and using chemometrics analysis. Here are few citations from his book:[10]

"a problem in chemometrics is that lots of people, often without a good mathematical or computational background, want to 'use' it. Often, I am surprised that people without any prior knowledge of this subject feel that they can pick it up in a workshop that 'should not last too long'. They want to walk in, then walk out and understand how to do pattern recognition in a couple of afternoons. This desire, unfortunately, is an important economic driving force in this subject. I say to my students that it may take a year or so just working through examples to learn the basis and gain sufficient feel for the subject. They accept this, but that is why they are giving up so much of their time to learn chemometrics. If they did not accept this, they would not be my students. The dilemma though is that for chemometrics to become widespread there should be a big user base. This is where the subject, and especially the application of pattern recognition, has a problem – almost like a split personality. Keep the subject theoretical and to an elite who are really good at maths and computing, and it is not widespread. Tell an analytical chemist in the lab that he or she cannot do any pattern recognition, and many will turn round, download a package, and put some data through and go away, even if he or she cannot understand the results. A few will get interested and learn but then they need to be in an environment where they have a lot of time – and many employers or even research supervisors will not allow this. So, most will either give up or try to cut corners. They will pay money for chemometrics, but not to spend a year or two learning the ropes, but rather to buy a package, that they believe does what they want, and go on a course that will teach them how to enter data and print out results in a couple of afternoons. The people that market these packages will make it easy for someone to take a series of spectra, import them into a package, view a graph on a screen, change the appearance with a mouse or a menu and incorporate into a nice report in Word that will be on their boss's (or their sponsor's) desk within a few hours. They won't gain much insight, but they will spread the word that chemometrics is a useful discipline. The course they go on will not really give them an insight into chemometrics (how can one in an afternoon?) but will teach them how to put data through a package and learn to use software and will catalyse the wider name recognition of the subject."

"Many of the early successes in chemometrics were in quite narrowly defined areas, NIR spectroscopy being one of them." "However since many NIR problems are quite straightforward from the chemometrics point of view (see in this text Case Study 2 – NIR of Food), in many cases no harm has come, and this at least demonstrates the tremendous power of multivariate methods for simplifying and visualizing data, even if the difficult part is the spectroscopic data handling, and so NIR spectroscopy can be considered correctly as an early success story and an important historic driving force of the subject."

"The problem is that over the past decade new sources of data have come on-stream, and this is particularly confusing many analytical chemists. The development of metabolic profiling, e.g. using coupled chromatography, mass spectrometry and nuclear magnetic resonance spectroscopy, has had a very fast development, with improved, more sensitive, and automated instruments. It looks easy, but it is not."

"The potential application of chemometrics to analytical data arising from problems in biology and medicine is enormous, but often the experimentalists have little understanding of how to acquire and handle these data. They want to learn but have only the odd afternoon or downloaded package with which to learn. They are funded to obtain data not to spend a year learning about Matlab. They usually want quick fixes."

"The biologists are anxious to be first to publish their 'marker compounds' and to claim that their work is a success and see data analysis as the afterthought that can be done on a Friday afternoon once all experiments are complete. So they will turn to the user-friendly packages and afternoon workshops and learn how to use the mouse and the menu and get a graph for incorporation into their report and then move on to the next project. Many do not realize that the methods they are using probably were developed for different purposes. Most chemometrics methods have their origins in traditional analytical chemistry, where there are often underlying certainties, for example in calibration we know what answer we are aiming for and as such just want to get our multivariate method as close as possible to the known answer. In some of the original applications of chemical pattern recognition such as spectroscopy we know what the underlying groups of compounds are and want our methods to classify spectra as effectively as possible into these groupings. We aim for 100 % accuracy and the original algorithms were considered to be better the more accurate the answer. With nice reproducible spectra, a known solution, and no hidden factors, this was possible. But there often is no certain answer in biology, for example, we cannot be sure that by measuring some compounds in a patient's serum that we can predict whether they will develop kidney disease within the next five years: we are uncertain whether there will be an answer or not. We are testing hypotheses as well as trying to obtain accurate predictions, and now do not just want to predict properties with a high degree of accuracy, but also to determine whether there really is sufficient information in the analytical data to detect the desired trend. Overfitting involves overinterpreting data and seeing trends that are not really there. Many biologists do not have a feel for whether data are overfitted or not. One can start with purely random data and by a judicious choice of variables end up with graphs that look as if two arbitrarily selected groups are separate. Most people when submitting a paper for publication will actively seek out the graph that 'looks better' even if it is misleading."

All these problems need careful data preparation, understanding the chemometric methods and careful analysis. One of the problems is the lack of training data which can be analyzed. It this text I have prepared many examples which should be solved and compared with the solutions included. Of course, only a **small part of chemometrics** related to the **principal component analysis and calibration** is presented here. For more complex problems one should go to the scientific literature, both the books and good scientific publications.

Before presenting chemometric methods of analysis a review of matrix operations will be presented below.

2 Matrix operations

2.1 Simple matrix operations

A matrix is a rectangular table of numbers and a vector is one column or row of numbers (column vector and row vector). Examples are shown in Eq. (2.1):

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \tag{2.1}$$

where matrix **A** has three columns and four rows **A**(3×4) and vector **v**(4) has four rows.

Basic operations include multiplication by a constant:

$$b\mathbf{A} = \begin{bmatrix} ba_{1,1} & ba_{1,2} & ba_{1,3} \\ ba_{2,1} & ba_{2,2} & ba_{2,3} \\ ba_{3,1} & ba_{3,2} & ba_{3,3} \\ ba_{4,1} & ba_{4,2} & ba_{4,3} \end{bmatrix} \qquad b\mathbf{v} = \begin{bmatrix} bv_1 \\ bv_2 \\ bv_3 \\ bv_4 \end{bmatrix} \tag{2.2}$$

and addition:

$$\begin{aligned} \mathbf{A} + \mathbf{B} &= \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{bmatrix} + \begin{bmatrix} b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,1} & b_{2,2} & b_{2,3} \\ b_{3,1} & b_{3,2} & b_{3,3} \\ b_{4,1} & b_{4,2} & b_{4,3} \end{bmatrix} = \begin{bmatrix} a_{1,1}+b_{1,1} & a_{1,2}+b_{1,2} & a_{1,3}+b_{1,3} \\ a_{2,1}+b_{2,1} & a_{2,2}+b_{2,2} & a_{2,3}+b_{2,3} \\ a_{3,1}+b_{3,1} & a_{3,2}+b_{3,2} & a_{3,3}+b_{3.3} \\ a_{4,1}+b_{4,1} & a_{4,2}+b_{4,2} & a_{4,3}+b_{4,3} \end{bmatrix} \\ \mathbf{v} + \mathbf{z} &= \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} + \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} v_1 + z_1 \\ v_2 + z_2 \\ v_3 + z_3 \\ v_4 + z_4 \end{bmatrix} \end{aligned} \tag{2.3}$$

The **identity** matrix, **I**, is a **square** matrix whose diagonal elements are equal to 1:

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

and **diagonal** matrix is a square matrix in which only diagonal elements are different from zero:

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & 0 & 0 \\ 0 & a_2 & 0 & 0 \\ 0 & 0 & a_3 & 0 \\ 0 & 0 & 0 & a_4 \end{bmatrix} \tag{2.5}$$

For matrix **transposition** operation is defined as swapping the columns and rows around and is denoted as **A**' or $\mathbf{A}^{\mathrm{T}}$:

$$\mathbf{A}=\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{bmatrix} \quad \mathbf{A}'=\mathbf{A}^{\mathrm{T}}=\begin{bmatrix} a_{1,1} & a_{2,1} & a_{3,1} & a_{4,1} \\ a_{1,2} & a_{2,2} & a_{3,2} & a_{4,2} \\ a_{1,3} & a_{2,3} & a_{3,3} & a_{4,3} \end{bmatrix} \tag{2.6}$$

2.2 Multiplication

Multiplication of matrices is possible only if the number of columns in the first matrix equals number of rows in the second matrix:

$$\mathbf{A}=\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} b_{1,1} & b_{1,2} & b_{1,3} \\ b_{2,1} & b_{2,2} & b_{2,3} \\ b_{3,1} & b_{3,2} & b_{3,3} \end{bmatrix}$$
$$\mathbf{C}=\mathbf{AB}=\begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,1} & c_{3,2} & c_{3,3} \\ c_{4,1} & c_{4,2} & c_{4,3} \end{bmatrix} \tag{2.7}$$

For the multiplication of $\mathbf{A}(I\times J)$ by $\mathbf{B}(J\times K)$ gives $\mathbf{C}(I\times K)$ and the elements are calculated as:

$$c_{i,k}=\sum_{j=1}^{J} a_{i,j}b_{j,k} \tag{2.8}$$

Multiplication of matrices is noncommutative, that is, in general:

$$\mathbf{AB}\neq\mathbf{BA} \tag{2.9}$$

even if the matrix dimensions allow such an operation. Multiplying more than two matrices it is not important which neighboring matrices are multiply first, but the general matrix order should be kept:

$$\mathbf{ABC}=(\mathbf{AB})\mathbf{C}=\mathbf{A}(\mathbf{BC}) \tag{2.10}$$

Matrix multiplication is also distributive:

$$\mathbf{A}(\mathbf{B}+\mathbf{C})=\mathbf{AB}+\mathbf{AC} \tag{2.11}$$

Multiplication of the matrix by the **identity** matrix does not change its value:

$$\mathbf{AI}=\mathbf{IA}=\mathbf{A} \tag{2.12}$$

Product of vectors is a multiplication of the row by the column. For two vectors **v** and **w**:

$$\mathbf{v}=\begin{bmatrix} v_1 \\ v_2 \\ \dots \\ v_N \end{bmatrix} \quad \mathbf{w}=\begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_N \end{bmatrix} \tag{2.13}$$

their product is a scalar i.e. one value (**scalar** or **dot product of vectors**):

$$\mathbf{v}'\mathbf{w}=v_1w_1+v_2w_2+\dots+v_Nw_N \tag{2.14}$$

For square matrices one can usually find **inverse** that is when original matrix is multiplied by its inverse one obtains the identity matrix:

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I} \tag{2.15}$$

If matrix **A** is not square then so-called **pseudoinverse**, $\mathbf{A}^{+}$ can be used. It is defined as:

$$\mathbf{A}\mathbf{A}^{+}\mathbf{A} = \mathbf{A} \tag{2.16}$$

If the number of columns is less than the number of rows, i.e. for **A**(*I*,*J*), *J* < *I*, the pseudoinverse is:

$$\mathbf{A}^{+} = \left(\mathbf{A}'\mathbf{A}\right)^{-1}\mathbf{A}' \tag{2.17}$$

and if number of columns exceeds the number of rows *J* > *I*:

$$\mathbf{A}^{+} = \mathbf{A}'\left(\mathbf{A}\mathbf{A}'\right)^{-1} \tag{2.18}$$

Where **A'A** or **AA**' are square matrices.

It can be noticed that the pseudoinverse was already used in the classical least-squares method, because Jacobian matrix **X** is not square. To obtain pseudoinverse the following operations are made:

$$\begin{aligned} &\mathbf{Y} = \mathbf{X}\mathbf{b} \\ &\mathbf{X}'\mathbf{Y} = \mathbf{X}'\mathbf{X}\mathbf{b} \\ &\left(\mathbf{X}\mathbf{X}'\right)^{-1}\mathbf{X}'\mathbf{Y} = \left(\mathbf{X}\mathbf{X}'\right)^{-1}\left(\mathbf{X}'\mathbf{X}\right)\mathbf{b} \\ &\left(\mathbf{X}\mathbf{X}'\right)^{-1}\left(\mathbf{X}\mathbf{X}'\right) = \mathbf{I} \\ &\mathbf{b} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Y} = \mathbf{X}^{+}\mathbf{Y} \end{aligned} \tag{2.19}$$

Although in simple cases matrix calculations can be carried out on paper in practice all the matrix operations are carried out using various programs widely available (Excel, Matlab, Mathematica, Maple, FORTRAN, etc.).

The matrix is **symmetric** if:

$$\mathbf{A}' = \mathbf{A} \tag{2.20}$$

normal if:

$$\mathbf{A}'\mathbf{A} = \mathbf{A}\mathbf{A}' \tag{2.21}$$

and **orthogonal** if:

$$\mathbf{A}'\mathbf{A} = \mathbf{A}\mathbf{A}' = \mathbf{I} \tag{2.22}$$

In this case matrix **A** is **orthonormal**. An interesting property of orthogonal matrices is that their inversion is equivalent to the transposition:

$$\mathbf{A}^{-1} = \mathbf{A}' \tag{2.23}$$

Vectors are orthogonal if their scalar product is zero:

$$\mathbf{v}'\mathbf{w} = 0 \tag{2.24}$$

Vector norm $\|\mathbf{v}\|$ is the vector length calculated from the vectors' scalar product:

$$\begin{aligned} \|\mathbf{v}\| &= \sqrt{\left(\mathbf{v},\mathbf{v}\right)} \\ \left(\mathbf{v},\mathbf{v}\right) &= \mathbf{v}'\mathbf{v} = \sum_{i=1}^{N} v_i^2 \end{aligned} \tag{2.25}$$

Matrix **A**(4×3), in Eq. (2.1) consists of three vectors,

$$\mathbf{A}=\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \\ a_{4,1} & a_{4,2} & a_{4,3} \end{bmatrix}=\begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3\end{bmatrix}$$
$$\mathbf{a}_1=\begin{bmatrix} a_{1,1} \\ a_{2,1} \\ a_{3,1} \\ a_{4,1} \end{bmatrix}\ \mathbf{a}_2=\begin{bmatrix} a_{1,1} \\ a_{2,2} \\ a_{3,2} \\ a_{4,2} \end{bmatrix}\ \mathbf{a}_3=\begin{bmatrix} a_{1,3} \\ a_{2,3} \\ a_{3,3} \\ a_{4,3} \end{bmatrix} \tag{2.26}$$

In general, matrix can be composed of J vectors:

$$\mathbf{A}=\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & ... & a_{1,J} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & .. & a_{2,J} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & ... & a_{3,J} \\ ... & ... & ... & ... & ... & ... \\ a_{I,1} & a_{I,2} & a_{I,2} & a_{I,4} & ... & a_{I,J} \end{bmatrix}=\begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 & \mathbf{a}_4 & ... & \mathbf{a}_J\end{bmatrix} \tag{2.27}$$

2.3 Rank

Matrix **rank** is defined as the maximum number of the linearly independent vectors $\mathbf{a}_i$ which is lower or equal to the number of columns:

$$\text{rank}\left(\mathbf{A}(I\acute{}J)\right)\le J \tag{2.28}$$

For matrices column rank equals row rank equals rank. Therefore, matrix rank is the same for matrix **A** and its transpose **A**':

$$\text{rank}\left(\mathbf{A}\right)=\text{rank}\left(\mathbf{A}'\right) \tag{2.29}$$

Rank of the random matrix $n\times n$ is always n.

2.4 Eigenvalues and eigenvectors

For a square matrix **A**, one can write the following relation:

$$\mathbf{Av}=\lambda\mathbf{v} \tag{2.30}$$

where **v** is the **eigenvector** of matrix **A** and λ is the corresponding **eigenvalue**. Matrix **A**(N,N) can have no more than N eigenvalues which follow the **characteristic equation**:

$$\det\left(\mathbf{A}-\lambda\mathbf{I}\right)=0 \tag{2.31}$$

where **det** is the determinant. Eq. (2.31) defines an algebraic equation of N-th order.

Moreover, the following relation is fulfilled:

$$\det\left(\mathbf{A}\right)=\lambda_1\times\lambda_2\times...\times\lambda_N \tag{2.32}$$

Eigenvalues of a matrix might be real or complex but if the matrix is symmetric, **A**=**A**' its eigenvalues are real. It should be stressed that a **symmetric matrix has orthogonal eigenvectors**.

A normal (in particular, a symmetric) matrix can be transformed into a **diagonal** matrix using **similarity transformation**:

$$\mathbf{A}=\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} \tag{2.33}$$

where:

$$\mathbf{\Lambda} = \mathrm{diag}\left(\lambda_1, \lambda_2, ...\lambda_N\right) = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ ... & ... & ... & ... \\ 0 & 0 & 0 & \lambda_N \end{bmatrix} \tag{2.34}$$

and matrix **V** is composed of the eigenvectors of **A**.
For example, for a three dimensional matrix **A**:

$$\mathbf{A} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 4 \\ 0 & 4 & 9 \end{bmatrix} \tag{2.35}$$

the characteristic polynomial of **A** is:

$$\det\left(\mathbf{A} - \lambda\mathbf{I}\right) = \det\left(\begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 4 \\ 0 & 4 & 9 \end{bmatrix} - \lambda\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) = \det\begin{bmatrix} 2-\lambda & 0 & 0 \\ 0 & 3-\lambda & 4 \\ 0 & 4 & 9-\lambda \end{bmatrix} \tag{2.36}$$

which gives:

$$\left(2-\lambda\right)\left[\left(3-\lambda\right)\left(9-\lambda\right)-16\right] = -\lambda^3 + 14\lambda^2 - 35\lambda + 22 \tag{2.37}$$

and has roots, i.e. eigenvalues:

$$\lambda_1 = 11; \quad \lambda_2 = 2; \quad \lambda_3 = 1 \tag{2.38}$$

or in a matrix form with eigenvalues as diagonal elements

$$\Lambda = \begin{bmatrix} 11 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.39}$$

corresponding to the eigenvectors:

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}; \quad \mathbf{v}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \quad \mathbf{v}_3 = \begin{bmatrix} 0 \\ -2 \\ 1 \end{bmatrix} \tag{2.40}$$

or in the matrix form

$$\mathbf{V} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & -2 \\ 2 & 0 & 1 \end{bmatrix} \tag{2.41}$$

It can be easily checked that $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$.

Exercise 2.1.

Find eigenvalues and eigenvectors of the matrix A in Eq. (2.35). Matlab program is aprog.m and the data are in a.m and VV.m in Ex2-1.

It should be noticed that Matlab produces eigenvectors:

V =

0	1.0000	0
-0.8944	0	0.4472
0.4472	0	0.8944

However, this matrix may be transformed into the one in Eq. (2.32)
V =

0	1	0
-2	0	1
1	0	2

by multiplication of the first and the third column by (1/0.4472) and both matrices are equivalent that is both fulfill Eq. (2.33): A = V*Λ*inv(V).

2.5 Singular value decomposition and pseudorank

In the first book we have seen **singular value decomposition** for a square matrix. Let us look now at it in more detail. The matrix undergoing decomposition does not have to be square. Let us suppose matrix **A**(*I*×*J*) which has a rank *J*. The singular value decomposition allows to represent matrix **A** as a product of three matrices **U,** Σ, and **V'**:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' \tag{2.42}$$

where matrices **U**(*I*×*I*) and **V**(*J*×*J*) are orthogonal:

$$\mathbf{U}'\mathbf{U} = \mathbf{V}'\mathbf{V} = \mathbf{I} \tag{2.43}$$

and the matrix **Σ**(*I*×*J*) is diagonal containing so called singular value elements of matrix **A** from the largest to the smallest.

When the rank of matrix **A** is *R*, the dimensions of the matrices are: **U**(*I*×*R*), **Σ**(*R*×*R*), and **V**(*J*×*R*) (or **V'**(*R*×*J*).

Matrix **U** is composed of *R* eigenvectors $\mathbf{u}_i$ and matrix **V** of *R* eigenvectors $\mathbf{v}_i$:

$$\mathbf{U}\mathbf{U}' = \mathbf{V}\mathbf{V}' = \mathbf{I} \tag{2.44}$$

The vectors **u** correspond to the eigenvalues λ_i of the matrix **AA'**:

$$\mathbf{A}\mathbf{A}'\mathbf{u} = \lambda\mathbf{u} \tag{2.45}$$

and vectors **v** to the eigenvalues λ_i of the matrix **A**'**A**:

$$\mathbf{A}'\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \tag{2.46}$$

Singular values of matrix **A** are nonnegative ($\sigma_i \geq 0$):

$$\mathbf{\Sigma} = \mathrm{diag}\left(\sigma_1, \sigma_2, ...\sigma_R\right)$$

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & ... & 0 \\ 0 & 0 & 0 & \sigma_J \end{bmatrix} \tag{2.47}$$

$$\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_R$$

Singular values of **A** are simply the square roots of the eigenvalues of matrix **AA**':

$$\sigma_i = \sqrt{\lambda_i} \text{ or } \lambda_i = \sigma_i^2 \tag{2.48}$$

The **condition number** of matrix **A** is the ratio of the largest to the lowest singular values:

$$\text{cond}(A) = \frac{\sigma_{\max}}{\sigma_{\min}} \tag{2.49}$$

Although matrix **Σ** might contain all nonzero diagonal values, that is mathematically its rank might be *J*, many **singular values may be very small** and the condition number very large. This might indicate that the smallest singular values correspond to the random noise. This suggests that the smallest singular values could be neglected because they model noise only. Although the mathematical rank of the matrix is still *J*, the **pseudo**- (or **effective**) **rank** might be $R < J$. In such a case all the singular values from $R+1$ to *J* should be zero and a new set of matrices is obtained: **U**(*I*×*R*) and **V**(*J*×*R*) or **V**'(*R*×*J*), and **Σ**(*R*×*R*).

In many practical cases we can predict the rank of matrix **A**. For example, when this matrix contains *I* spectra measured at *J* wavelength for the mixtures containing only two chemical components its rank should not be greater than two.

It can be added that the pseudoinverse of **A** might be calculated using singular value decomposition:

$$\mathbf{A}^+ = \mathbf{V}\mathbf{\Sigma}^+\mathbf{U}' \tag{2.50}$$

where $\mathbf{\Sigma}^+$ is the pseudoinverse of **Σ** obtained by replacing the nonzero values by their inverse:

$$\mathbf{\Sigma}^+ = \begin{bmatrix} 1/\sigma_1 & 0 & 0 & 0 \\ 0 & 1/\sigma_2 & 0 & 0 \\ 0 & 0 & ... & 0 \\ 0 & 0 & 0 & 1/\sigma_J \end{bmatrix} \tag{2.51}$$

When the condition number of the matrix **A** exceeds the computer precision or matrix pseudo rank is lower than *J* matrix inversion using full matrix $\mathbf{\Sigma}^+$ will introduce numerical noise as the smallest singular value σ_J in **Σ** will become the largest $1/\sigma_J$ in □+. In such a case number of elements in $\mathbf{\Sigma}^+$ must be reduced. For example, when the rank of **A** is 2, matrix $\mathbf{\Sigma}^+$, Eq. (2.51), must be simplified and all the values for $J > 2$ must be replaced by zeros:

$$\mathbf{\Sigma}^+ = \begin{bmatrix} 1/\sigma_1 & 0 & 0 & 0 & 0 \\ 0 & 1/\sigma_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & ... & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.52}$$

In the further analysis using chemometrics methods the measurement matrix **X** will be represented as a product of **scores T=UΣ** and **loadings P'=V**' obtained as:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' = (\mathbf{U}\mathbf{\Sigma})(\mathbf{V}') = \mathbf{T}\mathbf{P}' \tag{2.53}$$

Another algorithm, NIPALS (Non-linear iterative partial least-squares), may also be used for decomposition of matrix **X** into scores and loadings.[3,6]

Exercise 2.2.

To better understand the above matrix operations let us look at an example of the matrix **X**(10,8) shown in Table 2.1 (matrix elements are shown in bold). The numerical data are in Ex2-2 in Xdata.m and the program in prog.m which also uses subroutine pca.m.
Let us apply the singular value decomposition, Eq. (2.42), to **X**. The Matlab program is matrixop.m. This operation is only a different representation of the matrix **X**=**U*****S*****P**', Eq. (2.42). No information about **X** is lost. Using Matlab operation: [U,S,P]=svd(X) the following three matrices, shown in Table 2.2 - 2.5 are obtained. The matrix which will be later used **T**=**U*****S** and is also shown in Table 2.5; of course **X**=**T*****P**', Eq. (2.53).

Table 2.1. Example of the matrix **X**(10,8). Matrix elements are in bold.

	1	2	3	4	5	6	7	8
1	**0.318**	**0.413**	**0.335**	**0.196**	**0.161**	**0.237**	**0.290**	**0.226**
2	**0.527**	**0.689**	**0.569**	**0.346**	**0.283**	**0.400**	**0.485**	**0.379**
3	**0.718**	**0.951**	**0.811**	**0.521**	**0.426**	**0.566**	**0.671**	**0.526**
4	**0.805**	**1.091**	**0.982**	**0.687**	**0.559**	**0.676**	**0.775**	**0.611**
5	**0.747**	**1.054**	**1.030**	**0.804**	**0.652**	**0.695**	**0.756**	**0.601**
6	**0.579**	**0.871**	**0.954**	**0.841**	**0.680**	**0.627**	**0.633**	**0.511**
7	**0.380**	**0.628**	**0.789**	**0.782**	**0.631**	**0.505**	**0.465**	**0.383**
8	**0.214**	**0.402**	**0.583**	**0.635**	**0.510**	**0.363**	**0.305**	**0.256**
9	**0.106**	**0.230**	**0.378**	**0.440**	**0.354**	**0.231**	**0.178**	**0.153**
10	**0.047**	**0.117**	**0.212**	**0.257**	**0.206**	**0.128**	**0.092**	**0.080**

Table 2.2. Values of the matrix **U**(10,10) obtained from the singular value decomposition.

	1	2	3	4	5	6	7	8	9	10
1	**-0.15682**	**-0.20366**	**-0.16646**	**0.22393**	**0.36709**	**-0.20434**	**-0.72932**	**-0.11853**	**-0.36190**	**0.05136**
2	**-0.26502**	**-0.31242**	**0.15887**	**-0.48203**	**0.12140**	**0.33160**	**-0.00835**	**-0.15572**	**0.01562**	**0.65215**
3	**-0.37388**	**-0.36521**	**-0.40298**	**0.08792**	**0.13487**	**0.05403**	**0.20461**	**-0.46557**	**0.40624**	**-0.33460**
4	**-0.44543**	**-0.28630**	**0.10637**	**-0.04256**	**-0.53305**	**0.29815**	**-0.23403**	**0.38529**	**-0.13238**	**-0.33574**
5	**-0.45618**	**-0.06586**	**0.32555**	**0.18790**	**-0.14653**	**-0.71472**	**0.15783**	**0.06886**	**0.15078**	**0.24819**
6	**-0.40957**	**0.21099**	**0.05733**	**0.24063**	**0.60071**	**0.27312**	**0.31132**	**0.41729**	**-0.13413**	**-0.04532**
7	**-0.32775**	**0.41552**	**-0.48175**	**-0.24909**	**-0.24176**	**-0.12374**	**0.19570**	**-0.19757**	**-0.51875**	**0.07479**
8	**-0.23454**	**0.46683**	**0.55750**	**-0.14022**	**0.06535**	**0.12211**	**-0.22733**	**-0.47858**	**0.06479**	**-0.30412**
9	**-0.14843**	**0.38273**	**-0.33415**	**-0.30455**	**0.05290**	**-0.08488**	**-0.38390**	**0.34513**	**0.58719**	**0.07606**
10	**-0.08165**	**0.24423**	**-0.09759**	**0.66354**	**-0.31676**	**0.36555**	**-0.11044**	**-0.17444**	**0.16557**	**0.42410**

Table 2.3. Values of the matrix **S**(10,8) obtained from the singular value decomposition. Singular values are the diagonal elements.

	1	2	3	4	5	6	7	8
1	**5.00589476**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
2	**0**	**0.7495972**	**0**	**0**	**0**	**0**	**0**	**0**
3	**0**	**0**	**0.001453**	**0**	**0**	**0**	**0**	**0**
4	**0**	**0**	**0**	**0.001246**	**0**	**0**	**0**	**0**
5	**0**	**0**	**0**	**0**	**0.0007272**	**0**	**0**	**0**
6	**0**	**0**	**0**	**0**	**0**	**0.0005257**	**0**	**0**
7	**0**	**0**	**0**	**0**	**0**	**0**	**0.0003955**	**0**
8	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0.0001682**
9	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
10	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Table 2.4. Values of the matrix **P**(8,8) = **V**(8,8) obtained from the singular value decomposition.

	1	2	3	4	5	6	7	8
1	**-0.31738**	**-0.45261**	**-0.20945**	**-0.02083**	**-0.50115**	**0.22069**	**0.46343**	**0.36809**
2	**-0.45351**	**-0.37280**	**-0.06707**	**-0.02464**	**0.39682**	**-0.09756**	**0.27618**	**-0.63795**
3	**-0.45413**	**0.14220**	**0.14513**	**0.49760**	**0.44984**	**0.37853**	**-0.08726**	**0.38940**
4	**-0.36477**	**0.58974**	**0.41687**	**-0.14569**	**-0.38709**	**0.18671**	**0.23257**	**-0.29217**
5	**-0.29570**	**0.46664**	**-0.58725**	**-0.29508**	**0.19169**	**-0.37026**	**0.15630**	**0.25418**
6	**-0.30467**	**0.01603**	**-0.32196**	**0.48903**	**-0.44909**	**-0.22582**	**-0.50352**	**-0.24062**
7	**-0.32603**	**-0.22352**	**0.54535**	**-0.17778**	**-0.02342**	**-0.61881**	**-0.18239**	**0.31303**
8	**-0.26029**	**-0.14204**	**-0.10426**	**-0.61016**	**0.01454**	**0.43889**	**-0.57968**	**-0.00646**

Table 2.5. Values of the matrix **T**(10,8) obtained from the singular value decomposition, **T**=**U***S**.

	1	2	3	4	5	6	7	8
1	**-0.78504**	**-0.15266**	**-0.00024**	**0.00028**	**0.00027**	**-0.00011**	**-0.00029**	**-0.00002**
2	**-1.32666**	**-0.23419**	**0.00023**	**-0.00060**	**0.00009**	**0.00017**	**0.00000**	**-0.00003**
3	**-1.87160**	**-0.27376**	**-0.00059**	**0.00011**	**0.00010**	**0.00003**	**0.00008**	**-0.00008**
4	**-2.22979**	**-0.21461**	**0.00015**	**-0.00005**	**-0.00039**	**0.00016**	**-0.00009**	**0.00006**
5	**-2.28357**	**-0.04937**	**0.00047**	**0.00023**	**-0.00011**	**-0.00038**	**0.00006**	**0.00001**
6	**-2.05027**	**0.15816**	**0.00008**	**0.00030**	**0.00044**	**0.00014**	**0.00012**	**0.00007**
7	**-1.64070**	**0.31147**	**-0.00070**	**-0.00031**	**-0.00018**	**-0.00007**	**0.00008**	**-0.00003**
8	**-1.17409**	**0.34994**	**0.00081**	**-0.00017**	**0.00005**	**0.00006**	**-0.00009**	**-0.00008**
9	**-0.74302**	**0.28689**	**-0.00049**	**-0.00038**	**0.00004**	**-0.00004**	**-0.00015**	**0.00006**
10	**-0.40873**	**0.18307**	**-0.00014**	**0.00083**	**-0.00023**	**0.00019**	**-0.00004**	**-0.00003**

It should be noticed that singular values are always positive and are displayed as diagonal values in the decreasing order. As it will be shown in the next section limitations will be introduced to the dimension of the matrices. First, let us look into the eigenvalues which can be calculated as squares of the singular values, Eq. (2.48), which are:

$$\lambda_i = \sigma_i^2 \tag{2.54}$$

They are shown in Table 2.6.

Table 2.6. Eigenvalues (**SS**) determined from the matrix elements.

No	Eigenvalues
1	**25.05898235**
2	**0.56189599**
3	**0.00000211**
4	**0.00000155**
5	**0.00000053**
6	**0.00000028**
7	**0.00000016**
8	**0.00000003**

It is also obvious from Table 2.5 that the values in columns of the matrix **T** decrease with increase of the principal component, PC, number because they are obtained by multiplication by matrix **S** which contains diagonal elements in a decreasing order.

It can be noticed that the sum of the eigenvalues equals to the sum of the elements of entire matrix **X**:

$$\sum_{i=1}^{J} \lambda_i = \sum_{i=1}^{I} \sum_{j=1}^{J} \left(x_{i,j} \right)^2 \tag{2.55}$$

As it will be shown in the next chapter the first two eigenvalues constitute 99.15% of the sum of all the eigenvalues. This means that further components (beyond the second) have low significance (0.85%) and might be neglected and only two first values kept. In such a case the matrices **U**, **S, P**, and **T**=**U***S** might be limited to two columns, Table 2.7-2.11.

Table 2.7. Matrix **U**(10,2) limited to two columns according to the values of the eigenvalues.

	1	2
1	**-0.15682**	**-0.20366**
2	**-0.26502**	**-0.31242**
3	**-0.37388**	**-0.36521**
4	**-0.44543**	**-0.28630**
5	**-0.45618**	**-0.06586**
6	**-0.40957**	**0.21099**
7	**-0.32775**	**0.41552**
8	**-0.23454**	**0.46683**
9	**-0.14843**	**0.38273**
10	**-0.08165**	**0.24423**

Table 2.8. Matrix **S**(2,2) limited to two columns.

	1	2
1	**5.0058948**	**0**
2	**0**	**0.7495972**

Table 2.9. Vector of eigenvalues of the matrix **SS** limited to two values.

1	**25.05898235**
2	**0.56189599**

Table 2.10. Matrix **P**(8,2) limited to two columns.

	1	2
1	**-0.317379**	**-0.452612**
2	**-0.453514**	**-0.372799**
3	**-0.454126**	**0.142204**
4	**-0.364766**	**0.589742**
5	**-0.295701**	**0.466640**
6	**-0.304669**	**0.016029**
7	**-0.326035**	**-0.223520**
8	**-0.260287**	**-0.142039**

Table 2.11. Matrix **T**(10,2) limited to two columns.

	1	2
1	**-0.785044**	**-0.152663**
2	**-1.326664**	**-0.234190**
3	**-1.871602**	**-0.273758**
4	**-2.229786**	**-0.214606**
5	**-2.283565**	**-0.049366**
6	**-2.050269**	**0.158160**
7	**-1.640705**	**0.311473**
8	**-1.174090**	**0.349936**
9	**-0.743022**	**0.286894**
10	**-0.408728**	**0.183073**

Finally, a new matrix $\hat{\mathbf{X}}$ can be predicted from the truncated matrices **T** and **P**, $\hat{\mathbf{X}} = \mathbf{T}\,\mathbf{P}'$. It is shown in Table 2.12.

Table 2.12. Values of the matrix $\hat{\mathbf{X}}(10,8) = \mathbf{T}\,\mathbf{P}'$, obtained from the truncated singular value decomposition.

	1	2	3	4	5	6	7	8
1	**0.3182537**	**0.4129412**	**0.3347994**	**0.1963261**	**0.1608995**	**0.2367312**	**0.2900749**	**0.2260209**
2	**0.5270528**	**0.6889669**	**0.5691696**	**0.3458110**	**0.2830131**	**0.4004391**	**0.4848849**	**0.3785777**
3	**0.7179138**	**0.9508550**	**0.8110132**	**0.5212510**	**0.4256875**	**0.5658303**	**0.6713979**	**0.5260383**
4	**0.8048215**	**1.0912451**	**0.9820857**	**0.6867890**	**0.5592054**	**0.6759060**	**0.7749570**	**0.6108674**
5	**0.7471003**	**1.0540333**	**1.0300059**	**0.8038547**	**0.6522155**	**0.6949393**	**0.7555564**	**0.6013949**
6	**0.5791280**	**0.8708644**	**0.9535713**	**0.8411431**	**0.6800697**	**0.6271877**	**0.6331074**	**0.5111941**
7	**0.3797495**	**0.6279662**	**0.7893794**	**0.7821630**	**0.6305033**	**0.5048638**	**0.4653067**	**0.3828134**
8	**0.2142470**	**0.4020109**	**0.5829472**	**0.6346407**	**0.5104733**	**0.3633174**	**0.3045768**	**0.2558963**
9	**0.1059685**	**0.2300174**	**0.3782230**	**0.4402226**	**0.3535880**	**0.2309740**	**0.1781247**	**0.1526492**
10	**0.0468607**	**0.1171144**	**0.2116477**	**0.2570561**	**0.2062903**	**0.1274609**	**0.0923390**	**0.0803832**

In further calculations matrix **T** is arranged so that the largest absolute values of its vectors are positive. If they are not the signs of **T** and **P** corresponding vectors are changed. It of course does not affect the values of the matrix $\hat{\mathbf{X}}$ elements.

3 Principal component analysis, PCA

Principal component analysis let us answer the question: **how many factors** influence the obtained results? The answer might be found **without identifying** what are the factors, only finding how many there are. Usually, these factors are related to composition but other factors may also exist, e.g. temperature, electrical potential, etc.

3.1 Determination of the number of principal components, PC

As it was mentioned in Section 1.3 the original data matrix is denoted as **X**(I×J). Let us look at the spectroscopic (UV/VIS) study of the mixture of several components (of course this method is not limited to the spectroscopy). In such a case the rows of **X** contain I spectra (absorbances) measured at J columns (wavelengths), Eq. (3.1):

	λ_1	λ_2	λ_3	λ_4	...	λ_{J-1}	λ_J
spectrum 1	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	...	$x_{1,J-1}$	$x_{1,J}$
spectrum 2	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	...	$x_{2,J-1}$	$x_{2,J}$
...	...	...	...	...	...	...	...
spectrum I	$x_{I,1}$	$x_{I,2}$	$x_{I,3}$	$x_{I,4}$	...	$x_{I,J-1}$	$x_{I,J}$

(3.1)

This matrix might also represent UV/VIS/IR or mass spectra of chromatographic elution obtained at different times, currents in voltammetric studies i.e. currents at different potentials, etc. In the case of the UV/VIS spectroscopy we expect a relation, the Beer's law, between absorbances and concentrations **C**(I×K) where K is the number of chemical components in the mixture. The Beer's law can be written as:

$$\mathbf{X} = \mathbf{CS} + \mathbf{E} \tag{3.2}$$

and **S**(K×J) is the matrix of the spectra of pure chemical components (that is specific absorptivites a multiplied by the cell length, l, $S = a\ l$ and **E**(I×J) is the error matrix. **E** should contain only the random errors. As there are K chemical components the **X** matrix effective rank, R, should not exceed K. Eq. (3.2) is schematically illustrated in Fig. 3.1.

However, as we do not know the concentrations (which should be determined from the data analysis) another approach is used in the PCA. In this case we can use an abstract mathematical transformation in which matrix **X**(I×J) is presented as a product of **scores T**(I×R) and **loadings P'**(R×J), [or **P(**J×R)], Eq. (2.53), where the matrix effective rank, that is number of principal components, R, is smaller or equal to the number of chemical components, K, $R \le K$:

$$\mathbf{X} = \mathbf{TP}' + \mathbf{E} = \hat{\mathbf{X}} + \mathbf{E} = \sum_{r=1}^{R} \mathbf{t}_r \mathbf{p}_r' + \mathbf{E} \tag{3.3}$$

where $\mathbf{t}_r$ and $\mathbf{p}_r'$ are columns and rows of scores and loadings and $\hat{\mathbf{X}}$ is calculated using R principal components. For the individual elements x_{ij} one can write Eq. (3.3) as:

$$x_{ij} = \sum_{r=1}^{R} t_{ir} p_{jr} + e_{ij} \tag{3.4}$$

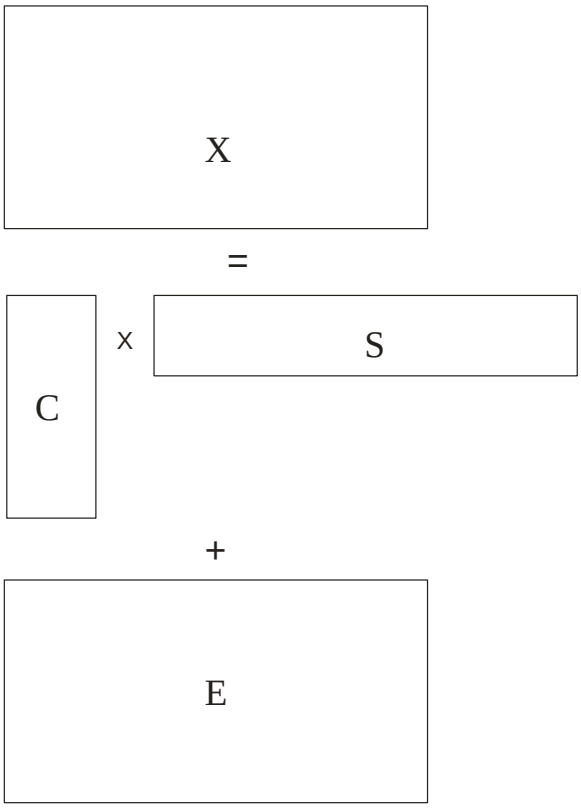


Fig. 3.1. Schematic illustration of the dependence of matrix **X**(*I*×*J*) on concentration **C**(*I*×*K*) and spectra of individual components **S**(*K*×*J*), based on the Beer's law, Eq. (3.2).

It should be stressed that this is an abstract mathematical transformation and matrices **T** and **P** are different form chemical matrices **C** and **S**. The matrix **pseudo-rank** is called **number of Principal Components**, **PC**s. Each Principal Component is characterized by one column score vector, $\mathbf{t}_j$ and one (transposed) row loading vector, $\mathbf{p}'_i$. **PCA allows us to determine the number of principal components** from the measurement matrix **X**. This number might be lower than number of compounds if the concentrations are linearly dependent or have negligible contributions. This produces so called **rank deficiency**; for example, there are five chemical components but two of them contribute negligibly to the total matrix and, in this case, the pseudo-rank is three.

In another example matrices **C** in Eq. (3.5) contain two linearly dependent columns and have rank of one (notice that for the left matrix col(2) = 2*col(1) and for the right matrix col(2) = 1 – col(1)). This is a bad choice for the calibration concentrations. It should also be noticed that Eq. (3.3) does not use any information about concentrations. PCA allows to determine **how many factors influence the spectra** in **X** without identifying its chemical origin.

$$\mathbf{C} = \begin{bmatrix} 0.1 & 0.2 \\ 0.2 & 0.4 \\ 0.3 & 0.6 \\ 0.4 & 0.8 \\ 0.5 & 1.0 \\ 0.6 & 1.2 \\ 0.7 & 1.4 \\ 0.8 & 1.6 \\ 0.9 & 1.8 \end{bmatrix} \quad \text{or} \quad \mathbf{C} = \begin{bmatrix} 0.1 & 0.9 \\ 0.2 & 0.8 \\ 0.3 & 0.7 \\ 0.4 & 0.6 \\ 0.5 & 0.5 \\ 0.6 & 0.4 \\ 0.7 & 0.3 \\ 0.8 & 0.2 \\ 0.9 & 0.1 \end{bmatrix} \tag{3.5}$$

As the matrices **T** and **P** contain only the important principal components, they may be used to calculate matrix of the predicted spectra, $\hat{\mathbf{X}}$:

$$\hat{\mathbf{X}} = \mathbf{TP}' \tag{3.6}$$

Eq. (3.3) is illustrated in Fig. 3.2 where matrix **X** may be represented as a sum of two multiplications of vectors $\mathbf{t_i}$ and $\mathbf{p'_i}$ or matrix multiplication of **T** and **P'** (plus matrix of residual errors **E**).

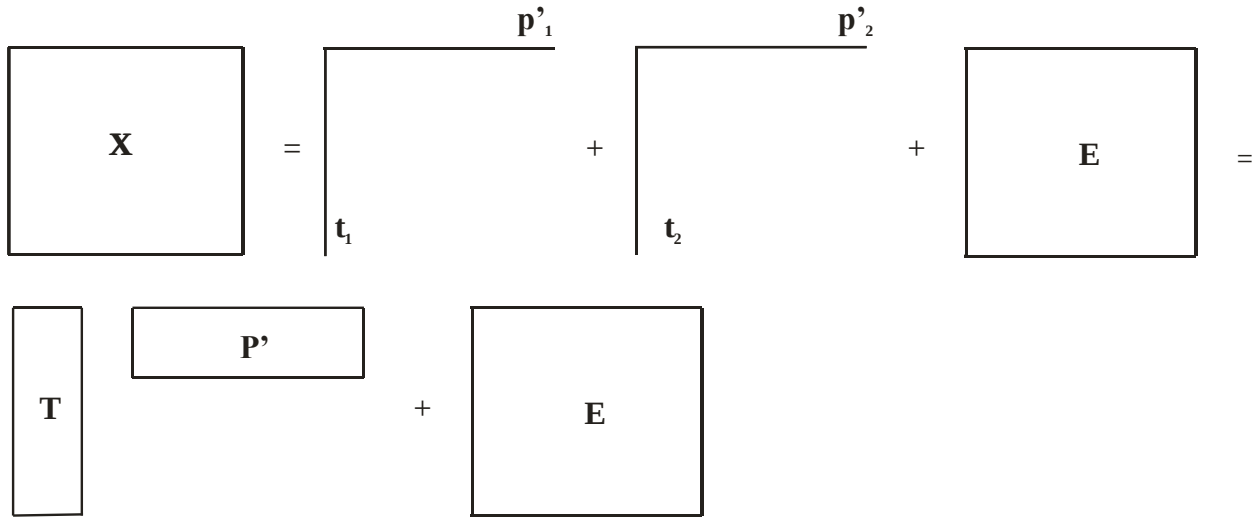


Fig. 3.2. Illustration of the decomposition of matrix $\mathbf{X}(I \times J)$ into scores $\mathbf{T}(I,R)$ and loadings $\mathbf{P}(R,J)$, according to Eq. (3.3); dimensions of error matrix is $\mathbf{E}(I \times J)$.

All **scores and loadings are orthogonal** which arises from the SVD method (they are eigenvectors), that is:

$$\sum_{i=1}^{I} t_{i,r_1} t_{i,r_2} = 0; \qquad \sum_{i=1}^{J} p_{r_1,j} p_{r_2,j} = 0 \tag{3.7}$$

where r_1 and r_2 are the principal components numbers and these relations hold for $r_1 \neq r_2$. Moreover, **loadings are normalized (orthonormal)**:

$$\sum_{j=1}^{J} p_{a,j}^2 = 1 \tag{3.8}$$

matrix **T'T** is diagonal (all elements are zero except diagonal) and **P'P** is an identity matrix:

$$\begin{aligned} \mathbf{T}'\mathbf{T} &= \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_R) \\ \mathbf{P}'\mathbf{P} &= \mathbf{I} \end{aligned} \tag{3.9}$$

The **size** of each PCA component is related to its eigenvalue; eigenvalues, λ_i, are squares of the singular values, σ_i, that is diagonal parameters of matrix **Σ**, $\lambda_i = \sigma_i^2$, Eq. (2.47)-(2.48). These values might be normalized and expressed in %:

$$\frac{\lambda_r}{\sum_{r=1}^{R} \lambda_r} 100\% \tag{3.10}$$

where the singular value λ_r is divided by the sum of all singular values. It can be noticed that the values of λ_r can also be obtained as the sum of squares of scores:

$$\lambda_r = \sum_{i=1}^{I} \mathbf{t}_{i,r}^2 \tag{3.11}$$

It should be noticed that **PCA decomposition is not unique** and introduces an ambiguity. In fact, Eq. (3.3) can be written as:

$$\mathbf{T}\mathbf{P}' = \mathbf{T}\mathbf{R}\mathbf{R}^{-1}\mathbf{P}' = (\mathbf{T}\mathbf{R})(\mathbf{P}\mathbf{R})' = \tilde{\mathbf{T}}\tilde{\mathbf{P}}' \tag{3.12}$$

where **R** is any orthogonal matrix (called here rotation matrix) for which inversion is equivalent to transposition that is $\mathbf{R}^{-1} = \mathbf{R}'$, Eq. (2.23). The new matrices $\tilde{\mathbf{T}}$ and $\tilde{\mathbf{P}}'$ preserve all properties of the original matrices:[6]

$$\begin{aligned} \tilde{\mathbf{T}}'\tilde{\mathbf{T}} &= \mathbf{T}'\mathbf{T} = \mathrm{diag}(\lambda_i) \\ \tilde{\mathbf{P}}'\tilde{\mathbf{P}} &= \mathbf{P}'\mathbf{P} = \mathbf{I} \end{aligned} \tag{3.13}$$

This property of the PCA is called **rotational ambiguity**. One cannot obtain pure spectra from the PCA analysis because there are infinite number of scores and loadings which can be obtained and which reproduce the original matrix **X** (experimental data), see Fig. 3.3, but actual orientation (rotation) of scores and loadings is not defined. Although scores and plots reflect the underlying chemistry (concentrations, spectra) they are not directly related to these parameters.

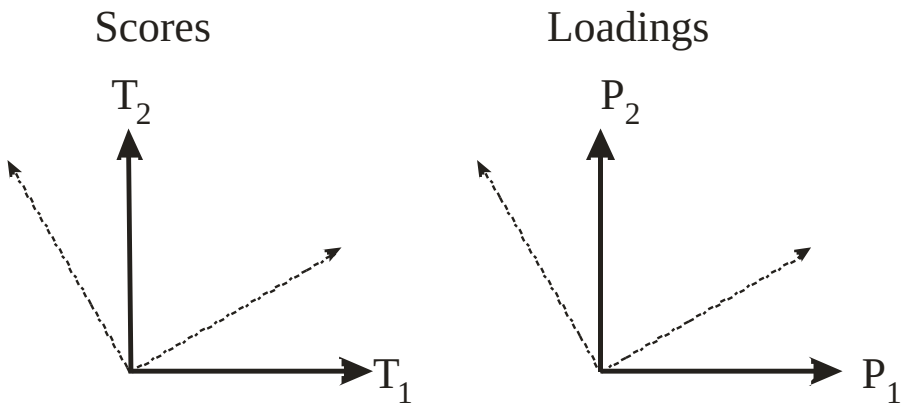


Fig. 3.3. Illustration of rotational ambiguity of scores and loadings in two-way (2D, bilinear) PCA. After rotation both scores and loadings stay orthogonal and reproduce the original matrix **X**.

Exercise 3.1.

Using data in Ex3-1[3] (file Xdata.m, it is also included in the Excel file Ex3-1.xlsx) containing 30 spectra measured at 28 wavelengths (λ), determine number of the principal component influencing the data. These data correspond to the spectra taken during elution in chromatography.

These spectra contained in **X**(30×28) are displayed in Fig. 3.4.

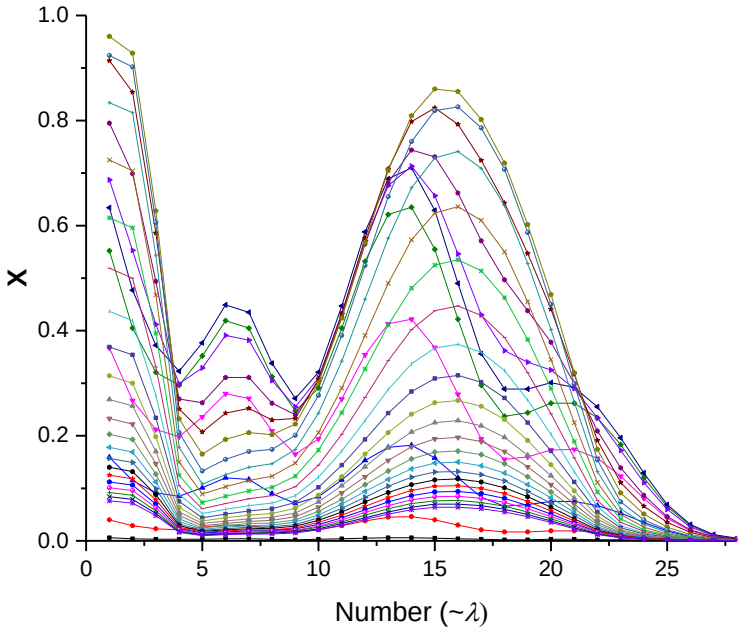


Fig. 3.4. 30 spectra in Exercise 3.1.

The Principal Component Analysis is carried out using program PCAtest.m using Matlab; it is located in folder Ex3-1 and all the PCA related programs are also in the folder PCA. The results are in file Ex3-1.xlsx. In the program, assuming maximal matrix rank of 4 (enter value of maxrank in PCAtest.m) and using raw data, the values of λ were calculated and are displayed in Table 3.1.

Table 3.1. Sizes i.e. eigenvalues of four first principal components, PC, for Exercise 3.1 using raw data.

PC	λ_i	%	Cumulative %
1	59.2056	97.05845195%	97.05845195%
2	1.7925	2.93857170%	99.99702366%
3	0.0018	0.00295404%	99.99997770%
4	1.4×10^{-5}	0.00002230%	100.0000000%
	sum		
	60.9999		

The results indicate that the first PC explains to 97.06% of the total variation and the second for 2.94%. Cumulative % is simply the sum of all lower PCs. Using simple statistics one can reject principal components which contribute less than, e.g. 5% (or 1%). This would suggest that only one PC can describe the experimental spectra. However, in this case such rejection would be incorrect as the original data were not centered prior to PCA and the reason why only one PC was found might be due to its greater size in the experimental data. Centering the data (change preoption to 2) leads to a different result, Table 3.2.

Table 3.2. Sizes of first four principal components, PCs for Exercise 3.1 using centered data.

PC	λ_i	%	Cumulative %
1	22.60225	92.6572%	92.657%
2	1.790112	7.3385%	99.996%
3	0.001037	0.0043%	100.000%
4	0.000013	0.0001%	100.000%
	sum		
	24.39341		

For the centered data the first PC corresponds to 92.7% and the second for 7.3% of the total variation, therefore, both components are important. Centering data reduced the total sum of eigenvalues from 61.9, Table 3.1 to 24.4, Table 3.2. Other components are negligible as they contribute only 0.0044%. The matrix has an effective rank of 2. Therefore, there are only **two PCs** in the data presented in Fig. 3.4, which corresponds to two important concentrations. There is no general rule whether the raw of centered data should be used, the choice depends on the nature of the experimental data and on the one's experience.[3] However, usually data are centered.

If there is a meaningful **sequential order** in data sets, e.g. in chromatography or in the presence of chemical reactions, plots of scores **T** as functions of data number (here proportional to time) might be presented, they are shown in Fig. 3.5 for raw and centered data.

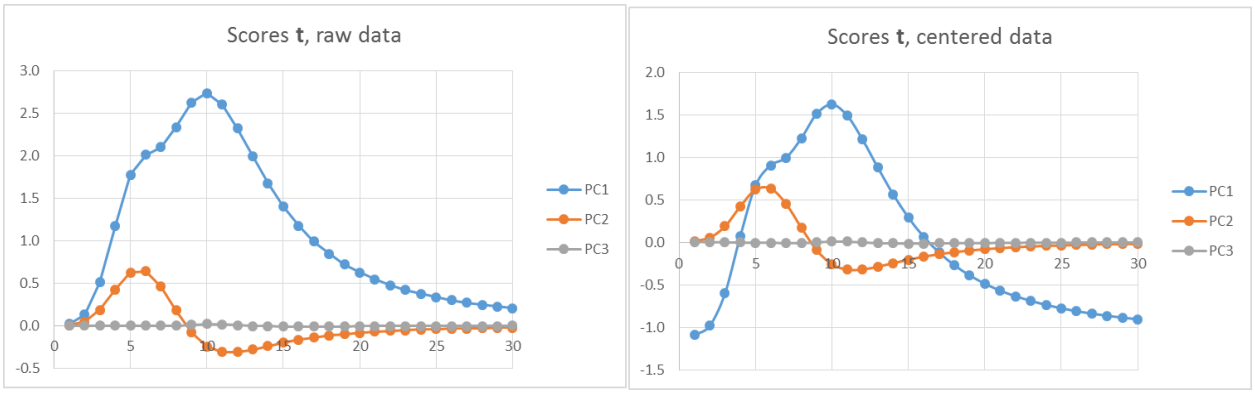


Fig. 3.5. Plots of scores **T** for three first principal components PC1 ($\mathbf{t}_1$), PC2 ($\mathbf{t}_2$), and PC3 ($\mathbf{t}_3$) versus data (spectrum) number, proportional to the elution time.

In this case the first principal component relates to the magnitude of the signal while the second PC relates to the difference between the two components in the mixture, being positive for the fastest eluting compound and negative for the slowest compound. The third PC is negligible. It is clear that these plots are different from the concentrations or elution profiles as scores are abstract matrix components.

The plot of loadings is shown in Fig. 3.6.

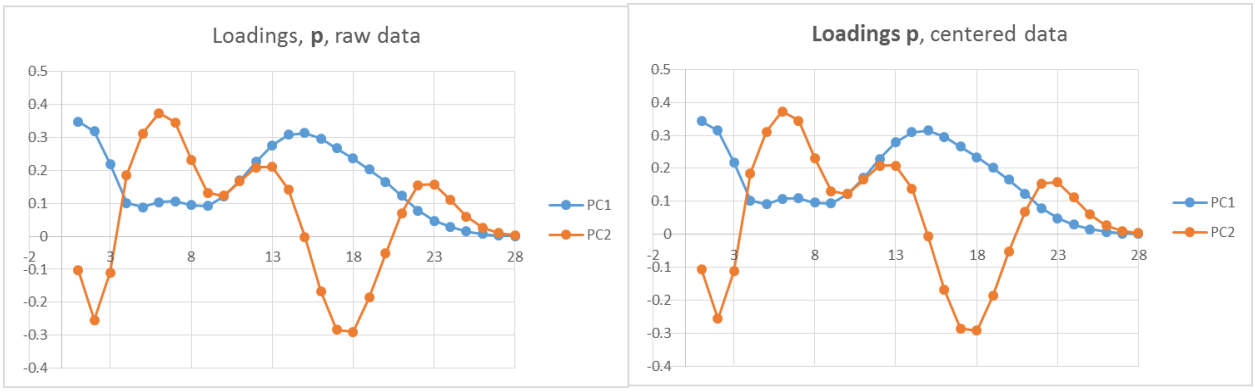


Fig. 3.6. Plot of loadings **P** for the two principal components PC1 ($\mathbf{p}_1$) and PC2 ($\mathbf{p}_2$) versus data (spectrum) number, proportional to the elution time for data in Exercise 3.1.

Spectra of the individual components: faster eluting A and slower eluting B, are presented in Fig. 3.7. They were be obtained from the measurements of pure components.

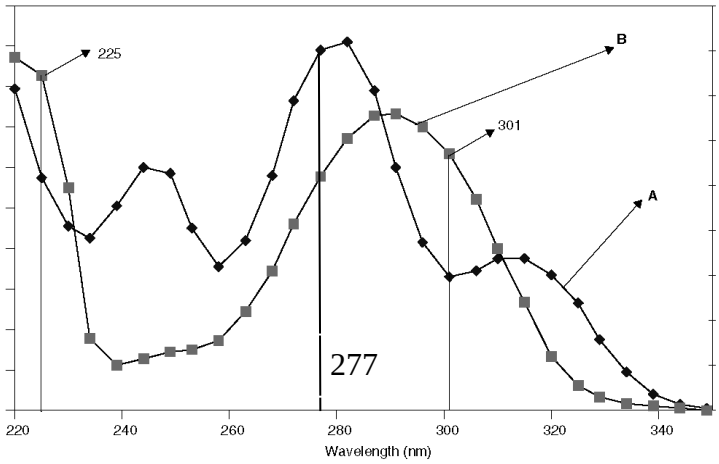


Fig. 3.7. Spectra of the faster A and slower B eluting compound for the chromatographic analysis in Exercise 3.1.[3]

It is evident that around λ = 277 nm (point No 13) both compounds absorb. The plot at this wavelength versus time shows the elution profile in absorbance (not concentration) units. It is shown in Fig. 3.8.

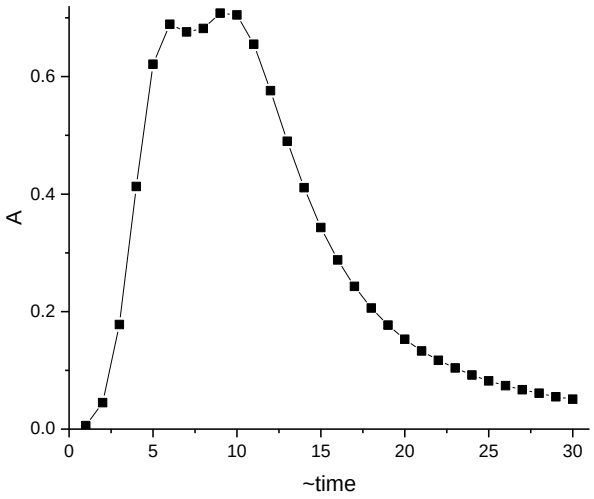


Fig. 3.8. Absorbance elution profile for λ = 277 nm (point No 13) versus time (from Fig. 3.4) for two overlapping peaks.

In general, in the case where there is a **sequential order** in the data **scores relate to the elution profiles** (**concentrations**) while **loadings relate to the components spectra**.[3]

Scores, **T**, **plots** are very often used in the data analysis. In such plots scores for PC2 are plotted versus PC1. An example of such a plot is presented in Fig. 3.9.

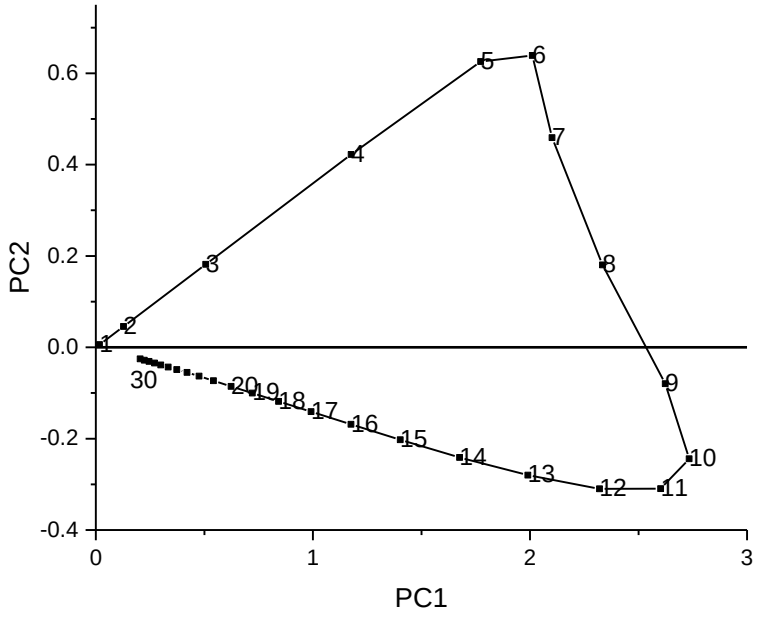


Fig. 3.9. Scores (**T**) plot of $\mathbf{t}_2$ (PC2) versus $\mathbf{t}_1$ (PC1) for row data in Exercise 3.1.

The scores plot has some interesting characteristics:

- linear zones represent regions where pure components prevail; in our case for points 1-5 and 13-30,
- curved portions represent regions where two compounds exist: points 6 to 10,
- values close to the origin represent zones of low intensity (low concentration),
- number of bends can provide information about the number of compounds in a complex mixture; there are two bends at points 6 and 10 corresponding to two compounds.

The **loadings plot**, presents loadings of the second PC2 ($\mathbf{p}_1$) versus those of the first PC1 ($\mathbf{p}_1$). They are displayed, for raw data, in Fig. 3.10.

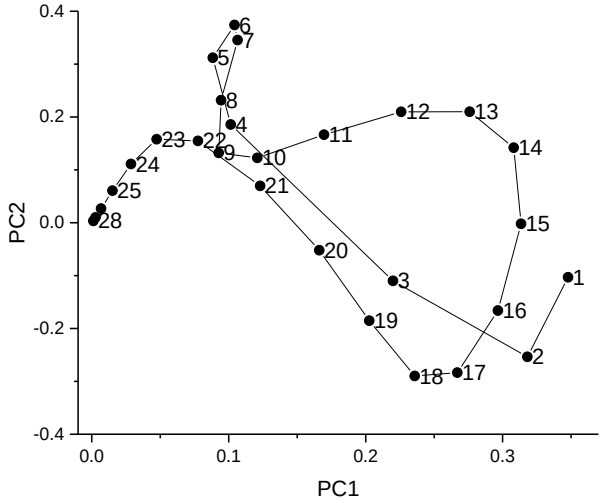


Fig. 3.10. Loadings (**P**) plot of PC2 ($\mathbf{p}_2$) versus PC1 ($\mathbf{p}_1$) for the raw data in Exercise 3.1.

The loadings plot can be explained by comparison with the spectra of pure compounds. The upper part of the plot, points 4-14 (234-282 nm) and 22-30 (320-349 nm) correspond to the zone where absorbance of A is more important while the bottom half of the plot, points 2-3 (225-230 nm) and 16-20 (291-310 nm) correspond to B. The peak, point 6 (244 nm, see Fig. 3.7), for the component A is clearly visible. Loadings provide information about which wavelengths are associated with which compound.[3]

Using obtained scores and loadings for the matrix rank found from the above analysis (R = 2) the calculated values of $\hat{\mathbf{X}}$ can be obtained using Eq. (3.6) and are shown in Fig. 3.11. It should be noticed that large amount of data in **X** (840 data points) was explained by only two principal components that is by much smaller matrices **T**(60) and **P**(56) containing 116 data points. It is clear that large reduction of the experimental data was obtained.

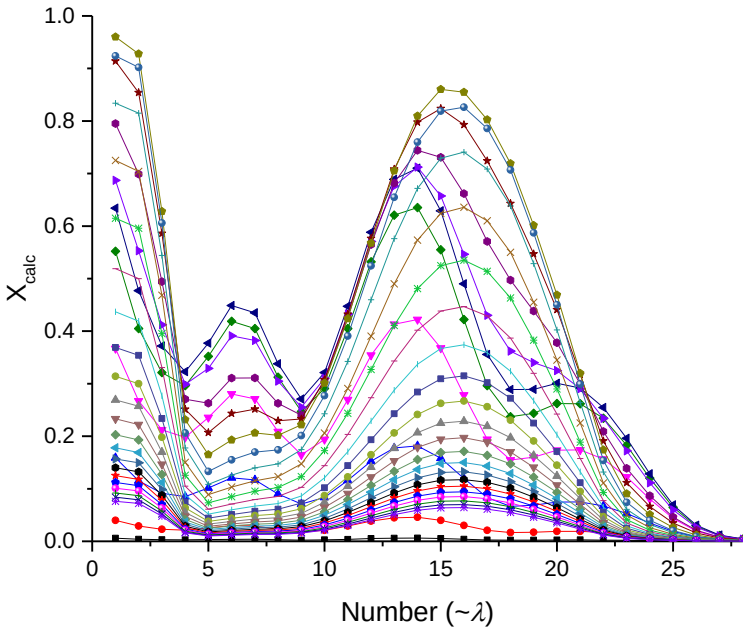


Fig. 3.11. Values of $\hat{\mathbf{X}}$ calculated from Eq. (3.6) for two PCs (matrix rank of 2) using centered data.

Comparison of the experimental **X** and calculated (model) $\hat{\mathbf{X}}$ allows for the calculation of the **residual sum of squares RSS_R** calculated using *R* principal components:

$$\mathrm{RSS_R} = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{i,j} - \hat{x}_{i,j}\right)^2 = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(e_{i,j}\right)^2 \qquad (3.14)$$

This parameter is an analog of the residual sum of squares in the univariate (classical) regression. The root mean square RMS_R for *R* PCs is calculated dividing by the number of the degrees of freedom. Because there are *I*J* experimental points which in the above case is 28*30=840 this value is usually not corrected by the loss of the degree of freedom due to number of PCs:

$$\mathrm{RMS_R} = \sqrt{\frac{\mathrm{RSS_R}}{I * J}} \qquad (3.15)$$

For the data in Exercise 3.1 the values of RSS_r are calculated for *r* principal components (here from 1 to 4) using PCAcross.m and the results obtained are shown in Table 3.3.

Table 3.3. Dependence of RSS_r on the number of principal components used, *r* (centered data).

Number of PCs, *r*	RSS_r
1	1.7944
2	0.00187
3	6.86×10^{-5}
4	5.50×10^{-5}

It is clearly seen that residual sum of squares decreases with the increase of the number of PCs used. However, this does not indicate that we should use here three PCs as it has been shown above that two PCs explain more that 99.99% of the variance. The third (and further) PCs are simply modeling the noise in the data and do not contain information about the concentrations.

3.2 Preprocessing of data

Data can be analyzed directly, that is **raw data** can be used in the calculations. Such an analysis might be carried out in spectroscopy where deviation above baseline is studied. However, usually we are interested in the deviation from the mean. In such a case the data are mean **centered** that is mean of each **column** of **X** matrix is calculated and this value is subtracted from each column element:

$$x_{i,j}^{cent} = x_{i,j} - \bar{x}_j \tag{3.16}$$

where $\bar{x}_j$ is the mean of column j

$$\bar{x}_j = \frac{1}{I}\sum_{i=1}^{I} x_{i,j} \tag{3.17}$$

Most of all data analysis is carried out as mean centered.

Another method of data preprocessing is **standardization**. The centered data (in columns) are divided by their standard deviation:

$$x_{i,j}^{std} = \frac{x_{i,j} - \bar{x}_j}{\left[\dfrac{\sum_{i=1}^{I}\left(x_{i,j} - \hat{x}_j\right)^2}{I}\right]} \tag{3.18}$$

where the sum of squares was divided by I that is the population standard deviation (number of columna) was used. Standardization might be important in some cases. For example, the sample might contain larger concentration of some compounds but with their variation not very significant. We might be interested in concentration of some minor compounds. If standardization is not performed, PCA will be dominated by the compounds having the highest concentration.[3]

Another type of preprocessing is a row scaling (or normalization):

$$x_{i,j}^{rs} = \frac{x_{i,j}}{\sum_{j=1}^{J} x_{i,j}} \tag{3.19}$$

Scaling the rows is useful if the absolute concentrations of samples cannot easily be controlled. An example might be biological extracts: the precise amount of material might vary unpredictably, but the **relative proportions of each chemical can be measured**.[3]

Plots for one set of data (No 14, $\mathbf{x}_{i,14}$) from Fig. 3.4 as raw, centered, and standardized are shown in Fig. 3.12.

Plots of all the data with different preprocessing are displayed in Fig. 3.13-3.15. First, Fig. 3.13 shows the raw spectra and their mean value (thick blue line). The mean value is subtracted

from each spectrum and the obtained centered data are displayed in Fig. 3.14. Finally, the standardized spectra are shown in Fig. 3.15. Centered and standardized spectra do not resemble the original spectra but they contain all the pertinent information necessary for the PCA.

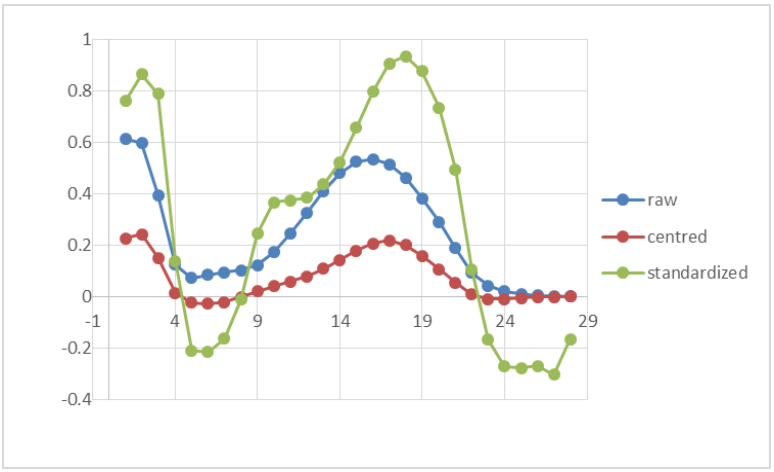


Fig. 3.12. Plot of the data series No. 14, $\mathbf{x}_{i,14}$ from Fig. 3.4 in three formats: raw, column centered, and standardized.

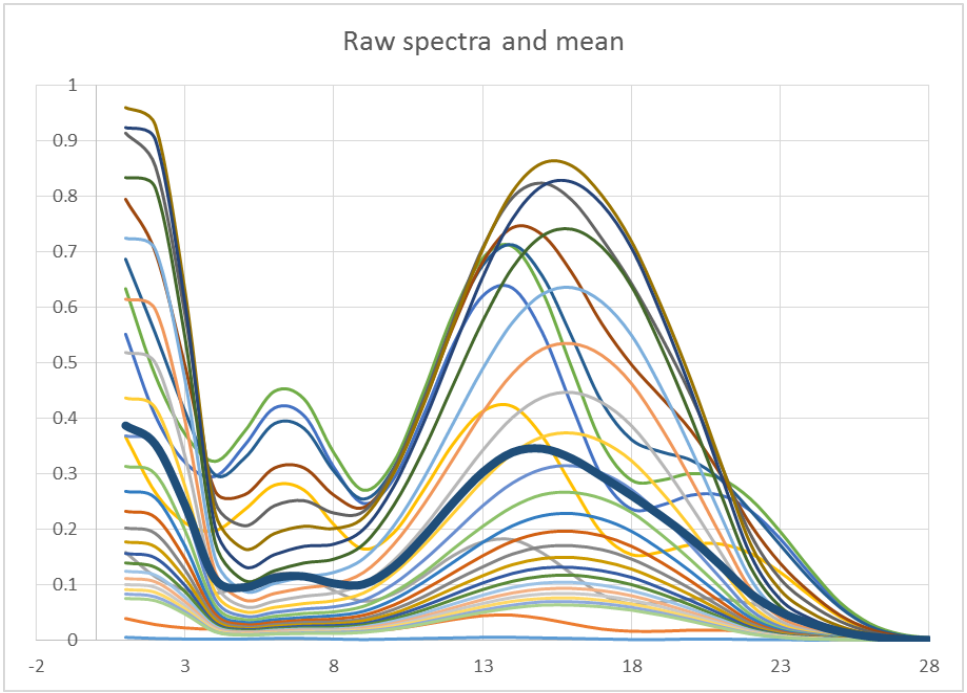


Fig. 3.13. Plots of the raw spectra, the mean value is shown as thick blue line.

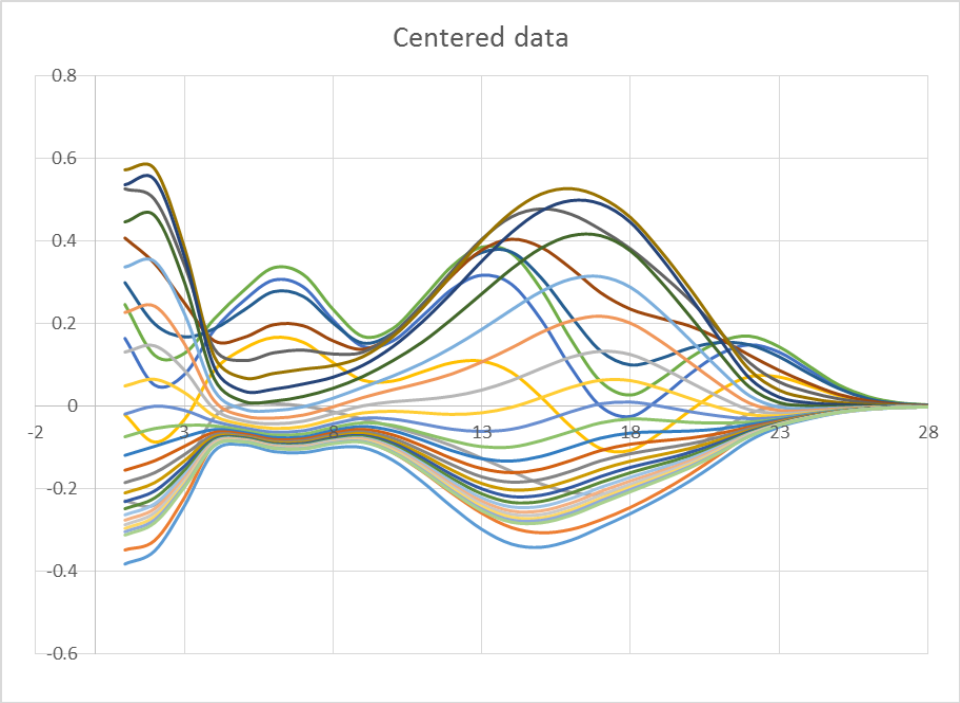


Fig. 3.14. Centered spectra from Fig. 3.13.

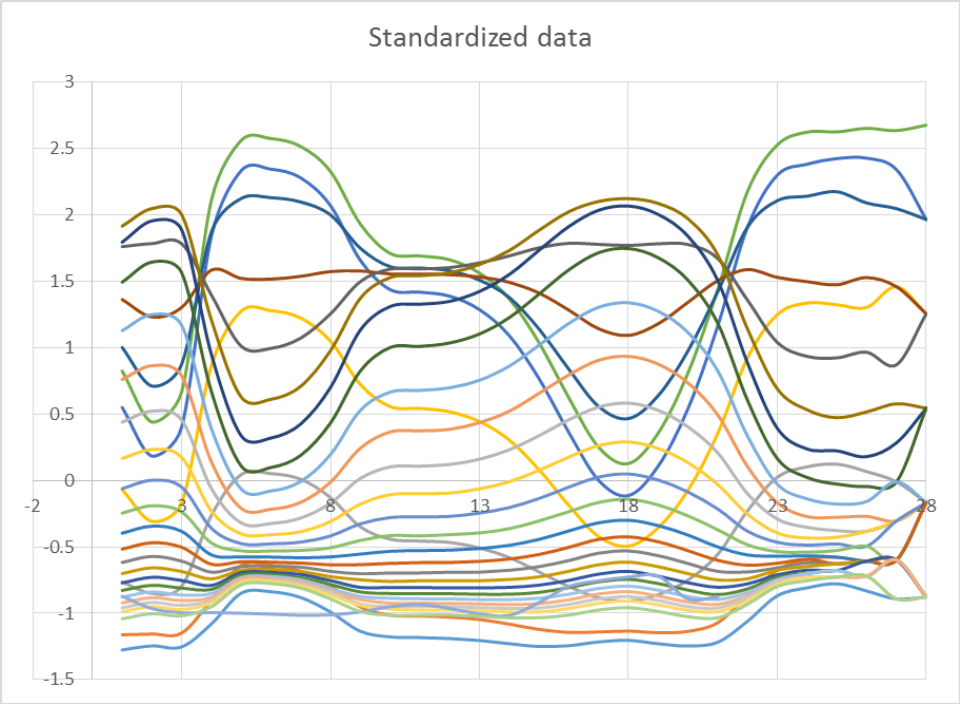


Fig. 3.15. Standardized spectra from Fig. 3.13.

Data preprocessing influences values of scores and plots and their plots. This will be shown in the following exercise.

Exercise 3.2.

Data **X** presented below in Table 3.4 (Ex3-2.xlsx, Xdata.m in folder Ex3-2) contain 10 rows and eight columns and represent a portion of the chromatographic UV/VIS elution profile.[3] Determine number of PCs and compare the scores and loadings plots for different data preprocessing.

Table 3.4. UV/VIS spectra obtained during elution in HPLC method.[3]

	A	B	C	D	E	F	G	H
1	0.318	0.413	0.335	0.196	0.161	0.237	0.29	0.226
2	0.527	0.689	0.569	0.346	0.283	0.400	0.485	0.379
3	0.718	0.951	0.811	0.521	0.426	0.566	0.671	0.526
4	0.805	1.091	0.982	0.687	0.559	0.676	0.775	0.611
5	0.747	1.054	1.03	0.804	0.652	0.695	0.756	0.601
6	0.579	0.871	0.954	0.841	0.680	0.627	0.633	0.511
7	0.380	0.628	0.789	0.782	0.631	0.505	0.465	0.383
8	0.214	0.402	0.583	0.635	0.510	0.363	0.305	0.256
9	0.106	0.230	0.378	0.440	0.354	0.231	0.178	0.153
10	0.047	0.117	0.212	0.257	0.206	0.128	0.092	0.080

The spectra in Table 3.4 are displayed in Fig. 3.16.

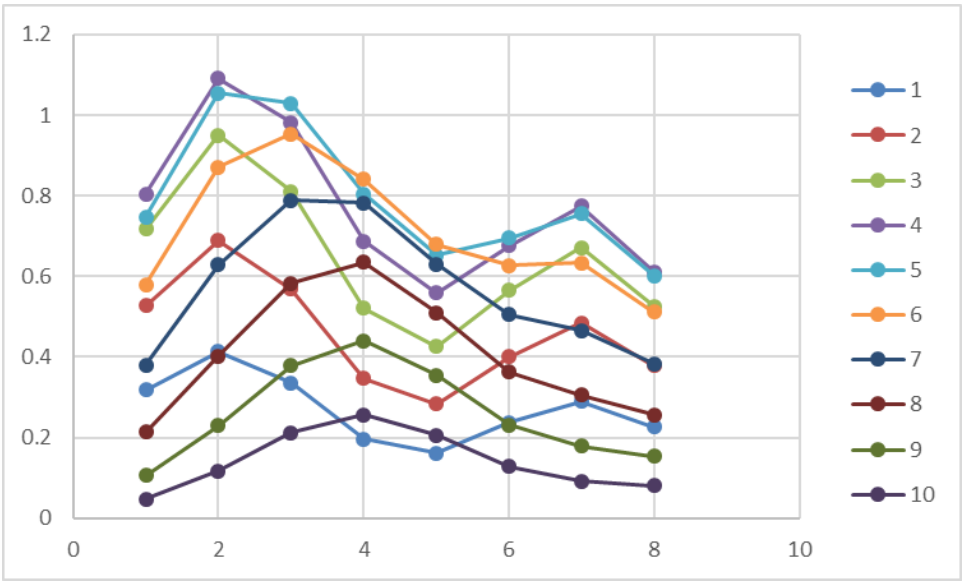


Fig. 3.16. Raw spectra from chromatographic elution, Table 3.4.

An example of elution profile for the first column in Table 3.4 (corresponding to one wavelength, the first data column $\mathbf{x}_1$) is displayed in Fig. 3.17.

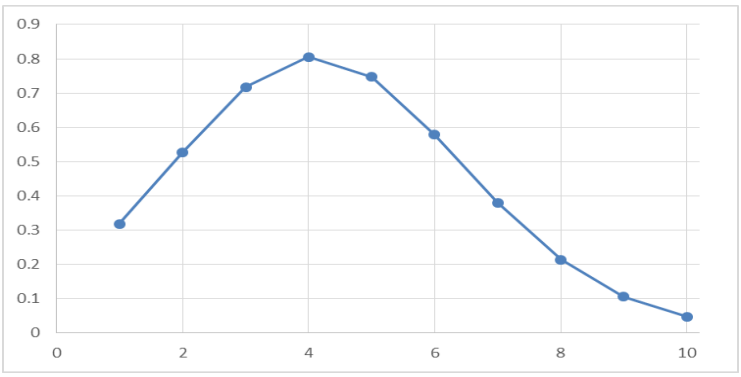


Fig. 3.17. Elution profile i.e. absorbance versus time at one wavelength corresponding to the first data column $\mathbf{x}_1$ in Table 3.4.

Analysis of the eigenvalues (PCAtest.m) is shown in Table 3.5.

Table 3.5. PCA on raw, centered and standardized data from Table 3.4.

raw data		centered		standardized	
λ_i	%	λ_i	%	λ_i	%
25.05898	**97.807%**	4.074459	**89.68%**	70.4548	**88.07%**
0.561896	**2.193%**	0.469006	**10.32%**	9.54513	**11.93%**
2.11×10^{-6}	0.000%	2.01×10^{-6}	0.000%	4.6×10^{-5}	0.000%
sum		sum		sum	
25.62088		4.543468		79.9999	

Analysis of the raw data suggests that there is only one principal component but the analysis using data centered or standardized strongly indicate that there are two PCs and the third PC is completely negligible. The plots of scores **T** for three first PCs and raw and centered data are shown in Fig. 3.18.

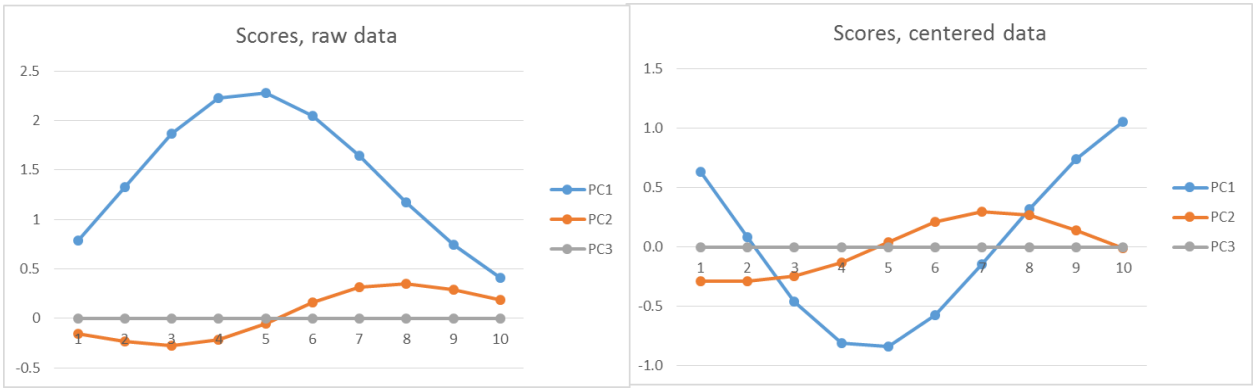


Fig. 3.18. Plots of three first scores, $\mathbf{t}_i$, for the raw and centered data in Table 3.4.

The scores and loading plots for the two first principal components for the raw data are shown in Fig. 3.19 and 3.20.

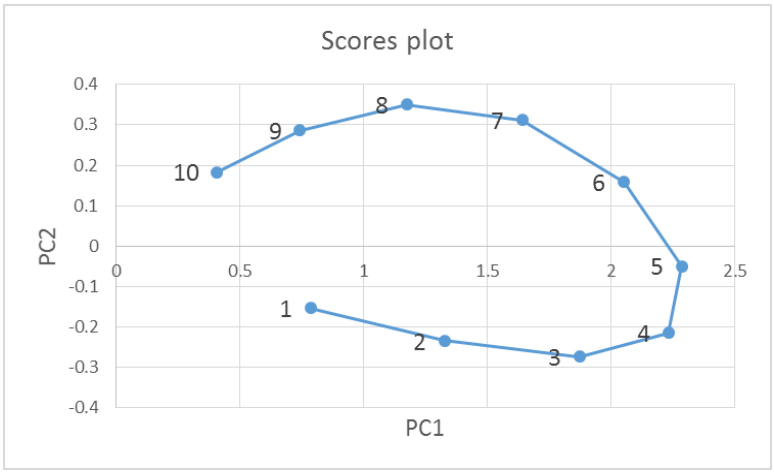


Fig. 3.19. Scores plot of $\mathbf{t}_2$ (PC2) versus $\mathbf{t}_1$ (PC1) for the raw data in Table 3.4.

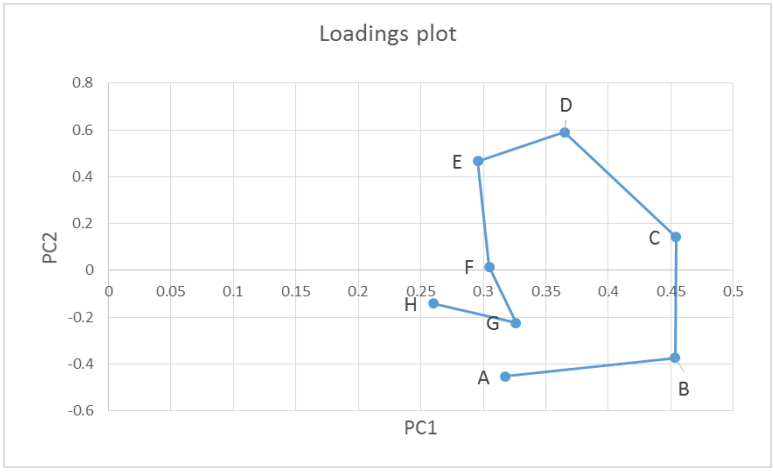


Fig. 3.20. Loadings plots $\mathbf{p}_2$ (PC2) versus $\mathbf{p}_1$ (PC1) for the raw data in Table 3.4.

The above plots suggest the presence of two components; in Fig. 3.20 points A, B, and G correspond mainly to one component and points D and E to another.

Let us look now at the similar plots for the centered data, Fig. 3.21- 3.22.

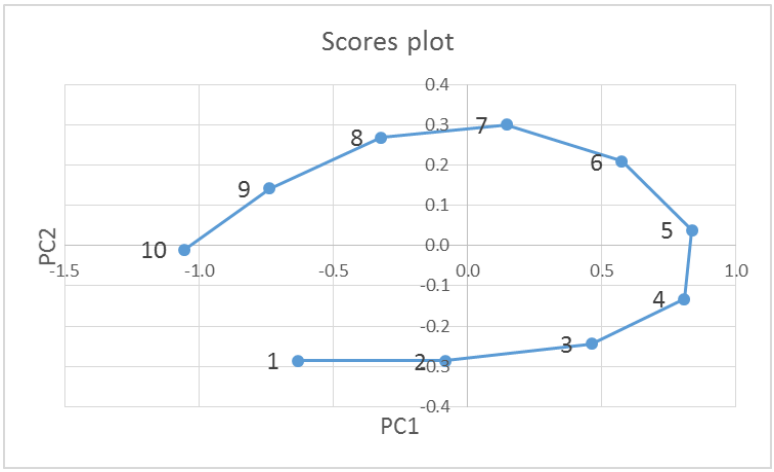


Fig. 3.21. Scores plot for the centered data from Table 3.4.

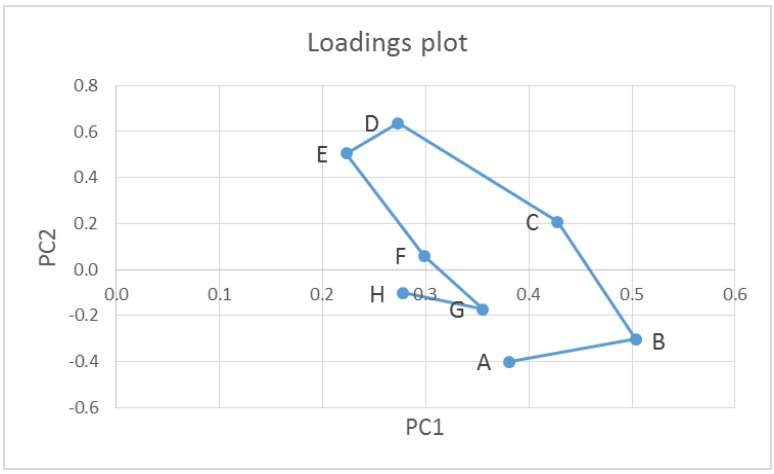


Fig. 3.22. Loadings plot for the centered data from Table 3.4.

In this case the scores, Fig. 3.21, are centered on the origin because the data are centered on mean and the sum of each column of **X** is zero. Nevertheless, the general shape is similar. However, the shape of loading,
Fig. 3.22, is changed. Similar plots for the standardized data are shown in Fig. 3.23 and 3.24.

Fig. 3.23. Scores plot for the standardized data from Table 3.4.

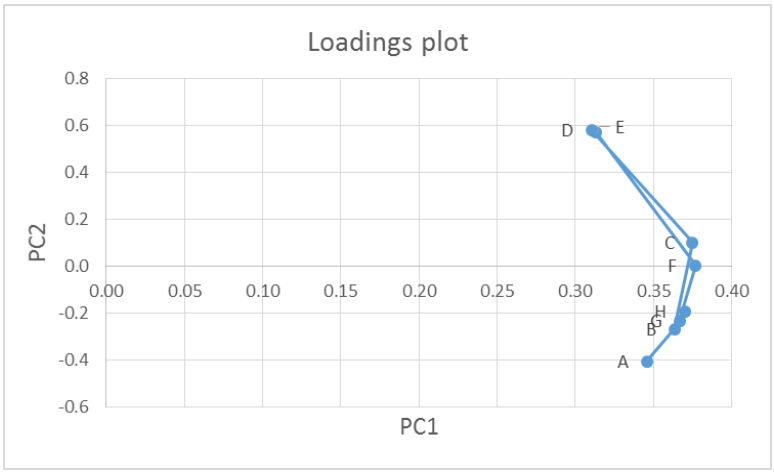


Fig. 3.24. Loadings plot for the standardized data from Table 3.4

The general shape of the scores plot for standardized data is similar to that of the centered data. However, there is a complete change of the loadings plot, it forms a part of a semicircle. Because standardization puts the variables on the same scale and the variables of low magnitude have the same importance as those of large magnitude. For two significant principal components the points lie on a semicircle and for three PCs they lie on a sphere. The results in Fig. 3.24 confirm visually that there are two important PCs. This figure confirms as well that points A, B, and G correspond mainly to one component and points D and E to another.

Standardization is recommended when different components are present in very different concentrations, change very little, or when different measurements are carried out on a different scales. In the standard measurements mean-centered data are used although in some cases raw or standardized data are analyzed.

3.3 Cross-validation

The significance of each PC can be tested using cross-validation. It is based on auto-prediction of the experimental data and these data are used **to predict a sample which was removed from the data set**. First, the row number 1 is left out from the data matrix **X**, the PCA analysis is carried out on matrix **X** containing rows 2, 3, …, *I*, that is $I - 1$ rows. Next data for row 1 are predicted. Then the row number 2 is left out from the original matrix **X** which contains containing rows 1, 3, …, *I*, the values for the row 2 are predicted and the sum of squares computed. This procedure is continued until the last row *I*.

The PCA, after omitting one row in data file, produces new scores **T** and loadings **P** for each row *i*. The vector of predicted scores for the left out row *i*, $\hat{\mathbf{t}}_i$, is calculated using standard multiple regression equation from Eq. (3.3) and (3.6), $\mathbf{x}_i = \hat{\mathbf{t}}_i \mathbf{P}'$:

$$\hat{\mathbf{t}}_i = \mathbf{x}_i \mathbf{P} \left(\mathbf{P}'\mathbf{P} \right)^{-1} = \mathbf{x}_i \mathbf{P} \tag{3.20}$$

This equation is simplified because matrix **P** is orthonormal and:

$$\left(\mathbf{P}'\mathbf{P} \right)^{-1} = \mathbf{I} \tag{3.21}$$

where **I** is the unitary matrix and $\hat{\mathbf{t}}_i$ is the predicted row of scores. Next, the vector ${}^{r,cv}\hat{\mathbf{x}}_i$ for matrix **X** is calculated for sample *i* and for *r* PCs:

$$ {}^{r,cv}\hat{\mathbf{x}}_i = {}^{r}\hat{\mathbf{t}}_i \; {}^{r}\mathbf{P}' \tag{3.22} $$

where *r* is the number of PCs used in the model, and vector ${}^{r}\mathbf{t}_i(1\times r)$ is calculated from Eq. (3.20).

Finally, **Predicted Residual Error Sum of Squares**, **PRESS**, is calculated:

$$ \text{PRESS}_r = \sum_{i=1}^{I}\sum_{j=1}^{J}\left({}^{r,cv}\hat{x}_{i,j} - x_{i,j}\right)^2 \tag{3.23} $$

where ${}^{r,cv}\hat{\mathbf{x}}_{i,j}$ are the values of **x** predicted after elimination of row *i* from the original data set, calculated for *r* PCs.

PRESS_r is a sum of squared differences between values ${}^{r,cv}\hat{x}_{i,j}$ predicted after sequential elimination of one data row from analysis and the observed experimental values $x_{i,j}$. This parameter decreases down to the correct number of PCs and then stays almost constant or increases. The plot of PRESS as a function of the number of PCs for the data in Exercise 3.1 is shown in Fig. 3.25.

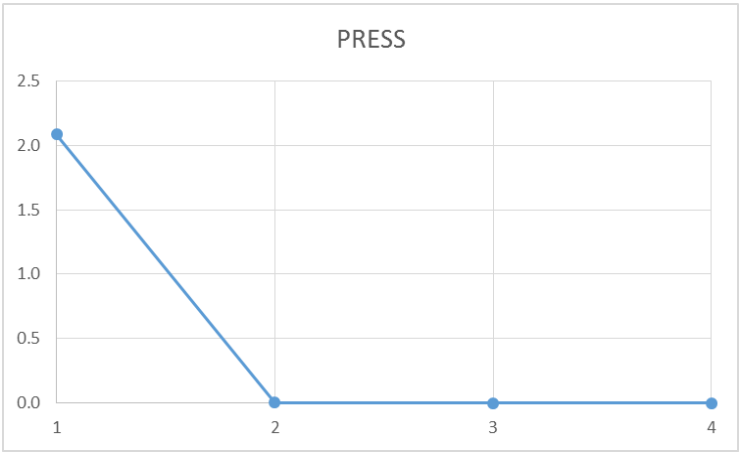


Fig. 3.25. Plot of the parameter PRESS as a function of the number of principal components for the centered data in Exercise 3.1.

It is evident that the parameter PRESS decreases up to two PCs and then changes very little which confirms that there are two PCs. Similar plot is obtained for RSS.

It is important that RSS, Eq. (3.14), and PRESS are presented in the same scale, preferably as raw data (although calculations can be are carried out on centered or standardized data). The values of RRS and PRESS can be easily calculated using program PCAcross.m.

To analyze the obtained results one can compare PRESS_r with $\text{RSS}_{r\text{-}1}$. If:

$$ \frac{\text{PRESS}_r}{\text{RSS}_{r-1}} > 1 \tag{3.24} $$

this means that an extra PC_r is modeling only noise and should not be retained, that is only *r* - 1 components should be kept.

Another method is to calculate ratio $PRESS_r/PRESS_{r-1}$. If:

$$\frac{PRESS_r}{PRESS_{r-1}} > 1 \tag{3.25}$$

only $r - 1$ PCs should be used. PRESS often starts to increase after the optimum number of components have been reached. It should be added that these tests may not always work well. The plot of PRESS and RSS for the data in Exercise 3.2 is shown in Fig. 3.26.

.

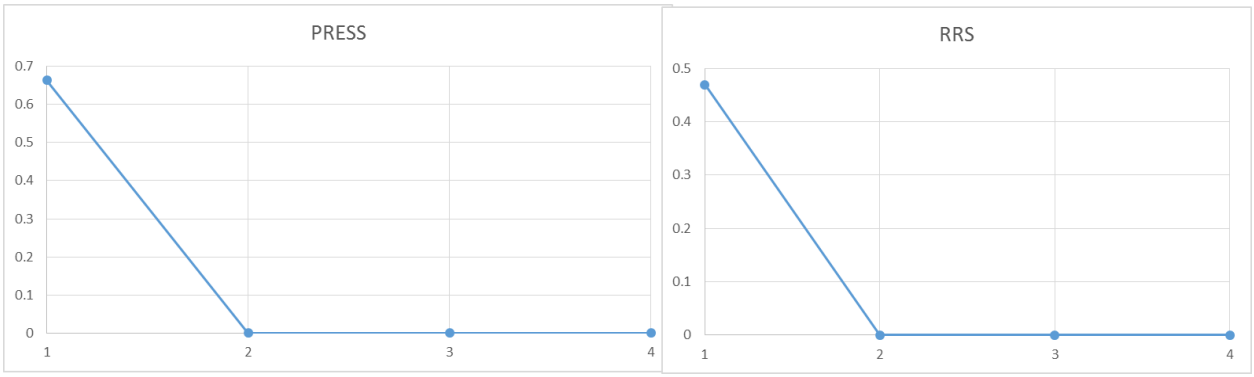


Fig. 3.26. Plot of PRESS and RSS as functions of the number of PCs for the centered data in Exercise 3.2.

Logarithmic scale is often used in plots to emphasize the importance of small variations, see Fig. 3.27, where the same data as in Fig. 3.26 are plotted.

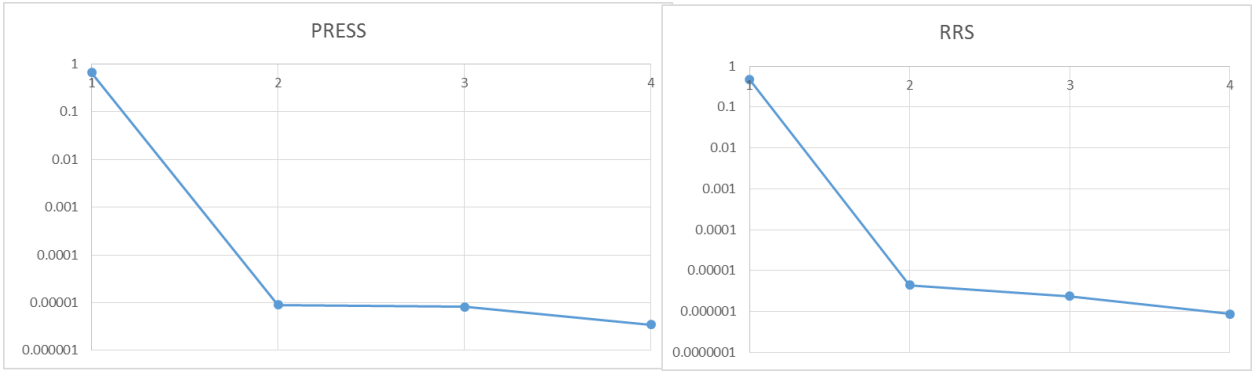


Fig. 3.27. Logarithmic plots of PRESS and RSS as functions of the number of PCs for the centered data in Exercise 3.2.

From Fig. 3.26 -3.27 it is obvious that the parameters PRESS and RSS decreases to PC=2 and then change very little which confirms the presence of two PCs. This statement is confirmed by the results in Table 3.6. These results indicate that after $r = 2$ $PRESS_r/RSS_{r-1}$ is larger than one and $PRESS_r/PRESS_{r-1}$ increases from 1.35×10^{-5} to 0.901, a value close to one. Therefore, there are only two important PCs in the data.

Table 3.6. Analysis of RSS and PRESS for the centered data in Table 3.4 in Exercise 3.2.

r	$PRESS_r$	RSS_r	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	0.66265	0.46901		
2	9.0×10^{-6}	4.4×10^{-6}	1.91128E-05	1.35276×10^{-5}
3	8.2×10^{-6}	2.4×10^{-6}	1.877288218	0.919088287
4	3.4×10^{-6}	8.7×10^{-7}	1.448287021	0.418730875

Exercise 3.3.
Determine number of PCs in the following data, file Xdata.m, **X**(10×8), and in Ex3-3.xlsx in folder Ex3-3.[3]

Table 3.7. Table of data to analysis containing 10 rows and 8 columns.[3]

	A	B	C	D	E	F	G	H
1	89.821	59.760	68.502	48.099	56.296	95.478	71.116	95.701
2	97.599	88.842	95.203	71.796	97.88	113.122	72.172	92.310
3	91.043	79.551	104.336	55.900	107.807	91.2290	60.906	97.735
4	30.015	22.517	60.330	21.886	53.049	23.127	12.067	37.204
5	37.438	38.294	50.967	29.938	60.807	31.974	17.472	35.718
6	83.442	48.037	59.176	47.027	43.554	84.609	67.567	81.807
7	71.200	47.990	86.850	35.600	86.857	57.643	38.631	67.779
8	37.969	15.468	33.195	12.294	32.042	25.887	27.050	37.399
9	34.604	68.132	63.888	48.687	86.538	63.560	35.904	40.778
10	74.856	36.043	61.235	37.381	53.98	64.714	48.673	73.166

The PCA in Matlab using program PCAtest.m gives the values of PCs (eigenvalues) displayed in Table 3.8.

Table 3.8. Results of PCA analysis on data in Table 3.7 using raw and centered data.

	Raw data			Centered data		
r	λ_r	%	Cumulative %	λ_r	%	Cumulative %
1	316522.12	96.9045%	96.905%	34552.51	79.7238%	79.724%
2	7324.62	2.2425%	99.147%	6625.63	15.2875%	95.011%
3	2408.63	0.7374%	99.884%	1826.65	4.2147%	99.226%
4	136.006	0.0416%	99.926%	135.54	0.3127%	99.539%
5	117.716	0.0360%	99.962%	117.72	0.2716%	99.810%
6	72.891	0.0223%	99.984%	55.45	0.1279%	99.938%
7	36.067	0.0110%	99.995%	18.10	0.0418%	99.980%
8	14.989	0.0046%	100.000%	8.66	0.0200%	100.000%

The analysis of raw data indicates existence of only one PC while that of centered data two or three components (for centered data the third component explains 4.2% of data close to the limit of typical value of 5%). Three components explain 99.22% of the total dependence.

As there is no order in the underlying concentrations (as in the case of chromatography) the scores plot does not show any order, Fig. 3.28, however the magnitude of scores decreases with the PC number, *r*, Fig. 3.29.

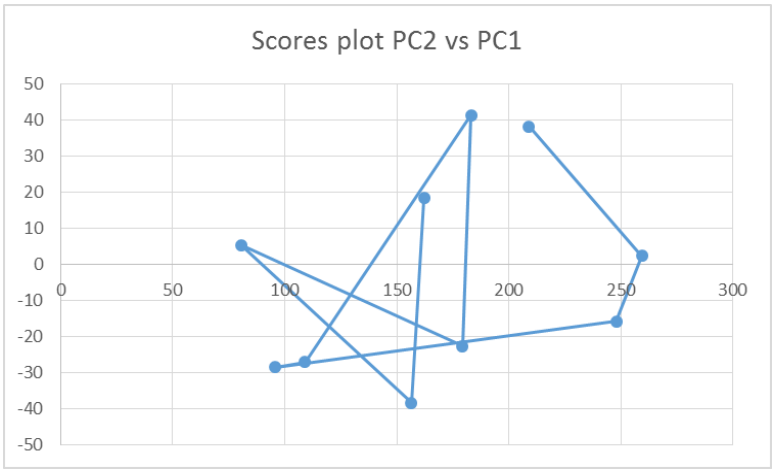


Fig. 3.28. Scores plot of $\mathbf{t}_2$ (PC2) vs. $\mathbf{t}_1$ (PC1) for raw data.

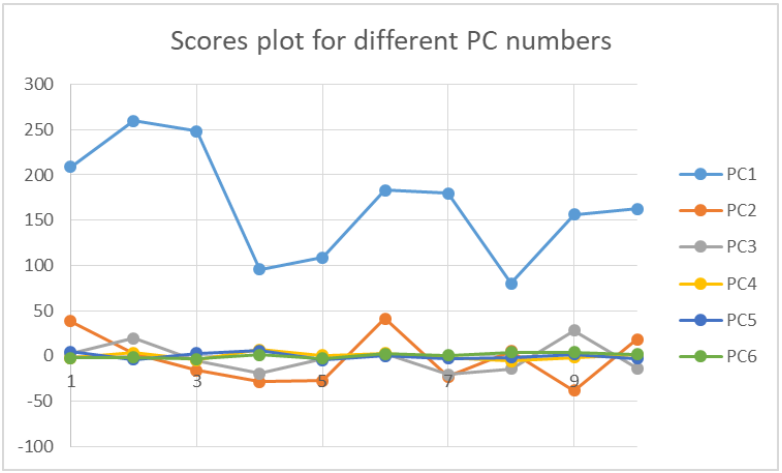


Fig. 3.29. Scores plots for different PC numbers, *r*, for raw data.

Next, further analysis using RSS and PRESS data was carried out using program PCAcross.m. The results are displayed below (for $r = 8$ values of zero are obtained as there are only 8 columns in **X** and the fit is perfect).

Table 3.9. Analysis of RSS and PRESS for the raw data in Table 3.7.

r	RSS	PRESS	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	10110.9141	12412.1		
2	2786.2979	4551.5	0.450	0.367
3	377.6687	830.9	0.298	0.183
4	241.6627	802.2	**2.124**	**0.965**
5	123.9470	683.7	2.829	0.852
6	51.0561	499.7	4.031	0.731
7	14.9890	336.2	6.586	0.673
8	0.0000	0.0	0.000	0.000

The test of the ratio of $PRESS_r/RSS_{r-1}$ shows that this value for $r = 4$ is 2.124 which is larger than 1. It suggests that three PCs should be conserved. However, the test $PRESS_r/PRESS_{r-1}$ is less conclusive although from $r = 4$ this ratio is relatively close to one.

Another part of the analysis is the plot logarithm of RSS and PRESS as functions of the number of principal components. Such a plot for the data studied is shown in Fig. 3.30.

It is clear that both parameters decrease quickly until $r = 3$ and then decrease more slowly. It should also be noticed that logarithmic plot increases importance of very small changes for $r > 3$.

Although the analysis of eigenvalues for the centered data is inconclusive suggesting between 2 and 3 PCs, test $PRESS_r/RSS_{r-1}$ and the logarithmic plots of RSS and PRESS suggest that there are **three PCs** which influence the experimental data.

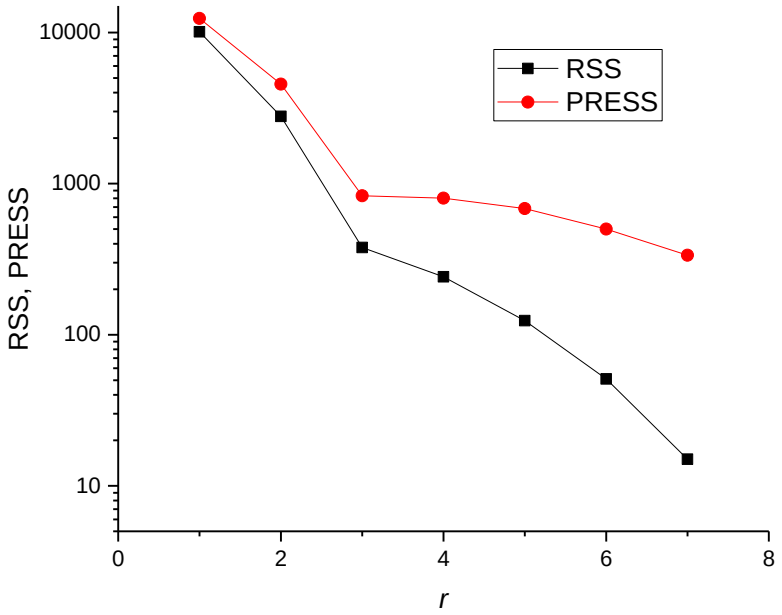


Fig. 3.30. Logarithmic plot of RRS and PRESS as functions of the number of PCs for centered data in Table 3.7.

Exercise 3.4.
Determine number of PCs in 21 spectra measured at 54 wavelengths in Fig. 3.31, file Xdata.m in folder Ex3-4 and file Ex3-4.xlsx.[6] They were registered during chemical reaction between species.

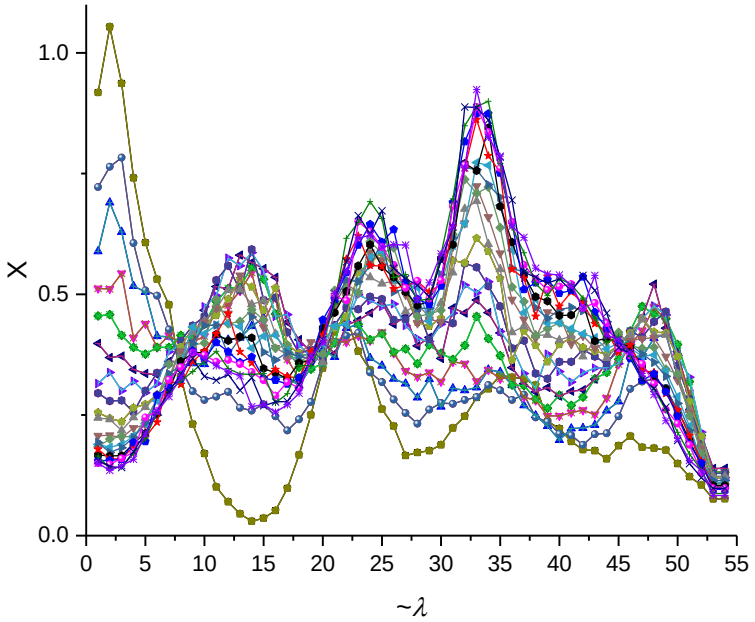


Fig. 3.31. Spectra of 21 mixtures of unknown number of compounds.

Table 3.10. PCA analysis of the raw and centered data in analysis of eigenvalues.

	Raw data			Centered data		
r	λ_r	%	Cumulative %	λ_r	%	Cumulative %
1	192.8	94.51%	94.51%	11.092	77.94%	77.94%
2	8.271	4.05%	98.57%	2.9725	20.89%	98.83%
3	2.808	1.38%	99.94%	0.0599	0.42%	99.25%
4	0.0593	0.03%	99.97%	0.0575	0.40%	99.66%
5	0.0573	0.03%	100.00%	0.491	0.34%	100.00%

The results presented in Table 3.10 show that using raw data the second PC contributes only 4% while using centered data its importance is 21%. This suggests existence of two PCs.

The scores' plots for different PCs are shown in Fig. 3.32. In this case they correspond to the gradual changes with time (spectrum number). Their magnitude for the first two PCs is much larger than for further PCs. The scores plot of $\mathbf{t}_2$ (PC2) versus $\mathbf{t}_1$ (PC1) is displayed in Fig. 3.33. The points are on a regular lines confirming presence of at least two PCs.

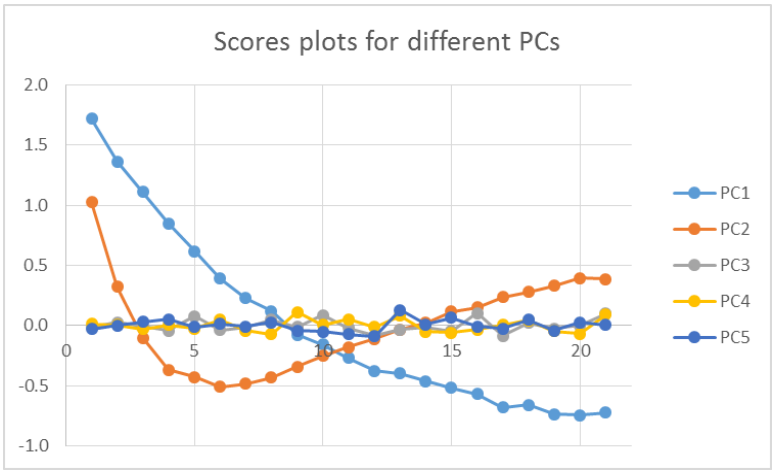


Fig. 3.32.Scores plots of $\mathbf{t}_r$ for different principal components 1 to 5 for the centered data.

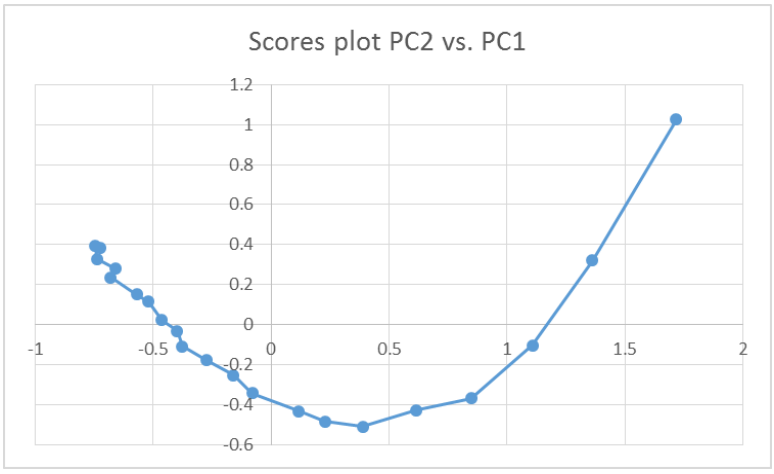


Fig. 3.33. Scores plot of $\mathbf{t}_2$ (PC2) versus $\mathbf{t}_1$ (PC1) for centered data in Exercise 3.4.

Further analysis of RRS and PRESS, is shown in Table 3.11 for raw data and in Table 3.12 for centered data. The plots of RSS and PRESS versus number of PCs are displayed in Fig. 3.34.

Table 3.11. Analysis of RRS and PRESS for raw data in Exercise 3.4.

r	RSS	PRESS	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	11.50	12.423		
2	3.230	4.698	0.409	0.378
3	0.422	0.576	0.178	0.123
4	0.363	0.573	**1.357**	**0.995**
5	0.305	0.541	1.493	0.945

Table 3.12. Analysis of RRS and PRESS for centered data in Exercise 3.4.

r	RSS	PRESS	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	3.4025	5.1819		
2	0.4291	0.5849	0.172	0.11287
3	0.3692	0.5813	**1.355**	**0.99387**
4	0.3117	0.5533	1.499	0.95187
5	0.2626	0.5260	1.688	0.95063

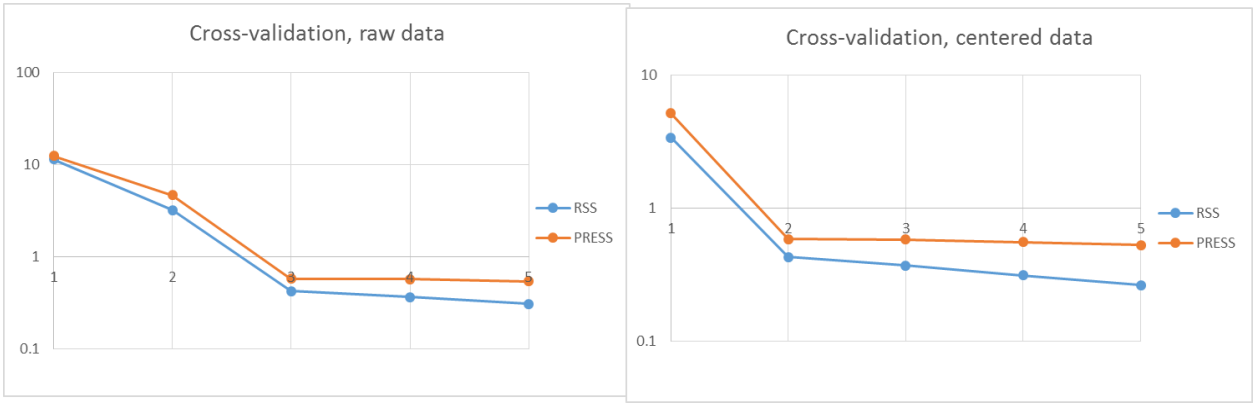


Fig. 3.34. Cross-validation of data in Exercise 3.4; logarithmic plot of RSS and PRESS for the raw and centered data.

Analysis of the RSS and PRESS plots,

Fig. 3.34, for the raw data indicates presence of three PCs while that for the centered data presence of two PCs. This fact is also confirmed by the analysis in Table 3.11 and 3.12 from the analysis of $PRESS_r/RSS_{r-1}$ and $PRESS_r/PRESS_{r-1}$ that here are three PCs for the raw and two for centered data. In such a case of conflicting results it is advisable to use the one for raw data that there are three PCs influencing the spectroscopic data.

Exercise 3.5.

Determine number of principal components for the data file Xdata.m in folder Ex3-5 and Ex3-5.xlsx. These data were displayed in Fig. 1.1.

First, PCA was carried out on the raw and centered data. The obtained eigenvalues of principal components are displayed in

Table 3.13.

Table 3.13. Sizes of the first five PCs using PCA on the data in Exercise 3.5.

	Raw data			Centered data		
PC_r	λ_r	%	Cumulative %	λ_r	%	Cumulative %
1	267.9820	96.4984%	96.4984%	16.1409	84.144%	84.14%
2	9.5795	3.4495%	99.9479%	2.9047	15.143%	99.287%
3	0.0566	0.0204%	99.9683%	0.0529	0.276%	99.5628%
4	0.0464	0.0167%	99.9850%	0.0428	0.223%	99.786%
5	0.0415	0.0150%	100.0000%	0.0411	0.214%	100.0000%
	sum			sum		
	277.7061			19.1825		

Results displayed for the raw (i.e. non-centered) data show that the second PC contributes only 3.45%, which is lower than 5% usually used in statistics. However, using centered data the second component contributes 15.14% and the first two components explain 99.29% of the variation. This confirms that there are only two important principal components influencing the spectra and shows that using centered data is advantageous in the analysis. Calculation of the spectra using only two PCs shows partial noise reduction. Comparison of the experimental (raw) and calculated using Eq. (3.6) (approximated) spectrum No 3 is shown in Fig. 3.35.

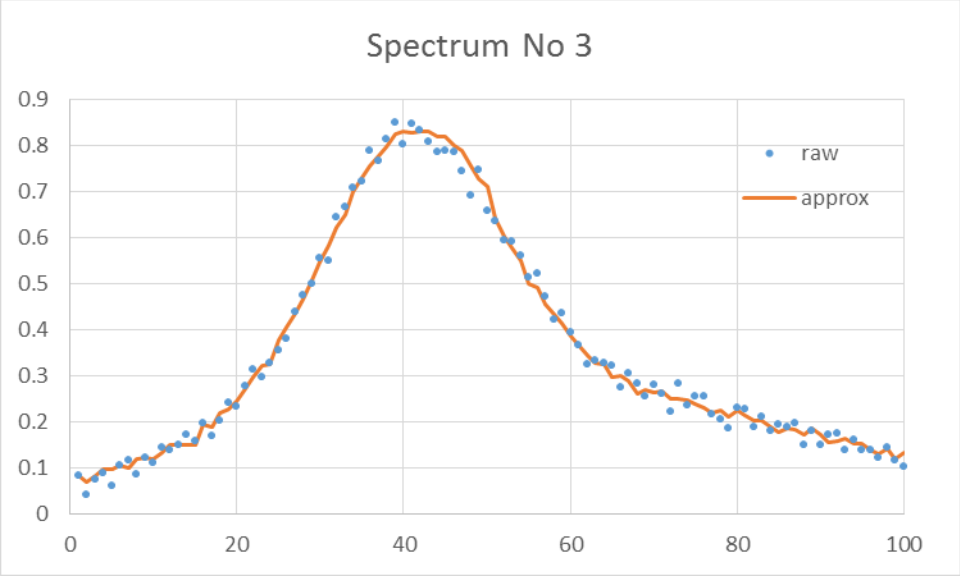


Fig. 3.35. Plot of the experimental, $\mathbf{x}$ (raw) and calculated, $\hat{\mathbf{X}}$ (approx) spectrum No 3, using two PCs.

The calculated spectra $\hat{\mathbf{X}}$ are shown in Fig. 3.36 and should be compared with raw data in Fig. 1.1 to see noise reduction.

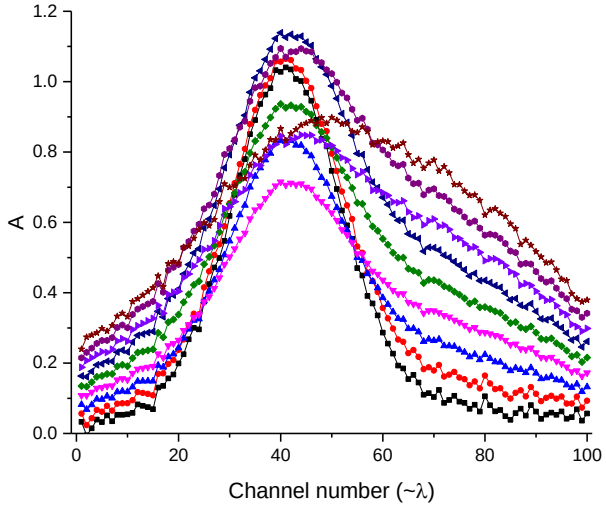


Fig. 3.36. Spectra calculated using two PCs; they should be compared with raw data in Fig. 1.1.

Scores plots are not interesting here as the concentrations of the different samples are random. However, the loadings plots are related to the spectra of the individual components in the analysis. They are presented in Fig. 3.37 and 3.38 for the raw data.

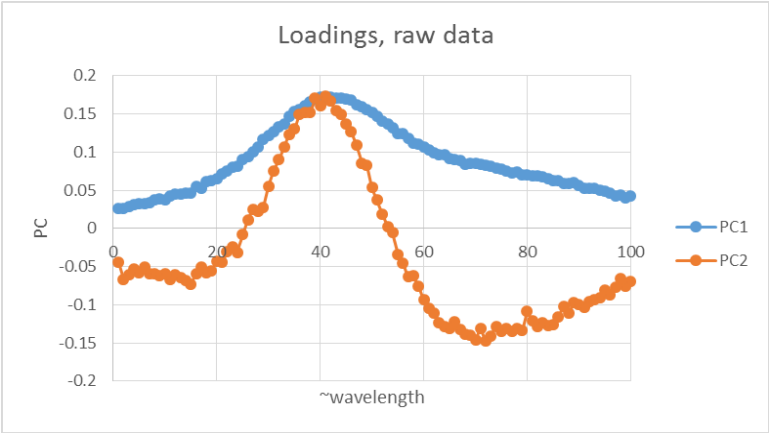


Fig. 3.37. Loadings plots $\mathbf{p}_1$ (PC1) and $\mathbf{p}_2$ (PC2) vs. number proportional to the wavelength using PCA and the raw data for Exercise 3.5.

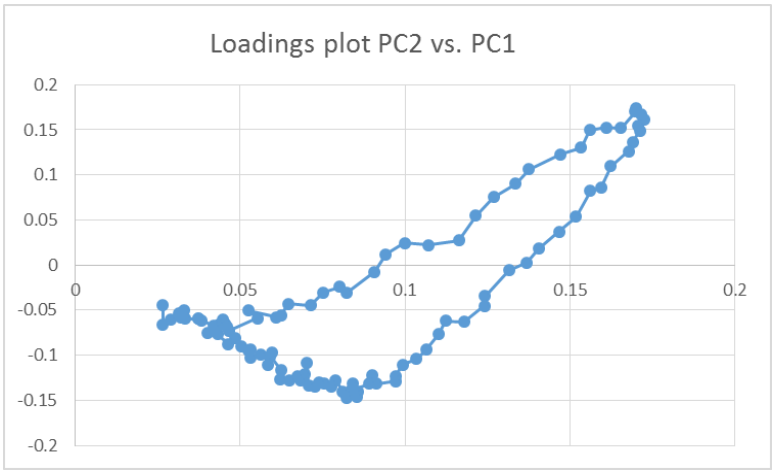


Fig. 3.38. Loadings plot of $\mathbf{t}_2$ (PC2) versus $\mathbf{t}_1$ (PC1) for raw data for Exercise 3.5.

One can also use cross-validation to obtain information on the number of important PCs. The calculated values of RRS and PRESS are shown in Table 3.14 and 3.15.

Table 3.14. Results of the cross-validation analysis for the raw data from.

r	RRS_r	$PRESS_r$	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	9.839	12.3730		
2	0.260	0.4405	0.04477	0.03560
3	0.203	0.4346	**1.67400**	0.98659
4	0.157	0.4329	2.13248	0.99591
5	0.115	0.4296	2.74453	0.99258

Table 3.15. Results of the cross-validation analysis for the centered data from Exercise 3.5.

r	RRS_r	$PRESS_r$	$PRESS_r/RSS_{r-1}$	$PRESS_r/PRESS_{r-1}$
1	3.124	5.3117		
2	0.219	0.5036	0.1612	0.0948
3	0.167	0.4966	**2.2629**	0.9862
4	0.124	0.4960	2.9783	0.9988
5	0.083	0.4890	3.9507	0.9858

Plots of RRS and PRESS as functions of the number of principal components, *r*, are shown in Fig. 3.39. These plots are for the raw data but those for centered data are similar.

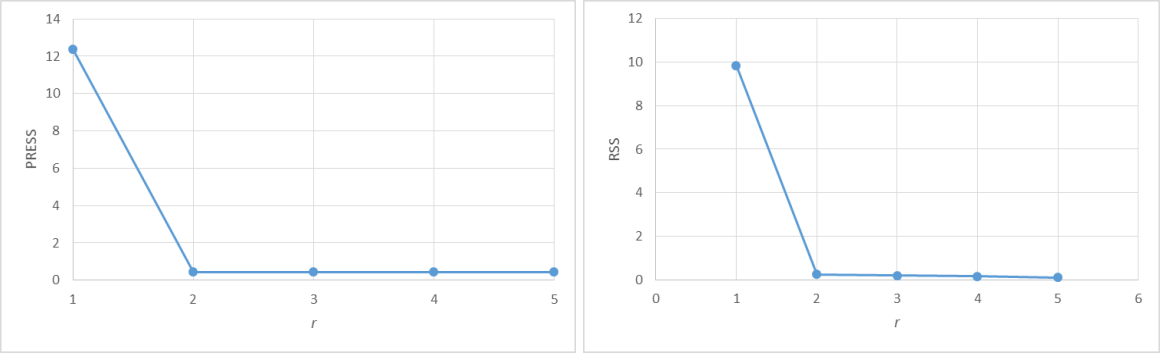


Fig. 3.39. Plots of PRESS and RSS as functions of the number of PCs using the raw data in Exercise 3.5.

These results show that by adding the second PC both RRS and PRESS decrease significantly, but addition of further components does not affect these parameters significantly. Moreover, the ratio of $PRESS_3/RSS_2$ is 1.7 or 2.3 for the raw and centered data, respectively, and from $r = 3$ $PRESS_r/PRESS_{r-1}$ is around one. These tests indicate that the first two PCs should be retained and the subsequent PCs model only random noise.

The above exercises show how to carry out principal component analysis to determine number of principal components influencing the experimental data. Detailed analysis of the eigenvalues (that is the magnitude of the PCs) should be carried out for the raw or centered data (in some cases standardized data). This analysis should be confirmed by the cross-validation. The understanding of the physical/analytical model is also necessary to decide the number of PCs. However, noise, spectral similarities and correlations between concentrations often make it hard to provide an exact estimate of the number of significant components.[3]

3.4 Exploratory data analysis

PCA can be used to find relations and differences in multivariate data. It is often applied in food and pharmaceutical industry to find the origin of samples, e.g. coffee, whisky, wine, beer, etc., and determination of possible counterfeiting. This might be better understood by exploring examples. There are few measures used for classification shown below but there are many more which are presented in the literature.[3,10]

3.4.1 Mahalanobis distance

To distinguish between groups of data (clusters) one of the measures is Mahalanobis distance (statistical distance).[3,10] It is a measure of a distance of a group of data form the reference sample (all data or another group). The Mahalanobis distances between a series of samples **y** and a class **x** is defined as:

$$d_{\mathbf{y}_i,\mathbf{x}} = \sqrt{(\mathbf{y}_i - \bar{\mathbf{x}})\mathbf{C}_\mathbf{X}^{-1}(\mathbf{y}_i - \bar{\mathbf{x}})'} \tag{3.26}$$

where $\mathbf{y}_i$ is the row vector of series of samples, $\mathbf{C_X}$ is the variance-covariance matrix of **X**, and $\bar{\mathbf{x}}$ is a vector of means of **X** columns. **y** can be identical to **x**, include **x**, or be an unknown sample set. Covariance matrix $\mathbf{C_X}$ is calculated using number of degrees of freedom equal to *N* (number of rows) i.e. using the population (not sample) statistics. It can be easily calculated in Matlab using function cov(**X**,1) as **Xc**'**Xc**/*N* where **Xc** is the centered (by the means of columns) matrix **X:**

$$\mathbf{C_x} = \frac{(\mathbf{X} - \bar{\mathbf{x}})'(\mathbf{X} - \bar{\mathbf{x}})}{N} \tag{3.27}$$

Typically, distances between samples *i* of the matrix **X** and mean of class **A** (which is a part of the matrix **X**) are calculated:

$$d_{i,\mathrm{A}} = \sqrt{(\mathbf{x}_i - \bar{\mathbf{x}}_\mathbf{A})\mathbf{C}_\mathbf{A}^{-1}(\mathbf{x}_i - \bar{\mathbf{x}}_\mathbf{A})'} \tag{3.28}$$

where $\mathbf{x}_i$ is a row vector for the sample *i*, $\bar{\mathbf{x}}_\mathbf{A}$ is the vector of means of class A, and $\mathbf{C}_\mathrm{A}$ is the variance-covariance matrix for group A. Samples and the group must have the same number of columns. Matrix $\mathbf{C}_\mathrm{A}$ is scaling the distances. Example of application of Mahalanobis distance will be presented in Exercise 3.7. When the variance-covariance matrix is the identity matrix (ones on the diagonal) the Mahalanobis distance becomes simple Euclidean distance:

$$d_{i,\mathrm{A}} = \sqrt{(\mathbf{x}_i - \bar{\mathbf{x}}_\mathbf{A})(\mathbf{x}_i - \bar{\mathbf{x}}_\mathbf{A})'} \tag{3.29}$$

3.4.2 SIMCA

The method of SIMCA is used in pattern recognition.[3,10,24] The acronym SIMCA means *soft independent modeling of class analogy*. Soft modeling means that two classes can overlap and an object can belong to two classes simultaneously. SIMCA belongs to one class classifiers.

The SIMCA method uses PCA as the first step. Very often, the logarithm of the data is taken and then the data are standardized to keep the same importance of all the parameters. Let us suppose one example where that there is one class of measurements **X** in the whole set of data **Y** (that is **X** is part of **Y**). PCA is performed on class **X** and matrices of scores, **T**, and loadings, **P**, are obtained. The new values of $\hat{\mathbf{Y}}$ are predicted using data **Y** and loadings of class **X**, see Eqns. (3.20) and (3.22). First, scores for **Y** are predicted using loadings **P**:

$$\mathbf{T_Y} = \mathbf{YP} \tag{3.30}$$

then new values are predicted (**PP'=I**):

$$\hat{\mathbf{Y}} = \mathbf{T_Y P}' \tag{3.31}$$

The difference between original **Y** and predicted $\hat{\mathbf{Y}}$ are calculated:

$$\mathbf{E_Y} = \mathbf{Y} - \hat{\mathbf{Y}} \tag{3.32}$$

and the square root of the sums of squares of columns of error matrix is obtained:

$$d_i = \sqrt{\sum_{j=1}^{J} E_y^2(i,j)} \tag{3.33}$$

This distance d_i presents the SIMCA distance from class **Y** to **X**. The subroutine simca.m from Brereton[3] can be used in calculations. Few examples of exploratory analysis and pattern recognition are illustrated in exercises below.

Exercise 3.6.

Let us investigate comparison of different chromatographic columns.[3] The aim of this analysis is to determine which columns behave in a similar fashion and which are different. Performance of eight commercial chromatographic columns were measured by determination of four peak characteristics: *k*' (capacity factor), *N* (number of theoretical plates), *N*(df) (peak width parameter), and A_s (asymmetry), for eight compounds denoted by a letter (P, N, A, C, Q, B, D, R). There are 32 parameters and the data are represented by a matrix **X(**8×32), its transposed version is presented in Table 3.16.

Let us apply PCA to these data. Because different parameters have very different values and units the data must be standardized. The results of the PCA analysis might be presented on the score plot of the second PC2, $\mathbf{t}_2$, versus the first PC1, $\mathbf{t}_1$. This is illustrated in Fig. 3.40.

Table 3.16. Parameters (32) of eight chromatographic columns for eight compounds **X'**(32×8).[3]

Parameter	Inertsil ODS	Inertsil ODS-2	Inertsil ODS-3	Kromasil C18	Kromasil C8	Symmetry C18	Supelco ABZ+	Purospher
Pk	0.25	0.19	0.26	0.3	0.28	0.54	0.03	0.04
PN	10 200	6 930	7 420	2 980	2 890	4 160	6 890	6 960
PN(df)	2 650	2 820	2 320	293	229	944	3 660	2 780
PA*s*	2.27	2.11	2.53	5.35	6.46	3.13	1.96	2.08
Nk	0.25	0.12	0.24	0.22	0.21	0.45	0	0
NN	12 000	8 370	9 460	13 900	16 800	4 170	13 800	8 260
NN(df)	6 160	4 600	4 880	5 330	6 500	490	6 020	3 450
NA*s*	1.73	1.82	1.91	2.12	1.78	5.61	2.03	2.05
Ak	2.6	1.69	2.82	2.76	2.57	2.38	0.67	0.29
AN	10 700	14 400	11 200	10 200	13 800	11 300	11 700	7 160
AN(df)	7 790	9 770	7 150	4 380	5 910	6 380	7 000	2 880
AA*s*	1.21	1.48	1.64	2.03	2.08	1.59	1.65	2.08
Ck	0.89	0.47	0.95	0.82	0.71	0.87	0.19	0.07
CN	10 200	10 100	8 500	9 540	12 600	9 690	10 700	5 300
CN(df)	7 830	7 280	6 990	6 840	8 340	6 790	7 250	3 070
CA*s*	1.18	1.42	1.28	1.37	1.58	1.38	1.49	1.66
Qk	12.3	5.22	10.57	8.08	8.43	6.6	1.83	2.17
QN	8 800	13 300	10 400	10 300	11 900	9 000	7 610	2 540
QN(df)	7 820	11 200	7 810	7 410	8 630	5 250	5 560	941
QA*s*	1.07	1.27	1.51	1.44	1.48	1.77	1.36	2.27
Bk	0.79	0.46	0.8	0.77	0.74	0.87	0.18	0
BN	15 900	12 000	10 200	11 200	14 300	10 300	11 300	4 570
BN(df)	7 370	6 550	5 930	4 560	6 000	3 690	5 320	2 060
BA*s*	1.54	1.79	1.74	2.06	2.03	2.13	1.97	1.67
Dk	2.64	1.72	2.73	2.75	2.27	2.54	0.55	0.35
DN	9 280	12 100	9 810	7 070	13 100	10 000	10 500	6 630
DN(df)	5 030	8 960	6 660	2 270	7 800	7 060	7 130	3 990
DA*s*	1.71	1.39	1.6	2.64	1.79	1.39	1.49	1.57
Rk	8.62	5.02	9.1	9.25	6.67	7.9	1.8	1.45
RN	9 660	13 900	11 600	7 710	13 500	11 000	9 680	5 140
RN(df)	8 410	10 900	7 770	3 460	9 640	8 530	6 980	3 270
RA*s*	1.16	1.39	1.65	2.17	1.5	1.28	1.41	1.56

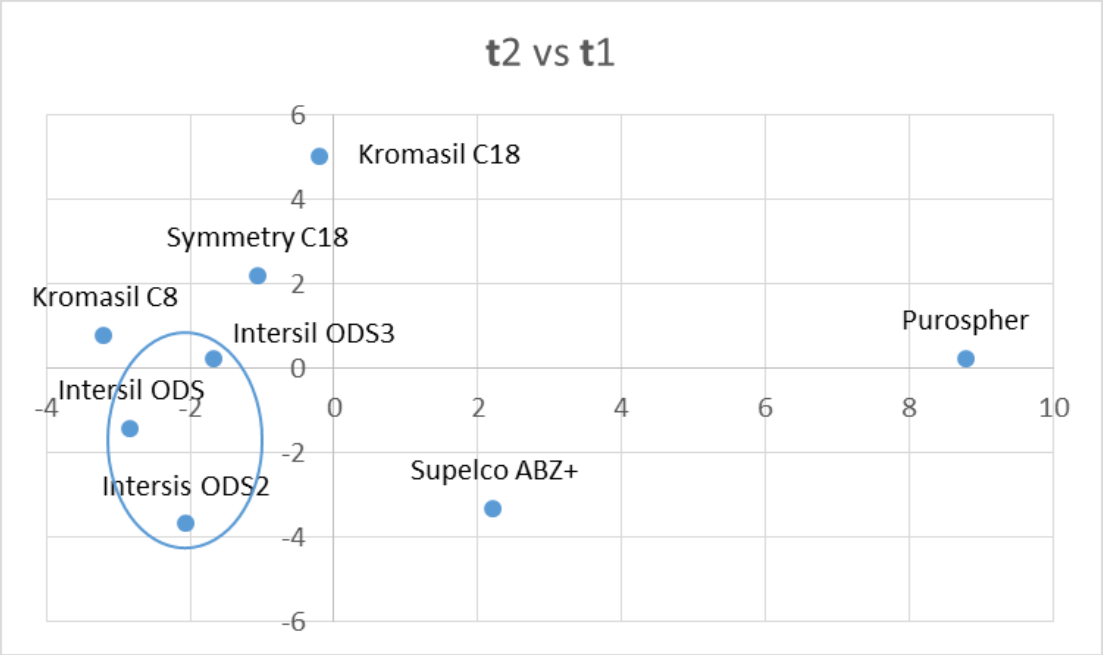


Fig. 3.40. Scores plot of PC2, $\mathbf{t}_2$, versus PC1, $\mathbf{t}_1$.

This plot suggests that closely clustering three Intersil columns behave similarly while Kromasil C8 and Purospher behave in opposite manner (are negatively correlated) and behavior of Purospher is different form the other columns. These facts might be important in the determination of which columns are best for different types of separations. For example, parameter which has high value for Purospher will have low value for Kromasil C8 (and vice versa). This would suggest that each column has a different purpose.

The scores plot was related to the columns but the loadings plot is related to the chromatographic parameters for different compounds. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$, is shown in Fig. 3.41. It can be concluded that loadings for *k* parameter for all compounds are closely clustered which suggests that this parameter does not vary much for different compounds and columns. Parameters As, N and N(df) show more variation but N and N(df) are more closely correlated. Parameters As and N are quite different and almost diametrically opposed suggesting that they measure opposite properties, e.g. high As corresponds to low N values.

Some parameters are in the middle of the loadings plots, such as NN. These behave atypically and are probably not useful indicators of column performance.

Most loadings are on an approximate large circles (ovals). This is because standardization is used, and suggests that we are probably correct in keeping only two principal components. The order of the compounds for both As and N reading clockwise around the circle are very similar, with P, D and N at one extreme and Q and C at the other extreme. This suggests that behavior is grouped according to chemical structure, and also that it is possible to reduce the number of test compounds by selecting one compound in each group.[3] These conclusions might be interesting in the chromatographic analysis. This analysis suggests that some tests or compounds can be omitted. Some measurements might be misleading as they are not typical of the overall pattern. If other PCs are important other plots might also be included in the analysis.

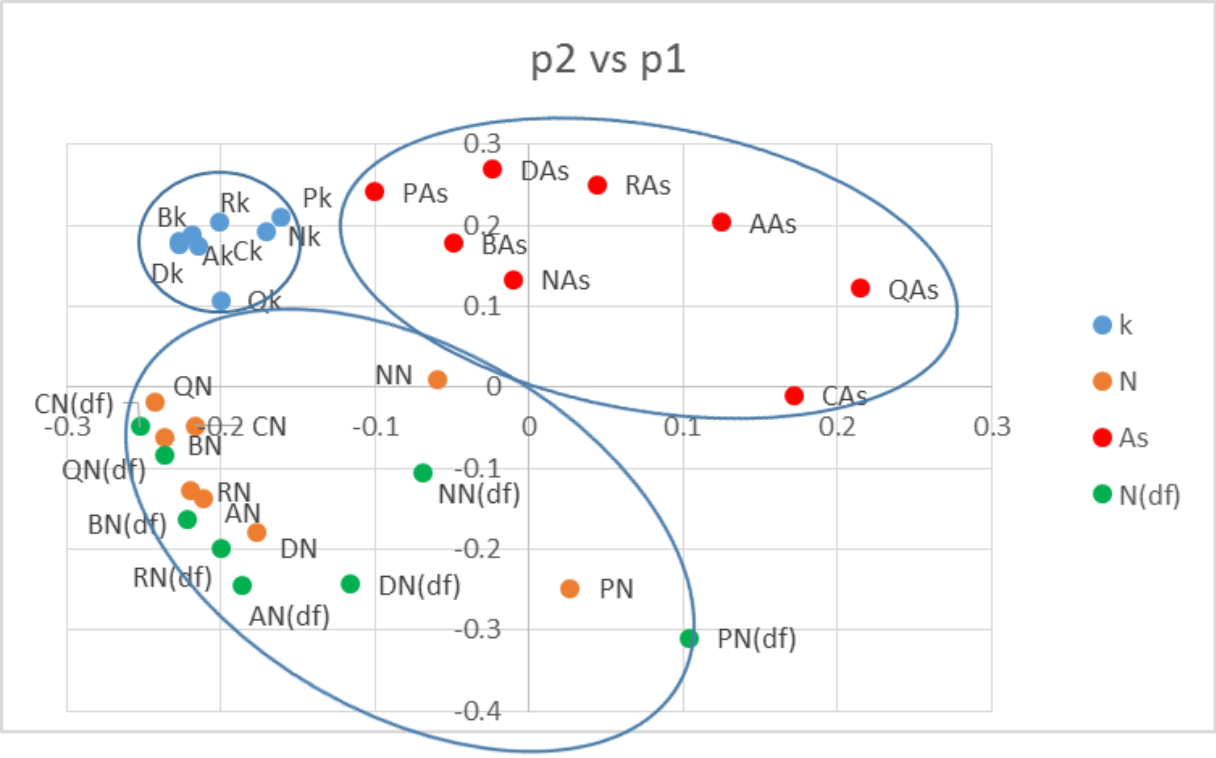


Fig. 3.41. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$.

Another method of comparison are the biplots where loadings and scores are presented together on one plot. To adjust the scales of both plots the scores are normalized using the following equation:[3]

$$\mathbf{t}_{i,r}^{\text{norm}} = \frac{\mathbf{t}_{i,r}}{\sum_{i=1}^{I} \frac{\mathbf{t}_{i,r}^2}{I}} \tag{3.34}$$

Biplot for the data in Exercise 3.6 is shown in Fig. 3.42. There is a lot of information in this plot which looks a little messy but it is possible to draw some conclusions.

Purospher lies at the extreme position along the horizontal axis, as does CAs. Hence we would expect CAs to have a high value for Purospher, which can be verified by examining Table 3.16. A similar comment can be made concerning DAs and Kromasil C18. These tests are good specific markers for particular columns.

Likewise, parameters at opposite corners to chromatographic columns will exhibit characteristically low values, for example, QN has a value of 2540 for Purospher. The chromatographic columns Supelco ABZ+ and Symmetry C18 are almost diametrically opposed, and good discriminating parameters are the measurements on the peaks corresponding to compound P (pyridine), PAs and PN(df). Hence to distinguish the behavior between columns lying on this line, one of the eight compounds can be employed for the tests.

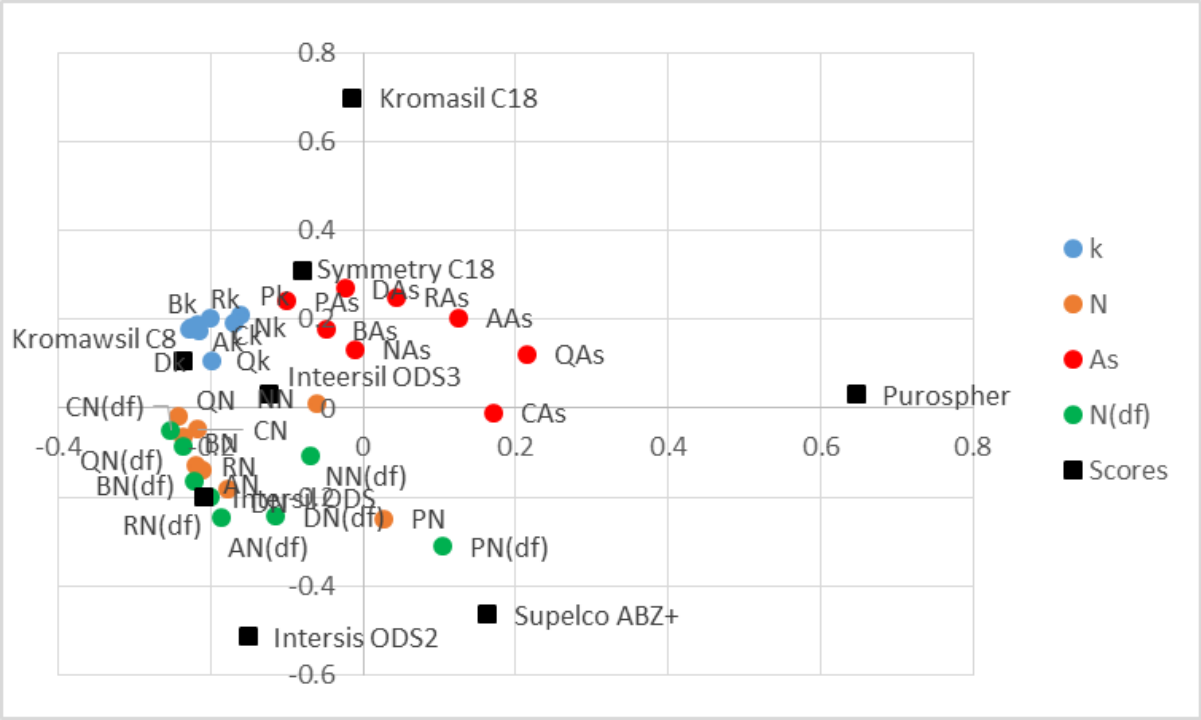


Fig. 3.42. Scores and loadings biplot for the data in Exercise 3.6.

Exercise 3.7.

Elemental analysis of 58 samples of pottery was carried out.[3] These samples were divided in two classes A (black carbon containing bulks) and B (clay). They are displayed in Table 3.17.

Perform PCA on these data. Can we distinguish between two classes of pottery?

Performing the PCA analysis gives loadings and scores. In this analysis standardization preoption was used as different elements have very different concentrations. The loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$, presents behavior of the elements and is displayed in Fig. 3.43 and the scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$ in Fig. 3.44.

Table 3.17. Results of the elemental analysis of pottery samples.

	Ti/%	Sr/ppm	Ba/ppm	m	Ca/%	Cr/ppm	Al/%	Fe/%	Mg/%	Na/%	K/%	Class
A1	0.304	181	1007	642	60	1.64	8.342	3.542	0.458	0.548	1.799	A
A2	0.316	194	1246	792	64	2.017	8.592	3.696	0.509	0.537	1.816	A
A3	0.272	172	842	588	48	1.587	7.886	3.221	0.54	0.608	1.97	A
A4	0.301	147	843	526	62	1.032	8.547	3.455	0.546	0.664	1.908	A
A5	0.908	129	913	775	184	1.334	11.229	4.637	0.395	0.429	1.521	A
E1	0.394	105	1470	1377	90	1.37	10.344	4.543	0.408	0.411	2.025	A
E2	0.359	96	1188	839	86	1.396	9.537	4.099	0.427	0.482	1.929	A
E3	0.406	137	1485	1924	90	1.731	10.139	4.49	0.502	0.415	1.93	A
E4	0.418	133	1174	1325	91	1.432	10.501	4.641	0.548	0.5	2.081	A
L1	0.36	111	410	652	70	1.129	9.802	4.28	0.738	0.476	2.019	A
L2	0.28	112	1008	838	59	1.458	8.96	3.828	0.535	0.392	1.883	A
L3	0.271	117	1171	681	61	1.456	8.163	3.265	0.521	0.509	1.97	A
L4	0.288	103	915	558	60	1.268	8.465	3.437	0.572	0.479	1.893	A
L5	0.253	102	833	415	193	1.226	7.207	3.102	0.539	0.577	1.972	A
C1	0.303	131	601	1308	65	0.907	8.401	3.743	0.784	0.704	2.473	A
C2	0.264	121	878	921	69	1.164	7.926	3.431	0.636	0.523	2.032	A
C3	0.264	112	1622	1674	63	0.922	7.98	3.748	0.549	0.497	2.291	A
C4	0.252	111	793	750	53	1.171	8.07	3.536	0.599	0.551	2.282	A
C5	0.261	127	851	849	61	1.311	7.819	3.77	0.668	0.508	2.121	A
G8	0.397	177	582	939	61	1.26	8.694	4.146	0.656	0.579	1.941	A
G9	0.246	106	1121	795	53	1.332	8.744	3.669	0.571	0.477	1.803	A
G10	1.178	97	886	530	441	6.29	8.975	6.519	0.323	0.275	0.762	A
G11	0.428	457	1488	1138	85	1.525	9.822	4.367	0.504	0.422	2.055	A
P1	0.259	389	399	443	175	11.609	5.901	3.283	1.378	0.491	2.148	B
P2	0.185	233	456	601	144	11.043	4.674	2.743	0.711	0.464	0.909	B
P3	0.312	277	383	682	138	8.43	6.55	3.66	1.156	0.532	1.757	B
P6	0.183	220	435	594	659	9.978	4.92	2.692	0.672	0.476	0.902	B
P7	0.271	392	427	410	125	12.009	5.997	3.245	1.378	0.527	2.173	B
P8	0.203	247	504	634	117	11.112	5.034	3.714	0.726	0.5	0.984	B
P9	0.182	217	474	520	92	12.922	4.573	2.33	0.59	0.547	0.746	B
P14	0.271	257	485	398	955	11.056	5.611	3.238	0.737	0.458	1.013	B
P15	0.236	228	203	592	83	9.061	6.795	3.514	0.75	0.506	1.574	B
P16	0.288	333	436	509	177	10.038	6.579	4.099	1.544	0.442	2.4	B
P17	0.331	309	460	530	97	9.952	6.267	3.344	1.123	0.519	1.746	B
P18	0.256	340	486	486	132	9.797	6.294	3.254	1.242	0.641	1.918	B
P19	0.292	289	426	531	143	8.372	6.874	3.36	1.055	0.592	1.598	B
P20	0.212	260	486	605	123	9.334	5.343	2.808	1.142	0.595	1.647	B
F1	0.301	320	475	556	142	8.819	6.914	3.597	1.067	0.584	1.635	B
F2	0.305	302	473	573	102	8.913	6.86	3.677	1.365	0.616	2.077	B
F3	0.3	204	192	575	79	7.422	7.663	3.476	1.06	0.521	2.324	B
F4	0.225	181	160	513	94	5.32	7.746	3.342	0.841	0.657	2.268	B
F5	0.306	209	109	536	285	7.866	7.21	3.528	0.971	0.534	1.851	B
F6	0.295	396	172	827	502	9.019	7.775	3.808	1.649	0.766	2.123	B
F7	0.279	230	99	760	129	5.344	7.781	3.535	1.2	0.827	2.305	B
D1	0.292	104	993	723	92	7.978	7.341	3.393	0.63	0.326	1.716	B
D2	0.338	232	687	683	108	4.988	8.617	3.985	1.035	0.697	2.215	B
D3	0.327	155	666	590	70	4.782	7.504	3.569	0.536	0.411	1.49	B
D4	0.233	98	560	678	73	8.936	5.831	2.748	0.542	0.282	1.248	B
M1	0.242	186	182	647	92	5.303	8.164	4.141	0.804	0.734	1.905	B
M2	0.271	473	198	459	89	10.205	6.547	3.035	1.157	0.951	0.828	B
M3	0.207	187	205	587	87	6.473	7.634	3.497	0.763	0.729	1.744	B
G1	0.271	195	472	587	104	5.119	7.657	3.949	0.836	0.671	1.845	B
G2	0.303	233	522	870	130	4.61	8.937	4.195	1.083	0.704	1.84	B
G3	0.166	193	322	498	80	7.633	6.443	3.196	0.743	0.46	1.39	B
G4	0.227	170	718	1384	87	3.491	7.833	3.971	0.783	0.707	1.949	B
G5	0.323	217	267	835	122	4.417	9.017	4.349	1.408	0.73	2.212	B
G6	0.291	272	197	613	86	6.055	7.384	3.343	1.214	0.762	2.056	B
G7	0.461	318	42	653	123	6.986	8.938	4.266	1.579	0.946	1.687	B

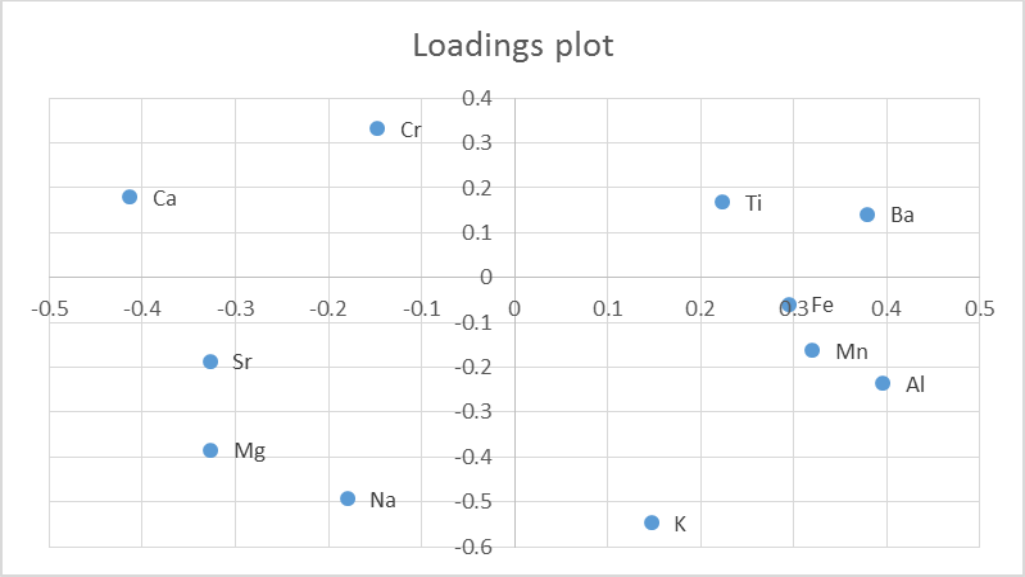


Fig. 3.43. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$.

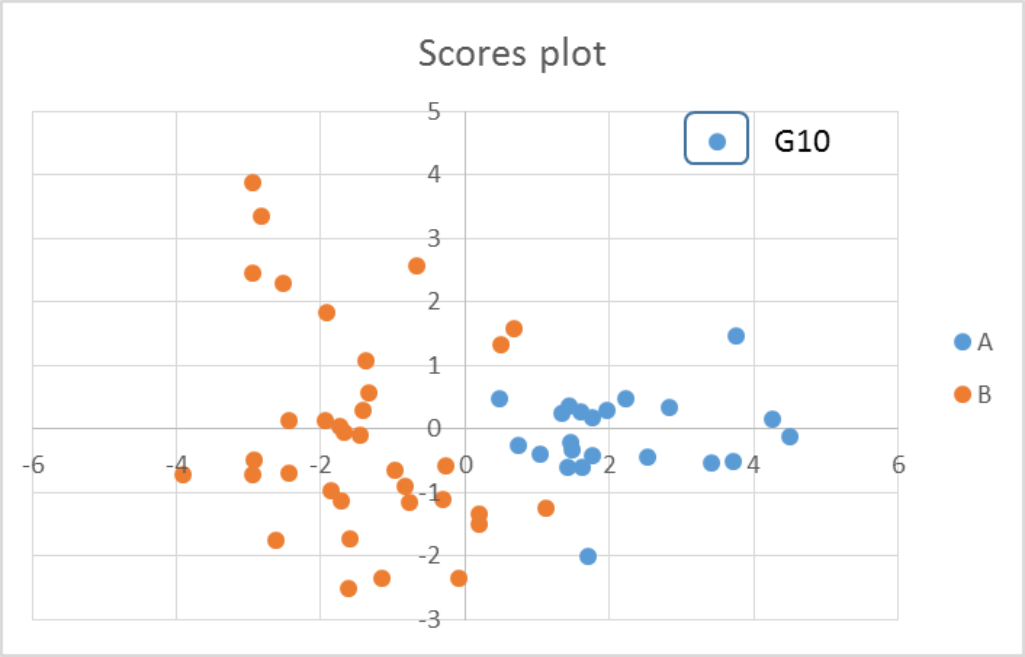


Fig. 3.44. Scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$.

Inspection of the scores plot indicates that the two classes of samples behave differently. There is a general difference between two classes although there is also some overlapping of data. Besides, point for sample G10 is evidently an outlier. The Mahalanobis distances are shown in Fig. 3.45 for two classes of results. They were calculated using program maha.m which calls the subroutine mahaldist.m.[3] Distinction between classes is well visible and G10 is evidently an outlier.

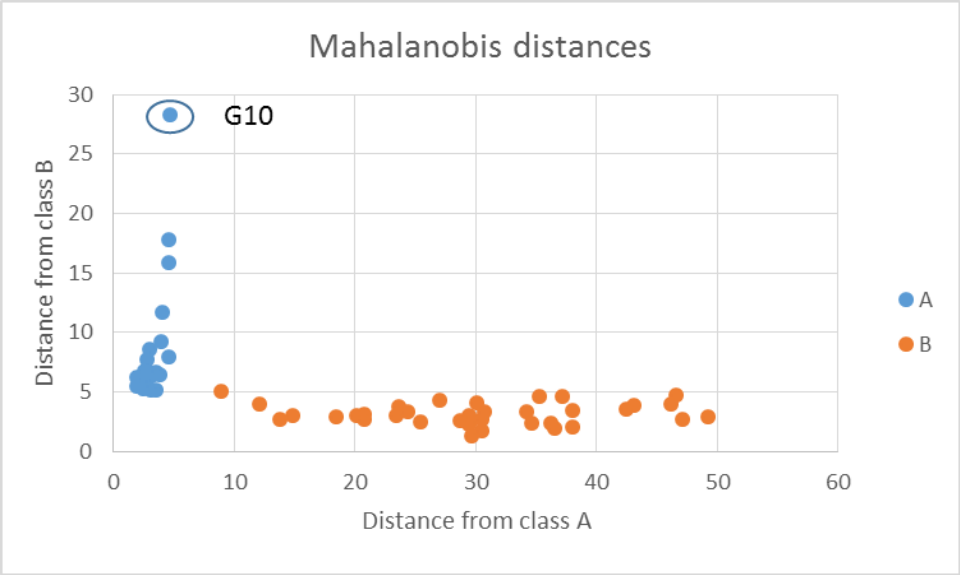


Fig. 3.45. Mahalanobis distances for classes A and B.

Next, the outlier G10 was removed from data set and the whole analysis was repeated. The scores and loadings plots are displayed in Fig. 3.36 and 3.37 and the Mahalanobis distances in Fig. 3.48

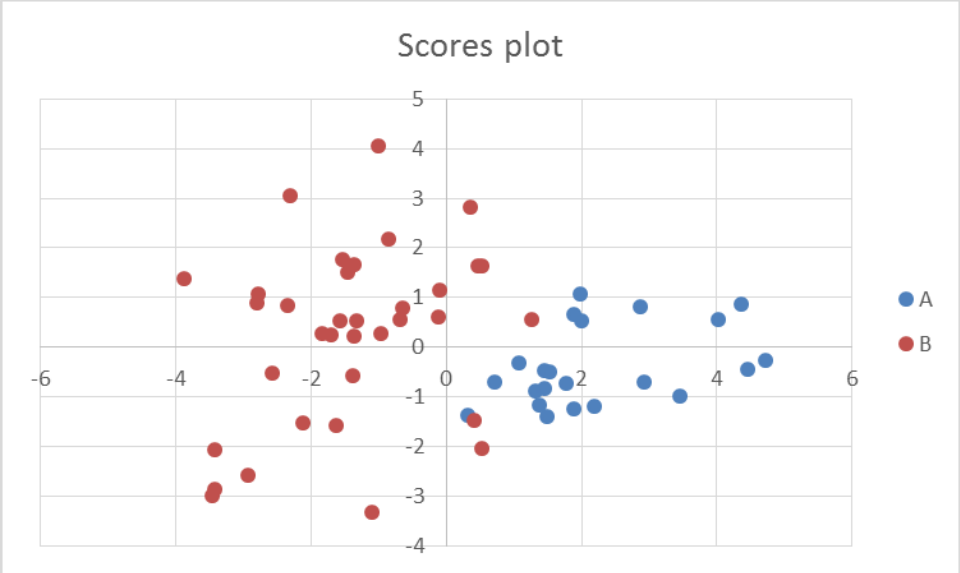


Fig. 3.46. Scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$ without outlier.

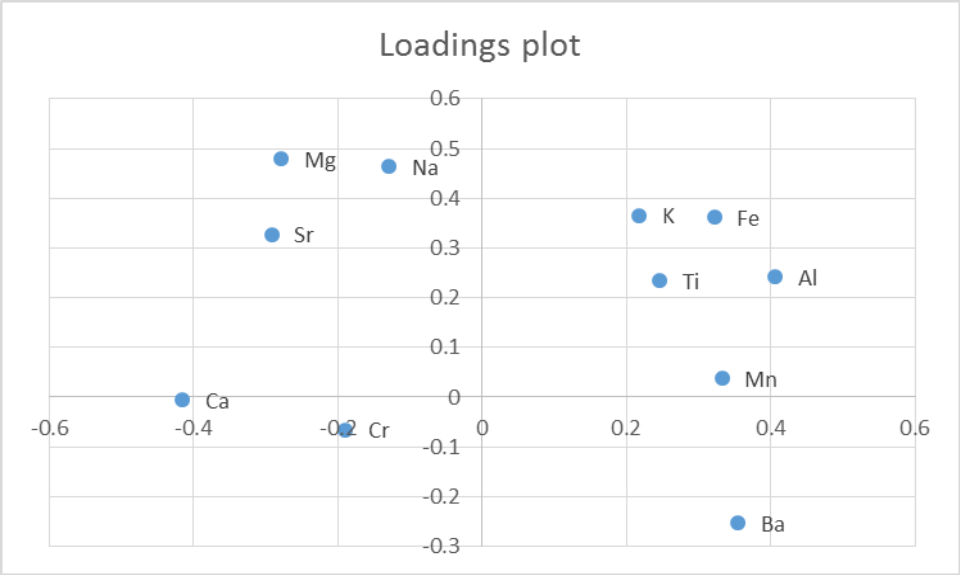


Fig. 3.47. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$ without outlier.

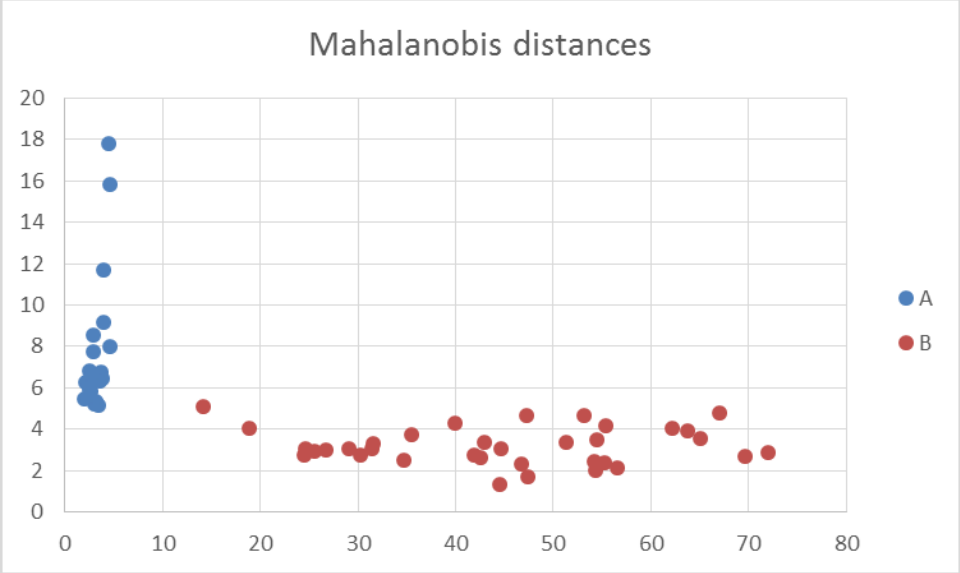


Fig. 3.48. Mahalanobis distances for classes A and B without outlier.

After deletion of the outlier all the plots changed. Although scores are still somewhat overlapping the Mahalanobis plot shows clear distinction of the two classes. This means that if new samples of pottery are found they could be easily classified. Distinction between classes could also be carried out using analysis of two elements lying on two opposite ends in the center of each group, e.g Ba and Mg. This is visible from the scores and loadings biplot, Fig. 3.49,

On the other hand, K and Na are approximately at right angles to the y axis and are poor for the discrimination of two classes.

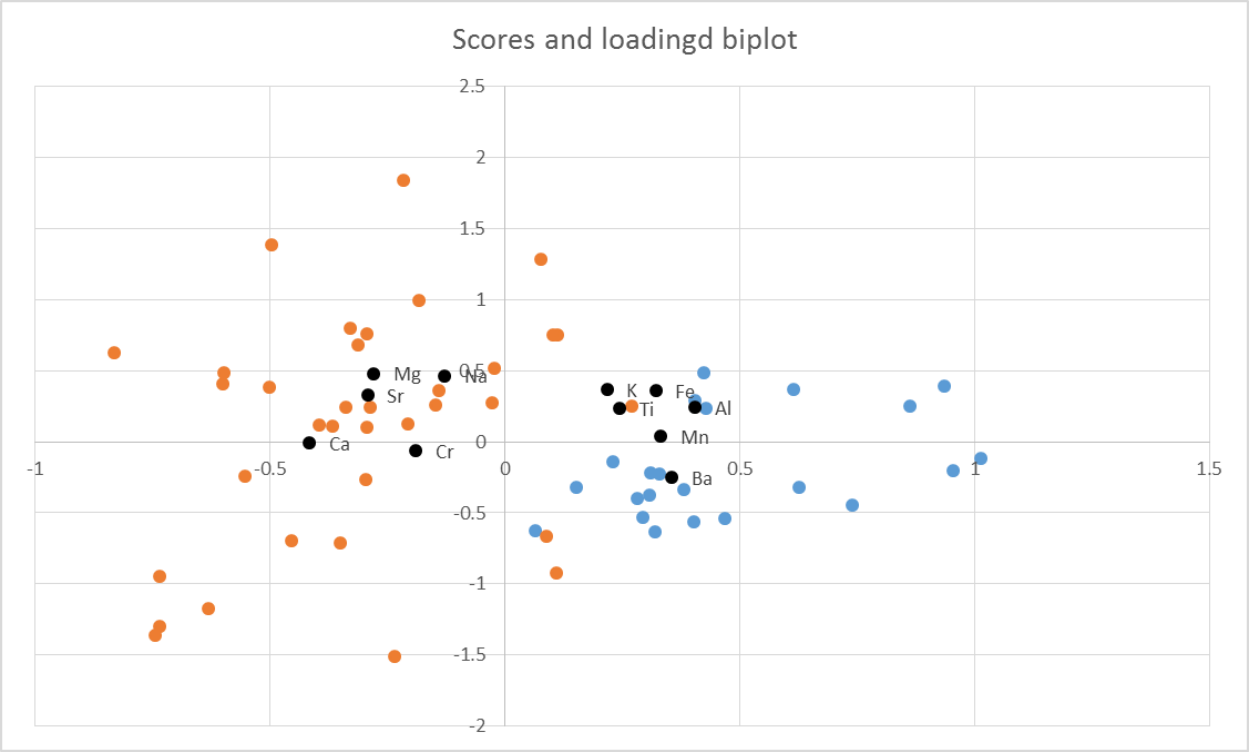


Fig. 3.49. Scores and loadings biplot without the outlier.

Exercise 3.8.

Data in this example come from the chemical analysis of the geological samples from Troodos area of Cyprus.[19] 143 rock samples were obtained from different locations of Troodos area. In this example concentrations of 8 oxides were determined. Are all the samples similar? Are there any outliers?

First, the PCA was conducted on all the data. As the numbers in **X** matrix are often very different, they were standardized. The scores plot is shown in Fig. 3.50 and loadings plot in Fig. 3.51. Scores plot suggests that points 65 and 66 are outliers which do not belong to the total group while points 129 and 130 are possible outliers. Such outliers should be carefully inspected to understand their origin. The outliers should be removed sequentially (not all at the same time). As samples 65 and 66 are very close to each other they might be remover together.

Next the PCA was carried out on 141 samples after removal of samples 65 and 66. The scores and loadings plots are displayed in Fig. 3.52 and Fig. 3.53, respectively. One can notice that samples 129 and 130 might also be outliers. In the next step samples 129 and 130 were removed from the analyzed data.

The results of the PCA on 139 samples (without samples 65, 66, 129, and 130) are displayed below in Fig. 3.54 and Fig. 3.55.

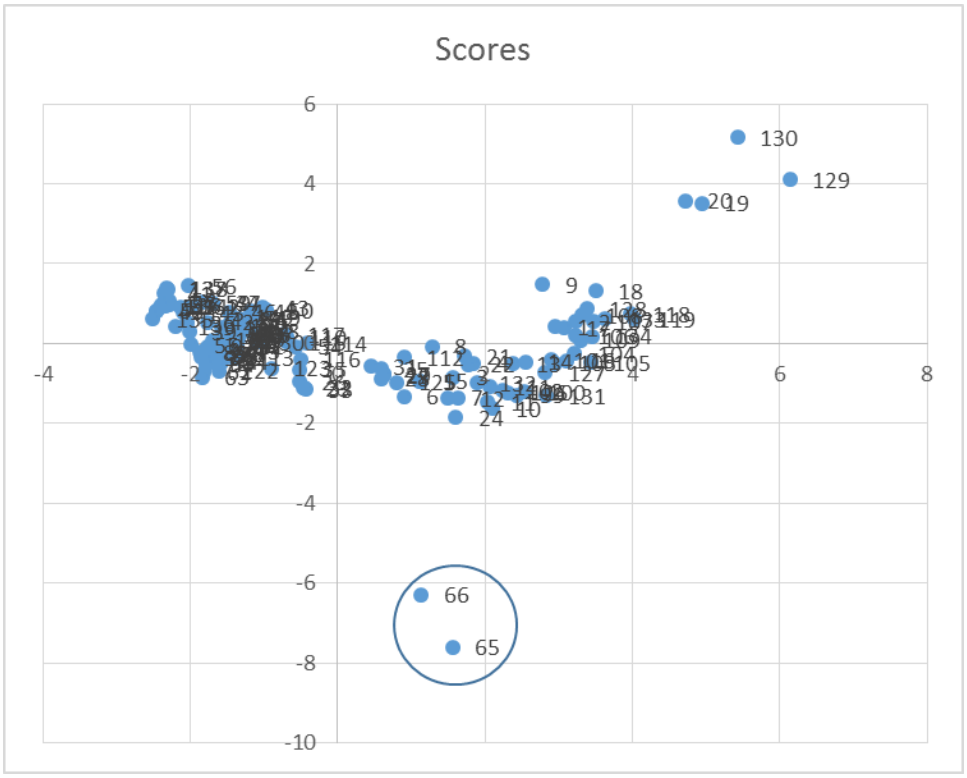


Fig. 3.50. Scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$ for data in Exercise 3.8.

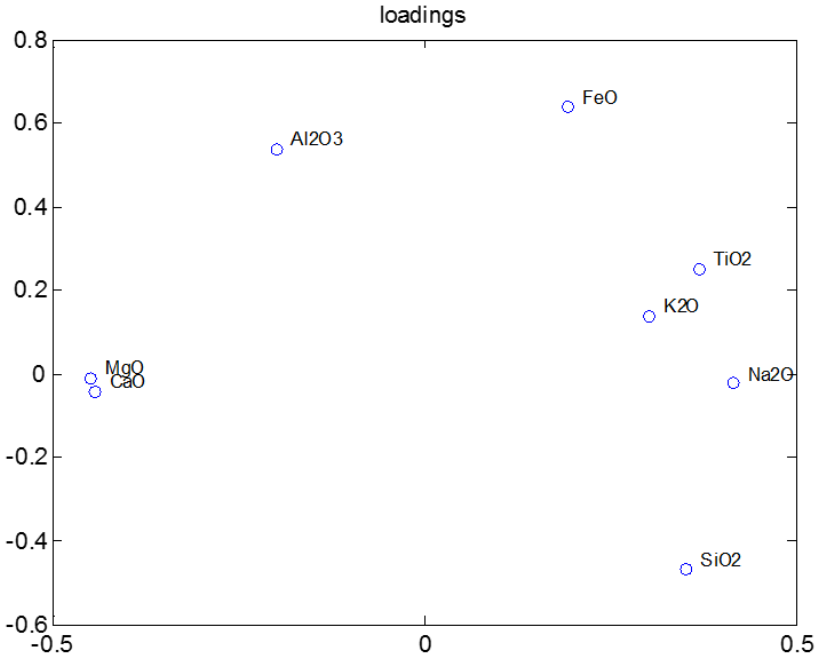


Fig. 3.51. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$.

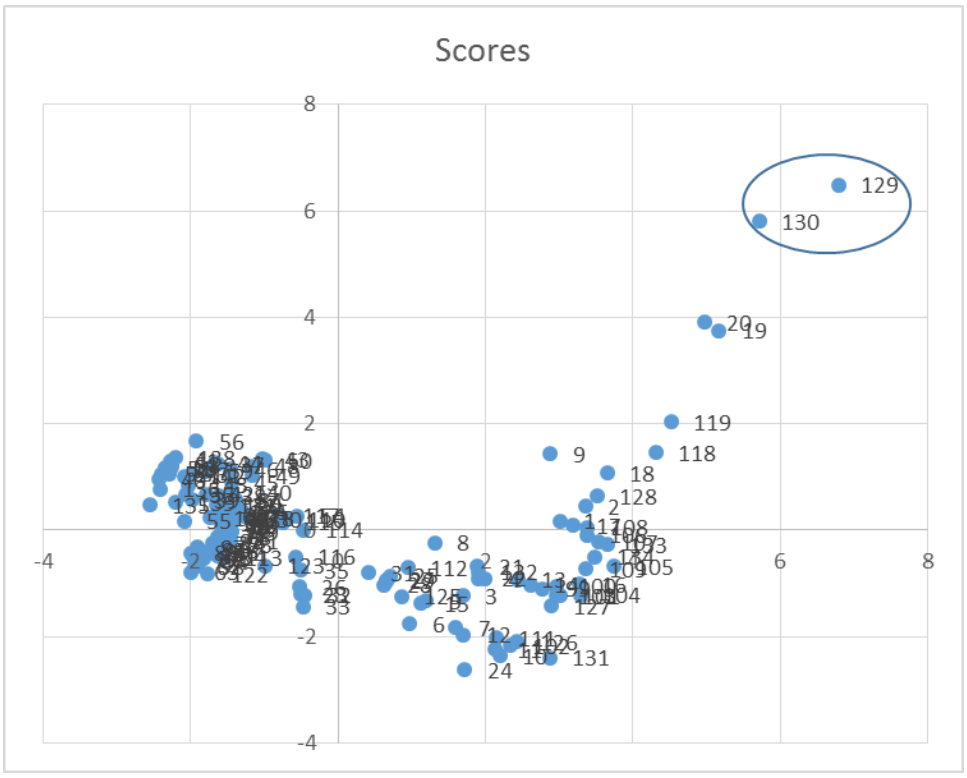


Fig. 3.52. Scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$ for data in Exercise 3.8 after removal of points 65 and 66.

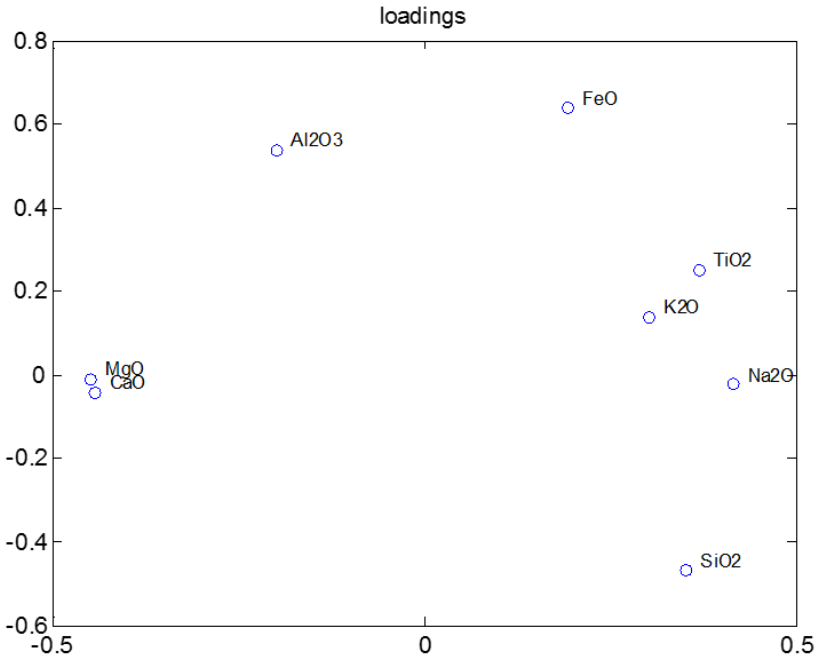


Fig. 3.53. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$ after removal of points 65 and 66.

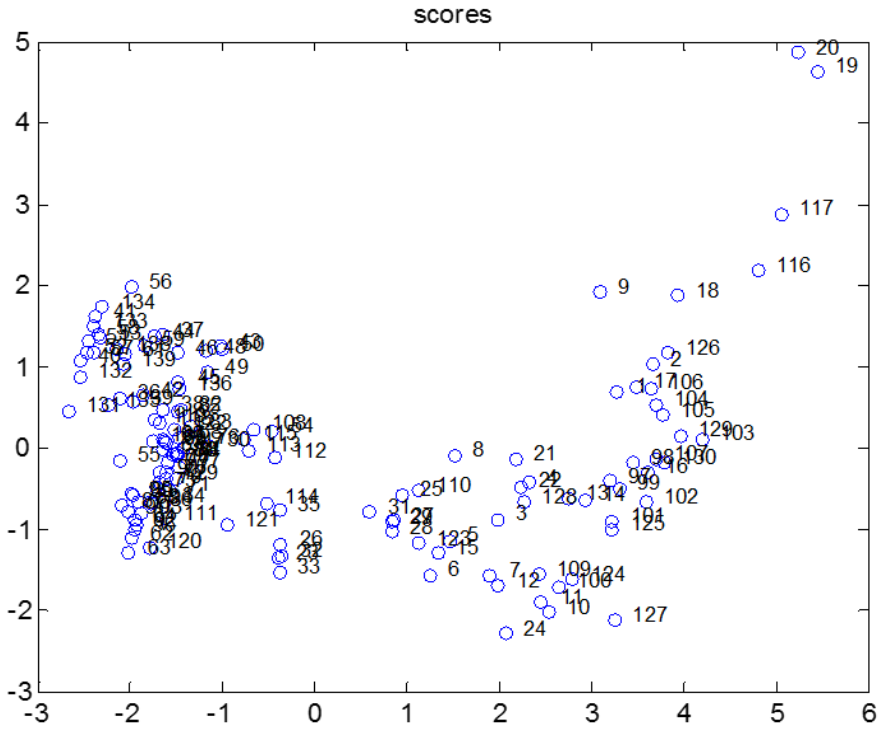


Fig. 3.54. Scores plot of PC2, $\mathbf{t}_2$, vs. PC1, $\mathbf{t}_1$ for data in Exercise 3.8 after removal of points 65, 66, 129, and 130.

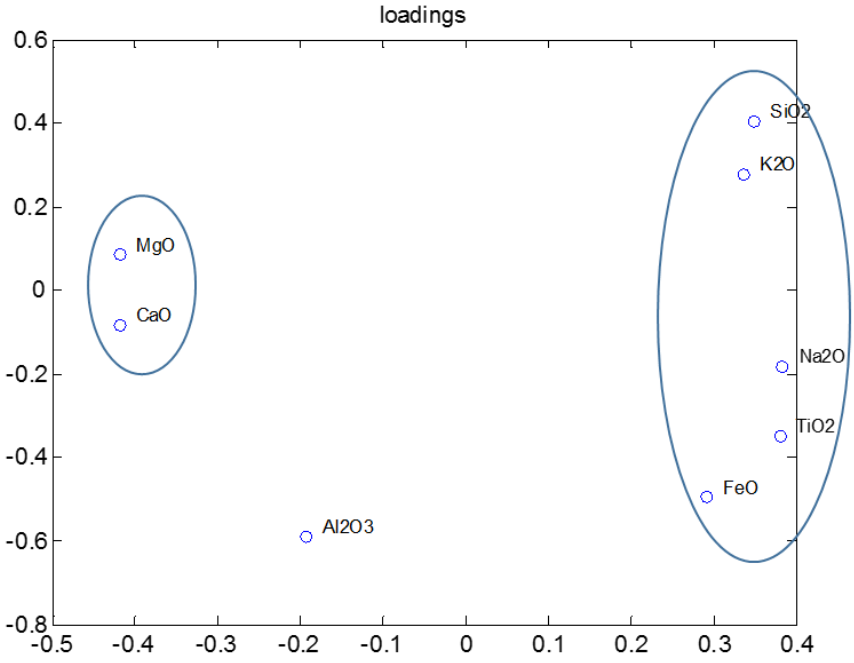


Fig. 3.55. Loadings plot of PC2, $\mathbf{p}_2$, vs. PC1, $\mathbf{p}_1$ after removal of points 65, 66, 129, and 130.

Loadings plots show that there are two main groups of variables: MgO and CaO in one and the rest, i.e. 5 in the other (except one lonely Al_2O_3). This means that despite of 8 variables used there are two underlying geochemical phenomena. This grouping was not visible when all the points (with outliers) were used in the model but appeared after removing the outliers. This analysis revealed hidden grouping leading to a new geological hypothesis.[19,25]

It is also interesting to compare variance explained as a function of the number of PCs. This is shown in Fig. 3.56.

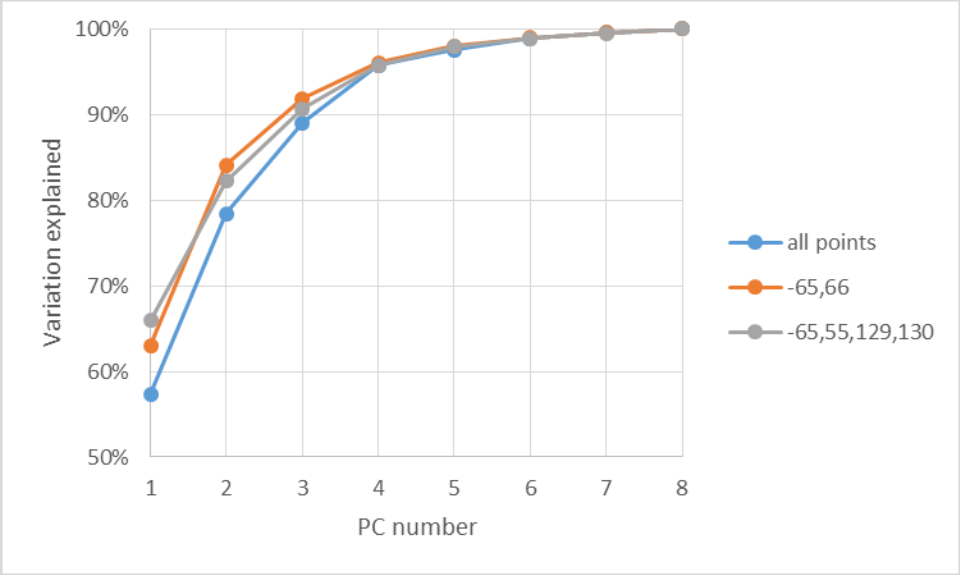


Fig. 3.56. Data variation explained (% cumulative) versus number of PCs using all points, after removing of two outliers, and after removing of four outliers.

Improvement in modeling is observed after removing two or four points but the two results without outliers are similar.

Exercise 3.9.

The next model contains measurements of petal and sepal dimensions of three types of irises: setosa, virginica, and versicolor.[19] Can these types of irises be distinguished by such measurements?

Example of how the measurements of these parameters were carried out is displayed in Fig. 3.57. These dimensions are in file Ex3-9.xlsx and Xdata.m and contain training and test data. Application of the PCA to all the data leads to the scores and loading plots in Fig. 3.58 and 3.59.

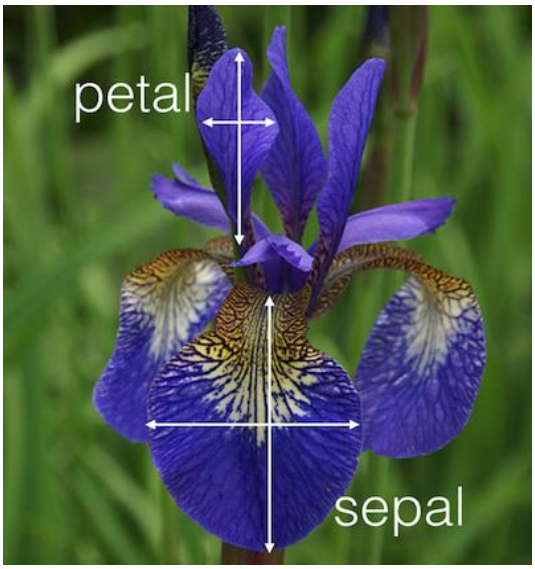


Fig. 3.57. Petal and sepal dimensions of irises.

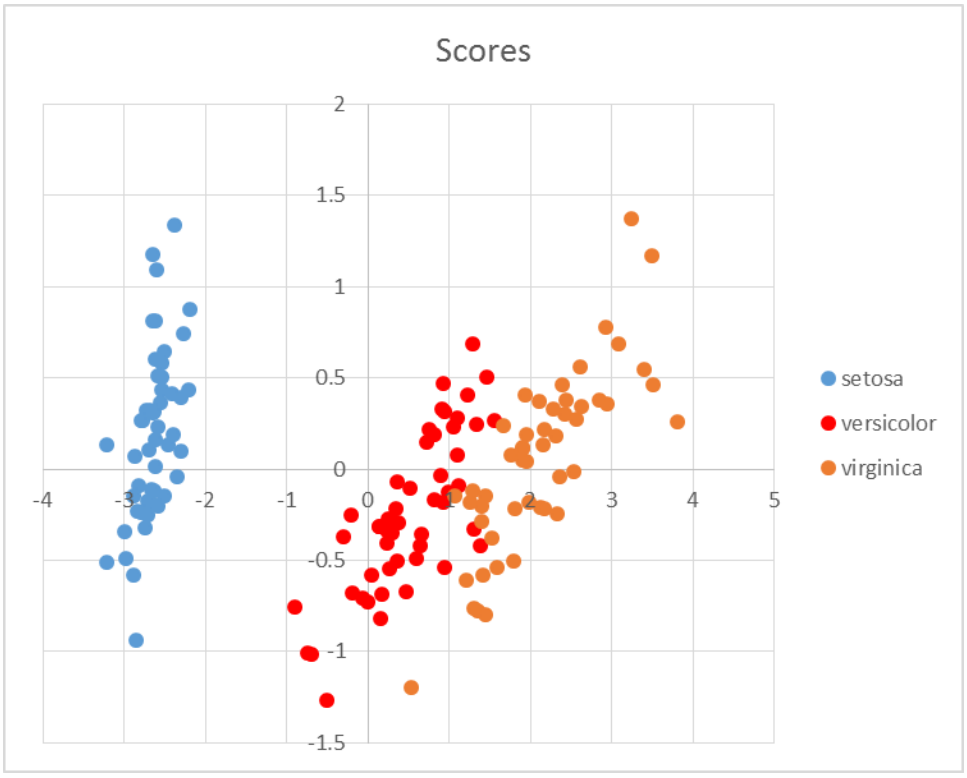


Fig. 3.58. Scores plot for all the data containing petal and sepal measurements of three types of irises.

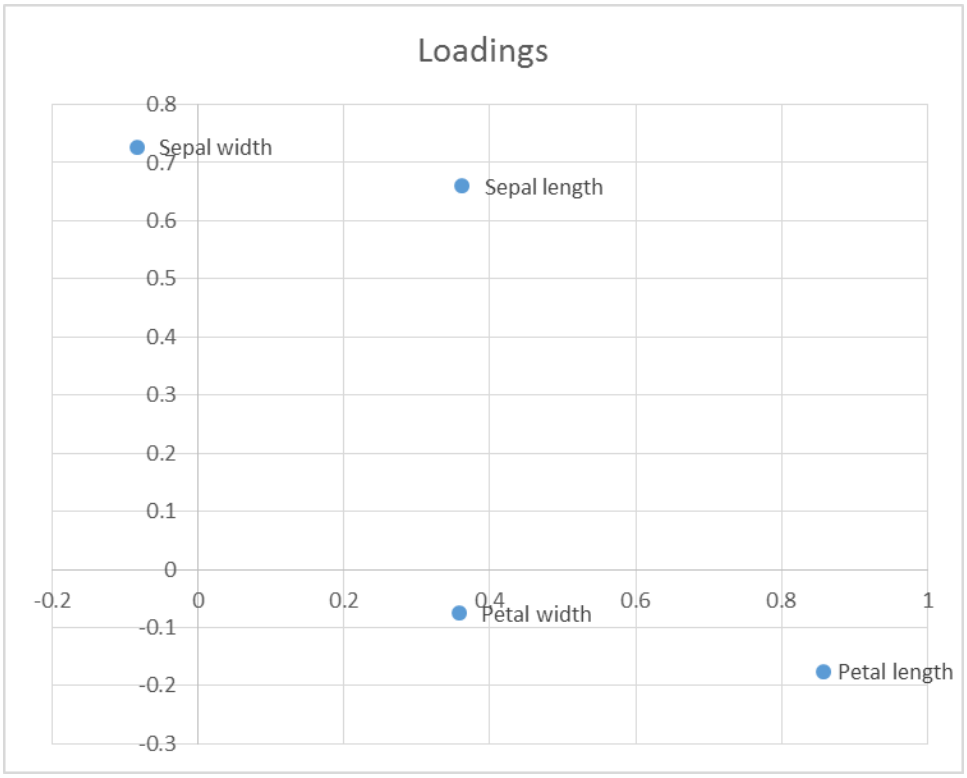


Fig. 3.59. Loadings plot of PCA of the iris dimensions.

Scores plot shows that the results for setosa are different from versicolor and virginica. However, although the centers of the scores for versicolor and virginica are different there is some overlapping of the scores.

Next, Mahalanobis distances between the three iris types were calculated. These plots are shown in Fig. 3.60. They clearly indicate that setosa might be easily distinguished from versicolor and virginica but distinction between versicolor and virginica contain some overlapping of data.

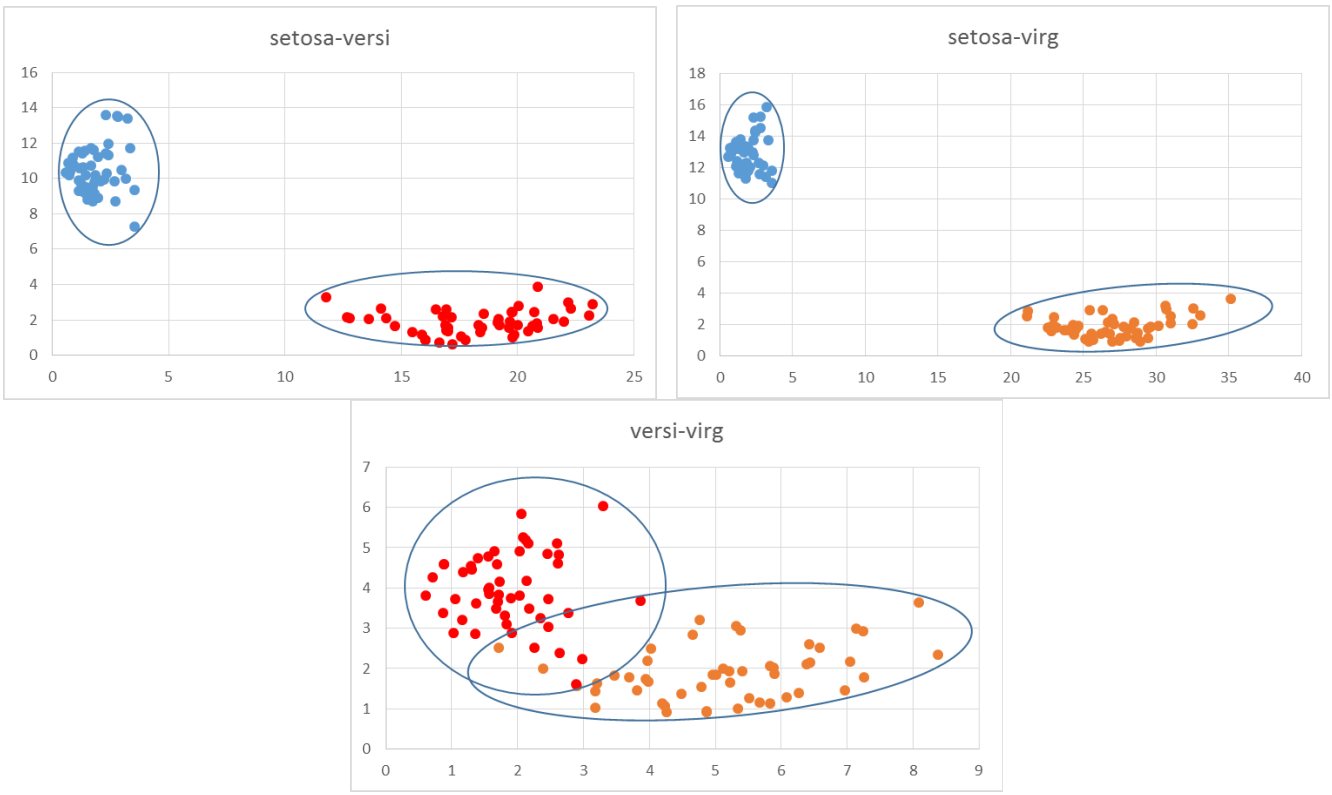


Fig. 3.60. Mahalanobis distances between three types of irises.

The total data contain training and test sets of data, 25 measurements each. The PCA was carried separately on training and test sets and compared together in Fig. 3.61.

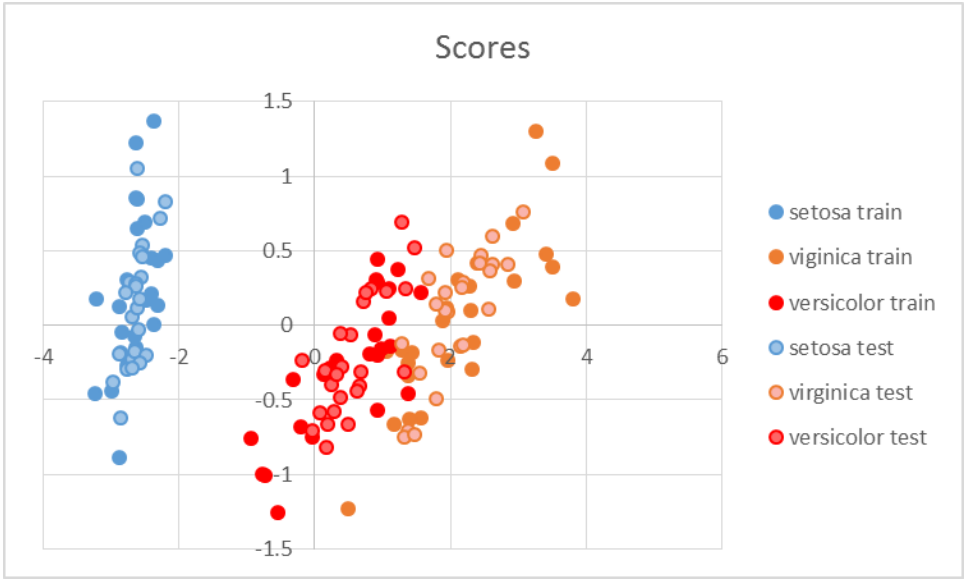


Fig. 3.61. Comparison of scores of the training and test sets of three types of irises.

Three plots in Fig. 3.60 can also be displayed on one 3D plot, Fig. 3.62.

Three plots in Fig. 3.60 can also be displayed on one 3D plot, Fig. 3.62, where setosa is clearly distinct from other classes.

This comparison shows clearly that versicolor and virginica sets are indistinguishable confirming that they follow the same model.

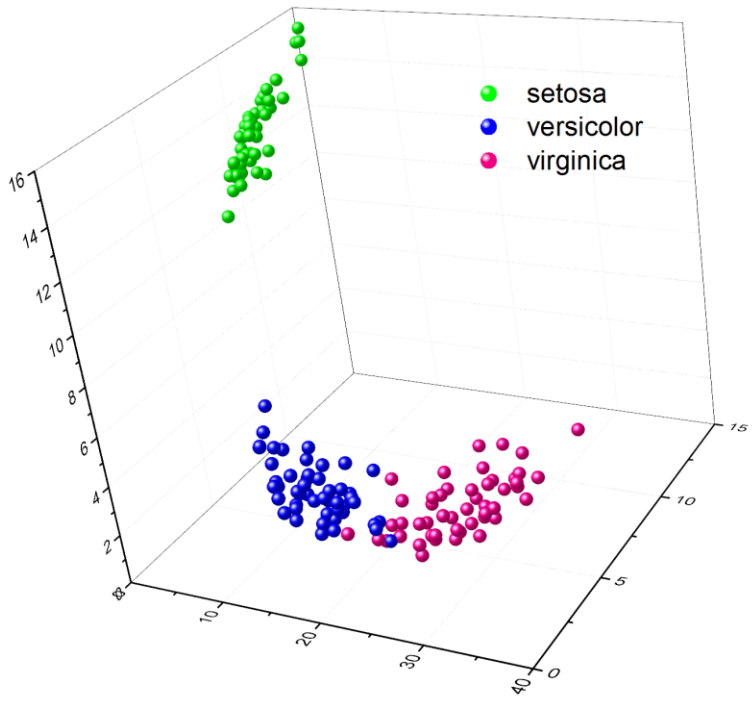
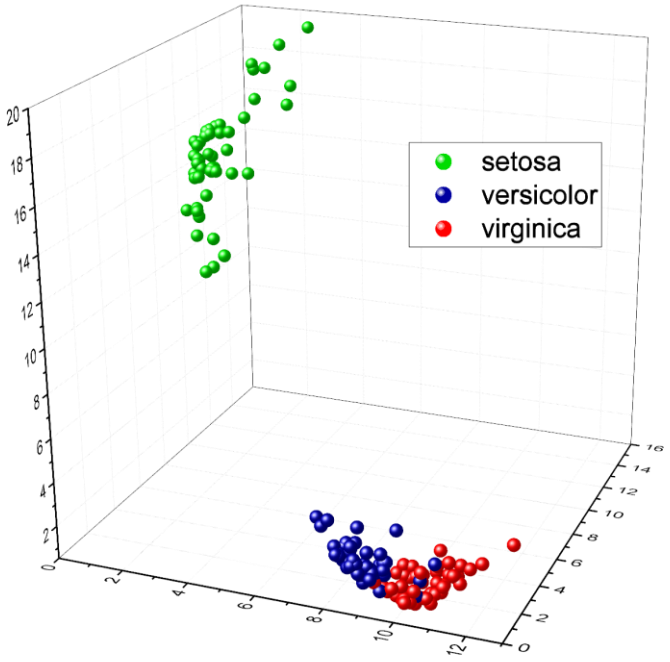


Fig. 3.62. 3D plot of the Mahalanobis distances.

Finally, applications of SIMCA method to distinguish (simcatest.m) to distinguish classes is displayed in Fig. 3.63. This analysis confirms that versicolor and virginica displays some overlapping, similarly to the methods presented above.


setosa
versicolor
virginica
20
18
16
14
12
10
8
6
4
2
0
2
4
6
8
10
12
16
14
12
10
8
6
4
2
0

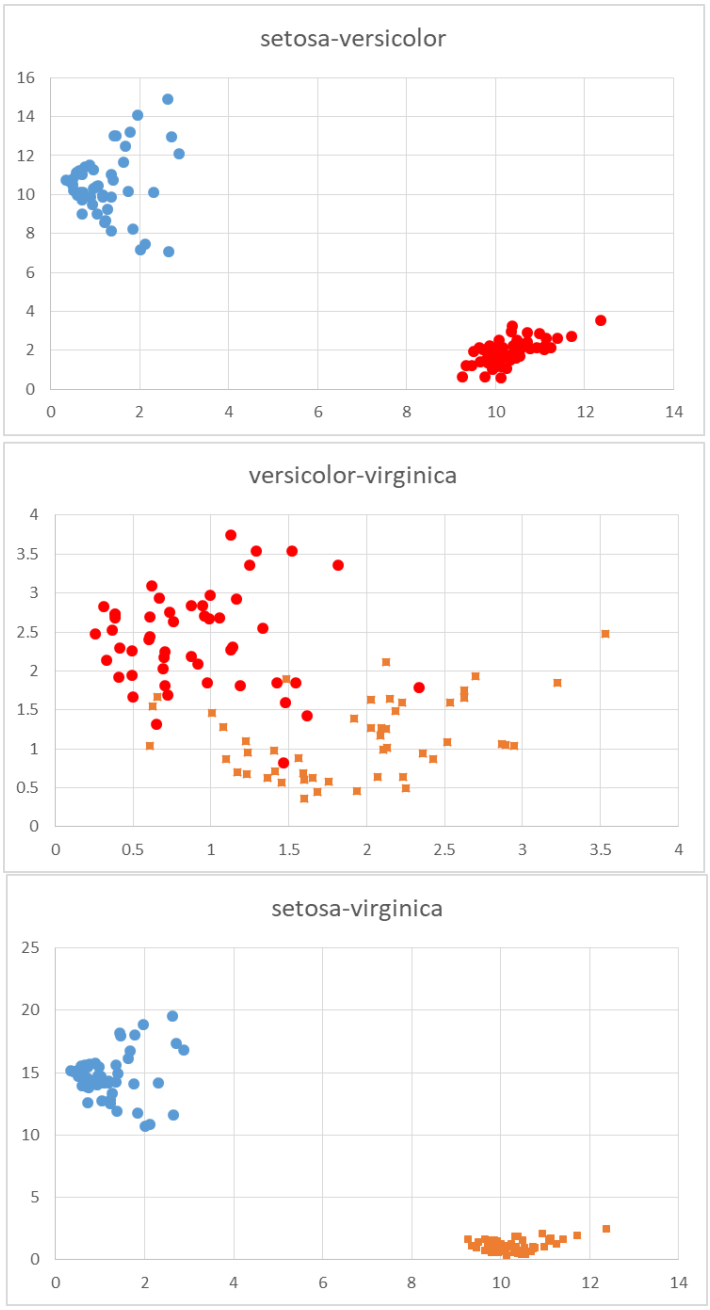


Fig. 3.63. Example of application SIMCA to distinguish three types of irises, 3D and 2D plots.

Exercise 3.10.

This example presents comparison of fresh, F, and stored, S, turnips (vegetable also called swede or rutabaga). To determine if they can be distinguished their extracts were analyzed by gas chromatography. The area under eight peaks was measured for 7 fresh and 7 stored turnips. It is presented in data file Xdata.m and in Ex3-10.xlsx. Is it possible to classify the data in two classes F and S using GC? Apply SIMCA to test samples, one fresh and one stored presented in Xtest.m. Can they be classified to F and S classes?

First, the data were transformed by taking logarithms and the classical PCA performed on standardized data. The scores and loadings plots are shown in Fig. 3.64. It is clear that the score for the point #7 is an outlier which should be removed from the analysis. The data file after removal of the point F #7 are in XS1.m.

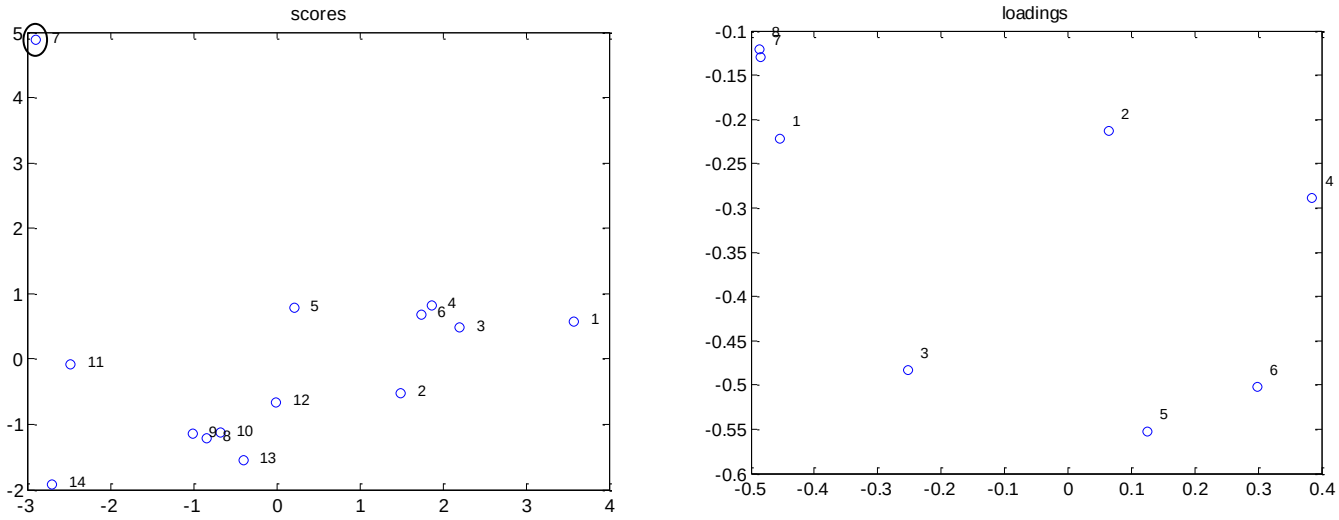


Fig. 3.64. Scores and loadings plots for the first two PCs.

Next, the PCA was performed on the new data set. The scores and loadings are displayed in Fig. 3.65.

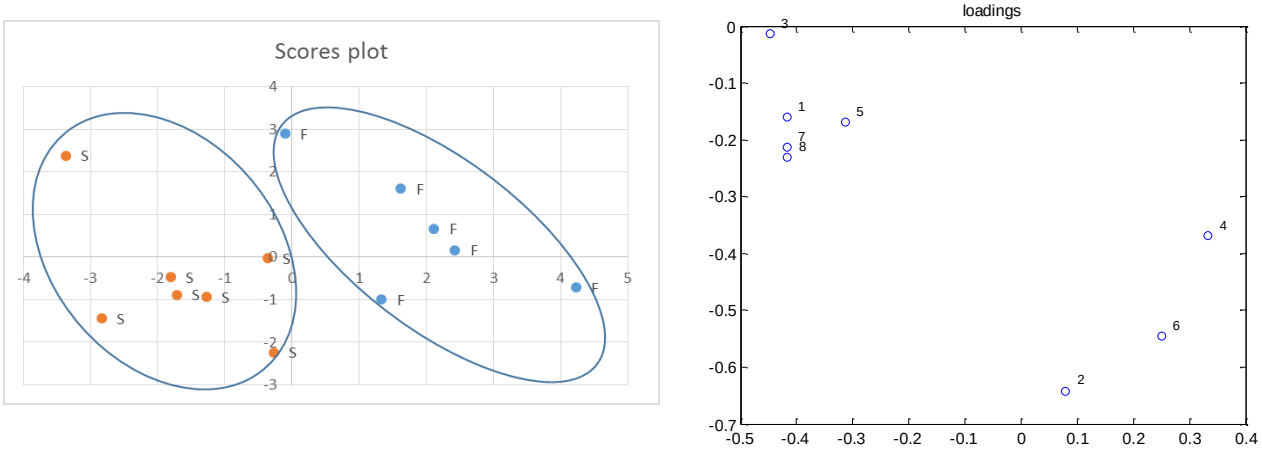


Fig. 3.65. Scores and loadings plots for the first two PCs without an outlier.

From Fig. 3.65 it is visible that scores for fresh and stored turnips form two groups.

Analysis using SIMCA was performed using simcatest.m which calls simca.m (using data without an outlier). The plot of class distances from class F and class S is displayed in Fig. 3.66. Although there are no obvious outliers or samples belonging to two classes, the sample #2 is slightly dubious and could be removed from the model. In all cases samples lying close to two classes should be carefully checked to find the possible reasons of their behavior. To check if the

model predictions are correct distances from the test set (F and S sample) were calculated as well. They are shown in Fig. 3.66 as larger squares. There is no doubt in classifying these samples as F and S, respectively.

The result obtained after removal of point #2 is displayed in Fig. 3.67. As above, test set distances were also added. The two classes are well separated and chromatographic method can be used to distinguish between fresh and stored product.

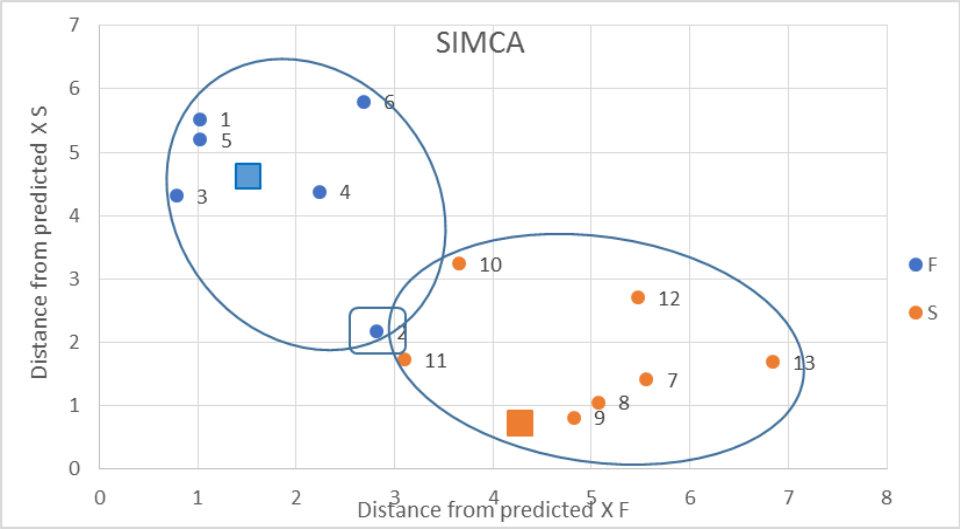


Fig. 3.66. Class distances calculated using SIMCA; the test set – squares

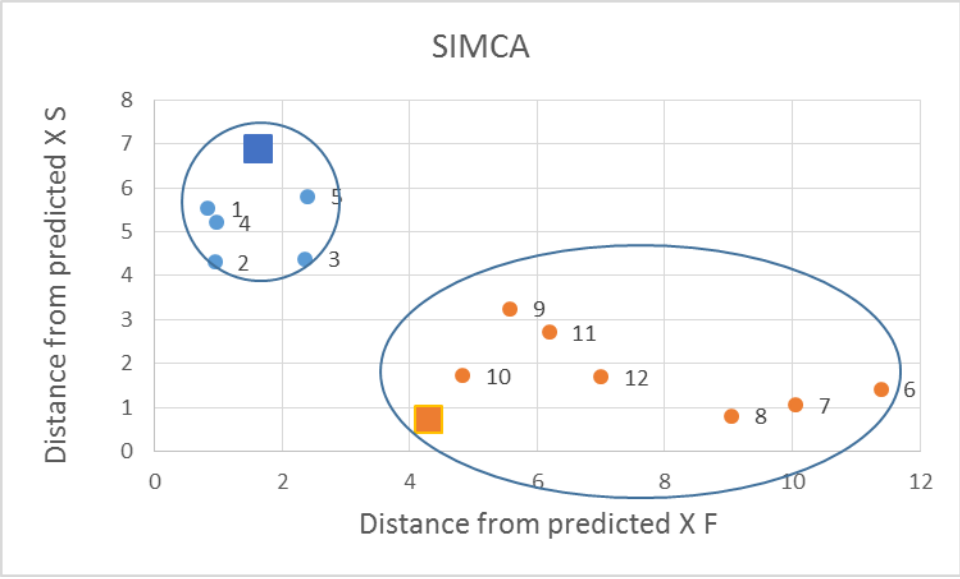


Fig. 3.67. Results of SIMCA analysis after removal of point #2; the test set - squares.

4 Calibration

The analysis presented above uses only the measurements, e.g. spectra, and does not allow to determine the concentrations.

To determine concentrations, one uses a **training data set** (i.e. **calibration** set) for which concentrations are known for several spectra. Usually, mixture of species are used as it is not always possible (or desirable) to work with pure components. Habitually, training sets give relatively good auto-predictions as the model was constructed using these data sets. The next step is to test the quality of predictions using another independent **test data set**. This set contains other mixtures with known concentrations and is used to determine their concentrations using knowledge from the training set. It is normal that the predictions of the training set are worse than auto-prediction for the training set. Determination of the quality of prediction from the test set is called **validation**. If only training set is available then cross-validation for this set can be used.

Finally, validated or cross-validated model is applied to the unknown data samples to determine the concentrations.

There are two main methods which allow determination of concentrations for multivariate analysis: Principal Components Regression, PCR, and Partial Least Squares, PLS. They will be presented in the subsequent chapters. These methods base the predictions of concentrations on changes in the data, not absolute values of absorbances (like in the classical models). If the concentrations of components change in the same way, e.g. using dilution, this method will detect only one component. The wavelengths chosen might be selected randomly and there is no limit on the number of wavelengths. The advantage of these methods is that they can be used in analysis of very complex mixtures since only knowledge of constituents of interest is required.

It should be added that the above methods assume linear relations between the measured signal and concentrations.

5 Principal Components Regression, PCR

Principal Components Regression, PCR, uses principal components analysis, PCA, in analysis of the experimental data that is distribution of the data matrix into scores and loadings. As we have already seen scores and loadings are abstract matrix quantities but PCR uses regression, also called transformation or rotation, to **convert principal component scores into concentrations**. PCR might be used not only in spectroscopy but to calibrate other properties, for example the drug activity to molecular parameters of the drug, material properties to its structural parameters, etc. This method first uses the PCA to determine scores and loadings and then scores are regressed against concentrations.

Matrix **X(***I*×*J*), in spectroscopy, for multicomponent mixture of compounds, might be calculated knowing the concentrations **C**(*I*×*K*) and spectra of individual components **S**(*K*×*J*) where *K* is the number of absorbing components. Using the Beer's law one can write, see Eq. (3.2):

$$\mathbf{X} = \mathbf{C}\mathbf{S} + \mathbf{E} \tag{3.2}$$

from which predicted spectra of individual components may be easily obtained:

$$\hat{\mathbf{S}} = \left(\mathbf{C}'\mathbf{C}\right)^{-1}\mathbf{C}'\mathbf{X} \tag{5.1}$$

In the PCR the PCA is used. The aim of the PCR calibration procedure is to determine unknown concentrations, **C**, from the spectra, **X**. First, the **training set** for which concentrations are known is used to build up the model.

Distribution of matrix **X**(*I*×*J*) into scores **T**(*I*×*R*) and loadings **P**(*R*×*J*) for the number of principal components $R \le K$, **X** = **T P'**, see Eq. (3.6), allows for the calculation of the scores **T** and a **rotation** or **transformation matrix R**(*R*×*K*):

$$\mathbf{C} = \mathbf{T}\mathbf{R} + \mathbf{E} \tag{5.2}$$

This equation presents regression of concentration to scores. Matrix **R** can be obtained from the above equation if the concentrations are known, using pseudoinverse, Eq. (2.17), of matrix **T**:

$$\mathbf{R} = \left(\mathbf{T}'\mathbf{T}\right)^{-1}\mathbf{T}'\mathbf{C} \tag{5.3}$$

When the number of PC equals number of compounds, *R* = *K*, rotation matrix is square, **R**(*K*,*K*). Finally one can rearrange PCA analysis into:

$$\hat{\mathbf{X}} = \mathbf{T}\mathbf{P}' = \left(\mathbf{T}\mathbf{R}\right)\left(\mathbf{R}^{-1}\mathbf{P}'\right) = \hat{\mathbf{C}}\hat{\mathbf{S}} \tag{5.4}$$

which gives Eq. (3.2), where

$$\hat{\mathbf{C}} = \mathbf{T}\mathbf{R} \quad \text{and} \quad \hat{\mathbf{S}} = \mathbf{R}^{-1}\mathbf{P}' \tag{5.5}$$

It is evident that the predicted **concentrations are related to the scores**, **T**, and the **spectra to the loadings**, **P**. With the help of the rotation matrix **R**, calculated from Eq. (5.3), one can obtain the predicted concentrations, $\hat{\mathbf{C}}$, and the individual spectra, $\hat{\mathbf{S}}$, Eq. (5.5).

In calculations of the predicted spectra, $\hat{\mathbf{S}}$, using Eqs. (5.1) and (5.5) for the raw data very similar values are obtained. However, smaller errors are obtained using Eq. (5.1) when data were centered. In the case of standardized data Eq. (5.5) gives standardized spectra while Eq. (5.1) gives correct values. When the concentrations in the training set are known the spectra of individual components may be calculated using $\hat{\mathbf{X}}$ predicted for *R* principal components:

$$\hat{\mathbf{S}} = \left(\mathbf{C}'\mathbf{C}\right)^{-1}\mathbf{C}'\hat{\mathbf{X}} \tag{5.6}$$

In conclusion, spectra of the analyzed species can be obtain using rotation matrix, Eq. (5.5) and Eqs. (5.1) or (5.6).

5.1 Determination of the concentration from the analytical spectra

Using the training set matrices: **X** and **C** PCA is performed and matrices **T**, **P**, and **R** calculated, then one can apply the PCR to determine unknown concentrations $\boldsymbol{c}_u$ (vector) or $\mathbf{C}_u$ (matrix) from the spectrum $\boldsymbol{x}_u$ or spectra $\mathbf{X}_u$ of unknown sample. First, scores vector $\hat{\boldsymbol{t}}_u$ or matrix $\hat{\mathbf{T}}_u$ must be determined from the loadings, **P**, of the training set, **X** = **TP'**, then from the relation $\mathbf{X}_u = \hat{\mathbf{T}}_u\mathbf{P}'$ $\mathbf{X}_u = \hat{\mathbf{T}}_u\mathbf{P}'$ and taking into account that loadings are orthonormal $\mathbf{P}'\mathbf{P} = \mathbf{I}$:

$$\hat{\mathbf{T}}_u = \mathbf{X}_u\mathbf{P} \tag{5.7}$$

Next, the unknown concentration(s) $\hat{\mathbf{C}}_u$, are determined using the calculated scores, $\hat{\mathbf{T}}_u$ and the rotation matrix **R** of the training set, Eq. (5.3):

$$\hat{\mathbf{C}}_u = \hat{\mathbf{T}}_u\mathbf{R} = \mathbf{X}_u\mathbf{P}\mathbf{R} \tag{5.8}$$

Of course, if the concentrations $\mathbf{C}_u$ are known these data are used as a test set.

5.2 Model validation: self-prediction

The simplest way of validation is to compare the predicted parameters (concentrations) with the known values used in the PCR analysis. This process is called **self or auto-prediction**. The comparison of the experimental $\mathbf{c}_i$ and auto-predicted, Eq. (5.5), $\hat{\mathbf{c}}_i$ concentrations allows to determine root mean square error of self-prediction concentrations, RMS_{sp}:[3]

$$\mathrm{RMS}_{\mathrm{sp}}(k,r) = \sqrt{\frac{\sum_{i=1}^{I}\left(c_{i,k} - \hat{c}_{i,k}\right)^2}{I - r - f}} \tag{5.9}$$

where the RMS error is calculated, for each compound, *k*, separately, and number principal components *r*, *I* is the number of spectra, and *f* the loss of degrees of freedom due to preprocessing; $f = 0$ for the raw data, $f = 1$ for the centered data (mean calculated), and $f = 2$ for the standardized data (mean and standard deviation calculated).

This error might be also presented in % as:

$$\mathrm{RMS}_{\mathrm{sp}}(k,r)(\%) = 100\frac{\mathrm{RMS}_{\mathrm{sp}}(k,r)}{\bar{c}_k}\,\% \tag{5.10}$$

where $\bar{c}_k$ is the average concentration of the component *k*:

$$\bar{c}_k = \frac{\sum_{i=1}^{I} c_{ik}}{I} \tag{5.11}$$

This is the simplest method of testing. However, the models are built using all the spectra in the training data set and then the same training set is predicted. Although the statisticians do not

recommend using of self-prediction to determine the number of principal components, the analytical chemists might know or have a good intuitive feeling of the noise level and they might be able to interpret the self-predictive errors in a physically meaningful manner.[3]

5.3 Quality of the prediction of the measurement matrix X

The quality of modeling may be tested comparing the experimental and calculated spectra (or more general measurements), Eq. (3.14), to determine the **residual sum of squares**, $\mathrm{RSS_X}$, from the experimental, **X**, and the calculated, $\hat{\mathbf{X}}$, matrices using *r* principal components:

$$\mathrm{RSS_X}(r) = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{i,j} - \hat{x}_{i,j}\right)^2 \tag{5.12}$$

The root mean square is obtained by division by the number of degrees of freedom $I \times J - r$ but because $I \times J$ is large and *r* small this number is often used as $I \times J$:

$$\mathrm{RMS_X}(r) = \sqrt{\frac{\mathrm{RSS_X}}{I\,J}} \tag{5.13}$$

This value might be presented as percentage (for the raw data):

$$\mathrm{RMS_X}(r)(\%) = 100\,\frac{\mathrm{RMS_X}(r)}{\bar{x}} \tag{5.14}$$

5.4 Model validation: cross-validation

The basis of the cross-validation were discussed in Section 3.3. It is sometimes called "leave-one-out cross-validation" because from the training set containing *I* samples (spectra) one sample *i* is left out leaving $I - 1$ data set and then the model is used to predict concentrations of the removed sample *i*. This process is repeated for all the data sets. If the data set contains replicate spectra of the same sample each pair of replicates should be left out together. Such an analysis is usually carried out for centered data but might be also used for other preprocessing methods. The procedure used is as follows:

1) From the original data set **X**(*I*,*J*) and **C**(*I*,*K*) remove one data set $i = 1$ obtaining **XX**(I–1,*J*) and **CC**(I-1,*K*) for $i = 2 \ldots I$. Choose number of PCs in the model (the process will be repeated for different number of PCs, from small to large)
2) Perform PCA on matrix **XX**(I-1,*J*) with one deleted row *i*, obtain scores, **T**, and loadings, **P**. Obtain matrix **R** for the above data using standard regression technique: $\mathbf{R} = (\mathbf{T'T})^{-1}\,\mathbf{T'C}$, Eq. (5.3).
3) Calculate predicted scores for the deleted data row, $\mathbf{x}_1$: $\hat{\mathbf{t}}_1 = \mathbf{x}_1\,\mathbf{P}$, Eq. (3.20), where the loadings **P** were obtained using data without this row.
4) Calculate cross-validated concentrations for the removed data ${}^{cv}\hat{\mathbf{c}}_i = \hat{\mathbf{t}}_i\,\mathbf{R}$, Eq. (5.5).
5) Repeat these operations for $i = 2, \ldots I$
6) Calculate PRESS and $\mathrm{RMS_{cv}}$ (see below)
7) Increase number PCs by one and repeat all the operations.

The parameter PRESS (Predicted Residual Error Sum of Squares) for r PCs from 1 to R is calculated using an analog of Eq. (3.23) for concentrations, but in the examples below it is calculated as an average value, divided by the number of samples I:

$$\mathrm{PRESS}_r = \frac{\sum_{i=1}^{I}\sum_{k=1}^{K}\left(c_{i,k} - {}^{r,cv}\hat{c}_{i,k}\right)^2}{I} \tag{5.15}$$

The root mean square of the concentrations of the component k and r principal components, obtained using cross-validation, $\mathrm{RMS}_{\mathrm{cv}}(k,r)$ is:

$$\mathrm{RMS}_{\mathrm{cv}}(k,r) = \sqrt{\frac{\sum_{i=1}^{I}\left(c_{i,k} - {}^{r,cv}\hat{c}_{i,k}\right)^2}{I}} \tag{5.16}$$

This error is divided by the number of samples I because each sample in the original data set represents an additional degree of freedom no matter how many PCs were used or how the data were preprocessed.[3]

5.5 Model validation: test set

The best method of validation is to check the model used above to determine concentrations of an independent test set for which concentrations are known. It can be achieved in calibration if calibration data are divided in two parts, one as a **training set** containing I spectra and another as a **test set** containing L spectra. The training set is used to construct the multivariate model and the test set is used to test prediction of the "unknown" (test) data.

The standard deviations of the concentrations (root mean square) in the test set are determined using equation similar to Eq. (5.16):

$$\mathrm{RMS}_{\mathrm{test}}(k,r) = \sqrt{\frac{\sum_{i=1}^{L}\left(c_{i,k} - {}^{r,test}\hat{c}_{i,k}\right)^2}{L}} \tag{5.17}$$

$\mathrm{RMS}_{\mathrm{test}}(k,r)$ is determined for concentration k using r principal components and ${}^{r,test}\hat{c}_{i,k}$ are the calculated concentrations for the test set.

Below several exercises will be present to allow better understanding of the theory shown above. The data are in the Matlab files. The solutions were obtained using PCA and PCR programs included. Their solutions are in the corresponding Excel files in different folders. The readers should try to repeat the calculations using these data and the programs (see Section 9.1) to see if the same results as those included in Excel files were obtained.

Exercise 5.1.

10 spectra of the mixtures of two compounds were recorded at eight wavelengths.[3] They are presented in Table 5.1, Fig. 5.1, and in the files Ex5-1.xlsx and Xdata.m, containing matrix **X**(10×8), that is 10 spectra at 8 wavelengths. The corresponding concentrations are presented in Table 5.2, and in the files Ex5-1.xlsx and Cdata.m containing **C**(10×2), that is concentrations of two species for 10 mixtures.

Determine the number of principal components, PC, in the spectra, carry out principal components regression, PCR, and determine the estimated concentrations and the spectra of two compounds. Use self and cross-validation and data centering.

Table 5.1. Spectra of two compounds measured at 8 wavelengths and 10 compositions of two compounds.

	~Wavelength							
Spectrum number	1	2	3	4	5	6	7	8
1	0.07	0.124	0.164	0.171	0.184	0.208	0.211	0.193
2	0.349	0.418	0.449	0.485	0.514	0.482	0.519	0.584
3	0.63	0.732	0.826	0.835	0.852	0.848	0.877	0.947
4	0.225	0.316	0.417	0.525	0.586	0.614	0.649	0.598
5	0.533	0.714	0.75	0.835	0.884	0.93	0.965	0.988
6	0.806	0.979	1.077	1.159	1.249	1.238	1.344	1.322
7	0.448	0.545	0.725	0.874	1.005	1.023	1.064	1.041
8	0.548	0.684	0.883	0.992	1.166	1.258	1.239	1.203
9	0.8	0.973	1.209	1.369	1.477	1.589	1.623	1.593
10	0.763	1.019	1.233	1.384	1.523	1.628	1.661	1.625

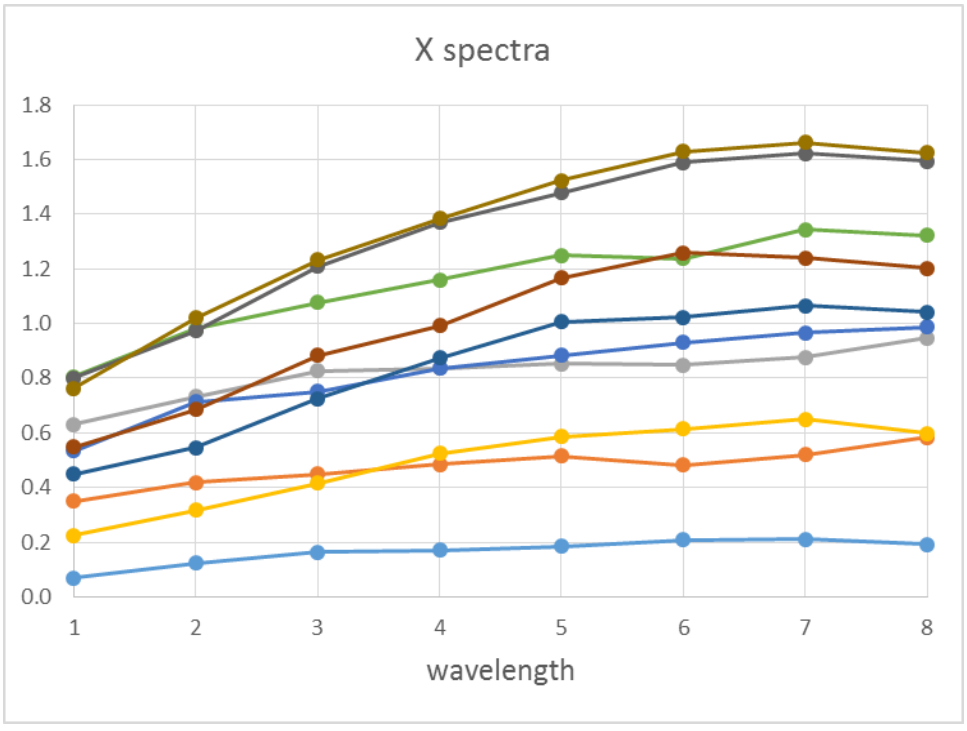


Fig. 5.1. Ten spectra registered at 8 wavelengths for Exercise 5.1.

Table 5.2. Concentrations of two compounds A and B (in ppm) for 10 mixtures corresponding to the spectra in Table 5.1.

No	A	B
1	1	1
2	2	5
3	3	9
4	4	2
5	5	6
6	6	10
7	7	3
8	8	4
9	9	8
10	10	7

Performing the PCA on matrix **X** gives the results shown in Table 5.3.

Table 5.3. Results for the PCA on matrix **X** using data centering in Exercise 5.1.

PC	λ_i	%	Cumulative %
1	11.13000	98.278%	98.278%
2	0.18169	1.604%	99.883%
3	0.00604	0.053%	99.936%
4	0.00453	0.040%	99.976%
5	0.00272	0.024%	100.000%
	sum		
	11.32498		

The PCA analysis shows PC1 contributes 98.3% and PC2 1.6% tom the total variance of **X**. This would suggest that only one PC is important. However, we know that there are two true components in the analysis and small size of PC2 does not mean that it should be completely ignored.

The cross-validation analysis was also carried out. The plots of PRESS and RMS_{cv} for different numbers of PCs using program PCRcross.m are illustrated in Fig. 5.2 and the values in Table 5.4. It is clearly seen that these parameters decrease up to PC2 and then stay constant. This suggests that two PCs should be used. PC1 and PC2 explain 99.88% of the total variance and in the subsequent PCR two PCs will be used in the further analysis.

Using PCR for two PCs gives the concentrations shown in Table 5.5.

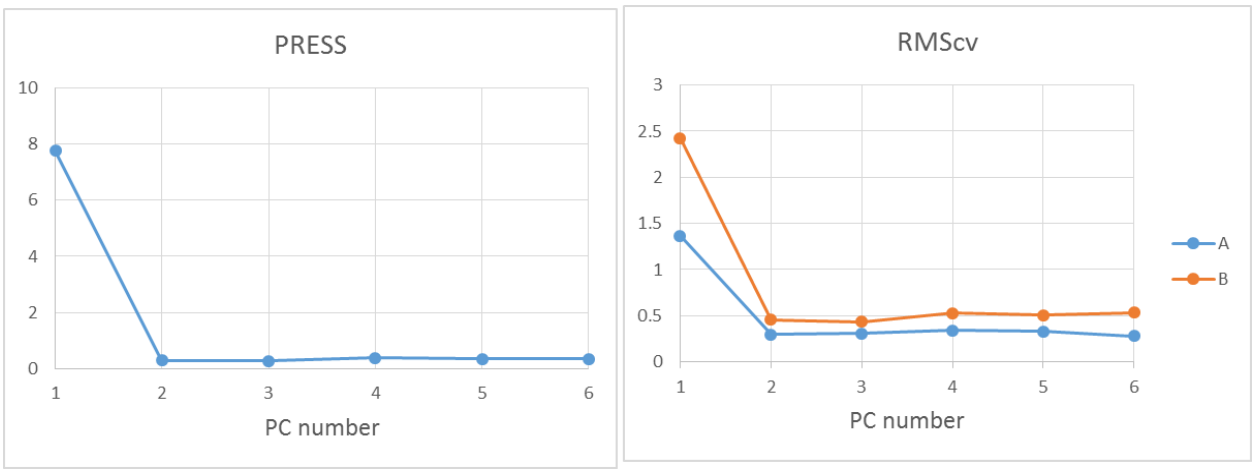


Fig. 5.2. Plots of the parameters PRESS and RMS_{cv} for different number of principal components in Exercise 5.1.

Table 5.4. Results of cross-validation of concentrations for data in Exercise 5.1.

	RMScv		**PRESS**
r	1	2	
1	1.3664	2.4263	7.7542
2	0.29891	0.45529	0.2966
3	0.30681	0.43435	0.2828
4	0.34152	0.52676	0.3941
5	0.32829	0.50482	0.3626
6	0.27901	0.53348	0.3624

Table 5.5. Concentrations of two components A and B auto-predicted using PCR analysis.

A	B
1.0387	1.0936
1.9512	4.9378
3.1893	8.8220
4.2119	1.5234
4.6033	6.6373
5.9997	9.8885
6.8775	3.1888
7.9304	4.1468
9.4014	7.4790
9.7966	7.2827

The values of RMS_{sp} (self-prediction of the training set) are 0.258 and 0.396 for species A and B separately. They are smaller than those obtained for cross-validations, RMS_{cv}, 0.299 and 0.455,

Table 5.4. Such a behavior is expected as cross-validation uses prediction of the removed concentrations

The obtained results are relatively close to the experimental values. The absolute, (C_{calc} - C_{exp}), and relative. (C_{calc} - C_{exp})/C_{exp} 100%, errors of the auto-predicted concentrations are shown in Fig. 5.3.

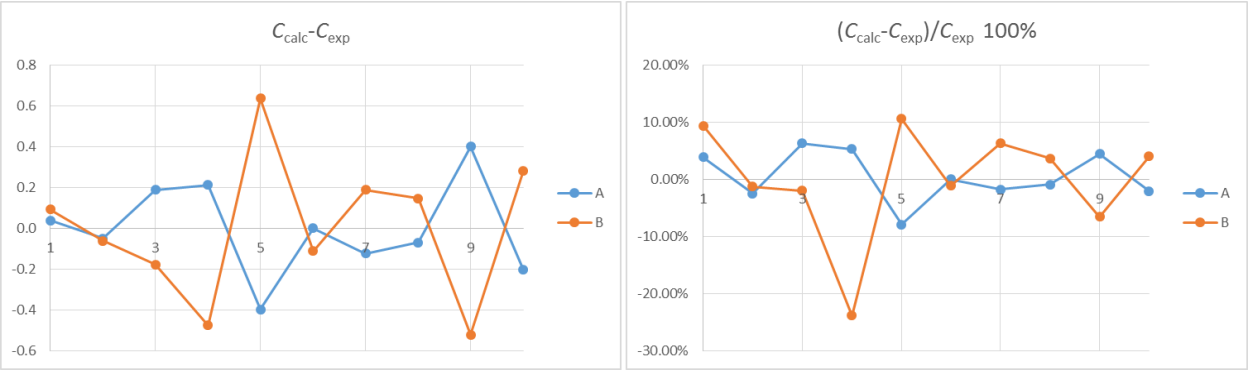


Fig. 5.3. Absolute and relative errors of the auto-predicted concentrations in Exercise 5.1.

It can be noticed that concentrations of species A are determined with smaller error (maximal value 7.9%) than species B (maximal error 23.8%). In general all the errors except one for species B are ≤10%. This might be expected as the second PC2 contributes only 1.6% to the total sum of eigenvalues.

Comparison of predicted and experimental concentrations is illustrated in Fig. 5.4. Good linearity is found with larger deviations for species B (smaller determination coefficient R^2).

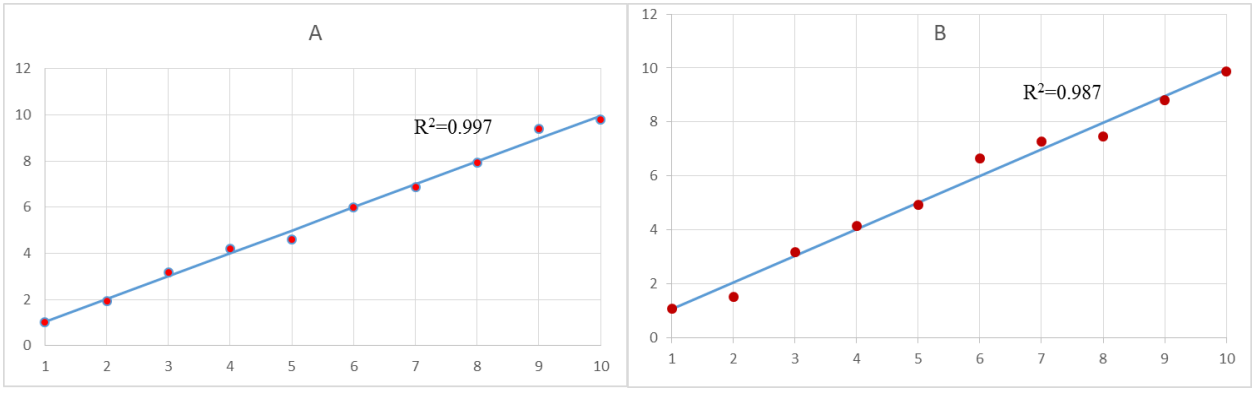


Fig. 5.4. Plots of the predicted (calculated) versus experimental concentrations for species A and B.

Finally, the spectra of both components are calculated using Eq. (5.6). They are shown in Fig. 5.5.

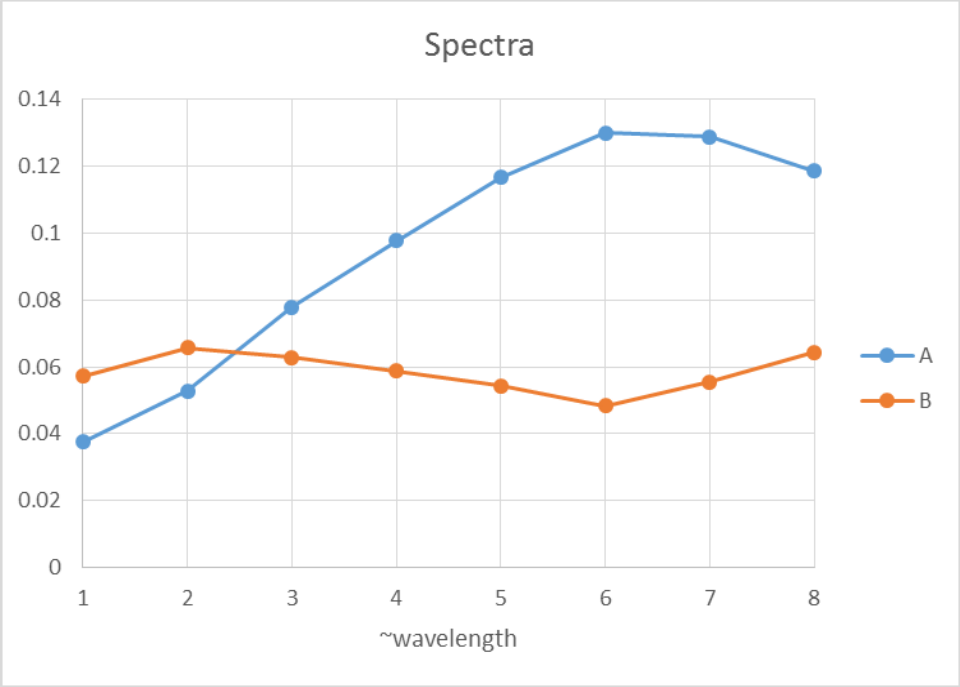


Fig. 5.5. Spectra of two components obtained from the PCR analysis.

The PCR analysis and cross-validation allowed us to determine that there are two PCs in agreement with the number of species. PCR also allowed for auto-prediction of the concentrations and determination of the spectra of pure components.

Exercise 5.2.

The spectroscopic analysis was performed to determine concentrations. First, the training set was measured, it contained 9 spectra measured at 100 wavelengths, **X**(9,100), in file Xdata.m, obtained for different concentrations of two compounds, **C**(9,2), in Cdata.m. Next the validation set was measured, Xtest(5,100), file XVtest.m, for 5 different sets of concentrations, Ctest(5,2), file CVtest.m. All the original data are in the Excel file Ex5-2.xlsx. The spectra **X** are the same as in Exercise 3.5 and Fig. 1.1.

Using PCR on the training set determine number of principal components, spectra of these components, concentrations using auto-validation. Next, use the knowledge from the training set to predict concentrations of the validation set and compare them with the experimental values. These data were prepared from the assumed spectra **S** of two compounds and concentrations[6] using Beer's law **X** = **C S**, Eq. (3.2), by adding Gaussian noise $N(0,1)*0.02$ to each absorbance, where $N(0,1)$ are the normally distributed random numbers with mean of 0 and standard deviation of 1. This presents the added random noise.

The PCA analysis for matrix **X** was already presented in Exercise 3.5 and it indicates the presence of two PCs in agreement with two known concentrations. Further analysis using cross-validation shows that the parameters RMS_{cv} and PRESS using PCRcross.m that these values decrease up to $r = 2$ and they stay relatively constant, see Fig. 5.6. These results are also shown in Table 5.6. The root-mean square of self-prediction of concentrations, RMS_{sp}, shows similar behavior, Fig. 5.7. Self-prediction analysis was used to predict concentrations and it was carried

out using PCR program PCRtest.m. These concentrations for centered data and two PCs are presented in Table 5.7.

Table 5.6. Dependence of the root mean-square errors of self-prediction and cross-validation of the concentrations on the number of PCs for the training set.

	RMSsp		RMScv		PRESS
PC	A	B	A	B	
1	0.03651	0.17483	0.04214	0.19972	0.04167
2	0.00399	0.0065	0.00528	0.00901	0.00011
3	0.00405	0.00689	0.00542	0.00897	0.00011
4	0.00453	0.00743	0.00546	0.00898	0.00011
5	0.0044	0.00827	0.00537	0.00932	0.00012

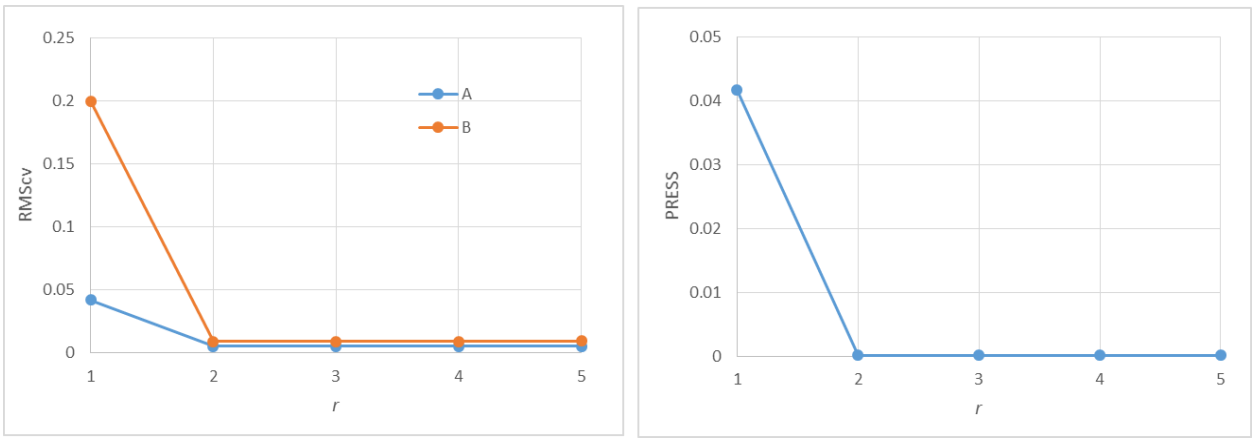


Fig. 5.6. Dependence of the root mean square of concentrations, RMS_{cv} and PRESS as a function of the number of principal components used in calculations.

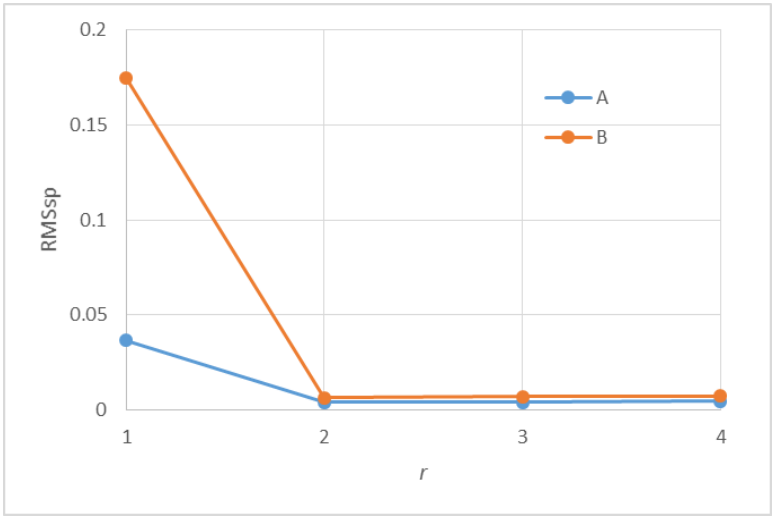


Fig. 5.7. Dependence of the root mean square of self-predicted concentrations, RMS_{sp} as a function of the number of principal components used in calculations.

Table 5.7. Comparison of the experimental, $c_{i,k}$ and self-predicted $\hat{c}_{i,k}$ concentrations obtained using PCR method for two PCs and the centered data, together with their RMS_{sp} (absolute and relative) errors, Eqs. (5.9) and (5.10).

$c_{i,k}$		$\hat{c}_{i,k}$	
0.1	0.9	0.106	0.896
0.2	0.85	0.194	0.852
0.3	0.55	0.298	0.553
0.4	0.35	0.400	0.357
0.5	0.5	0.498	0.492
0.6	0.6	0.602	0.604
0.7	0.25	0.703	0.241
0.8	0.4	0.799	0.403
0.9	0.1	0.900	0.103
	RMS_{sp}	0.0040	0.0065
	RMS_{sp} %	0.80%	1.30%

The relative standard deviations, RMS_{sp}, of the two self-predicted concentrations are 0.8% and 1.3%, respectively.

PCR method allows also to determine spectra of two components from the loadings, **P**, and the rotation matrix, **R**, using Eq. (5.5) or using Eq. (5.6). The calculated spectra (symbols) are compared with the theoretical (lines) used in simulations of the training spectra in Fig. 5.8. It is evident that the PCR analysis approximates well the spectra of two compounds present in the analysis.

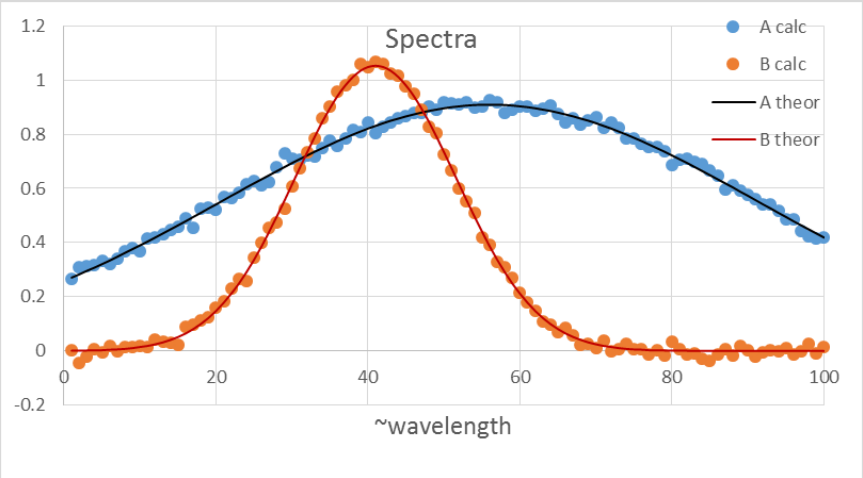


Fig. 5.8. Comparison of the theoretical (lines) and the calculated (symbols) spectra using Eq. (5.6) from the PCR analysis.

The “unknown” concentrations of the validation test were calculated using the information from the training set in program PCRpred.m. The validation spectral data are in the file XVtest.m

and the concentrations in CVtest.m. The known and calculated concentrations using centered data are shown in Table 5.8.

Table 5.8. Comparison of the known analytical and predicted (calculated) concentrations of the validation set using centered data.

Analytical concentrations		Predicted concentrations	
A	B	A	B
0.720	0.560	0.717	0.564
0.300	0.500	0.306	0.492
0.250	0.350	0.256	0.340
0.600	0.600	0.598	0.595
0.350	0.650	0.360	0.640
	RMS_{test}	0.0051	0.0065
	RMS_{tes} %	1.1%	1.2%

Comparison of the RMS values of the concentrations is displayed in Table 5.9.

Table 5.9. Standard deviations of the concentrations of two components A and B obtained using auto-prediction and cross-validations for the training set and for the validation set.

	CA	CB
RMS_{sp} self-prediction	0.003993	0.006501
RMS_{sp} self-prediction %	0.8%	1.3%
RMS_{cv} cross-validation	0.005285	0.009008
RMS_{cv} cross-validation %	1.1%	1.8%
RMS_{test} validation	0.005114	0.006541
RMS_{test} validation %	1.1%	1.2%

Analysis of the above example was carried out assuming two principal components corresponding to two species present in the sample. Auto-prediction shows relative standard deviations of 0.8% and 1.3%. Cross-validation shows little larger errors of 1.1% and 1.8%. However, validation using a validation data set shows errors very close to the auto-prediction of 1.1% and 1.2%. These results show that using PCR the unknown concentrations as well as the spectra of two components can be easily obtained.

Exercise 5.3.

To determine the concentrations during the reaction: A + B $\rightarrow$ C the flow injection analysis (FIA) was used and the UV/VIS spectra were recorded as function of time. In order to determine the concentration of three components, A, B, and C, a series of 25 three component mixtures were prepared as a training set (under the conditions where there was no reaction).[3] The measured spectra, **X**, are in Fig. 5.9 and data file Xdata.m and Ex5-3.xslx. The corresponding concentrations, **C**, are in Cdata.m and Table 5.10. Carry out PCA and PCR analysis and determine auto-predicted concentrations. Use this knowledge to determine dependence of the

concentration as a function of time in spectra measured during reaction presented in file XVtest.m. Use data centering.

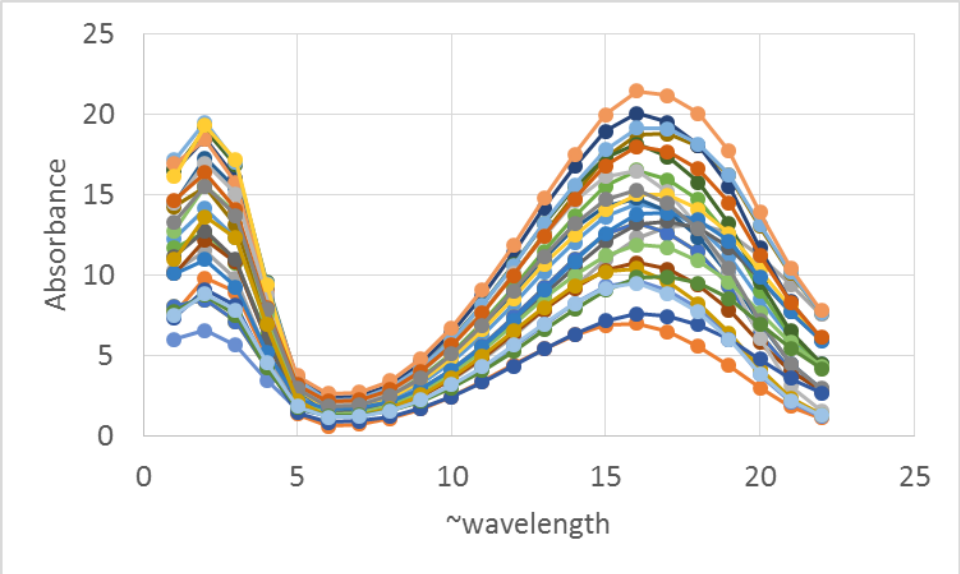


Fig. 5.9. Training spectra of 25 mixtures of three compounds measured at 22 wavelengths in Exercise 5.3.

Table 5.10. Concentration of three components A, B, and C, in 25 mixtures.

	A (mM)	B (mM)	C (mM)
1	0.276	0.090	0.069
2	0.276	0.026	0.013
3	0.128	0.026	0.126
4	0.128	0.153	0.041
5	0.434	0.058	0.126
6	0.200	0.153	0.069
7	0.434	0.090	0.041
8	0.276	0.058	0.041
9	0.200	0.058	0.098
10	0.200	0.121	0.126
11	0.357	0.153	0.098
12	0.434	0.121	0.069
13	0.357	0.090	0.126
14	0.276	0.153	0.126
15	0.434	0.153	0.013
16	0.434	0.026	0.098
17	0.128	0.121	0.013

18	0.357	0.026	0.069
19	0.128	0.090	0.098
20	0.276	0.121	0.098
21	0.357	0.121	0.041
22	0.357	0.058	0.013
23	0.200	0.026	0.041
24	0.128	0.058	0.069
25	0.200	0.090	0.013

The PCA analysis was carried up to 6 PCs. The results for centered data are shown in Table 5.11.

Table 5.11. Results for the PCA on training spectra, matrix **X,** using data centering in Exercise 5.3.

PC	λ_i	%	Cumulative %
1	3755.30	87.49%	87.49%
2	353.91	8.25%	95.74%
3	182.49	4.25%	99.99%
4	0.29	0.01%	100.00%
5	0.08	0.00%	100.00%
6	0.04	0.00%	100.00%
	sum		
	11.32498		

The PCA indicates that there are two or three PCs in the analyzed spectra. However,the presence of three PCs is confirmed by cross-validation. PRESS and RMS_{cv} were calculated using PCRcross.m program. They are displayed in Fig. 5.10 where decrease of these parameters is observed up to three PCs and after there are no important changes in their values.

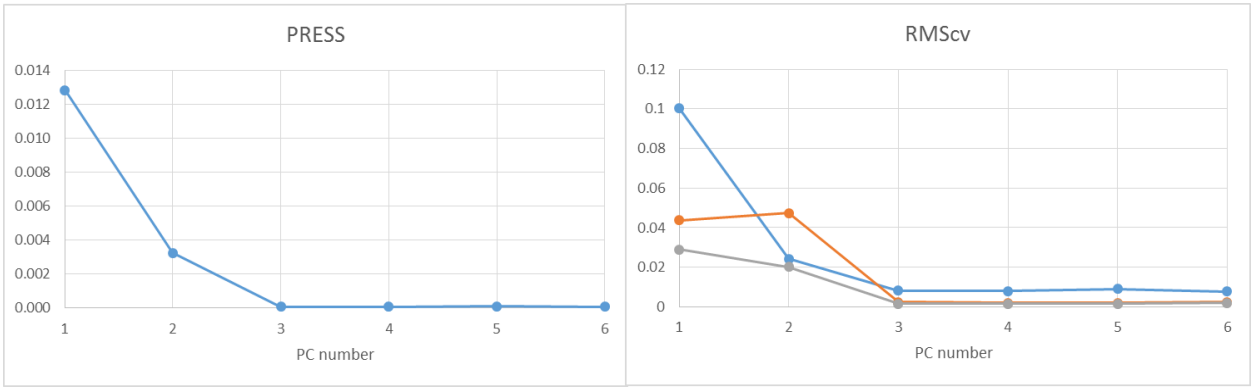


Fig. 5.10. Dependence of PRESS and RMS_{cv} on number of PC used.

Using three PCs and PCR program PCRtest.m concentrations for the training set were self-predicted. The values of RMS_{sp} for the self-prediction are displayed in Table 5.12.

Table 5.12. Results of the self-prediction for Exercise 5.3.

	A	B	C
RMSsp	0.0074	0.0023	0.0014
RMSsp %	2.7%	2.6%	2.0%

More detailed analysis of errors was obtained by plotting the absolute C_{calc} - C_{exp} and relative errors (C_{calc} - C_{exp})/ C_{exp} 100% of individual concentrations in Fig. 5.11.

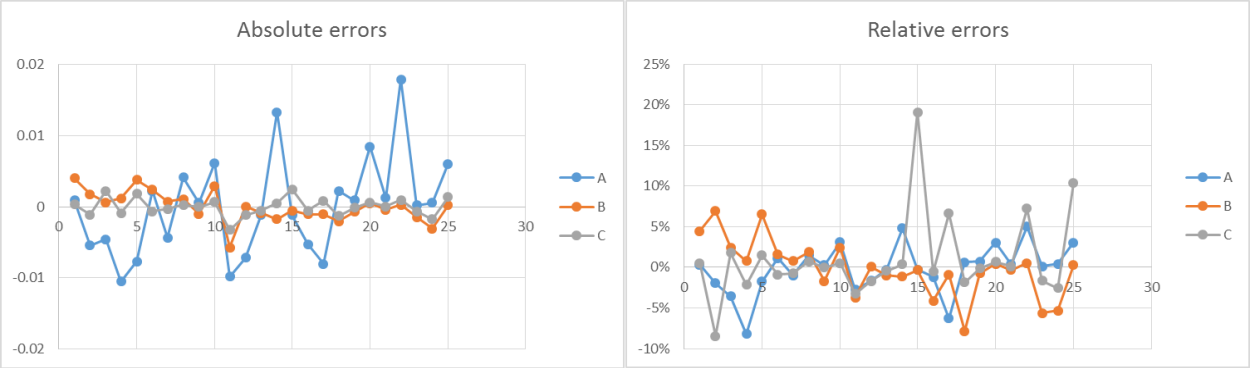


Fig. 5.11. Absolute and relative errors of the concentrations in the training set using PCR analysis (auto-prediction).

Absolute errors of absorbance are the largest for species A and all the relative errors are lower than 10% except for one set for species C. The good correlation between self-predicted and experimental concentrations for three species shown in Fig. 5.12. A very good linearity between these values (large values of R^2) was found confirming good quality of self-predictions.

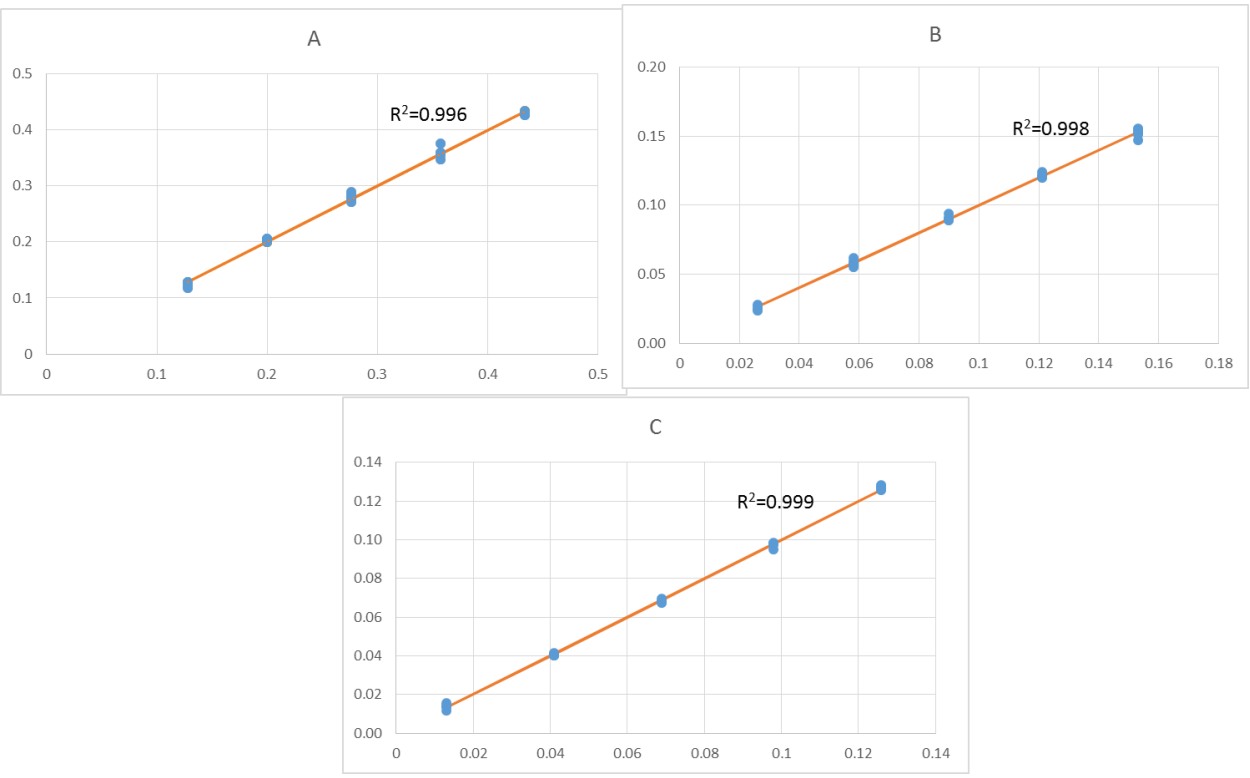


Fig. 5.12. Dependence of the calculated versus experimental concentrations for three components A, B, and C, determined using PCR analysis.

PCR analysis permits determination of the individual spectra of the absorbing components. They are presented in Fig. 5.13. It can be noticed that there are small difference in the relative absorbances of these compounds which makes determination more difficult. However, as the analysis is carried out at 22 wavelengths relatively good linearity between experimental and calculated concentrations is observed.

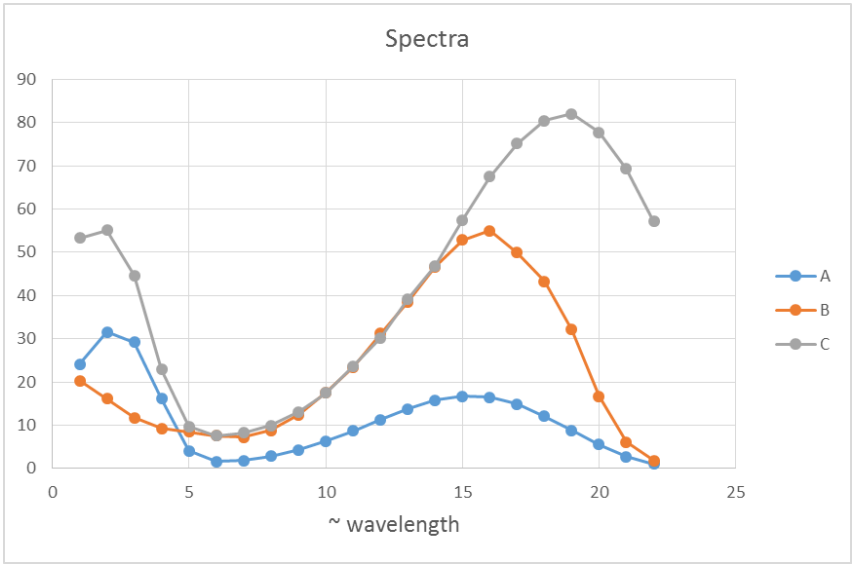


Fig. 5.13. Calculated spectra of three components, determined using the PCR analysis.

PCR analysis was applied to the spectra of reacting species in XVtest.m using program PCRpred.m and the concentrations of three components as a function of time were determined. They are presented in Fig. 5.14.

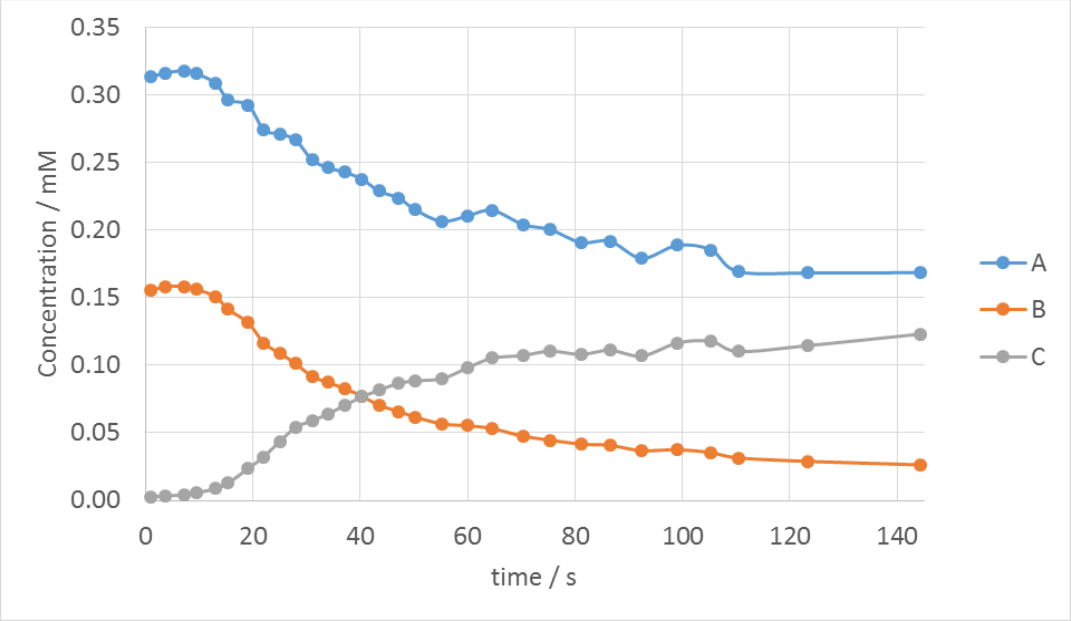


Fig. 5.14. Dependence of the concentrations of three reacting species on time using PCR analysis.

Obtained concentrations allow for the determination of the kinetics of chemical reaction taking place in solution by fitting the concentrations obtained above to the kinetic equations.

Exercise 5.4.
Nine training spectra of three compounds measured at 101 wavelengths were acquired. They are displayed in Fig. 5.15 and in files Xdata.m and Ex5-4.xlsx. The corresponding concentrations are included in Table 5.13 and file Cdata.m. Five spectra for the validation set are in file XVtest.m and the concentrations in Table 5.13 and file CVtest.m.

Carry out PCR analysis. Use auto-prediction, cross-validation and validation methods. Determine how precisely validation set is estimated. Use data centering,

First, the PCA was performed on the training set using centered data. The results are in Table 5.14. They show that the second PC contributes only 1.54% and the third only 0.21%. These results suggest that there is only one important PC in the data.
However, the cross-validation analysis shows that PRESS and RMS_{sp} decrease until the third PC and then they stay practically constant. These results are displayed in Fig. 5.16. They suggest that although the first PC explains 98.16% of the dependence three PCs should be used in the analysis of the experimental data, in agreement with three chemical components used.

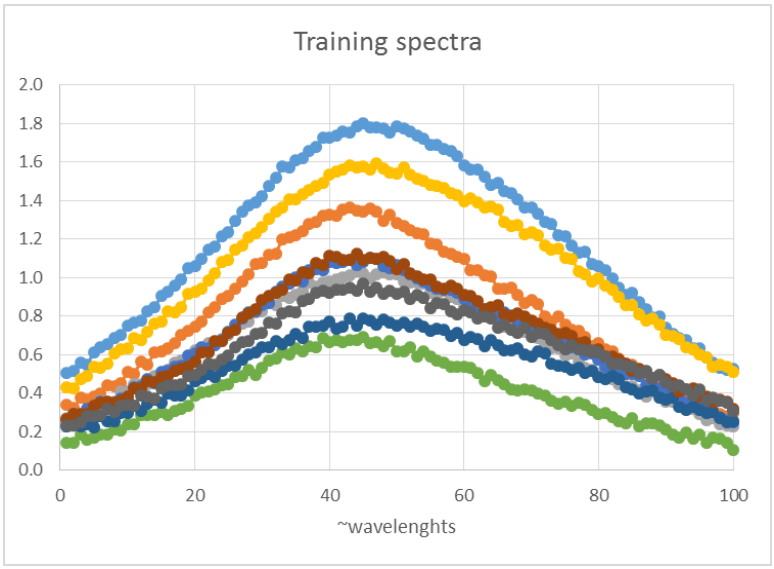


Fig. 5.15. Nine training spectra of three compounds.

Table 5.13. Concentrations of the training and validation sets.

	No	A	B	C
Calibration (training)	1	0.90	0.85	0.15
	2	0.80	0.35	0.25
	3	0.70	0.25	0.10
	4	0.60	0.95	0.15
	5	0.50	0.45	0.20
	6	0.40	0.15	0.15
	7	0.20	0.55	0.10
	8	0.30	0.65	0.25
	9	0.10	0.75	0.20
Validation	10	0.55	0.70	0.25
	11	0.75	0.50	0.15
	12	0.25	0.30	0.20
	13	0.35	0.80	0.15
	14	1.45	0.20	0.20

Table 5.14. PCA on the training set for centered data.

PC	λ_i	%	Cumulative %
1	56.9159	98.16%	98.16%
2	0.8915	1.54%	99.70%
3	0.1226	0.21%	99.91%
4	0.0282	0.05%	99.96%
5	0.0233	0.04%	100.00%
	sum		
	57.9815		

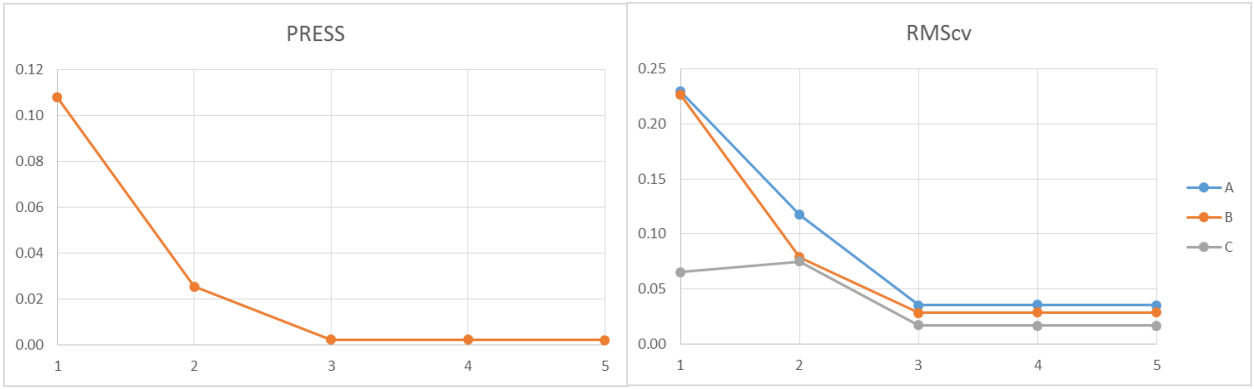


Fig. 5.16. The plots of the parameters PRESS and RMS_{cv} for cross-validation of the training set.

Self-prediction using the training set allows for calculation of concentrations and the spectra of three compounds. The RMS_{sp} of the auto-predicted concentrations for these compounds are shown in Table 5.15.

Table 5.15. RMS analysis of the auto-prediction in PCR analysis for centered data.

	A	B	C
RMS_{sp}	0.0175	0.0159	0.0058
RMSsp %	3.5%	2.9%	3.4%
Average concentration	0.50	0.55	0.17

The above analysis shows that the concentrations are predicted with the maximal error of 3.5% despite low contribution of the second and the third PC to the total variation. The plots of the self-predicted versus experimental concentrations are shown in Fig. 5.17. The correlation (R^2>0.99) is very good.

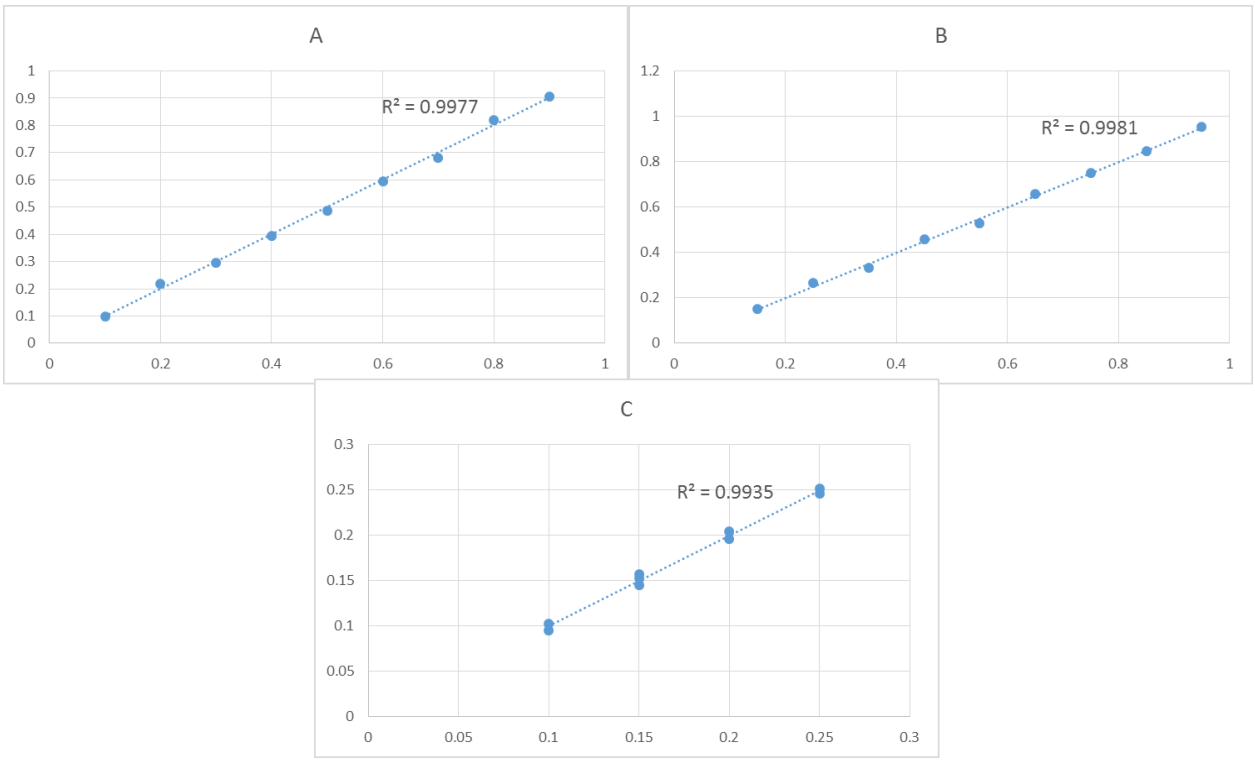


Fig. 5.17. Plots of the self-predicted versus experimental concentrations for three compounds present in the mixture.

The estimated spectra are shown in Fig. 5.18. Spectra for components A and B are estimated with low noise and the spectrum for C with larger noise. This effect is related to the fact that the average concentration of C is three times lower than that of components A and B.

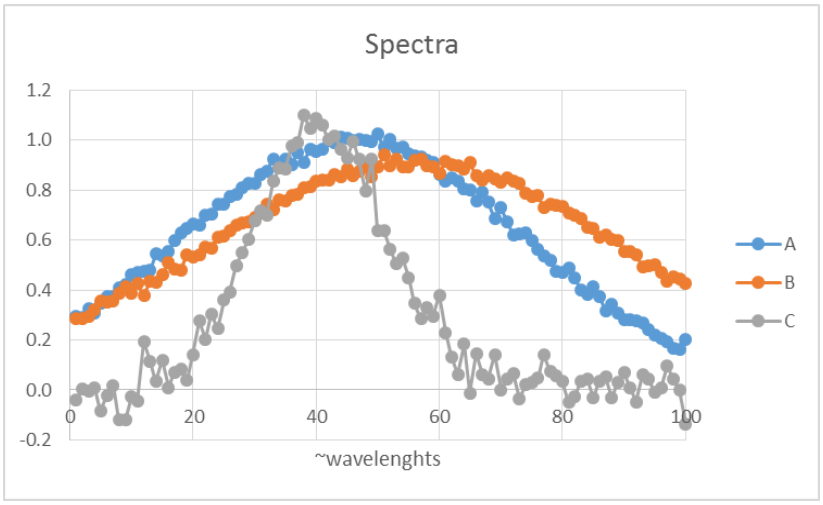


Fig. 5.18. Spectra of the three components obtained from the PCR analysis.

Finally, validation was carried out using validation data set and the loading **P** from the training set, Eqs. (5.7) and (5.8). The RMS_{test} results are shown in Table 5.16.

Table 5.16. RMS_{test} analysis of the concentrations in the validation set.

	A	B	C
RMS_{test}	0.01359	0.00927	0.00984
RMS_{test} %	2.0%	1.8%	5.2%
Mean concentration	0.67	0.50	0.19

These results show that the concentrations of species A and B were determined with the error of ~2% but those for species C with larger error of 5.2%. These errors are in agreement with much lower concentration of C in the mixture and low contribution off the PCs two and three. Details of all the calculations are in the file Ex1-9.xlsx.

Exercise 5.5.
Kinetics of the reaction $A \rightarrow B \rightarrow C$ was measured by registering spectra as functions of time.[6] First, spectra of the mixtures of these three components were measured under the conditions where there is no reaction. They are present in file Xdata.m and the corresponding concentrations in Cdata.m and in Ex5-5.xlsx. The training spectra are also displayed in Fig. 5.19.

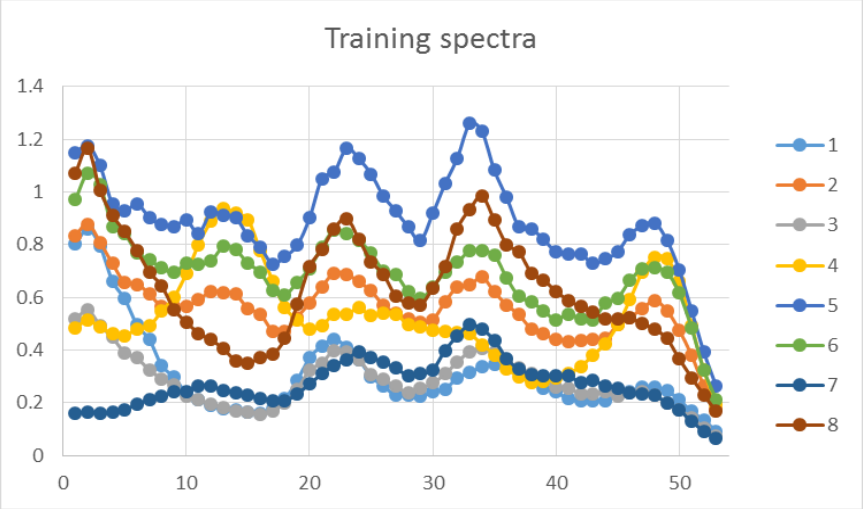


Fig. 5.19. Training spectra for different mixtures of three components.

Next, the spectra during the reaction were measured at times from zero to 21 s. They are in file XVtest.m and are displayed in Fig. 5.20. Cary out the PCR analysis and determine the concentrations as functions of time and the spectra of individual components.

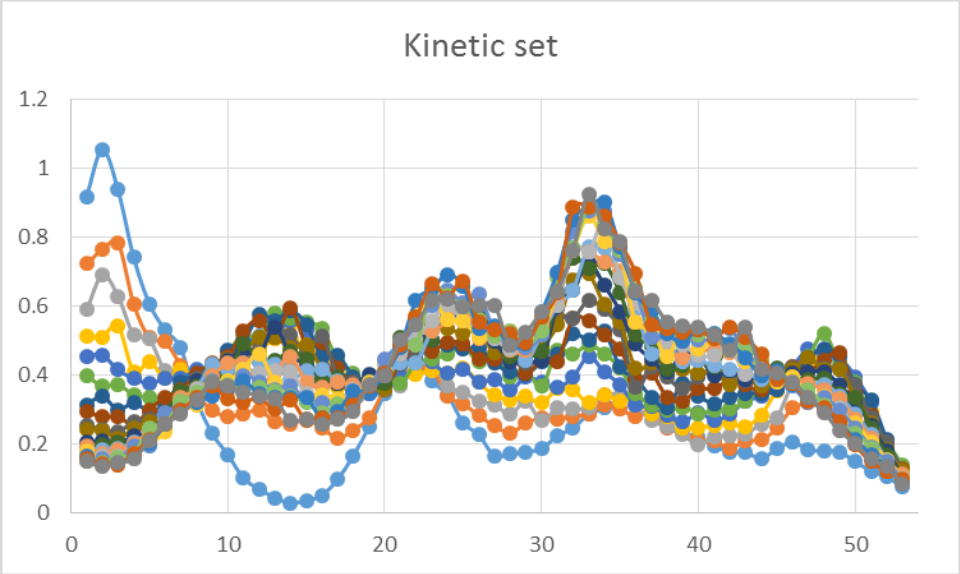


Fig. 5.20. Spectra measured at different times during chemical reaction.

Application of the PCA to the training data set gives the results shown in Table 5.17. They show the importance of two to three components, the third contributes 4.48% and the three components explain 99.88% of variation. The analysis using cross-validation is shown in Fig. 5.22 and shows that PRESS and RMS_{cv} parameters decrease down to 3 PCs and then they stay practically unchanged. The calculated spectra of the three components are displayed in Fig. 5.21. There is a good agreement between the calculated and theoretical spectra used to simulate the "experimental" spectra with noise.

Table 5.17. Results of the PCA for the training set.

PC	λ_i	%	Cumulative %
1	18.9118	82.076%	82.076%
2	3.07130	13.329%	95.405%
3	1.03170	4.478%	99.883%
4	0.01201	0.052%	99.935%
5	0.01053	0.046%	99.981%
6	0.00443	0.019%	100.000%
	sum		
	23.0417		

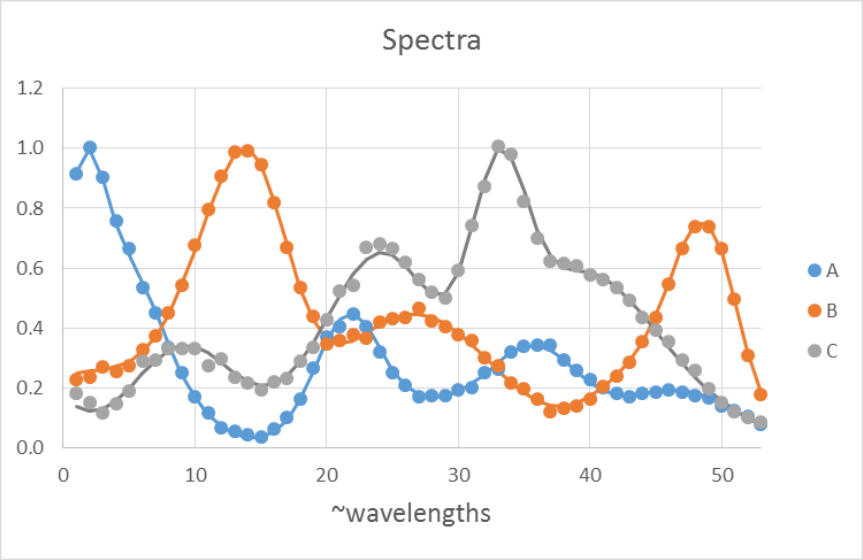


Fig. 5.21. Spectra of the three components in the training set; points – predictions, lines theoretical assumed in data preparation.

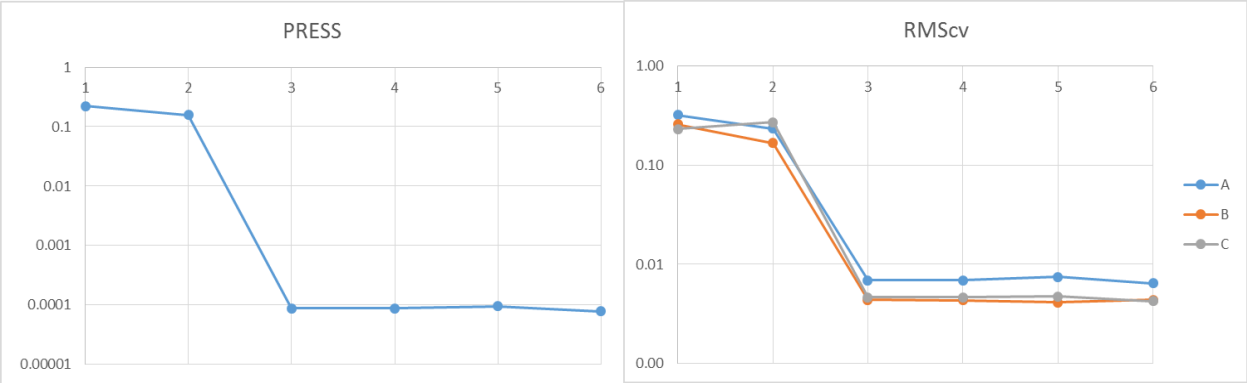


Fig. 5.22. PRESS and RMS_{cv} parameters as functions of the PC number obtained using cross-validation of training data set.

Next, loadings determined on training data were used to predict concentration of the test set containing unknown (validation) concentrations. They are shown in Fig. 5.23 and compared with the theoretical concentrations assumed for the simulation of spectra (with added noise). Very good agreement between the experimental and theoretical profiles was found. Analysis of the concentration profiles allows for the determination of the reaction kinetics spectroscopically.

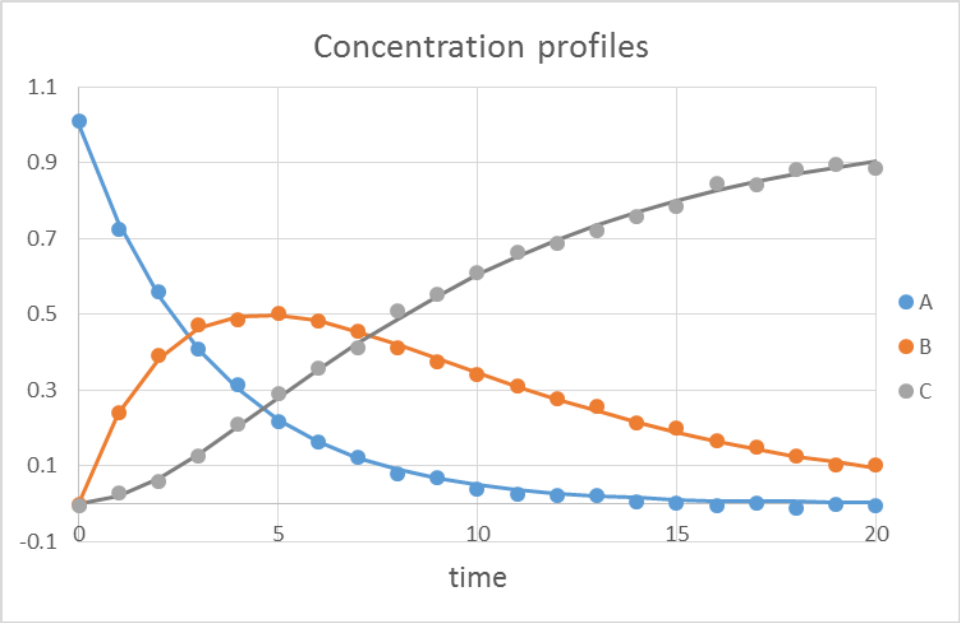


Fig. 5.23. Concentration spectra of the three components as function of time; points – calculated, lines - theoretical.

Exercise 5.6.

In order to analyze overlapping chromatograms a training set of known concentrations in Cdata.m and Table 5.18 was used and the obtained spectra are in Xdata.m and Fig. 5.24 (Gaussian noise was added). Next, the spectra as function of time during the chromatographic elution were recorded. They are in XVdata.m and Fig. 5.25. Using the PCR determine the chromatographic concentrations profiles vs. time.

Table 5.18. Concentrations of two components used in the training data set.

A	B
0.83	0.05
0.71	0.50
0.49	0.10
0.27	0.91
0.90	0.69
0.86	0.66
0.07	0.14
1.20	0.20

The PCA analysis is displayed in Table 5.19. It shows the presence of two PCs.

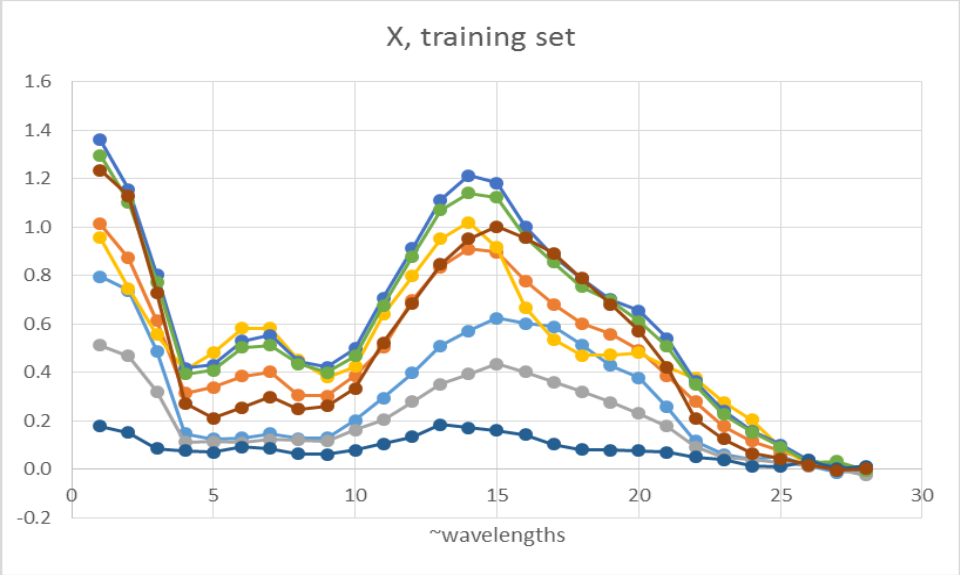


Fig. 5.24. Spectra obtained for the training set of data in Cdata.m.

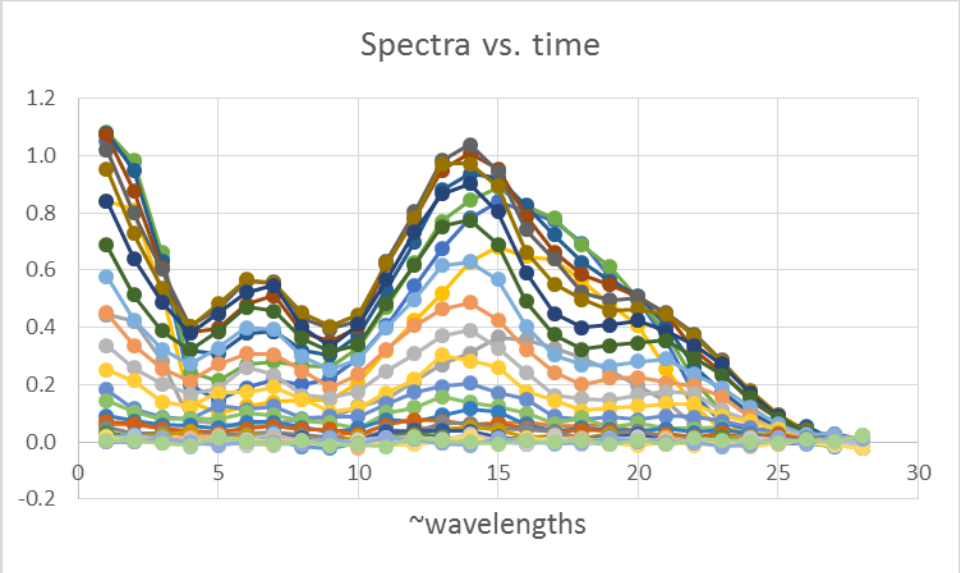


Fig. 5.25. Spectra recorded during the chromatographic analysis.

Table 5.19. Results of the PCA for the training data set and centered data.

PC	λ_i	%	Cumulative %
1	9.4637	93.234%	93.234%
2	0.6762	6.662%	99.896%
3	0.0041	0.040%	99.936%
4	0.0038	0.037%	99.973%
5	0.0027	0.027%	100.000%
	sum		
	10.1505		

This is also confirmed by the cross-validation. Both PRESS and RMS_{cv} decrease until PC = 2 and then stay constant or increase slightly, Fig. 5.26.

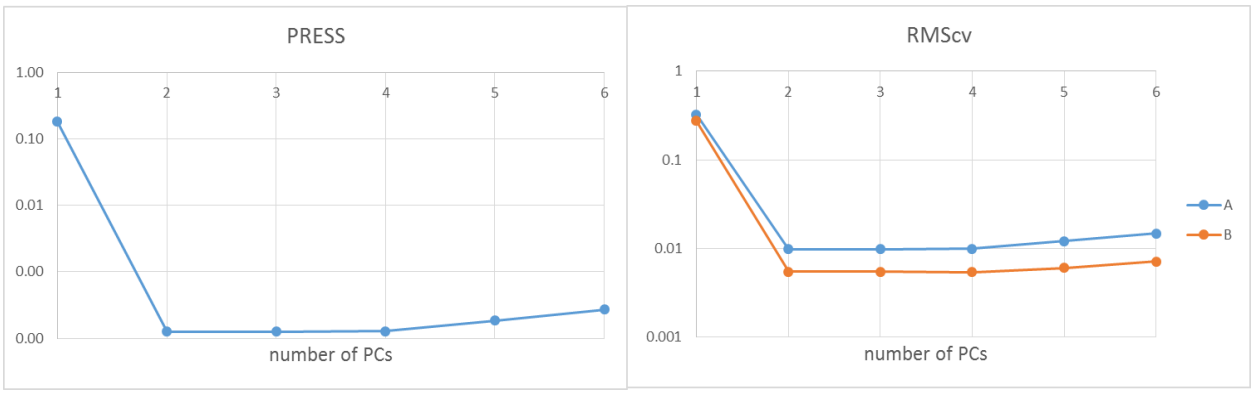


Fig. 5.26. Results of the cross-validation of the training set.

The PCR analysis permits determination of the spectra of two components present in the mixtures. The calculated spectra ($\hat{\mathbf{S}}$) are displayed in Fig. 5.27.

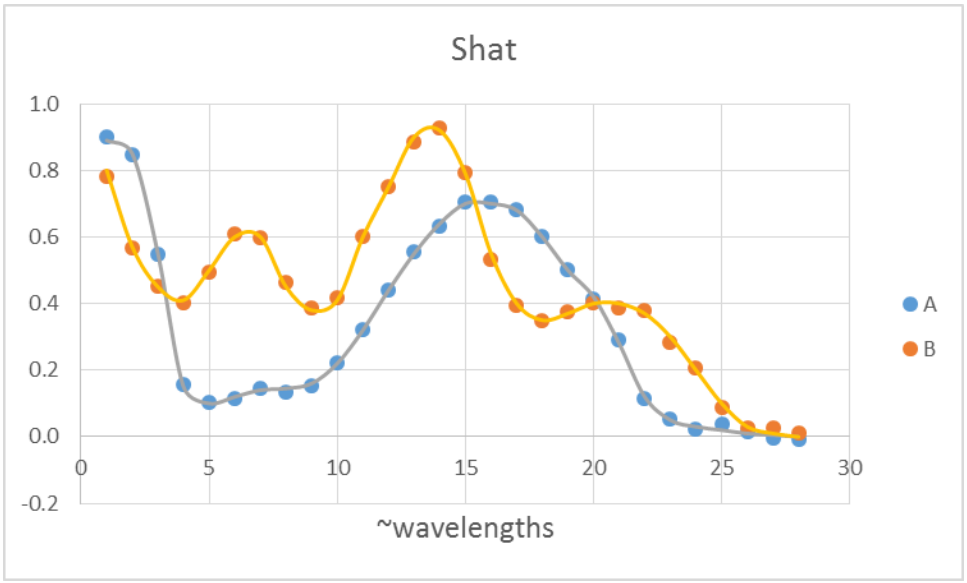


Fig. 5.27. Spectra of two components, $\hat{\mathbf{S}}$, obtained from PCR analysis from the training data set.

The root mean square of self-prediction, RMS_{sp}, is 0.0071 and 0.0043 for both components which corresponds to 1.07% and 1.06% relative value.

Finally, using PCR analysis concentrations of the analyzed test set were determined using program PCRpred.m. The calculated concentrations as a function of time are presented in Fig. 5.28. The calculated concentrations, points, are compared with the assumed concentrations, lines, used to simulate the spectra. A Gaussian absolute noise 0.01*N(0,1) was later added to the simulated spectra.

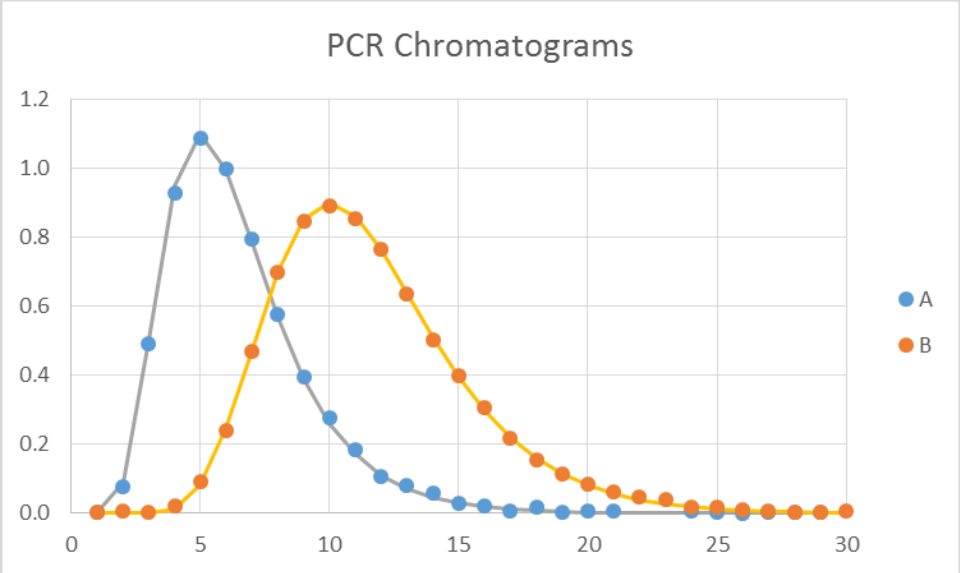


Fig. 5.28. Concentration profiles during the chromatographic analysis; points - calculated concentrations, lines – assumed concentrations used to simulate the spectra.

It is obvious that the PCR analysis of the overlapping spectra of the two compounds permits good resolution of the overlapping chromatograms.

6 Partial Least Squares (PLS)

PLS is considered as the most important regression technique for multivariate data analysis.[3,26] Although it is similar to PCR the decomposition is performed differently, using simultaneous decompositions of the spectra and concentrations. It takes advantage of the correlation between the spectral data **X** and the component concentrations **C**. It also takes into account errors in both the spectra and the concentrations (while PCR assumes that errors are only in the spectra **X**). The eigenvectors and scores calculated using PLS are different from those obtained using PCR. PLS is an important tool when there is only partial knowledge of the data. PLS is a good alternative to the more classical multiple linear regression and principal component regression methods because it is more robust. Besides in spectroscopy and chromatography it was very successful in other area as quantitative structure-activity relationships, QSAR.[27]

There are two implementations of PLS: PLS1 and PLS2. In PLS2 algorithm uses the concentration matrix of all species studied while in PLS1 only uses one concentration vector corresponding to one species and the procedure is repeated for each species separately.

6.1 PLS2

In PLS2 the scores, **T**, for the spectra, **X,** and for the concentrations, **C**, are common:

$$\mathbf{X} = \mathbf{T}\mathbf{P}' + \mathbf{E} \tag{6.1}$$

$$\mathbf{C} = \mathbf{T}\mathbf{Q}' + \mathbf{F} \tag{6.2}$$

where **P** and **Q** are the loadings of **X** and **C**, and **E** and **F** are the errors which should be minimized simultaneously. Centering of both **X** and **C** matrices is usually used but when concentration ranges or units of different components are different the standardization of both matrices is used. PLS is an iterative process until sum of square difference of old and new scores becomes small.

The algorithm used is as follows:[3]

1) Center or standardize **X** and **C** columns. Both components must be preprocessed in the same way.
2) Start with $\hat{\mathbf{C}}$ equal to 0 (for centered concentrations the predicted concentrations are equal to mean concentrations after the inverse preprocessing that is returning to original values).
3) Construct a vector **u** containing initial guess of concentrations e.g. one of the columns in the initial preprocessed concentration, matrix, **C**.
4) Calculate the vector **h**

$$\mathbf{h} = \mathbf{X}'\mathbf{u} \tag{6.3}$$

5) Calculate the guessed scores

$$^{new}\hat{\mathbf{t}} = \frac{\mathbf{X}\,\mathbf{h}}{\sqrt{\sum h^2}} \tag{6.4}$$

If it is the first iteration remember the scores, call them $^{initial}\mathbf{t}$.

6) Calculate the guessed loadings

$$\hat{\mathbf{p}} = \frac{\hat{\mathbf{t}}'\mathbf{X}}{\sqrt{\sum \hat{t}^2}} \tag{6.5}$$

7) Calculate concentrations loading's vector

$$\hat{\mathbf{q}} = \frac{\mathbf{C}'\hat{\mathbf{t}}}{\sum \hat{t}^2} \tag{6.6}$$

8) Calculate a new vector **u**

$$\mathbf{u} = \frac{\mathbf{C}\hat{\mathbf{q}}}{\sum q^2} \tag{6.7}$$

and return to step 4).

Check for convergence

9) If this is the second iteration, compare the new and old scores vectors for example, by looking at the size of the sum of square difference in the old and new scores, i.e. $\sum\left({}^{initial}\hat{t} - {}^{new}\hat{t}\right)^2$. If this is small the PLS component has been adequately modelled, set the PLS scores (**t**) and both types of loadings (**p** and **c**) for the current PC to $\hat{\mathrm{t}}$, $\hat{\mathbf{p}}$, and $\hat{\mathbf{q}}$. Otherwise, calculate a new value of **u** as in step 8) and return to step 4).

Compute the component and calculate residuals

10) Subtract the effect of the new PLS component from the data matrix to obtain a residual data matrix

$${}^{resid}\mathbf{X} = \mathbf{X} - \mathbf{t}\,\mathbf{p}' \tag{6.8}$$

11) Determine new concentrations estimated

$${}^{new}\hat{\mathbf{C}} = {}^{initial}\hat{\mathbf{C}} + \mathbf{t}\,\mathbf{q}' \tag{6.9}$$

and sum the contribution of all components calculated to give an estimated $\hat{\mathbf{c}}$. Calculate

$${}^{resid}\mathbf{C} = {}^{true}\mathbf{C} - \hat{\mathbf{C}} \tag{6.10}$$

12) To determine further components replace both **X** and **C** by residuals and return to step 3).

This process is performed using PLS2.m. Reader may follow details of calculations in pls3.m.[3] Very often PLS2 estimates of concentration are worse than those by PLS1, so a good strategy might be to perform PLS2 as a first step, which could provide further information such as which wavelengths are significant and which concentrations can be determined with a high degree of confidence, and then perform PLS1 individually for the most appropriate compounds.

6.2 PLS1

This method is similar to PLS2 and might be described by the following Eqns.

$$\mathbf{X} = \mathbf{T}\mathbf{P}' + \mathbf{E} \tag{6.1}$$

$$\mathbf{c} = \mathbf{T}\,\mathbf{q} + \mathbf{f} \tag{6.11}$$

where matrices **C**, **Q**, and **F** in Eq. (6.2) were replaced by vectors **c**, **q**, and **f** in Eq. (6.11). As in PLS2 the matrix of scores **T** is common to both the concentrations, **c**, and measurements, **X**. It should be noticed that scores, **T**, and loadings, **P**, obtained for PLS are different from **T** and **P** obtained using PCA because PCA does not take into account the **c** data. Here, a unique set of scores and loadings is obtained for each component in the analysis. It should also be noticed that although the scores, **T**, **are orthogonal**, as in PCA, the **loadings are neither normalized nor orthogonal**. One interesting feature of PLS1 is that knowledge of the number of principal components and the concentrations of one species allows to carry out the analysis and to predict unknown concentrations of one component from the measurement matrix $\mathbf{X}_u$ of these components.

Implementation of the PLS1 algorithm is described below:

1) First the data should be preprocessed by centering or standardization. Usually only centering is used.
2) Select one vector of concentrations corresponding to one species, **c**. Start with an estimate of $\hat{\mathbf{c}}$ which is a vector of 0 (equal to the mean concentration if the vector is centered).
3) Calculate vector **h**:

$$\mathbf{h} = \mathbf{X}'\mathbf{c} \tag{6.12}$$

4) Calculate the scores:

$$\mathbf{t} = \frac{\mathbf{X}\,\mathbf{h}}{\sqrt{\sum h^2}} \tag{6.13}$$

5) Calculate the loading of **x**

$$\mathbf{p} = \frac{\mathbf{t}'\mathbf{X}}{\sum t^2} \tag{6.14}$$

6) Calculate the loading *q* of *c* which is a scalar (in PLS2 it is a vector)

$$q = \frac{\mathbf{c}'\mathbf{t}}{\sum t^2} \tag{6.15}$$

7) Compute the contribution to the concentration *t q* and the contribution to *x*, **t p**
8) Subtract the effect of the new PLS component from the data matrix to get a residual data matrix:

$$^{resid}\mathbf{X} = \mathbf{X} - \mathbf{t}\,\mathbf{p}' \tag{6.16}$$

9) Determine the new estimation of concentrations:

$$^{new}\hat{\mathbf{c}} = ^{initial}\hat{\mathbf{c}} + \mathbf{t}\,\mathbf{q}' \tag{6.17}$$

and sum the contribution of all components calculated to give an estimated $\hat{\mathbf{c}}$. Note that the initial concentration estimate is 0 (or the mean) before the first component has been computed.

Calculate:

$$^{resid}\mathbf{c} = {}^{true}\mathbf{c} - {}^{new}\hat{\mathbf{c}} \qquad (6.18)$$

where $^{true}\mathbf{c}$ is, like all values of *c*, after the data have been preprocessed (such as centering).

10) If further components are required, replace both **X** and **c** by the residuals and return to step 3).

The PLS calculations might be followed in Matlab program PLS1.m.

Calculation of the unknown concentrations from $\mathbf{X}_u$ spectra may be achieved using PLS1pred.m or PLS2pred.m. Cross-validation is performed using PLScross.m. Below PLS 1 and PLS2 will be illustrated in a few examples.

Exercise 6.1.
Carry out PLS1 and PLS2 analysis for the data in Exercise 5.1.

Cross-validation using PLS applied the experimental data using PLScross.m confirms that there are only two principal components. The results of the parameters PRESS and RMS_{cv} are displayed in Fig. 6.1.

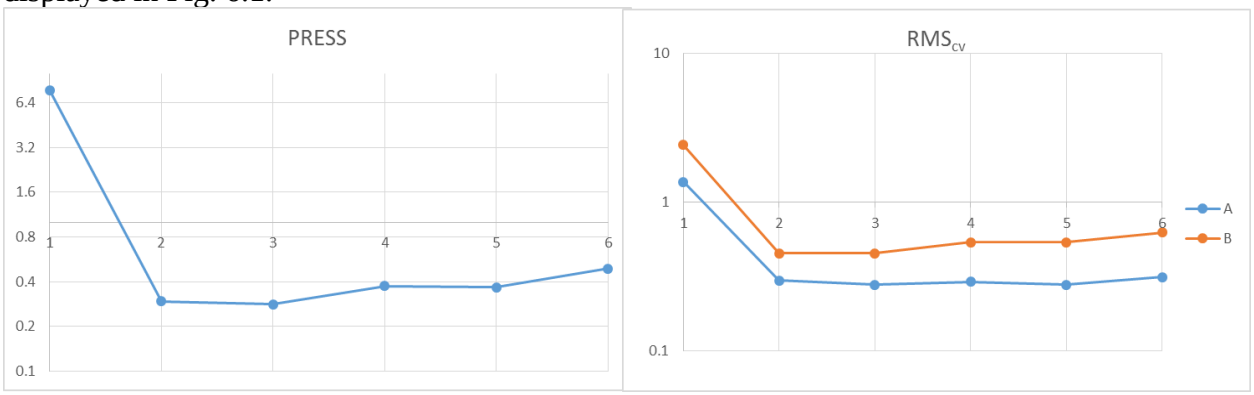


Fig. 6.1. Plots of PRESS and RMS_{cv} as functions of the number of principal components.

In performing PLS1 two concentration matrices CA.m and CB.m containing two vectors from the matrix **C**, each containing one column for one species, are used. The program PLS1 (PLS1A.m and PLS1B.m) is executed for each concentration vector separately. Comparison of RMS_{sp} of self-prediction of concentrations using PLS1 and PLS2 is shown in Table 6.1 and compared with the earlier found values from PCR in Exercise 5.1.

Table 6.1. Comparison of the root-mean square errors of self-prediction, RMS_{sp}, for three methods used.

compound	A	B
PCR	0.2576	0.3962
PLS1	0.2535	0.3908
PLS2	0.2542	0.3907

These results indicate that the errors of self-prediction are very similar with very little smaller values for the PLS method.

Exercise 6.2.
Use PLS method to analyze data in Exercise 5.2, center the experimental data.

Cross-validation using PLScross.m shows that there are two principal components, in agreement with two concentration components, see Fig. 6.2.

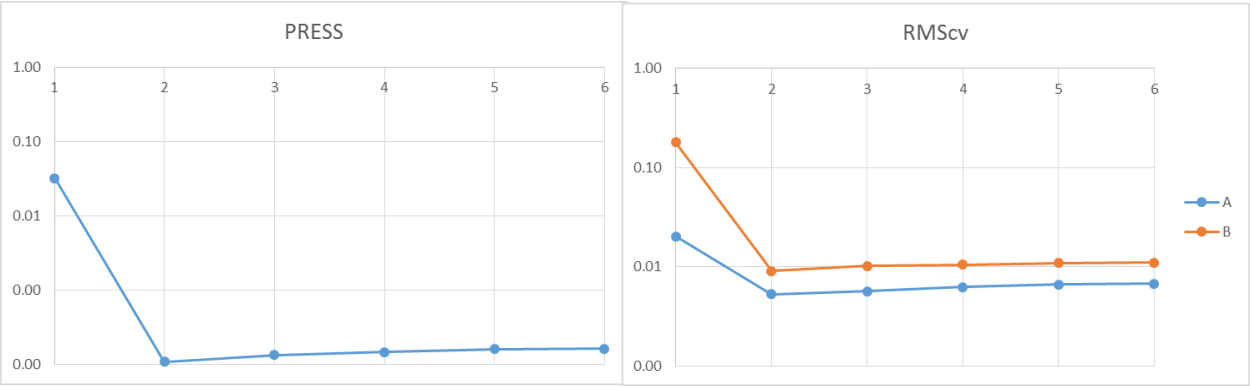


Fig. 6.2. Dependence of the cross-validation parameters PRESS and RMS_{cv} as functions of the number of principal components.

Comparison of the root mean squares of self-prediction, RMS_{sp}, of concentrations for the PLS (PLS1A.m, PLS1B.m, PKS2.m) and PCR is shown in Table 6.2. It is clear that all these methods give similar errors of prediction.

Table 6.2. Comparison of the root mean squares of self-prediction, RMS_{sp}, of concentrations for the determination using PCR, PLS2, and PLS2.

Component	A	B
PCR	0.0040	0.0065
PLS1	0.0040	0.0064
PLS2	0.0039	0.0064

PLS method can also be used to determine concentrations of the validation set. Comparison of the obtained results, RMS_{test} is presented in Table 6.3.

Table 6.3. Comparison of the root mean squares of validation, RMS_{test}, of concentrations for the determination using PCR, PLS2, and PLS2.

Component	A	B
PCR	0.0051	0.0065
PLS1	0.0060	0.0077
PLS2	0.0061	0.0077

In this case also the results are similar but those obtained using PCR are slightly better.

Exercise 6.3.
Carry out analysis of the data in Exercise 5.3 using PLS1 and PLS2.

First, the cross-validation is performed to determine the number of PCs in the data set. Dependence of the parameters PRESS and RMS_{cv} on the number of principal components is displayed in Fig. 6.3. These parameters decrease up to three PCs which suggests that only three principal components are important and the higher PCs approximate only the random noise.

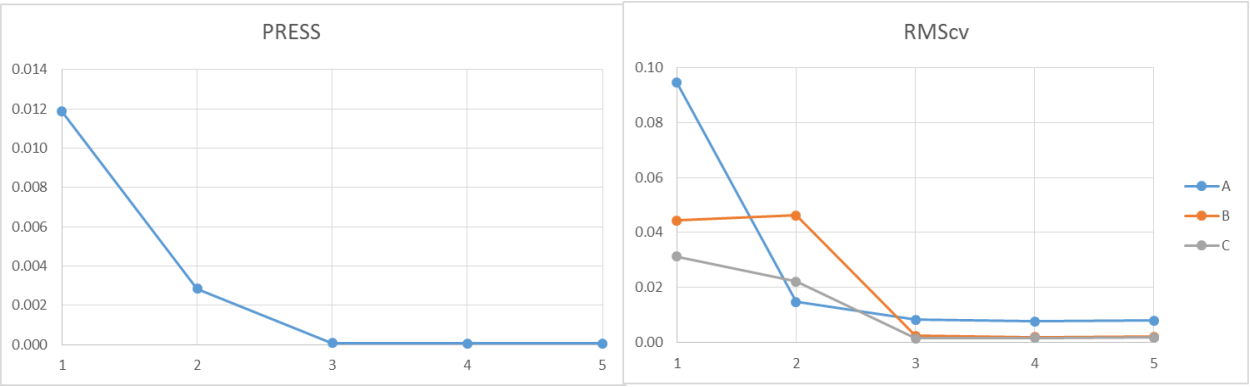


Fig. 6.3. Dependence of the cross-validation parameters PRESS and RMS_{cv} on the number of PCs using PLS method.

Comparison of the self-prediction errors RMS_{sp} of all three methods (PCR, PLS1, and PLS2) shows identical results as in Table 5.12, see file Ex6-3.xlsx and the predictions of the concentrations during the reaction are also practically the same. All these methods lead here to the same results.

Exercise 6.4.
Analyze data in Exercise 5.4 using PLS1 and PLS2.

Cross-validation of the experimental data, Fig. 6.4, indicates that there are three PCs in the system in agreement with three components analyzed.

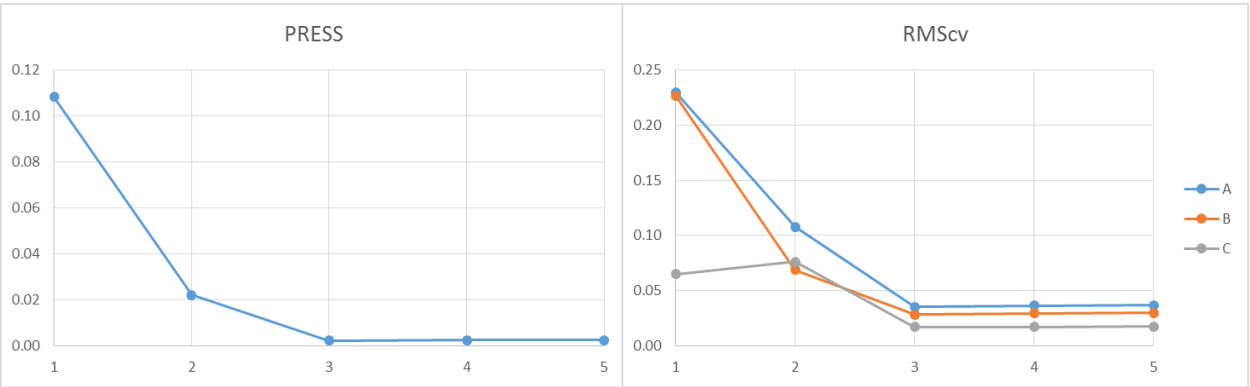


Fig. 6.4. Cross-validation analysis of the data using PLS.

Using three PCs and self-prediction of concentrations the following results were obtained, Table 6.4.

Table 6.4. Comparison of the root mean squares of self-prediction, RMS_{sp}, of concentrations for the determination using PCR, PLS2, and PLS2.

Component	A	B	C
PCR	0.0175	0.0159	0.0058
PLS1	0.0150	0.0135	0.0049
PLS2	0.0152	0.0143	0.0047

In this case the RMS_{sp} is smaller using PLS method.

Comparison of the results obtained for the validation test, RMS_{test}, is shown in Table 6.5. In general, the errors of the concentration determination are similar but for the component A they are slightly smaller using PCR.

Table 6.5. Comparison of the root mean squares of validation, RMS_{test}, of concentrations for the determination using PCR, PLS2, and PLS2.

Component	A	B	C
PCR	0.0136	0.0093	0.0098
PLS1	0.0150	0.0102	0.0107
PLS2	0.0150	0.0102	0.0109

Exercise 6.5.
Analyze data in Exercise 5.5 using PLS1 and PLS2.

The results of the cross-validation are presented in Fig. 6.5. They indicate that there are three PCs in agreement with three components present.

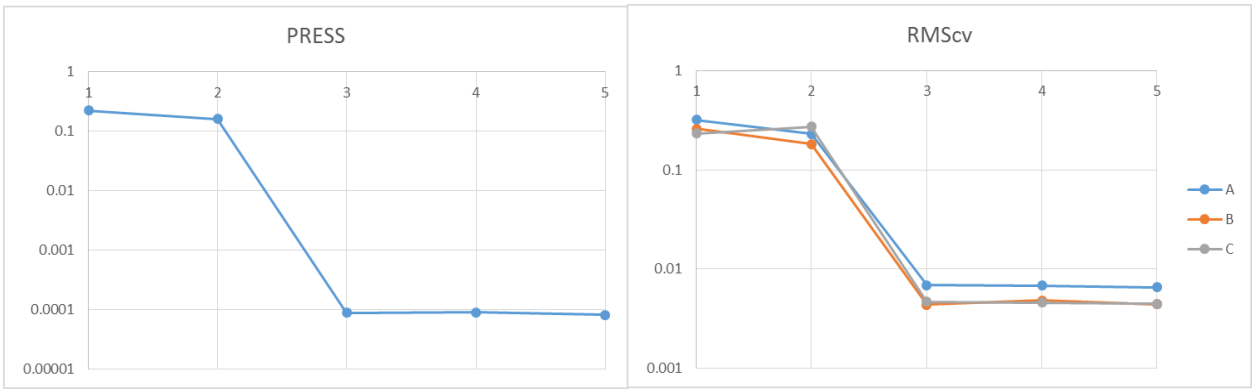


Fig. 6.5. Cross-validation analysis of the data using PLS.

Comparison of the results obtained for the self-prediction, RMS_{sp}, is shown in Table 6.6. The errors of the concentration determination by the three methods are similar.

Table 6.6. Comparison of the root mean squares of self-prediction, RMS_{sp}, of concentrations for the determination using PCR, PLS2, and PLS2.

Component	A	B	C
PCR	0.00361	0.00262	0.00271
PLS1	0.00359	0.00259	0.00269
PLS2	0.00360	0.00261	0.00269

Comparison of predictions for the unknown concentrations shows that the results obtained by three methods are also similar. Comparison of the predicted concentrations with those used for the simulations of the spectra is shown in Table 6.7.

Table 6.7. RMS_{test} values for the prediction of concentrations during chemical reaction, compared with the concentrations used in simulation of the spectra.

Component	A	B	C
PCR	0.020	0.014	0.023
PLS1	0.0096	0.0067	0.0113
PLS2	0.0096	0.0067	0.0113

The root mean squares errors of prediction of concentrations for PLS are smaller than those for PCR. Besides, errors for the species C are larger than those for A and B.

Exercise 6.6.
Analyze data in Exercise 5.6 using PLS1 and PLS2.

The cross-validation analysis, Fig. 6.6, shows that there are two principal components in agreement with the two concentrations.

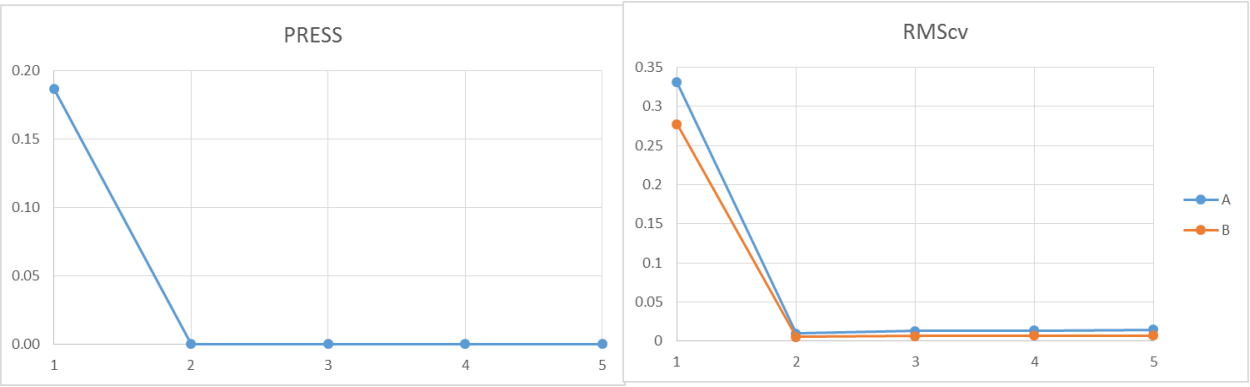


Fig. 6.6. Cross-validation analysis of the data using PLS.

Analysis of RMS_{sp} for PLS1 and PLS2 gives exactly the same results as in for PCR, Exercise 5.6.

Predicted concentrations were compared with those used in the simulation of the spectra. The corresponding RMS_{test} values are included in Table 6.8. It is evident that PLS techniques produce lower fitting errors than the PCR. Of course, on the chromatographic plot the differences between PCR and PLS are not noticeable.

Table 6.8. Comparison of the RMS_{test} values for the predicted concentrations.

Component	A	B
PCR	0.0184	0.0127
PLS1	0.00824	0.00568
PLS2	0.00720	0.00540

Exercise 6.7.
In the analysis of the complex mixture of 10 components spectra of 25 mixtures were measured at 27 wavelengths; they are in files Cdata.m and Xdata.m. Then, the spectra of 25 validation mixtures of these components were registered; they are in CVdata.m and XVdata.m. Analyze these mixtures using the PCR, PLS2, andPLS1 methods. The data and the results are included in the Excel file Ex6-7.xlsx. Use data centering.

This is the most difficult example because of the presence of so many components. The training set contains 25 spectra measured at 27 wavelength, they are in the matrix **X**(25,27) and the concentrations of these components in **C**(25,10).

Let us start first with the PCR. First, the number of principal components should be determined. It can be carried out in the PCRtest.m program which shows the values of λ for the principal components. They are displayed in Table 6.9. It is clear that the first four PCs explain 96.7% and 6 PCs explain 99.3% of the total variation. This means that the further PCs explain little variation. However, there are 10 components in the mixture and we would like to get all the possible information.

To further analyze the data the cross-validation in PCR was carried out. The plots of the parameters PRESS and RMS_{cv} are shown in Fig. 6.7.

Table 6.9. Sizes of the first 14 principal components, PCs using PCA.

PC	λ_i	%	Cumulative %
1	10.24642	79.269%	79.269%
2	1.078059	8.340%	87.609%
3	0.803598	6.217%	93.826%
4	0.375435	2.904%	96.730%
5	0.207632	1.606%	98.336%
6	0.130350	1.008%	99.345%
7	0.043285	0.335%	99.680%
8	0.021185	0.164%	99.843%
9	0.007670	0.059%	99.903%
10	0.006093	0.047%	99.950%
11	0.002491	0.019%	99.969%

12	0.001674	0.013%	99.982%
13	0.001635	0.013%	99.995%
14	0.000678	0.005%	100.000%
	sum		
	12.92621		

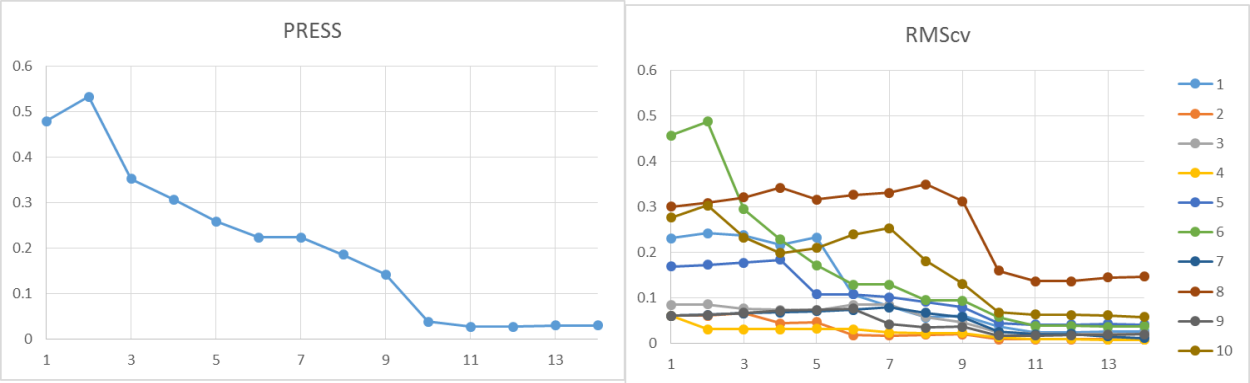


Fig. 6.7. Cross-validation analysis in PCR: plots of the parameters PRESS and RMS_{cv} versus the number of PCs.

Both parameters decrease up to PC =10 and then they are constant or increase slightly. These confirms that 10 PCs might be used in the analysis, in agreement with the number of compounds in the mixtures. Similar results were obtained using cross-validation in PLS; the results are shown Fig. 6.8.

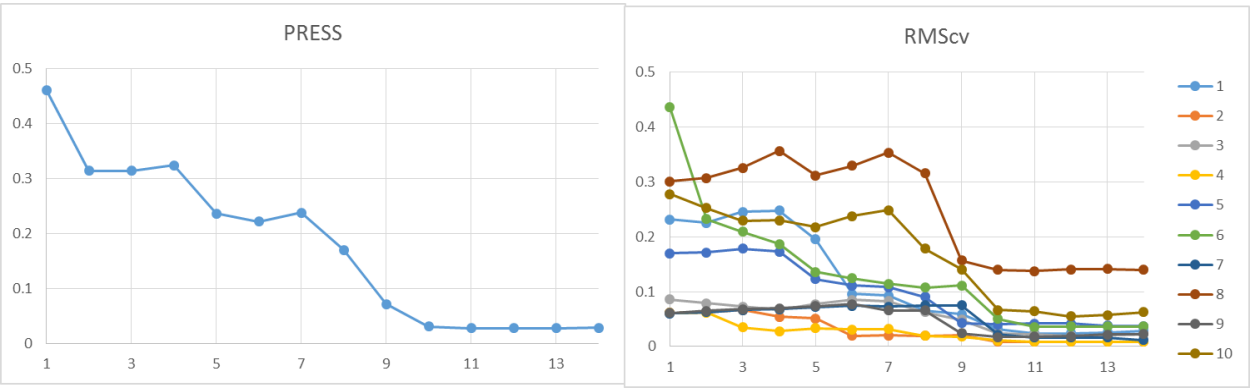


Fig. 6.8. Cross-validation analysis in PLS: plots of the parameters PRESS and RMS_{cv} versus the number of PCs.

Using 10 PCs the self-predicted concentrations were calculated and compared in Table 6.10. The values for PLS1 were calculated for each component separately.

Table 6.10. Results of the self-prediction of concentrations in the training set; the values of RMS_{sp} are given for all 10 components.

		1	2	3	4	5	6	7	8	9	10
PCR	RMSsp	0.0285	0.0072	0.0189	0.0102	0.0343	0.0413	0.0169	0.1221	0.0136	0.0498
	RMSsp%	6.3%	6.0%	11.2%	8.5%	10.2%	2.5%	14.1%	20.3%	11.3%	8.8%
PLS1	RMSsp	0.0137	0.0047	0.0129	0.0048	0.0208	0.0191	0.0074	0.0711	0.0111	0.0351
	RMSsp%	3.0%	3.9%	7.7%	4.0%	6.2%	1.2%	6.2%	11.9%	9.3%	6.2%
PLS2	RMSsp	0.0237	0.006	0.0179	0.0079	0.0284	0.0338	0.013	0.0962	0.0126	0.046
	RMSsp%	5.2%	5.0%	10.7%	6.6%	8.5%	2.1%	10.8%	16.0%	10.5%	8.2%

It can be noticed that the relative errors of self-prediction are the largest for the PCR and the smallest for the PLS1.

The comparison of the predicted and assumed concentrations for PCR is illustrated in Fig. 6.9. It is clear that the worst correlation was obtained for compound No 8 for which the largest RMS_{sp} is observed. The best correlations (>0.99) were found for components 1, 2, and 6. Of course better correlations were found for the results obtained using PLS, especially for PLS1.

PCR allows for the determination of the spectra of all 10 compound. They were calculated using Eq. (5.6) and compared with the assumed spectra in Fig. 6.10. It is evident that the spectra were well reproduced even in the cases where the self-prediction errors were higher.

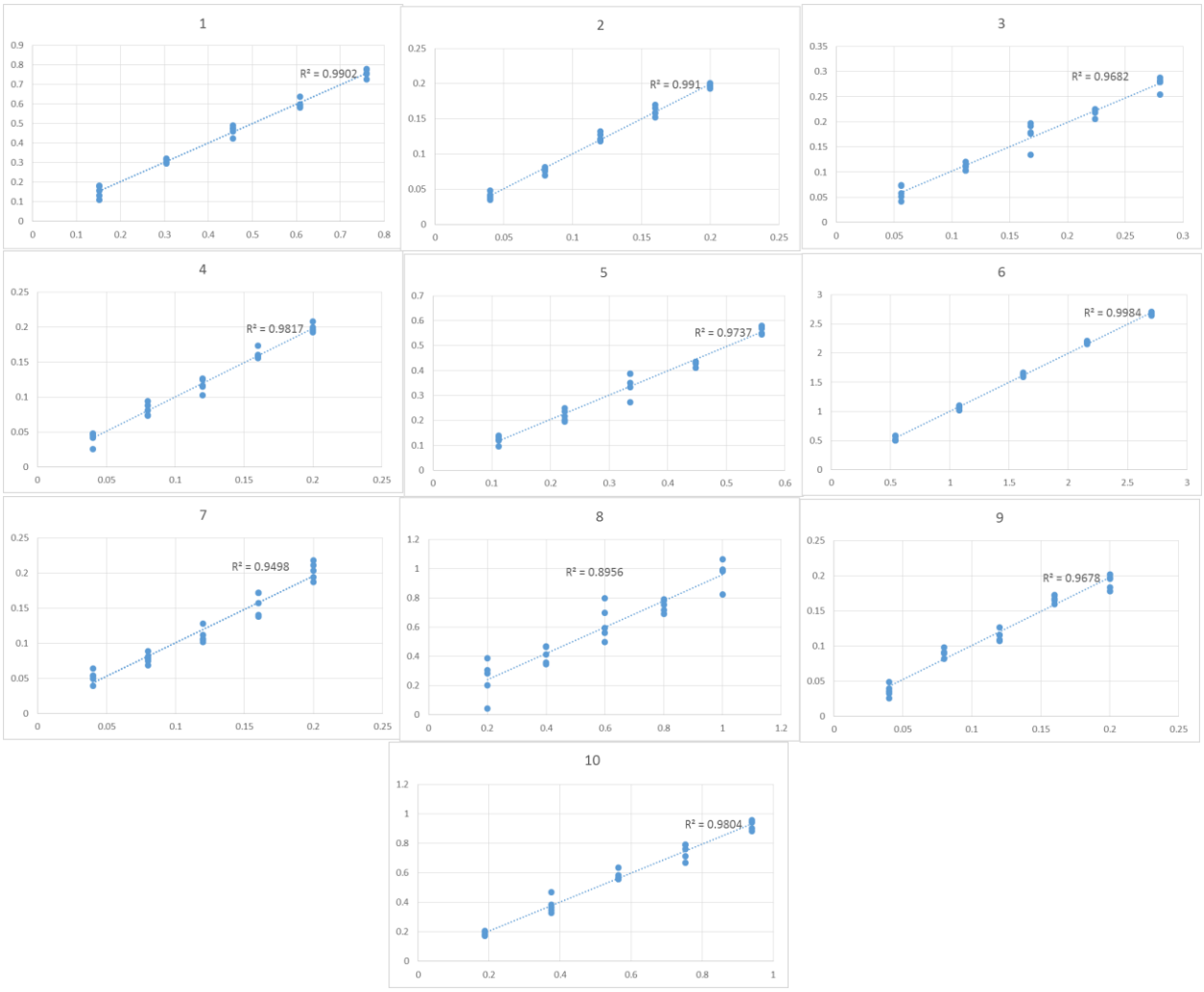


Fig. 6.9. Dependence of the self-predicted and assumed concentrations of 10 components obtained using PCR.

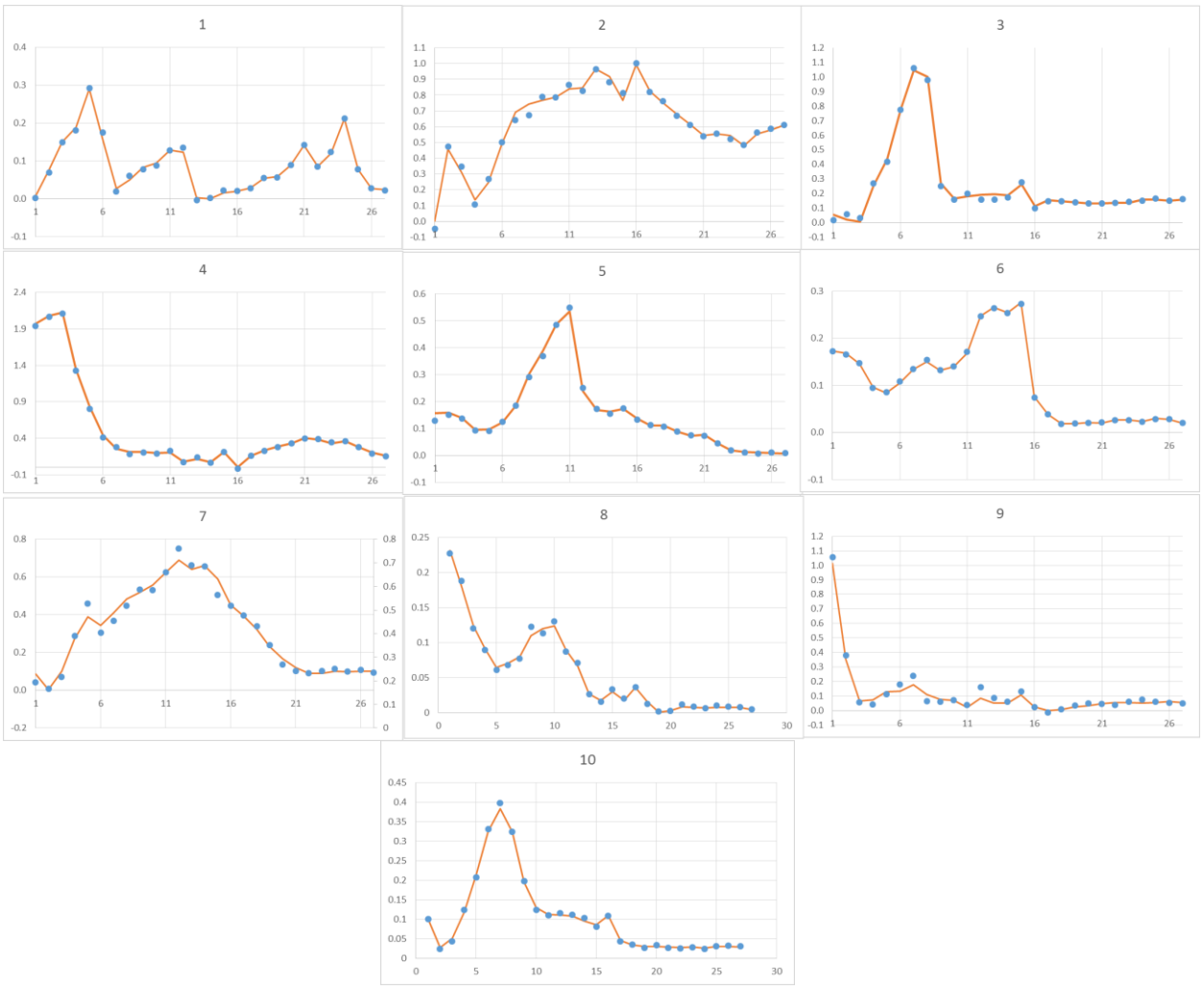


Fig. 6.10. Comparison of the self-predicted (points) and assumed (lines) spectra of the components form the PCR method.

Finally, using information from the training (test) set the concentrations were calculated for the validation set and compared with the assumed values. They are displayed in Table 6.11.

Table 6.11. Results of the prediction of concentrations in the validation set; the values of RMS_{test} are given for all 10 components.

		1	2	3	4	5	6	7	8	9	10
PCR	RMStest	0.033	0.0083	0.0342	0.0116	0.0486	0.0479	0.0313	0.14	0.0229	0.1167
	RMStest%	7.32%	6.99%	19.68%	9.44%	14.12%	2.96%	25.47%	24.21%	19.46%	21.61%
PLS1	RMStest	0.0238	0.0067	0.0225	0.0092	0.0376	0.0426	0.0244	0.1165	0.0189	0.0777
	RMStest%	5.29%	5.58%	13.11%	7.50%	10.99%	2.63%	19.82%	20.04%	15.96%	14.19%
PLS2	RMStest	0.0247	0.0064	0.0257	0.0087	0.0385	0.0378	0.0239	0.1055	0.0181	0.0878
	RMStest%	5.48%	5.39%	14.85%	7.08%	11.19%	2.33%	19.51%	18.24%	15.40%	16.20%

The errors of prediction are lower when PLS method is used but there are no significant differences between PLS1 and PLS2 here.

Exercise 6.8.

The next exercise contains the same concentrations as in the exercise above but as the spectra in Exercise 6.7 were simulated with the Gaussian noise added whereas the spectra in the present exercise are experimental coming from Brereton.[3]

Analyze the learning data **X**(25×27) and **C**(25×10) and then use the knowledge to validate spectra **Xtest**(25×27), predict the concentrations, and validate them versus the experimental concentrations in **Ctest**(25×10). All the data are in file pahdat.m. Both, the learning and validation spectra have the same dimensions and there are 10 components in the mixture. All the data are included in file pahdat.m.[3] Use PCR and PLS.

First, let us start with the PCA analysis. Using data centering and standardization the results shown in Table 6.12 were obtained.

Table 6.12. Results of PCA analysis on training data.

raw data		centered		standardized	
λ_i	%	λ_i	%	λ_i	%
183.6794	98.9273%	8.690716	84.332%	523.8251	77.740%
1.018452	0.5485%	0.727779	7.062%	63.6085	9.440%
0.503371	0.2711%	0.421568	4.091%	33.1315	4.917%
0.214053	0.1153%	0.213529	2.072%	26.8114	3.979%
0.178061	0.0959%	0.176635	1.714%	15.2296	2.260%
0.043008	0.0232%	0.042977	0.417%	5.7527	0.854%
0.013223	0.0071%	0.013165	0.128%	2.0830	0.309%
0.010914	0.0059%	0.008474	0.082%	1.6278	0.242%
0.00621	0.0033%	0.006169	0.060%	0.9356	0.139%
0.004317	0.0023%	0.004317	0.042%	0.8104	0.120%

In such a complex mixture the analysis of raw data suggests one PC. Assuming here that we accept that 1% of the explained data the centered and standardized data are explained by 5 PCs. This is much less than 10 components present. Let us look at the cross-validation using PCR. The plots of PRESS and RMScv are displayed in Fig. 6.11 and 6.12.

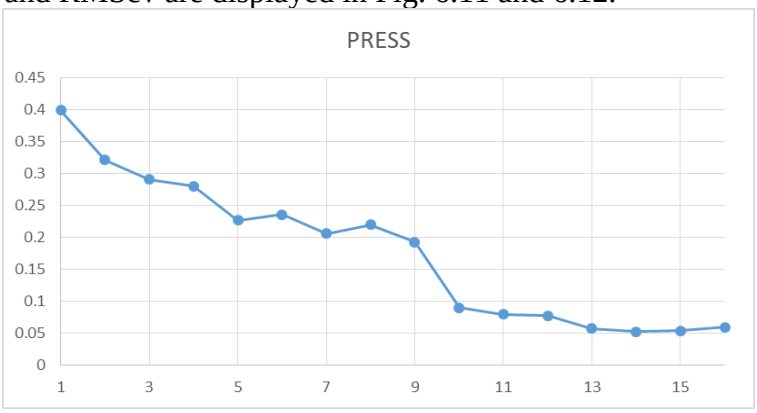


Fig. 6.11. Plot of the parameter PRESS vs. number of PCs for cross-validation of the training data using PCR.

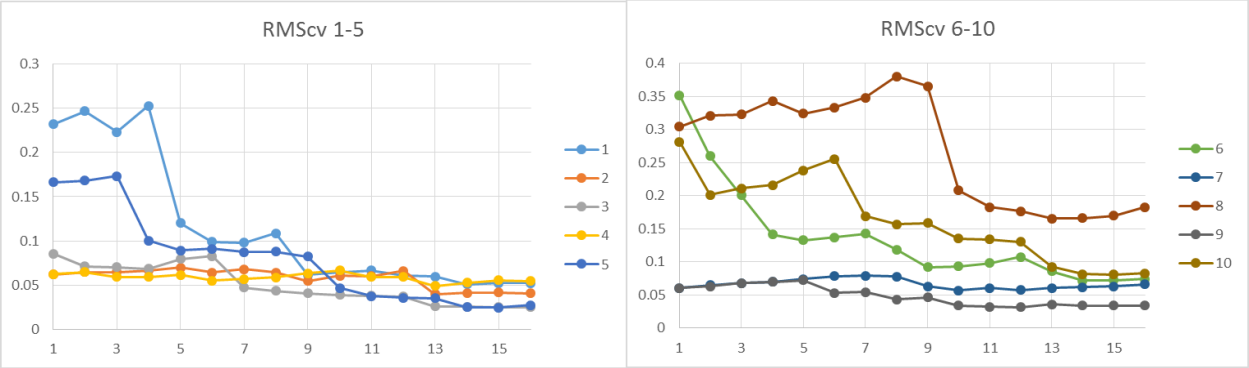


Fig. 6.12. Dependence of the root mean square of cross-validation of the learning data set, RMS_{cv}, versus number of PCs for 10 component analysis using PCR.

PRESS parameter shows gradual decrease with sharper step around 10 PC and then it stays constant. The behavior of RMS_{cv} is more complicated, some components display sharp decrease around 10 PCs but other more pronounced decrease at different values of PCs even increase, see Fig. 6.13.

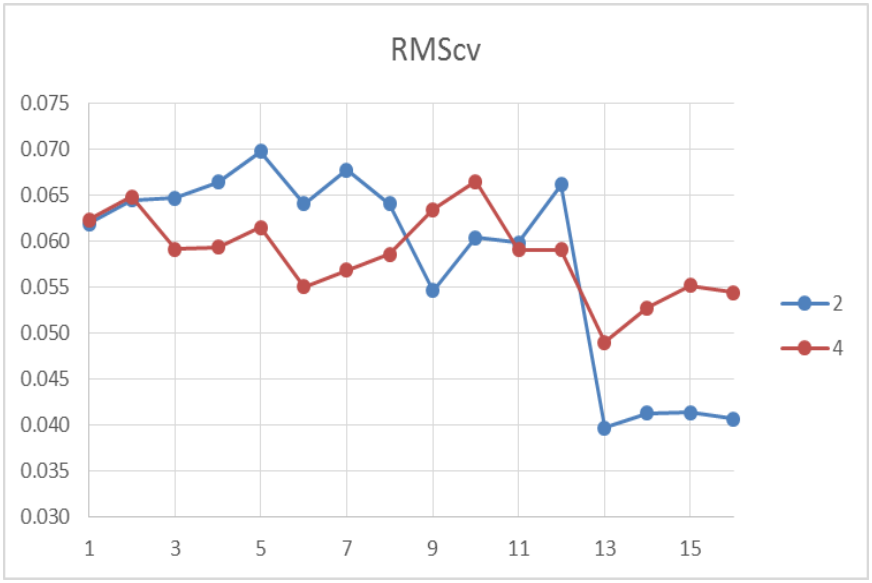


Fig. 6.13. Examples of the dependence of RMS_{cv} vs. number of PCs for two components from Fig. 6.12.

Cross-validation of the training set using PLS2 is displayed in Fig. 6.14 and 6.15.

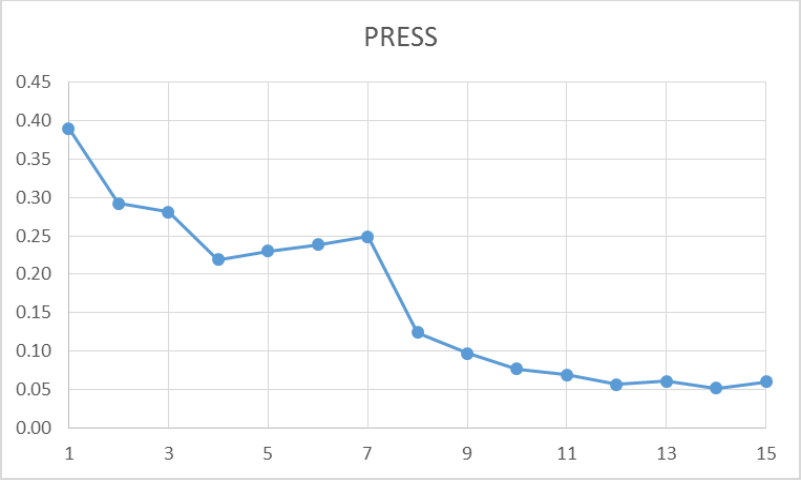


Fig. 6.14. Dependence of the parameter PRESS vs. number of PCs for validation of the training data using PLS analysis.

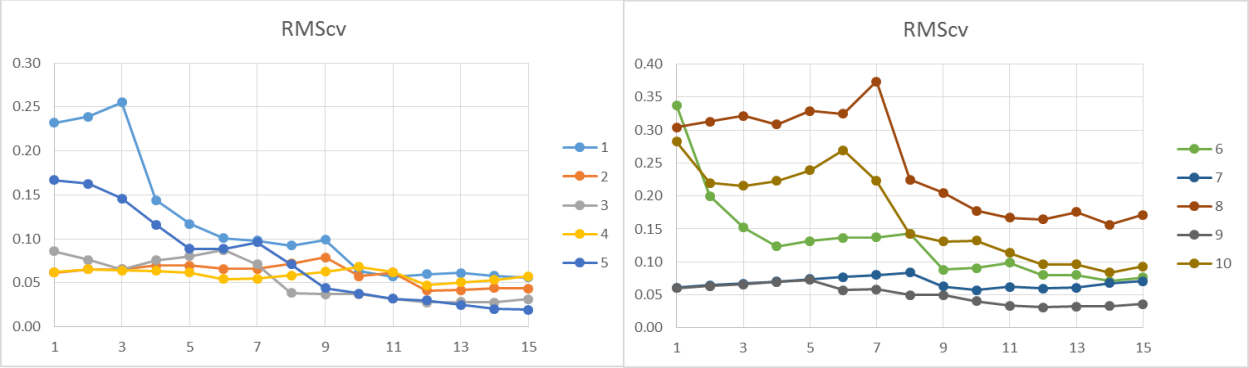


Fig. 6.15. Dependence of RMS_{cv} vs. number of PCs for 10 components in the training data set using PLS analysis.

The behavior of PRESS and RMS_{cv} does not indicate clearly the number of important PCs. Sharper decrease of PRESS is observed at PC=8 and then there is a gradual decrease and for RMS_{cv} it is observed for PC numbers between 5 and 10. Because we know that there are 10 components in further analysis, we will use 10 PCs and see how precisely these concentrations can be evaluated.

Results obtained for self-prediction of the training set, RMS_{sp}, using PCR, PLS1, and PL2 methods and different weighting method are shown in Table 6.13.

Table 6.13. Values of RMS_{sp} for the training data set for 10 components using various methods.

Method	1	2	3	4	5	6	7	8	9	10
PCR raw	10.29%	36.25%	15.81%	42.04%	9.04%	4.24%	31.86%	24.98%	21.21%	16.21%
PCR centered	10.56%	37.88%	15.95%	43.51%	9.44%	4.54%	32.70%	24.20%	21.40%	16.63%
PCR standardized	10.60%	41.27%	28.07%	39.15%	13.95%	4.98%	37.31%	24.77%	19.40%	26.04%
PLS2 raw	10.09%	33.64%	13.65%	43.13%	6.50%	4.03%	32.46%	18.95%	25.08%	14.49%
PLS2 centered	10.24%	34.08%	13.63%	44.58%	6.99%	4.25%	33.42%	18.62%	25.83%	14.77%
PLS2 standardized	10.25%	37.80%	22.17%	39.98%	14.33%	4.67%	31.89%	26.69%	19.03%	22.12%
PLS1 centered	5.46%	19.08%	7.86%	22.50%	3.54%	2.46%	21.92%	12.97%	16.50%	7.02%

Analysis of the results obtained using PCR indicates that standardization increases the determination errors while for raw and centered data the results are similar. The errors obtained using PLS2 are generally smaller than those for PCR but use of PLS1 is advantageous because the errors are much smaller. Centering of the data was chosen here.

Next, the training set was used to determine concentrations of the validation set which were compared with the analytical values. The results are shown in Table 6.14.

Table 6.14. Values of RMS_{test} for the validation data of 10 components.

Method	1	2	3	4	5	6	7	8	9	10
PCR raw	26.33%	40.33%	27.51%	59.95%	17.72%	5.20%	91.69%	52.72%	35.37%	20.51%
PCR centered	25.78%	39.79%	26.72%	59.88%	17.82%	5.30%	92.74%	49.33%	34.29%	20.62%
PCR standardized	29.11%	54.31%	49.70%	58.16%	23.27%	7.97%	88.25%	42.90%	34.74%	35.06%
PLS2 raw	20.15%	30.09%	18.35%	47.61%	10.84%	3.77%	70.47%	30.27%	32.17%	14.14%
PLS2 centered	19.76%	29.42%	17.80%	46.64%	11.15%	3.80%	71.82%	27.75%	31.03%	14.29%
PLS2 standardized	22.15%	38.30%	30.75%	46.27%	18.18%	6.09%	68.51%	36.62%	26.69%	23.04%
PLS1 centered	18.36%	49.12%	11.44%	42.93%	8.58%	3.77%	74.06%	18.17%	23.77%	14.04%

It is evident that PLS methods give lower errors of prediction of the validation data and PLS1 gives the lowest values except those for component No 2. The largest errors of prediction are for the lowest average concentrations of components, that is for No 2, 4, and 7 (average concentrations 0.098, 0.119 and 0.094, respectively). These large errors of prediction 49%, 43%, and 75%, respectively, indicate that these compounds cannot be determined in the mixture. This is illustrated in Fig. 6.16 for the relation between predicted and analytical concentrations for components 2 and 7.

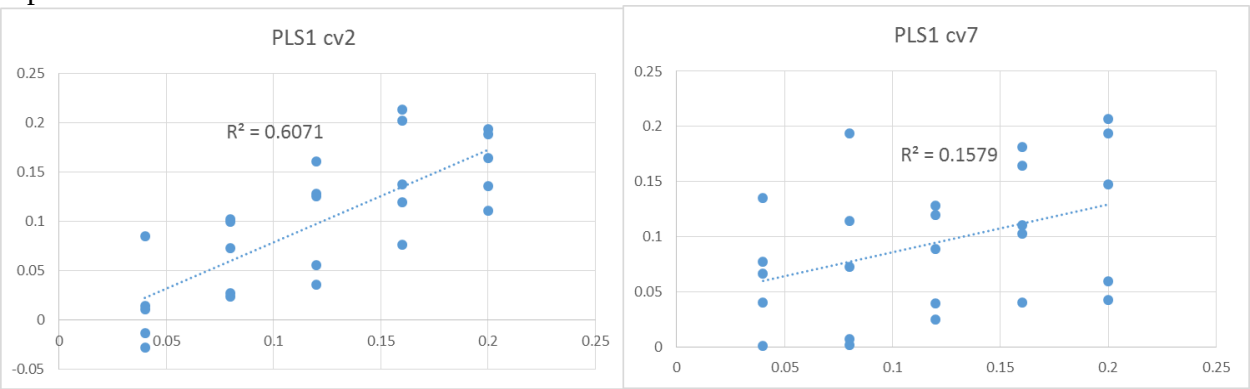


Fig. 6.16. Dependence of the predicted, using PLS1, on the analytical concentrations of the validation set for components 2 and 7. These concentrations cannot be quantitatively determined.

However, despite the complexity of the mixture (10 compounds at low and high concentrations) other components could be determined. Dependences of the predicted on analytical concentrations for components 5 and 7 are displayed in Fig. 6.17.

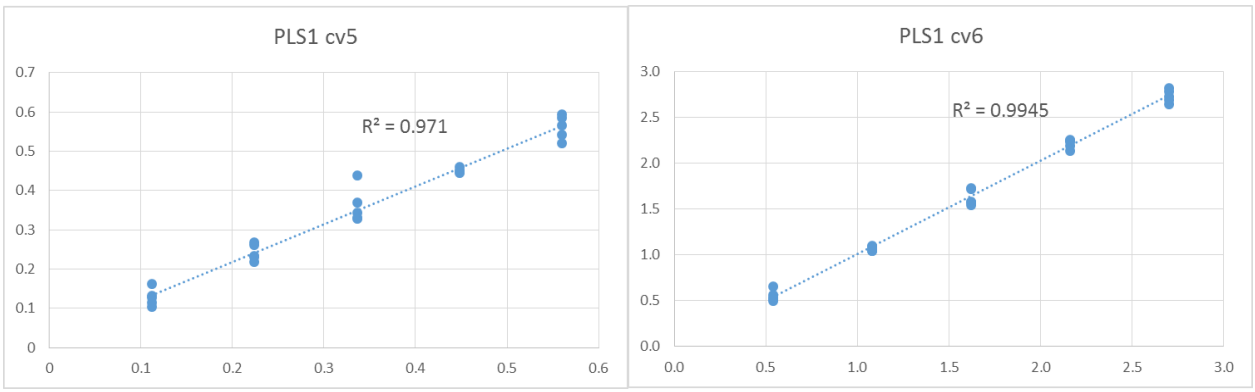


Fig. 6.17. Dependence of the predicted, using PLS1, on the analytical concentrations of the validation set for components 5 and 6. These concentrations can be easily determined in the mixture.

Despite of the problems with the validation the self-prediction analysis using PCR predicts the spectra of all compounds, Fig. 6.18, although some spectra display negative values of absorbance.

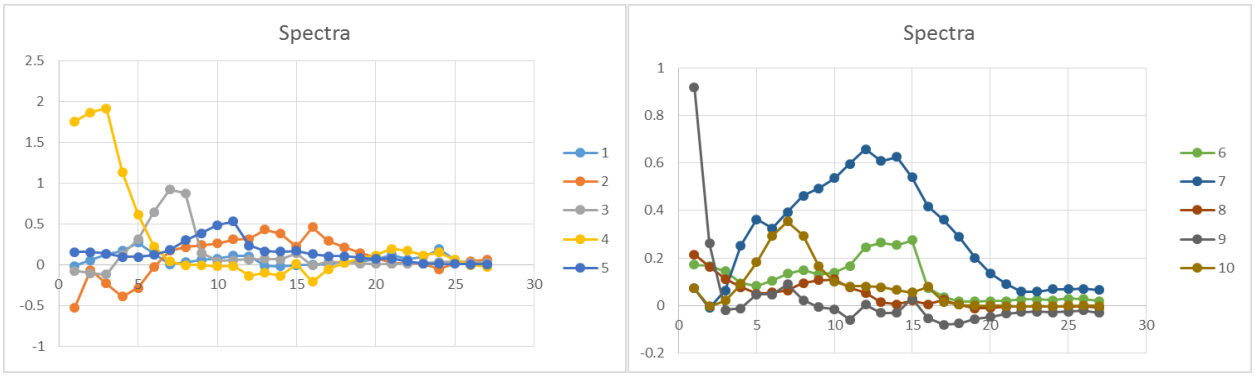


Fig. 6.18. Spectra of the components obtained using self-prediction PCR analysis for the training data set.

It should be added that in selecting the concentrations one should avoid collinearity of concentrations in the training set. Such a set would well auto-predict its concentrations; however such a collinearity would reduce the matrix rank and make it impossible to predict all the concentrations of the validation/test set.

Exercise 6.9.

In this exercise we will see the influence of the baseline on the results. The data from Exercise 5.1 will be used to which a baseline is added. Comparison of the spectra without the baseline and with the baseline are displayed in Fig. 6.19 where a linear baseline was chosen. However, any other baseline may be used.

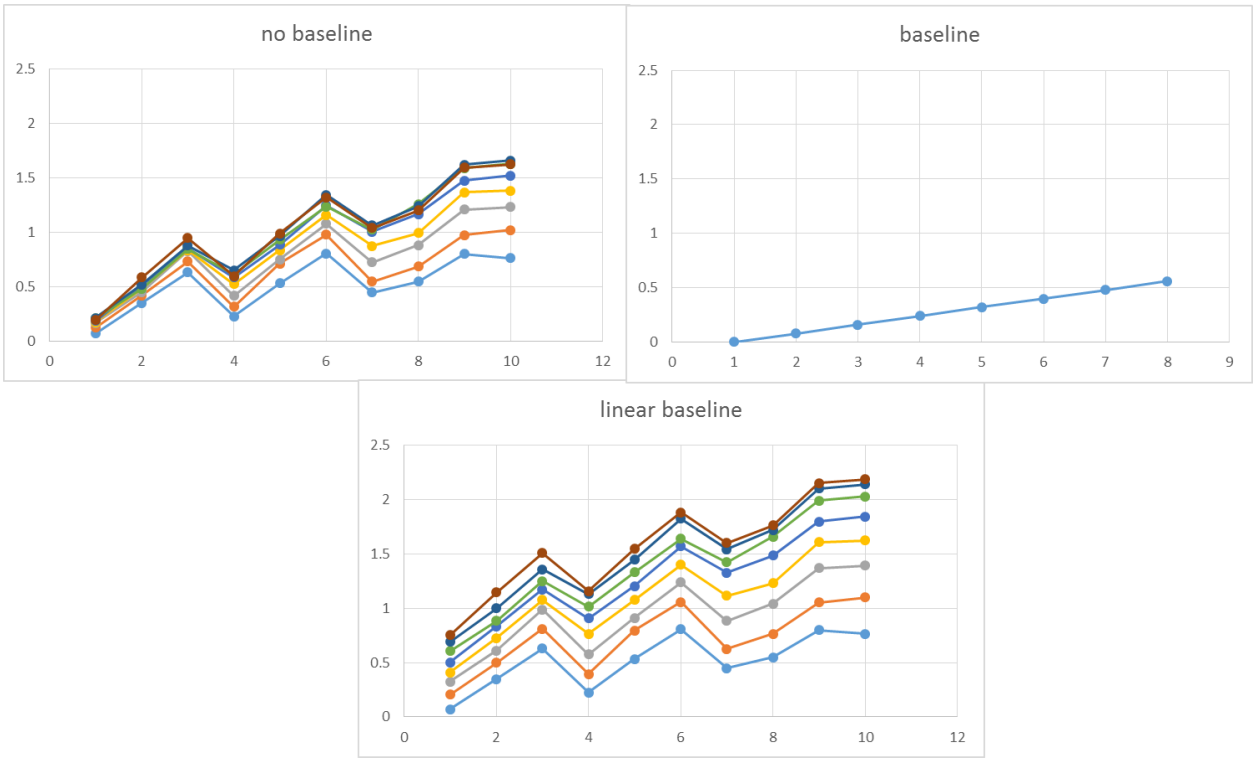


Fig. 6.19. Spectra without baseline, baseline added, and with baseline added.

Application of the PCR (2 PCs, centered data) gives the same results of concentrations as for the data without the baseline, see Table 5.1 and Ex6-9.xlsx. This is because the PCR and PLS are using the differences in spectra, not the absolute values. However, the baseline is added to the individual spectra, Fig. 6.20, because the sum of individual spectra must give the total spectrum, Fig. 6.19.

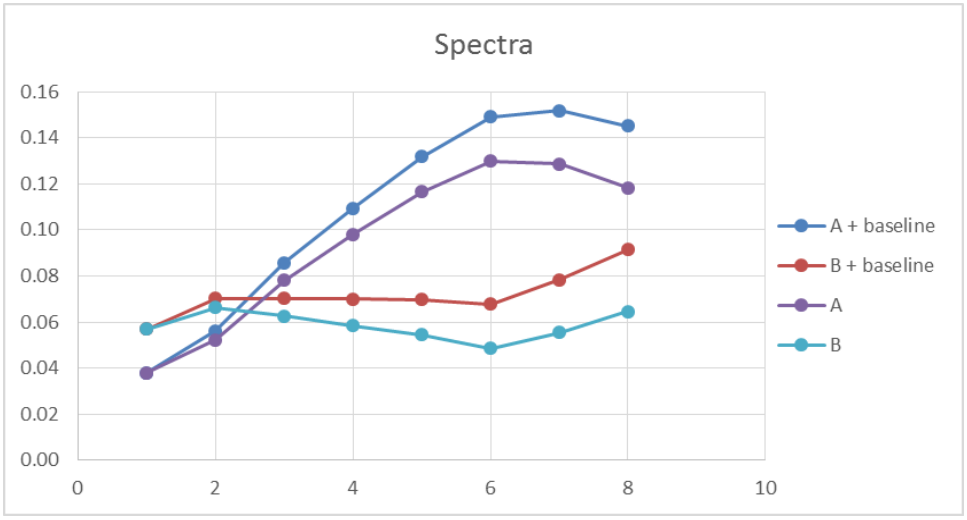


Fig. 6.20. Spectra of individual components obtained from the total spectra without and with the baseline.

It is clear that the presence of the baseline in the spectra (unknown but always the same) does not affect the determination of concentrations.

Exercise 6.10.

In this exercise determination of very different concentrations is illustrated; the average concentration of component 1 is 100 times smaller than that of the component 2, see Table 6.15.

Table 6.15. Concentrations of two species in Exercise 6.10.

1	2
0.001	0.90
0.002	0.85
0.003	0.55
0.004	0.35
0.005	0.50
0.006	0.60
0.007	0.25
0.008	0.40
0.009	0.10

Contribution of the component 1 to the total spectra is also very small. Comparison of all the spectra of two components and those of component 2 only is shown in

Fig. 6.21 where comparison of these spectra for the mixture No 6 and 9 is also presented.

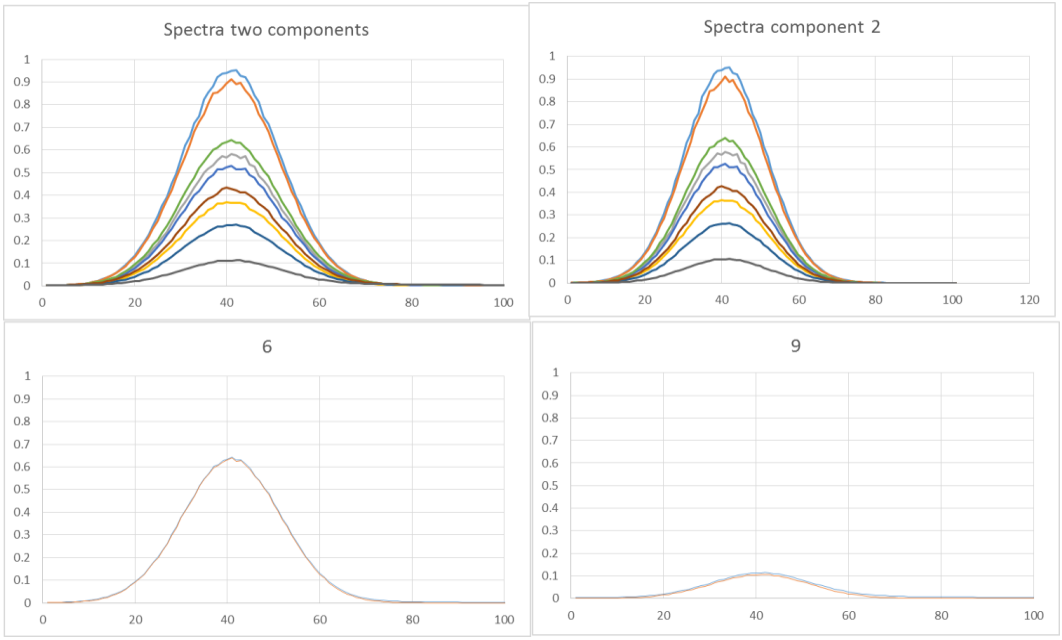


Fig. 6.21. Spectra of the mixture of two components and of the component 2 only. Below are the comparisons of spectra of component 2 and of the mixture for the spectra 6 and 9; the differences are not visible.

It is clear that the difference of the spectra of component 2 and of the mixture of 1 and 2 are hardly visible. In such a case there is an advantage of use of the standardization of the data. The results of the PCR and PLS analysis is shown in Table 6.16.

Table 6.16. Results of self-prediction using centered and standardized data in PCR and PLS1.

	RMS_{sp}	
Component	1	2
PCR centered	32.38%	0.51%
PCR standardized	0.25%	0.14%
PLS1 centered	14.13%	0.21%
PLS1 standardized	0.25%	0.14%

There is a dramatic decrease of the self-prediction errors of component 1 when the standardization is used.

The above examples show that the analysis using the multivariate analysis is much more complex than a simple regression and demand more tests but it allows to use information from the whole data set (spectra).

It should be noticed that the principal components regression is preferred by the statisticians while partial least squares method is preferred by the chemometricians.

7 Alternating Least Squares (ALS) method

Alternating least-squares is an alternative method for solving multiple component regression problems. It is not based on PCA but on the iterative least-squares method. It is usually used to implement non negativity (or other) constrains on the concentration and spectra.

The relation between the observed spectra and concentrations was shown in Eq. (3.2)

$$\mathbf{X} = \mathbf{C}\mathbf{S} + \mathbf{E} \tag{3.2}$$

Let us assume that the initial guess of concentrations $\hat{\mathbf{C}}$ is known (even random estimations might be sometimes used). Often as the first step the PCR is used. This will allow to estimate spectra $\hat{\mathbf{S}}$ using the least-squares method, Eq. (5.1):

$$\hat{\mathbf{S}} = \left(\hat{\mathbf{C}}'\hat{\mathbf{C}}\right)^{-1}\hat{\mathbf{C}}'\mathbf{X} = \hat{\mathbf{C}}^{+}\mathbf{X} \tag{7.1}$$

Next, the improved concentrations may be estimated using least-squares method:

$$\hat{\mathbf{C}} = \mathbf{X}\hat{\mathbf{S}}'\left(\hat{\mathbf{S}}\hat{\mathbf{S}}'\right)^{-1} = \mathbf{X}\hat{\mathbf{S}}^{+} \tag{7.2}$$

The calculations in Eqs. (7.1) and (7.2) are repeated until convergence is obtained. In each step the non-negativity condition might be implemented by replacing negative values by zero.

Applying ALS with the non-negativity constrains (program ALS.m in folder ALS in Ex6-8) to the data in Exercise 6.8 gives all positive spectra, Fig. 7.1 (compare with Fig. 6.18). However, errors of the calculated concentrations are larger.

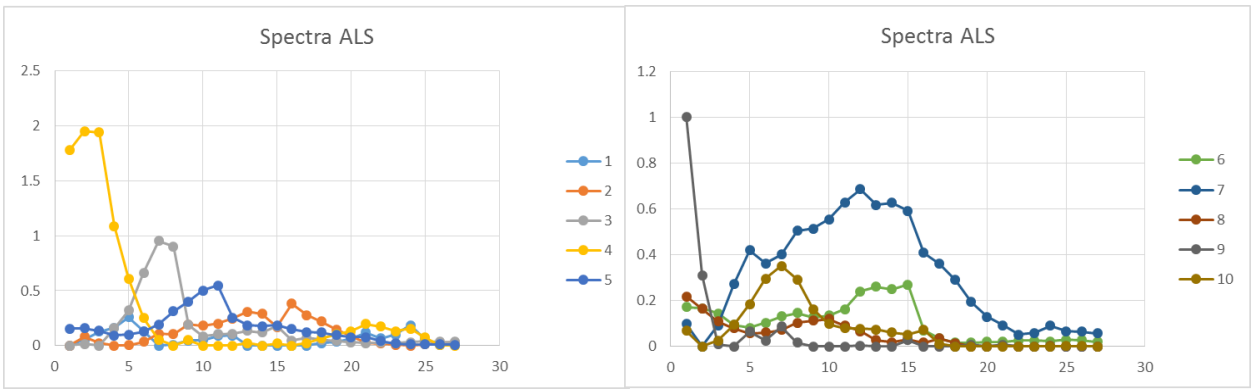


Fig. 7.1. Spectra in Exercise 6.8 calculated using ALS (compare with Fig. 6.18).

8 Multi-way analysis

8.1 Introduction

In the earlier parts of this book, we have considered two-way data described by matrices, e.g. UV/VIS spectra registered at J wavelengths for I samples. However, a single instrument can sometimes generate a table of results **for each sample**, for example in excitation and emission fluorescence spectra, there is a matrix $\mathbf{X}(J \times K)$ of fluorescences at J excitation wavelengths and measured at K emission wavelength for each sample. In this case for each sample one matrix $\mathbf{X}$ is obtained. Taking fluorescence spectra for I samples generates a three-way array $\underline{\mathbf{X}}(I \times J \times K)$. This is a three-dimensional array and to distinguish it from the two-dimensional matrices its symbol is underlined. Such three-dimensional arrays are also called tensors or cubes and may be analyzed using three-way methods. Of course, using higher dimensions (four, five) other multiway data may be obtained.

Another example of multi-way data is in environmental analysis where analytical samples are acquired from different locations (first-way) at multiple times (second way) and each sample is analyzed for several components (third way) producing three-way data cube. Three-way data might also be produced using special analytical methods GC/MS/MS or even using UV/VIS spectra as a function of time for samples of different compositions.

Mathematical methods of dealing with such problem were initially developed for psychometry and social sciences and adopted later to chemometrics.

Most chemists are probably unaware of the power of multi-way analysis. Multi-way analysis provides a new way to look at experimental design and the scientific method itself.[28] There are books on this topic,[28,29] many articles, and a Web sites where presentations, tools for Matlab, and example data are presented[30,31] or even YouTube courses.[32]

The analytical data must be first collected and the problems such as detection limits, missing data, and outliers dealt with. Then the data might be preprocessed differently. These are important problems and their detailed description might be found in the literature.[28]

8.2 Construction and properties of boxes

Let us look first into construction of boxes (or 3D arrays) describing three-way data. Two-way data described earlier are described by matrices, Fig. 8.1.

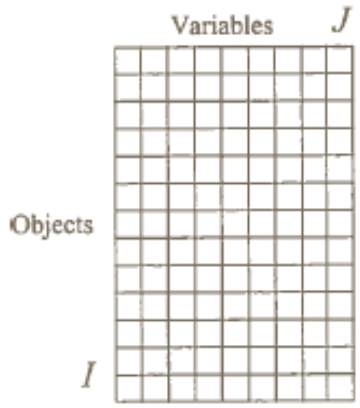


Fig. 8.1. Matrix presentation of two-way data, variables are for example wavelengths and objects are samples.

Such data contain variables, for example J wavelengths, and objects, for example I different samples or measurements in time. These axes are often called Mode 1 for objects (samples) I and Mode 2 for variables (spectra) J. However, for three-way data we have a matrix $\mathbf{X}(J\times K)$ for each analyzed sample, i. Let us suppose that we analyze excitation/emission fluorescence data that is each matrix $\mathbf{X}(J\times K)$ contains two-dimensional excitation/emission spectra for one analyzed sample. Examples of the spectra of tryptophan and mixture of tryptophan, tyrosine and phenylalanine are shown in Fig. 8.2 where there are 61 excitation and 201 emission wavelengths.

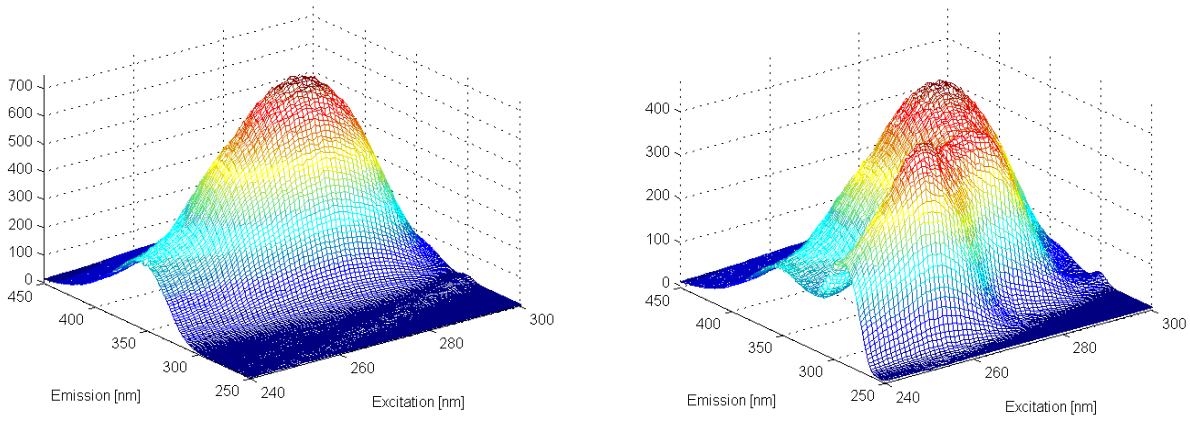


Fig. 8.2. An example of the 2D fluorescence spectrum of tryptophan (left) and a mixture of tryptophan, tyrosine and phenylalanine (right), $\mathbf{X}(61\times 201)$.

This means that for each composition (sample) i there is a matrix $\mathbf{X}(J\times K)$. We can stack I matrices together obtaining a box $\underline{\mathbf{X}}(I\times J\times K)$. This process is displayed in Fig. 8.3.

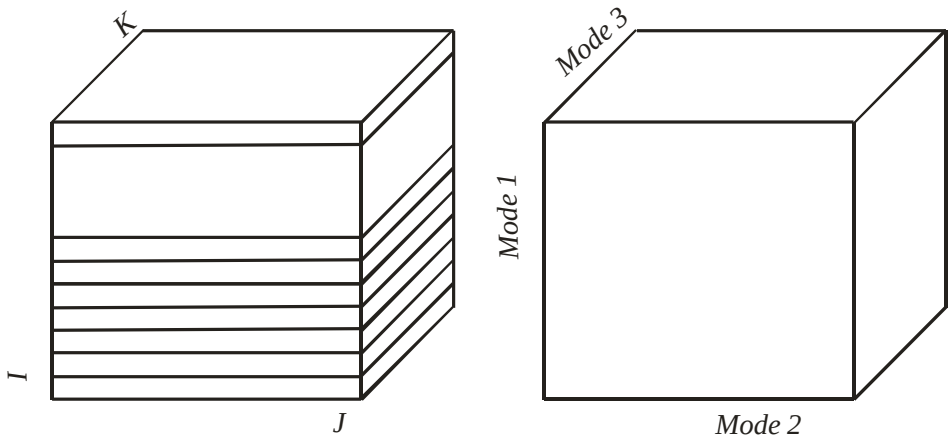


Fig. 8.3. Process of construction of a three-way data array (box).

Now the array X contains three modes. These modes can be samples, Mode 1 (*I*), excitation, Mode 2 (*J*), and emission, Mode 3 (*K*), in fluorescence or in chromatography: samples, spectra, and time. Each element of **X** has three indices, $x_{i,j,k}$. Often, the three way data are treated as two-way data with losing some additional information.

The simplest way of dealing with the 3D data is changing them into 2D matrices. This process is called **unfolding** or **matricization**, Fig. 8.4.

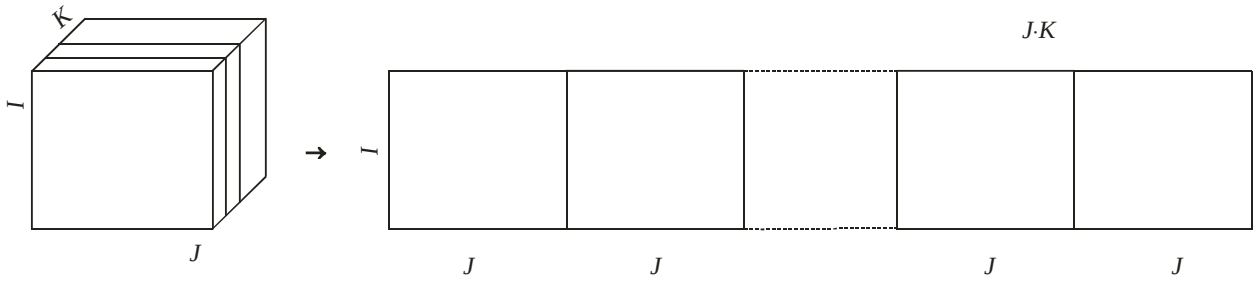


Fig. 8.4. Unfolding (matricization) of the cube $\underline{\mathbf{X}}(I\times J\times K)$ to $\mathbf{X}(I\times JK)$.

It can be noticed that the two-dimensional matrices $\mathbf{X}(J\times K)$ were initially stacked for *I* samples to get a cube (tensor) $\underline{\mathbf{X}}(I\times J\times K)$ but were unfolded as a series of *K* matrices ($I\times J$) to produce extended matrix $\mathbf{X}(I\times JK)$. In this process one large matrix $\mathbf{X}(I\times JK)$ with dimensions $I\times JK$ was created. After unfolding an ordinary two-way PCA can be applied to the matrix $\mathbf{X}(I\times JK)$. This process is illustrated in Exercise 8.1.

Exercise 8.1.

Four 2-dimensional spectra were recorded for different compositions of three component analyses. Therefore, there are four matrices $\mathbf{X}(5\times 6)$, each for different composition. They are displayed in Table 8.1.[3] The corresponding concentrations are shown in Table 8.2.

Table 8.1. Four matrices **X**(5×6) obtained for four different samples.[3]

				1						2						3						4			
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
1	390	421	871	940	610	525	488	433	971	870	722	479	186	276	540	546	288	306	205	231	479	481	314	268
2	635	357	952	710	910	380	1015	633	1682	928	1382	484	420	396	930	498	552	264	400	282	713	427	548	226
3	300	334	694	700	460	390	564	538	1234	804	772	434	328	396	860	552	440	300	240	264	576	424	336	232
4	65	125	234	238	102	134	269	317	708	364	342	194	228	264	594	294	288	156	120	150	327	189	156	102
5	835	308	1003	630	1180	325	1041	380	1253	734	1460	375	222	120	330	216	312	114	385	153	482	298	542	154

Table 8.2. Concentrations of three components A, B, and C, in four samples.[3]

	A	B	C
1	1	9	10
2	7	11	8
3	6	2	6
4	3	4	5

First, four matrices should be stacked together to obtain an array **X**(4×5×6) and unfolded to obtain one matrix **X**(4×5•6) = **X**(4×30). This unfolded matrix is displayed in Table 8.3.

Table 8.3. Unfolded cube **X**(4×5×6) to **X**(4×30).

				1						2						3						4						5			
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
1	390	421	871	940	610	525	635	357	952	710	910	380	300	334	694	700	460	390	65	125	234	238	102	134	835	308	1003	630	1180	325
2	488	433	971	870	722	479	1015	633	1682	928	1382	484	564	538	1234	804	772	434	269	317	708	364	342	194	1041	380	1253	734	1460	375
3	186	276	540	546	288	306	420	396	930	498	552	264	328	396	860	552	440	300	228	264	594	294	288	156	222	120	330	216	312	114
4	205	231	479	481	314	268	400	282	713	427	548	226	240	264	576	424	336	232	120	150	327	189	156	102	385	153	482	298	542	154

The classical PCA and PLS might be applied to the unfolded matrix. PLS analysis shows that there are only three principal components, in agreement with the number of chemical components. PLS analysis shows that the approximation reproduces practically exactly the experimental concentrations.

Unfolding introduces many variables; in this exercise 30 but in the case displayed in Fig. 8.2 unfolding of 6 samples produces matrix **X**(5×12261) where 61 excitation and 201 emission wavelengths produces 61×201 = 12261 variables! Besides, during unfolding some information contained in 3D **X(**5×61×201) array is lost.

Below some properties of cubes (or tensors will be presented).

8.3 Rank

As it was stated in Section 2.3 rank of two-way matrix is a minimal number of PCA components needed to reproduce matrix exactly. In two-way matrices the row rank equals to column rank equals to the rank. But this is not the case in three-way arrays.

Rank of three-way array is a minimum number of trilinear components needed to reproduce the experimental array **X**. Rank of random 2 × 2 matrix is always 2. For random 2×2×2 array the rank might be 2 or 3. Rank of cubes is not defined in the same way as of matrices. For larger random arrays e.g. 9×9×9 nobody knows what its rank might be.[32]

8.4 Three-way PARAFAC model

PARAFAC (**para**llel **fac**tor analysis) model was developed in 1970 by Harshman.[33] It is an extension of the PCA analysis to three-dimensional arrays (cubes). In PCA a matrix is decomposed as a sum of vector products (vertical scores and horizontal loadings), Fig. 3.2. For three-way data the 3D array is decomposed as a sum of triple products of vectors, Fig. 8.5.

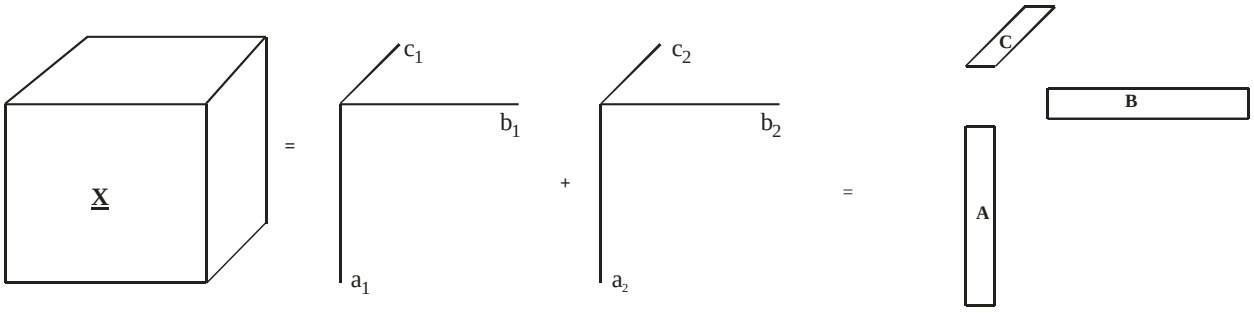


Fig. 8.5. Illustration of the decomposition of the three-way array $\underline{\mathbf{X}}(I \times J \times K)$ into three loadings **A**(*I*×*R*), **B**(*J*×*R*) and **C**(*K*×*R*) containing two factors (*R* = 2); for simplification the error array was omitted.

The elements of three-way array can be calculated as:

$$x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} + e_{ijk} \tag{8.1}$$

where *R* is called the pseudo-rank of $\underline{\mathbf{X}}(I \times J \times K)$. It is defined as the smallest number of trilinear PARAFAC components needed to fit $\underline{\mathbf{X}}$ without fitting the noise. By analog with Eq. (3.3) one can write for three-way data:

$$\underline{\mathbf{X}} = \underline{\hat{\mathbf{X}}} + \underline{\mathbf{E}} \tag{8.2}$$

where $\underline{\hat{\mathbf{X}}}$ is the calculated array using *R* PARAFAC components. Matrices **A**, **B**, and **C** are called in PARAFAC **model loadings** although one of them represents scores. The biggest advantage of the PARAFAC model in comparison with the PCA model is **uniqueness** of the solution (decomposition). In PCA there is rotational ambiguity, Eq. (3.12) and Fig. 3.3. However, in the decomposition of 3D arrays (cubes) there is no rotational degree of freedom and no ambiguity, therefore, we can get the real spectra and concentrations. The obtained model is the best model in the least-squares sense.

The **only non-uniqueness** that remains in a unique multilinear model is in **scaling and permutations** of factors. This means that scores (concentrations) and spectra (loadings) correspond to the real concentrations multiplied by a constant. Besides, one should find out which model component (and the corresponding spectra) correspond to which chemical component. This can be found out if the spectra of individual components are known (e.g. in fluorescence or chromatography) or one knows the order of elution in chromatography. Another propriety of PARAFAC is that the **loadings are not orthogonal** (but unique) and the subsequent **components do not decrease** as in PCA, see Fig. 3.5. If another factor is added to the model the whole model must be recalculated (in two-way PCA only new component is added without

changing the other, as they are orthogonal). Uniqueness of the solution means that the mixtures of the analytes can be separated and the concentrations and pure spectra or concentration profiles (in chromatography) determined.

The PARAFAC model is also **much less sensitive to the random noise** (see Section 8.8). Due to its uniqueness properties the PARAFAC model is ideally suited for curve resolution and certain kinds of calibration problems. Since the PARAFAC model is unique and coincides with several physical models (fluorescence spectroscopy, spectrally detected chromatography, etc.) it is possible to decompose such data into (chemically) meaningful parameters.

The condition of uniqueness of the PARAFAC model is related to the so called ***k*-rank**.[28,32] If R is the number of components, for loading **A**, k_A-rank is the maximal number of randomly chosen columns which have full rank ($\leq R$). It is never higher than the rank. The PARAFAC model is unique when sum of three k-ranks of three loadings fulfils the condition:

$$k_A + k_B + k_C \geq 2R + 2 \tag{8.3}$$

Of course, **A**, **B**, and **C** must vary adequately (replication of the same measurements is not good) for the model uniqueness. For example, 8-component PARAFAC model of a 6×6×6 array is unique (assuming that k-rank of each matrix is six) and 6 samples may furnish unique information about eight components (6 + 6 + 6 ≥ 2*8 + 2 =18). Of course in the classical PCA analysis from 6×6 matrix it is possible to extract at most 6 elements.

Let us compare a PARAFAC model of a cube $\underline{\mathbf{X}}(10\times100\times30)$ and its unfolded matrix (in the first mode) $\mathbf{X}(10\times3000)$. Each component of the cube has 10+100+30 = 140 parameters while the components of the unfolded matrix have 10+3000 = 3010 parameters. Therefore, there are many more parameters in the PCA model than in the PARAFAC model. For example, if there are three components, PARAFAC can model it easily with three components. PCA model can also model the unfolded matrix with three components but it will use many more parameters and might overfit the solution and the solution will be much more sensitive to noise.

Because the array $\underline{\mathbf{X}}$ is three-dimensional it is difficult to represent it in a matrix notation. Eq. (8.1) can be written for the matrix $\mathbf{X}_k$, Fig. 8.4 left, as:

$$\mathbf{X}_k = \mathbf{A}\mathbf{D}_k\mathbf{B}' + \mathbf{E} \tag{8.4}$$

where $\mathbf{D}_k$ is a diagonal matrix with the k^{th} row of **C** on its diagonal (elements: $c_{k,1}$, $c_{k,2}$,…, $c_{k,R}$). Transformation of the elements of matrix **C** into $\mathbf{D}_k$ is displayed below in Fig. 8.6. The elements different from zero are only on the diagonal of $\mathbf{D}_k$.

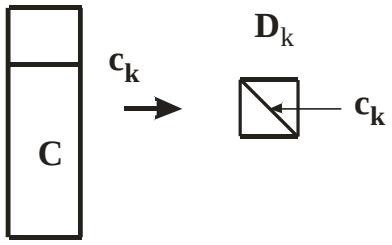


Fig. 8.6. Formation of the diagonal matrix $\mathbf{D}_k$ from **C.**

PARAFAC model works by sequential optimization of the loadings **A**, **B**, and **C** using **ALS** (alternating least-squares. Chapter 7) to minimize the differences between the experimental and calculated arrays:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \underline{\mathbf{X}} - \underline{\hat{\mathbf{X}}} \right\|^2 = \min_{\mathbf{A},\mathbf{B},\mathbf{C}} \sum_{r=1}^{R} \left\| \mathbf{X}_r - \hat{\mathbf{X}}_r \right\|^2 = \sum_{r=1}^{R} \left\| \mathbf{E}_r \right\|^2 \tag{8.5}$$

Starting from some initial values of **B** and **C** (guessed or random) the new values of matrix **A** are estimated:

$$\mathbf{A} = \left(\sum_{k=1}^{K} \mathbf{X}_k \mathbf{B} \mathbf{D}_k \right) \left[(\mathbf{B}'\mathbf{B}) * (\mathbf{C}'\mathbf{C}) \right]^{-1} \tag{8.6}$$

Then, elements of **B** and $\mathbf{D}_k$ (that is of matrix **C**) using:

$$\begin{gathered} \mathbf{B} = \left(\sum_{k=1}^{K} \mathbf{X}'_k \mathbf{A} \mathbf{D}_k \right) \left[(\mathbf{A}'\mathbf{A}) * (\mathbf{C}'\mathbf{C}) \right]^{-1} \\ \operatorname{diag} \mathbf{D}_k = \left[(\mathbf{B}'\mathbf{B}) * (\mathbf{A}'\mathbf{A}) \right]^{-1} \operatorname{diag} (\mathbf{A}'\mathbf{X}_k\mathbf{B}), \quad k = 1,\ldots,K \end{gathered} \tag{8.7}$$

where operator "*" is the Hadamard product (element-wise product) of two matrices **A(***I*,*J*) and **B**(*I*,*J*) defined as:

$$\mathbf{A} * \mathbf{B} = \begin{bmatrix} a_{1,1}b_{1,1} & . & . & . & a_{1,J}b_{1,J} \\ . & & & & . \\ . & & & & . \\ . & & & & . \\ a_{I,1}b_{I,1} & . & . & & a_{I,J}b_{I,J} \end{bmatrix} \tag{8.8}$$

Now. Eq. (8.5) can be rewritten as:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \underline{\mathbf{X}} - \underline{\hat{\mathbf{X}}} \right\|^2 = \min_{\mathbf{A},\mathbf{B},\mathbf{C}} \sum_{r=1}^{R} \left\| \mathbf{X}_r - \mathbf{A}\mathbf{D}_k\mathbf{B}' \right\|^2 = \sum_{r=1}^{R} \left\| \mathbf{E}_r \right\|^2 \tag{8.9}$$

Operations in Eqs. (8.6)-(8.7) are repeated until the parameters do not change and the sum of squares, Eq. (8.9), is at minimum.. Because PARAFAC model is based on the least-squares principle it might be sensitive to the initial choice of parameters and local minima might be found. It stops when no further changes are observed. It is possible in the PARAFAC program to decrease relative change criterion from default 10^{-6} to a smaller value or change the initial parameters (program allows for such changes in Option(1)). There are also other algorithms described in the literature. The starting values might be selected using Options(2). By default DTLD/GRAM method is used but with Options(2)=2 fit is using random orthogonalized values for initialization, that is each time new set of initial parameters is selected. One should keep in mind that if there are too many parameters the model might not converge or converge each time to a different solution. One of the tools used to estimate number of parameters is **split-half analysis** where the original data are split in half (sequential, blocks); both halves should produce the same loadings. Another method is core consistency described later.

One of the problems one can meet while using PARAFAC model is **two factor degeneracy**. It is a mathematical artifact not related to the "bad data". It appears when two components have the same value but opposite signs in one mode. Because of that they might cancel each other. This might cause correlation between loadings in other modes. It happens when PARAFAC model does not exist, e.g. a three and five component models might exist but not four component model.[28,32] With increasing number of iterations the problem will increase. Sometimes changing

of preprocessing of data or using non-negativity constrain can help. Alternatively, other model as Tucker model might be applied.

Before presenting more details about the PARAFAC model some applications will be presented which allow better understanding of the further discussion.

Exercise 8.2.

Apply PARAFAC model to the data in Exercise 8.1.

In Exercise 8.1 the 3D array was unfolded to apply classical PCA and PLS method. In this exercise we will use the experimental array **X**(4×5×6) in PARAFAC model. All the data, program file fac2.m and Excel file are in the folder Ex7-2. In was found earlier that there are three principal components corresponding to three chemical components. Program fac2.m first generates 3D graphs of four spectra. They are displayed in Fig. 8.7. They represent UV/VIS spectra in function of time.

Program stops after 80 iterations (it=80). The calculated loadings in three modes (i.e. **A**, **B**, **C**) are shown in Fig. 8.8. As there are four samples Mode #1 relates to the concentrations of three components that is the scores which are proportional to the concentrations. However, we do not know which column of scores corresponds to which component, see Table 8.4.

Inspection of scores shows that all the samples contain mixtures of three components. However, the second column decreases monotonically with the sample number and it might correspond to the component C, compare with Table 8.2. In the third column the first value is the smallest which suggests that it is component A. Similarly, the first column of scores corresponds to the component B. Plot the scores vs. analytical concentrations is presented in Fig. 8.9. The relations are linear which confirms that the attribution of scores to concentrations is correct.

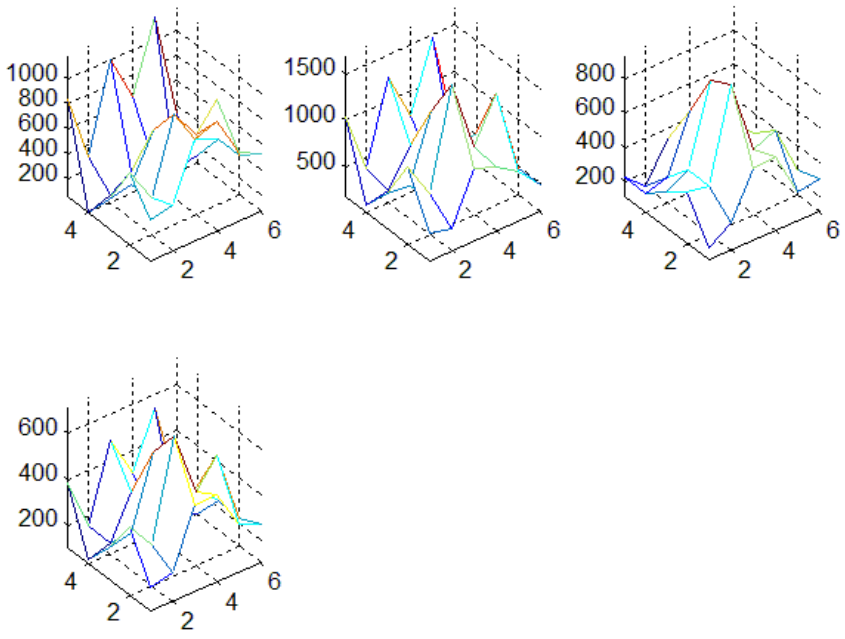


Fig. 8.7. 3D graphs of four spectra in Exercise 8.2.

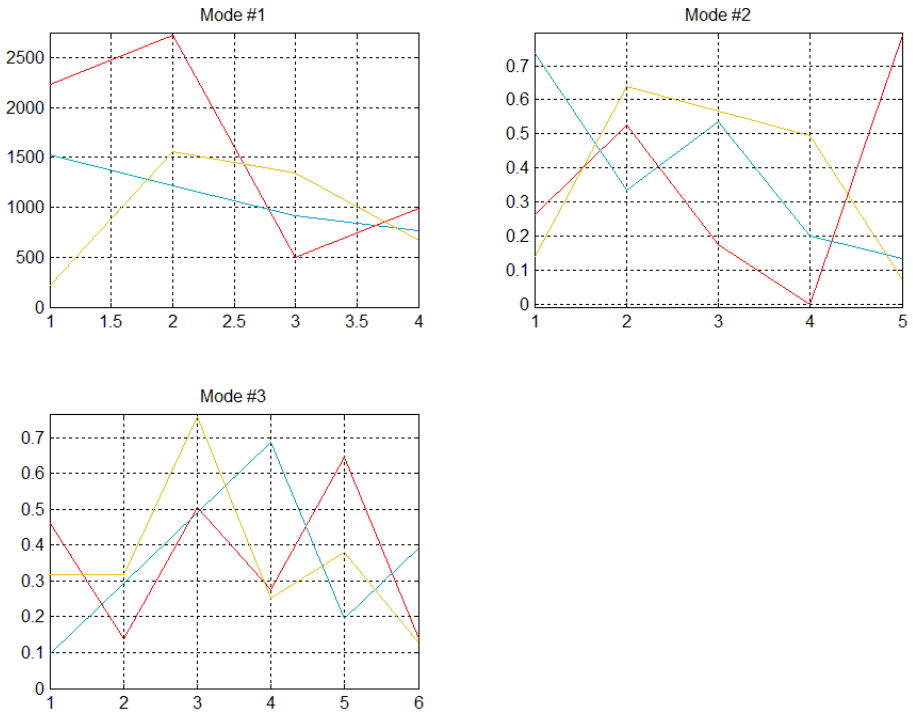


Fig. 8.8. Loadings A, B, and C (modes #1, #2, and #3, respectively) obtained from the PARAFAC model.

Table 8.4. Loadings A/1000 (scores) obtained from the PARAFAC model.

.

2.227027718	1.522882553	0.223054
2.721920171	1.218301956	1.561337
0.494896052	0.91372946	1.338287
0.989790039	0.761440685	0.669145

Attributed to concentrations:

B	C	A

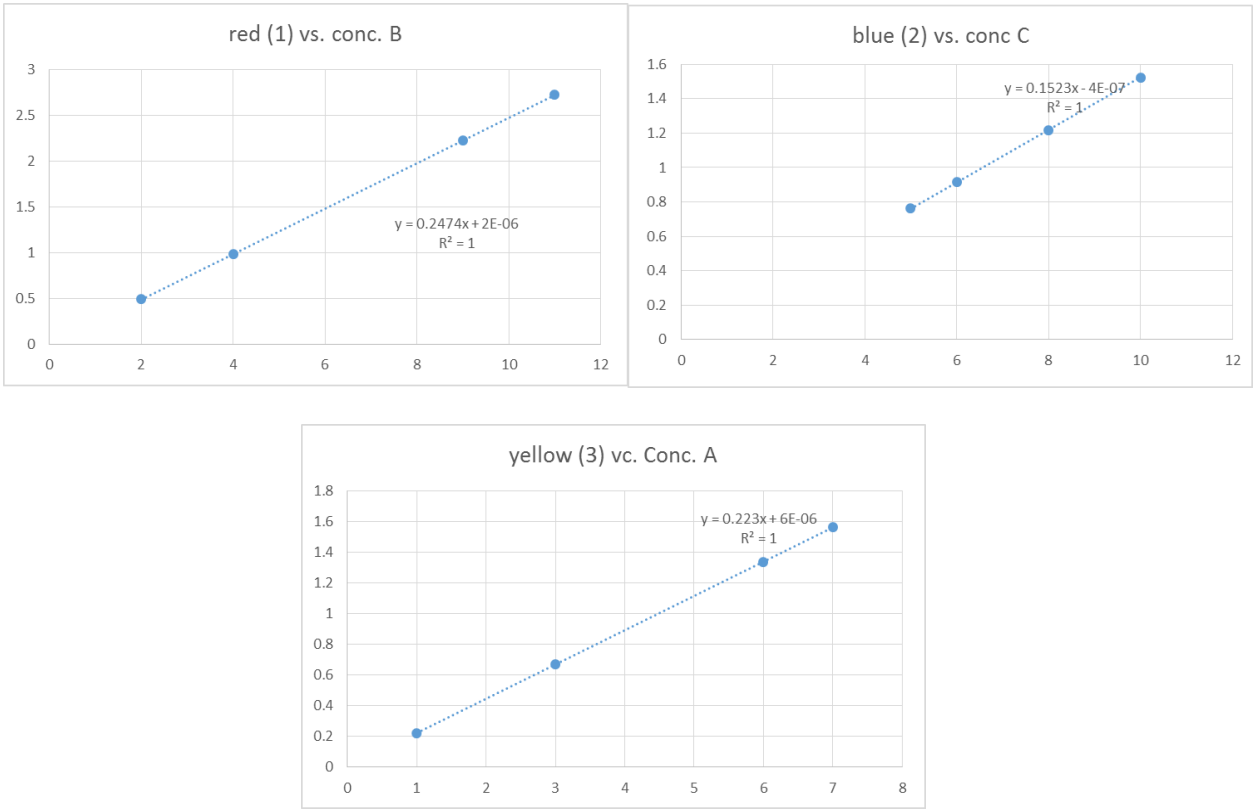


Fig. 8.9. Plots of the scores (divided by 1000) versus analytical concentrations; in this case the first scores (red curve in Fig. 5.8) correspond to the component B, second scores (blue) correspond to C, and third (yellow) scores correspond to A.

The correlations between the calculated scores (Mode #1) and concentrations are very good. The obtained regression equations allow to auto-predict concentrations. The predicted concentrations are displayed in Table 8.5. Comparison with Table 8.2 shows that the errors are very small.

Table 8.5. Auto-predicted concentrations of three components calculated from the relations between the scores and concentrations.

A	B	C
1.000006	9.000005	10.000010
7.000008	10.999997	7.999982
5.999994	2.000000	6.000006
2.999992	3.999998	5.000002

This analysis allowed also for the identification of the scores and chemical elements. It is interesting to notice that we did not need to know concentrations of the components as functions of time but the concentrations of the mixture injected.

8.5 Second order calibration

Second-order calibration[28,34] is a simple method of estimation of the concentrations of one component in a mixture. In simple terms if the concentration of the component in one sample is known it is possible to calculate concentrations of this component in all samples used in the PARAFAC model.

Let us suppose, that in Exercise 8.2 concentration of the component A (third column in scores in Table 8.4) is known for the first sample; from Table 8.2, $c_A(1) = 1$. Concentrations of A in other samples can be determined using simple proportionality. For example, concentration in other samples is:

$$c_A(i) = \frac{a_{3,i}}{a_{3,1}} c_A(1) \tag{8.10}$$

where $a_{3,i}$ are the elements of the third column of the scores matrix **A**. The results obtained are displayed in Table 8.6 and compared with the analytical values. The agreement is very good.

Table 8.6. Comparison of the concentrations of species A estimated using second order analysis with the analytical concentrations of A.

c A predicted	c A analytical
1.0000	1.0000
6.9998	7.0000
5.9998	6.0000
2.9999	3.0000

Similarly, if we know concentrations of other species B and C in the first sample, see Table 8.2, it is possible to predict concentrations in other samples using analog of Eq. (8.10) for B and C. Predicted concentrations in samples 2-4 are displayed in Table 8.7.

Table 8.7. Concentrations of species B and C predicted from the concentrations in the first sample using second order calibration.

c B predicted	c C predicted	c B analytical	c C analytical
9.00000	10.00000	9.00000	10.00000
10.99999	7.99997	11.00000	8.00000
2.00000	6.00000	2.00000	6.00000
4.00000	5.00000	4.00000	5.00000

In this case the agreement is excellent because the data **X** were simulated using these concentrations without adding random errors.[3]

8.6 Determination of the number of factors

Determination of the number of factors (principal components) in three way-analysis is different than in two-way analysis. Cross-validation used in PCA analysis is rarely used in the literature for three way data.

If two sets of data exist or one large set of data is divided in two parts then for the same number components PARAFAC model should produce the same loading vectors (except scores related to the concentrations). Of course different scaling and permutation of loadings can occur and must be taken into account.

Another method is the core consistency test.[28-32] Tucker proposed another decomposition of the array $\underline{\mathbf{X}}(I \times J \times K)$ into three loadings **A**(I×R), **B**(J×R) and **C**(K×R) and an array $\underline{\mathbf{G}}(R \times R \times R)$. This process is visualized in Fig. 8.10 and should be compared with PARAFAC decomposition, Fig. 8.5. The elements of array $\underline{\mathbf{X}}$ are calculated as (compare with Eq. (8.1):

$$x_{ijk} = \sum_{p=1}^{R} \sum_{q=1}^{R} \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} g_{pqr} + e_{ijk} \tag{8.11}$$

The difference between these two models is in the presence of the cube $\underline{\mathbf{G}}(R \times R \times R)$ in Tucker3 model. Of course, when the superdiagonal elements in $\underline{\mathbf{G}}$ (indicated as a dashed line in Fig. 8.10) are all equal to 1 and all other elements are 0, PARAFAC and Tucker3 models are identical.

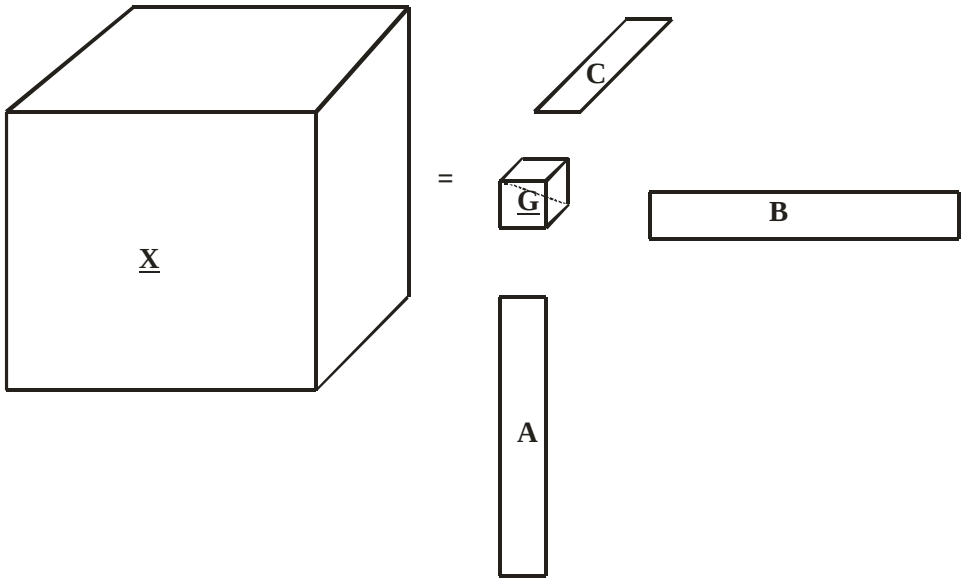


Fig. 8.10. Decomposition of the cube $\underline{\mathbf{X}}$ according to Tucker3 model, for simplification the error array was omitted.

However, when the number of components is too large and the noise is being fitted then the off-superdiagonal components will appear in array $\underline{\mathbf{G}}$. If all the core superdiagonal elements are one the core consistency is 100% and the PARAFAC model approximates data ideally. If the superdiagonal elements are different from one and off-diagonal elements different from zero

appear in **G** the core consistency is <100% (might also be negative). Core consistency is defined as:[35,36]

$$\text{Core Consistency} = 100\left[1 - \frac{\sum_{p}^{R}\sum_{q}^{R}\sum_{r}^{R}\left(g_{pqr} - t_{pqr}\right)^2}{R}\right] \tag{8.12}$$

where **T**($R\times R\times R$) is an array with superdiagonal equal to 1 and off-superdiagonal elements equal to zero.

Low values of core consistency mean that the model is invalid. For low number of model components the core consistency is 100% but if the number of components is too large core consistency becomes low. The general method is to start with low value of number of components, e.g. 1, and then increase it until core consistency becomes low. At the same time number of iterations increases. One can also find that starting with different initial values will produce different sets of loadings for the same number of components. When the number of components is not correct the loadings representing spectra of the chemical components will show incorrect spectra of the components. Such spectra should be compared with the spectra of pure components if possible.

The core consistency parameter is called "**corcondia**" in PARAFAC program.

Results obtained with different number of components from one to four (there are only four data sets) are shown in Table 8.8 and Fig. 8.11-8.14 where err is the sum of squares of approximation errors (3D analog of RRS_R, Eq. (3.14)).

Table 8.8. Results of the approximations of the data in Exercise 8.2 with different number of components, *R*, repeated for *R* = 4.

R	iterations	err	corcondia
1	4	2.12E+06	100
2	44	6.32E+05	99.9972505291239
3	215	7.02E-06	99.9999999952297
4	117	7.98E-06	-
4	97	6.18E-06	-

First, one can notice that with the increase of the number of factors the sum of squares err decreases from *R* = 1 to 3 and then does not change much. However, very large decrease is observed by going from two to three factors.

The loadings and core consistency factors (corcondia) are displayed in Fig. 8.11-8.14 for numbers of factors from one to four. Notice, that corcondia is 100% for one factor and cannot be calculated four factors (there are four samples only).

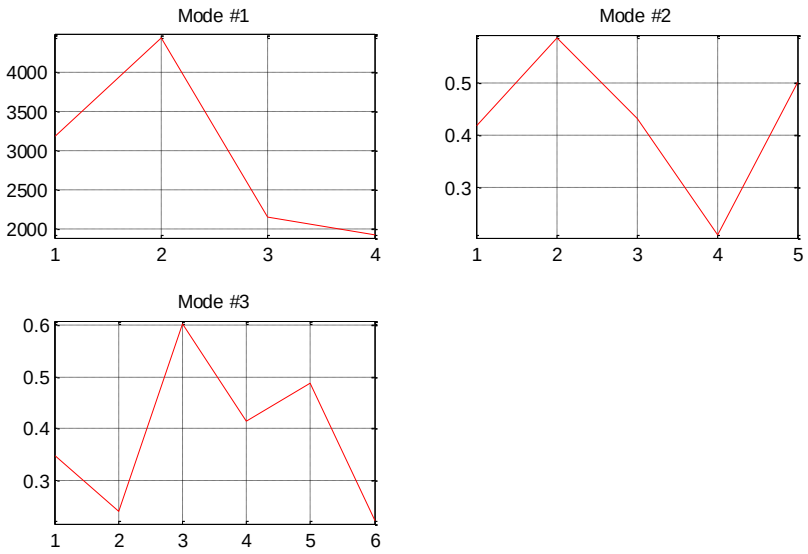


Fig. 8.11. Loadings calculated for one factor.

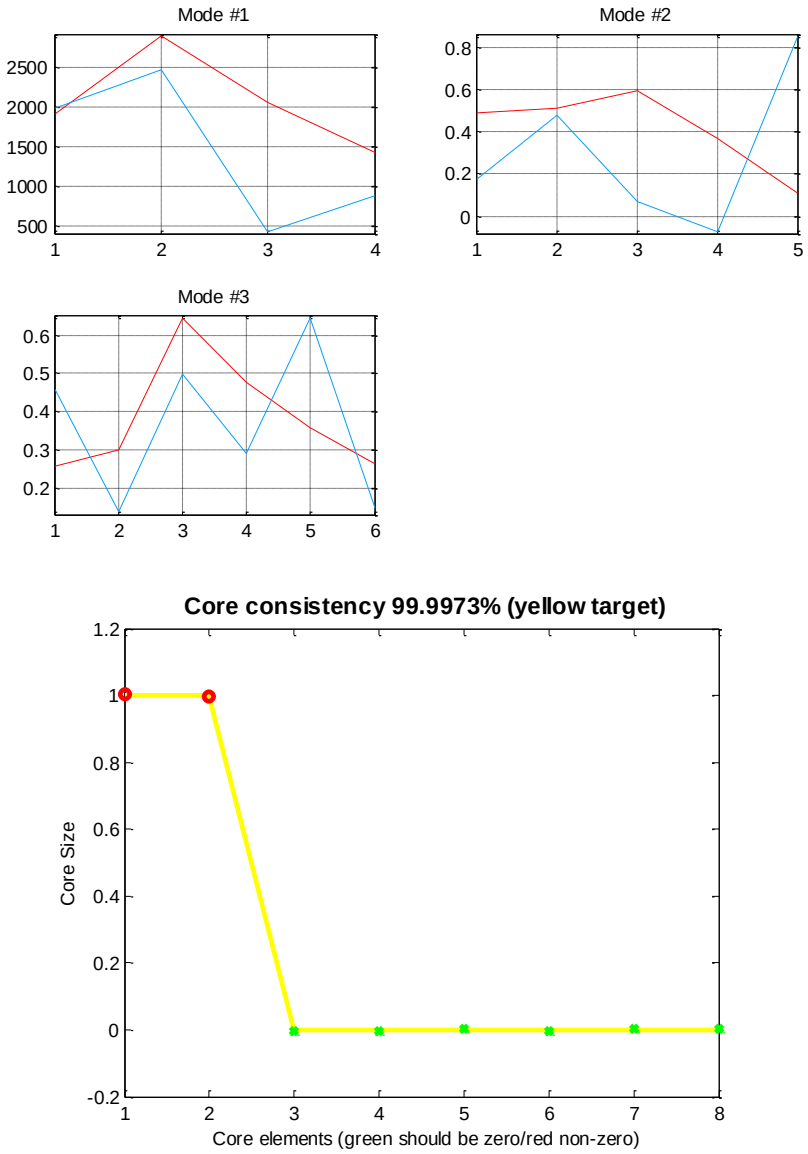


Fig. 8.12. Loadings and core consistency calculated assuming two factors.

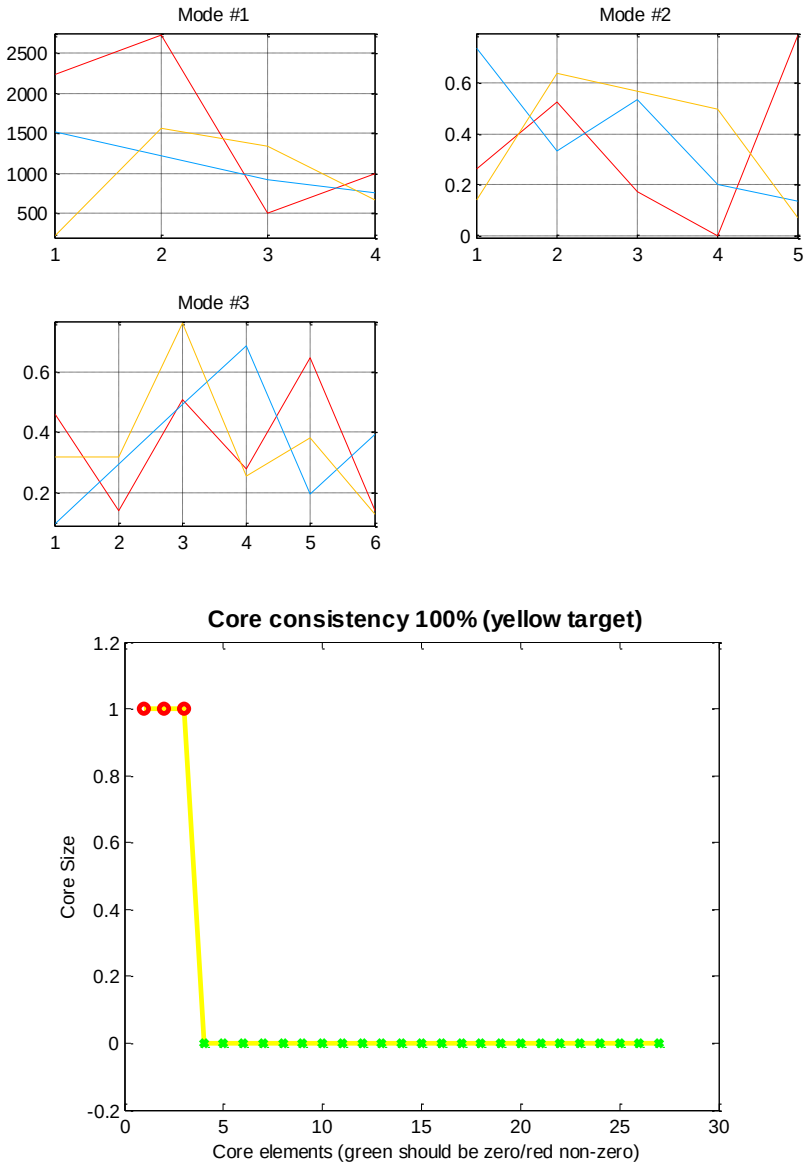


Fig. 8.13. Loadings and core consistency calculated assuming three factors.

PARAFAC can model data with one to four factors. In each case scores and loadings were calculated. However, for four factors the obtained solutions are different for different starting values although the sum of squares (err) stays similar. The program finds numerous local minima which suggests that that number of factors is too large and three factors should be used.

The core consistency plots for two and three factors are close to 100% but cannot be determined here for four factors.

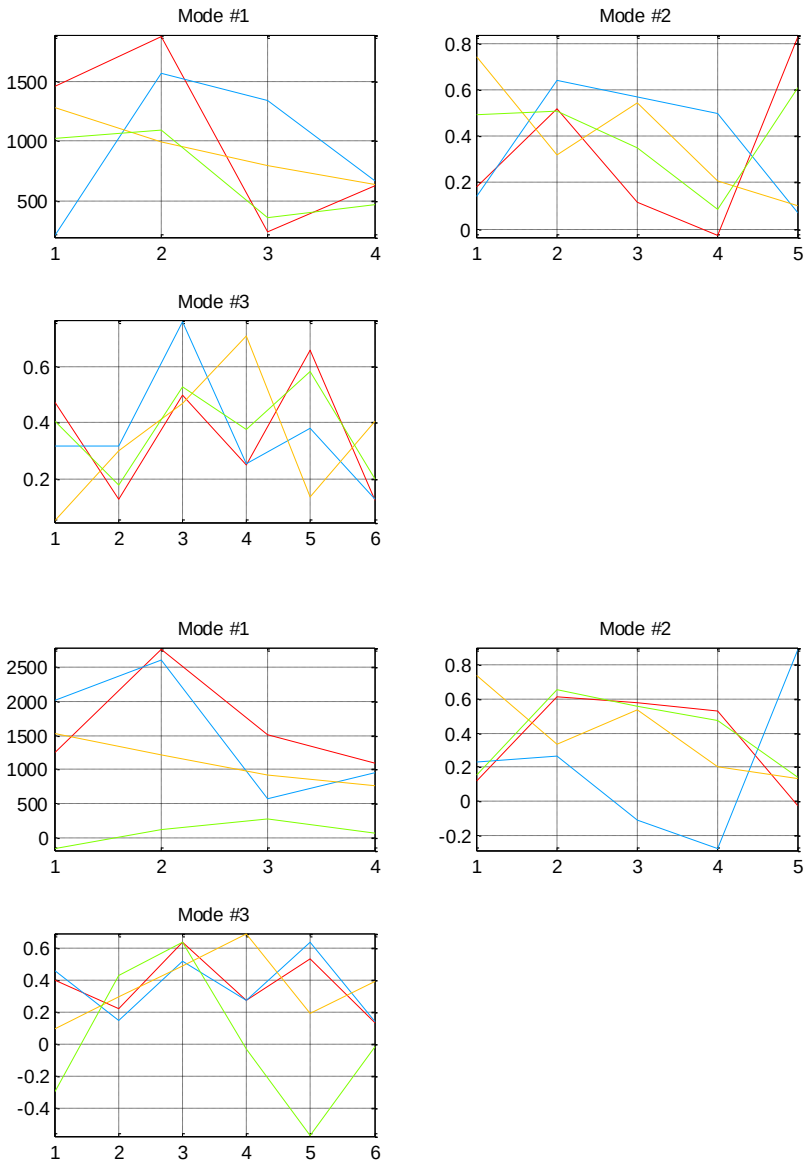


Fig. 8.14. Loadings calculated assuming four factors; PARAFAC model was started with different initial values.

Exercise 8.3.

Five spectra of excitation-emission fluorescence of pure components: tryptophan, phenylalanine, and tyrosine, and their mixtures are in datafile claus.mat.[30,31] This data file consists of array $\underline{\mathbf{X}}(5\times201\times61)$ containing fluorescence spectra at 61 excitation and 201 emission wavelengths for 5 samples, concentration matrix $\mathbf{y}(5\times3)$, scale wavelengths for emission EmAx and excitation ExAx axes and text files; readme contains legend to the concentrations matrix. To see what is in this file perform load and whos commands, see Table 8.9. The plot of the fluorescence spectra in array $\underline{\mathbf{X}}$ are shown in Fig. 8.15. The corresponding concentrations are shown in Table 8.10. It should be noticed that the first three samples contain pure components, however PARAFAC model does not use this information directly and five different mixtures could be used instead.

Table 8.9. Contains of the Matlab data file claus.mat.[30]

```
>> load claus
>> whos
  Name        Size                Bytes  Clas     Attributes

  EmAx        1x201                1608  double
  ExAx        1x61                  488  double
  X           5x201x61           490440  double
  evalme      1x229                 458  char
  readme      3x17                  102  char
  y           5x3                   120  double

>> readme
readme =

1.Column Trp in M
2.Column Tyr in M
3.Column Phe in M
```

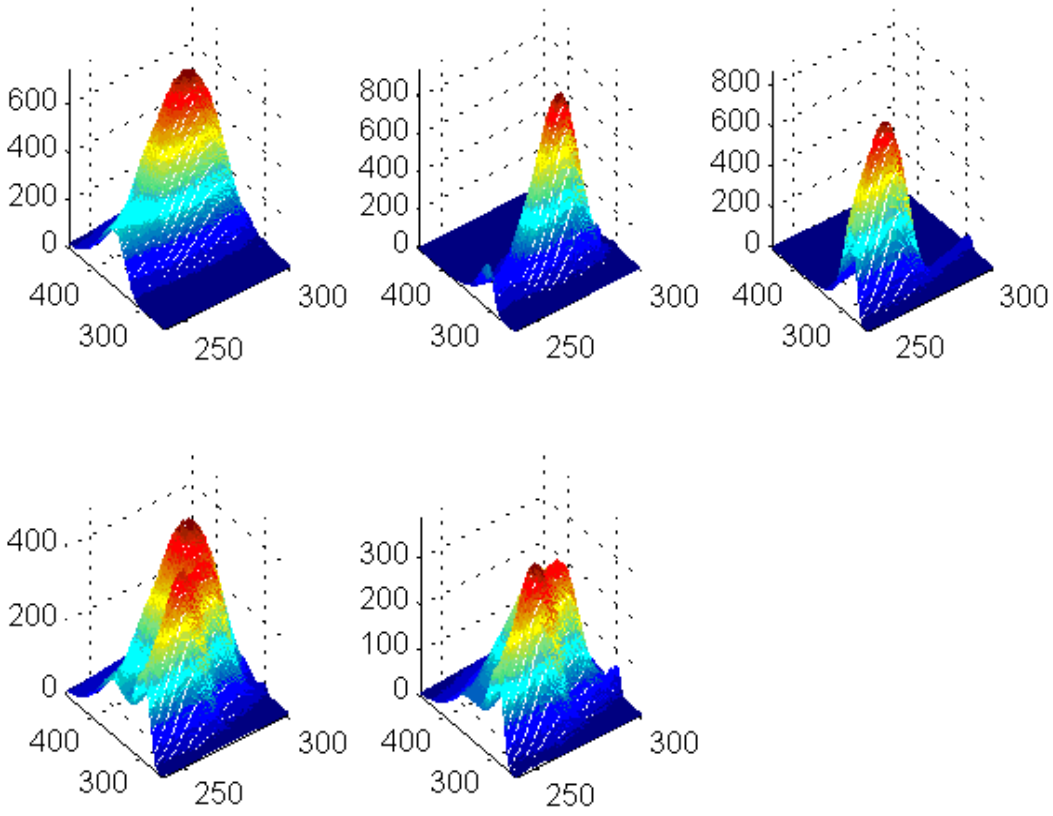


Fig. 8.15. Fluorescence spectra of tryptophan, phenylalanine, and tyrosine, and their mixtures in data file claus.mat.

Table 8.10. Concentrations of tryptophan (A), tyrosine (B), and phenylalanine (C) in mM.

A	B	C
0.0027	0	0
0	0.0133	0
0	0	0.9
0.0016	0.0054	0.355
0.0009	0.0044	0.297

PARAFAC program was executed for one to four factors using program flsc1.m and the results (loadings and core consistency) are displayed below. Some parameters characterizing the modeling are also shown in Table 8.11.

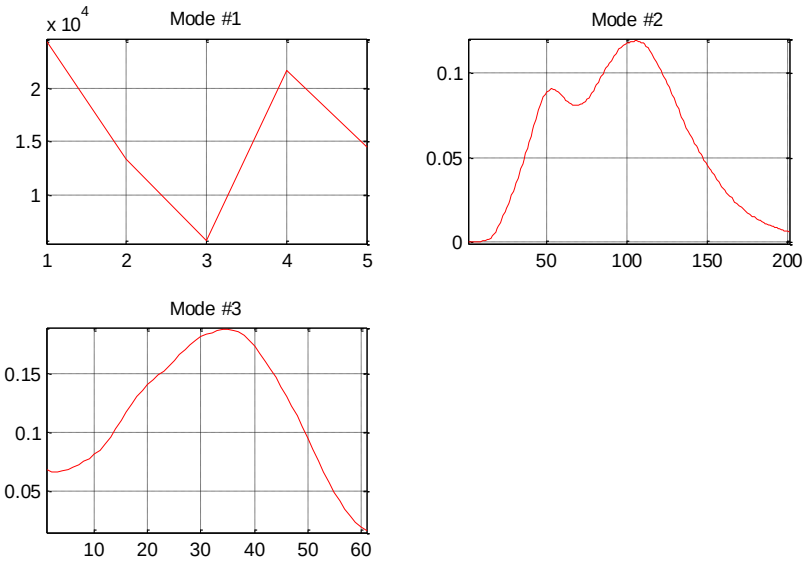


Fig. 8.16. Loadings obtained assuming one component.

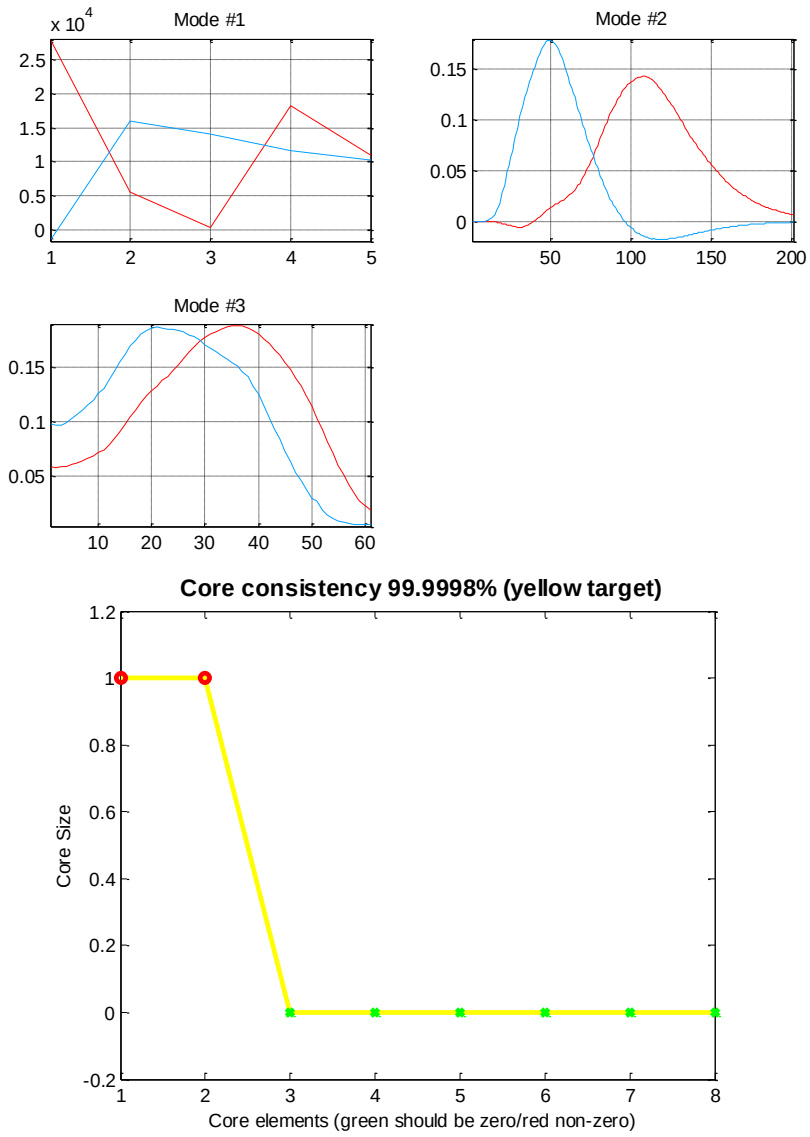


Fig. 8.17. Loadings and core consistency obtained assuming two components.

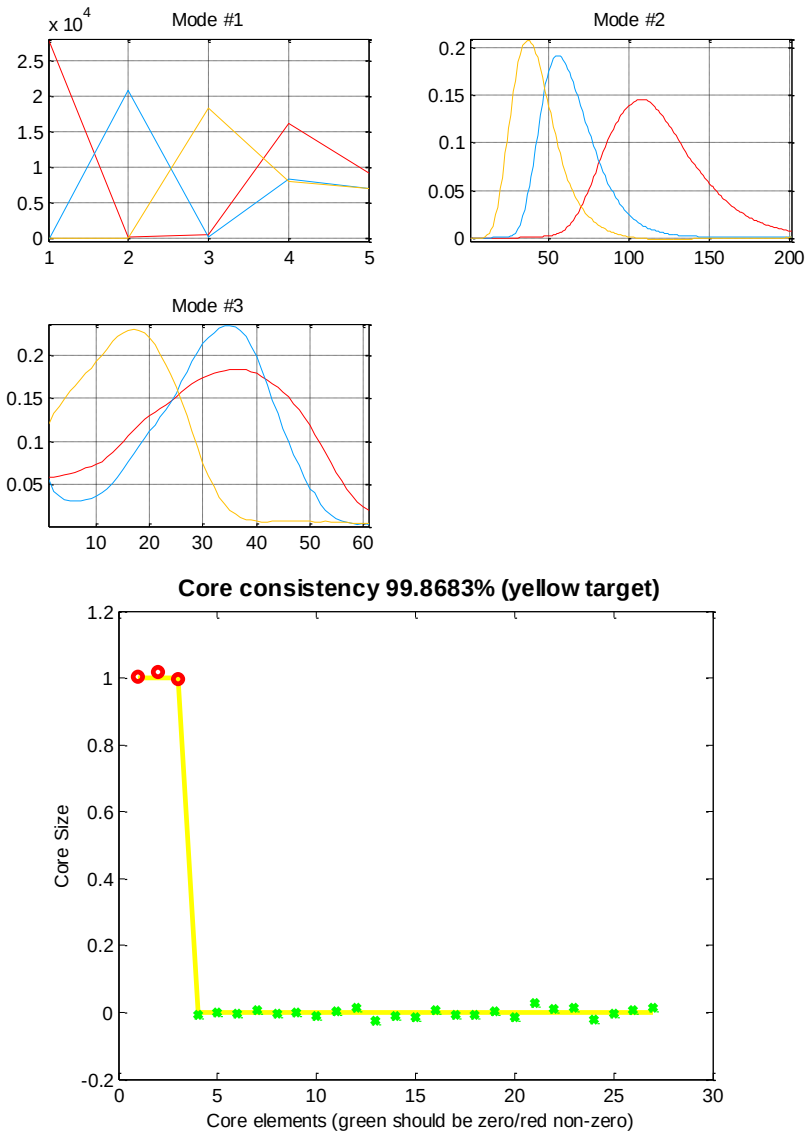


Fig. 8.18. Loadings and core consistency obtained assuming three components.

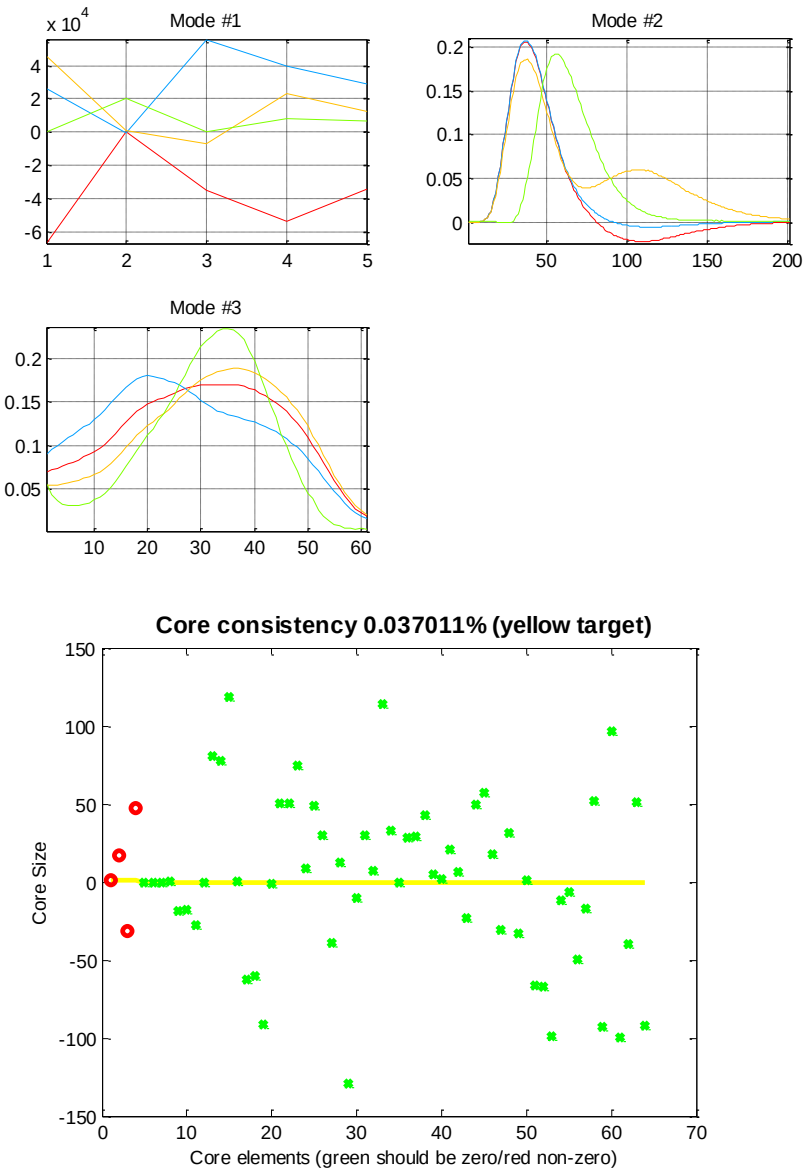


Fig. 8.19. Loadings and core consistency obtained assuming four components.

Table 8.11. Results of PARAFAC analysis for different number of components.

Factors	1	2	3	4
iterations	7	42	460	2048
SSQ	8.20E+08	3.05E+08	1.45E+06	1.32E+06
corcondia	100	99.9995	99.8683	0.037

PARAFAC model can be run for different number of components. However, although the residual sum of squares decreases with the number of factors core consistency (corcondia) stays ~100% up to three components and then decreases dramatically and is close to zero for four components. Besides, while increasing number of components from three to four the number of iterations increases. After repeating the modeling for four components program often cannot find minimum (it stops at 2500 iterations) or gives warning: “factors are highly correlated, decrease the number of components”. It also gives completely different loadings suggesting that some local minima were found. Repetition is important and in the case of doubt repetition of calculations with different initial values (Option(2)) should be carried out to check if the same solution is found each time. The obtained results indicate that only three components should be used, in agreement with the number of species.

When deciding number of factors one can compare excitation/emission fluorescence spectra with those of pure components to see if they are well reproduced.

Assuming that the concentrations of the three pure components are known concentrations of all components in other samples might be found using second order calibration. The obtained results are displayed in Table 8.12. Standard deviations of the first two components are very small but they are larger for the component III, compare with Table 8.10.

Table 8.12. Concentrations of three components auto-predicted using second order calibration and their standard deviations.

Component	I	II	III
	0.002700	-0.00008	-0.002
	0.000014	**0.01330**	0.000
	0.000048	0.00015	**0.900**
	0.001564	0.00536	0.392
	0.000889	0.00445	0.340
std	0.000036	0.00010	0.033
std%	3.5%	2.2%	10%

Concentrations can also be auto-predicted (if all concentrations are known) using linear regression between the scores and concentrations. Graphs for three components are shown in Fig. 8.20.

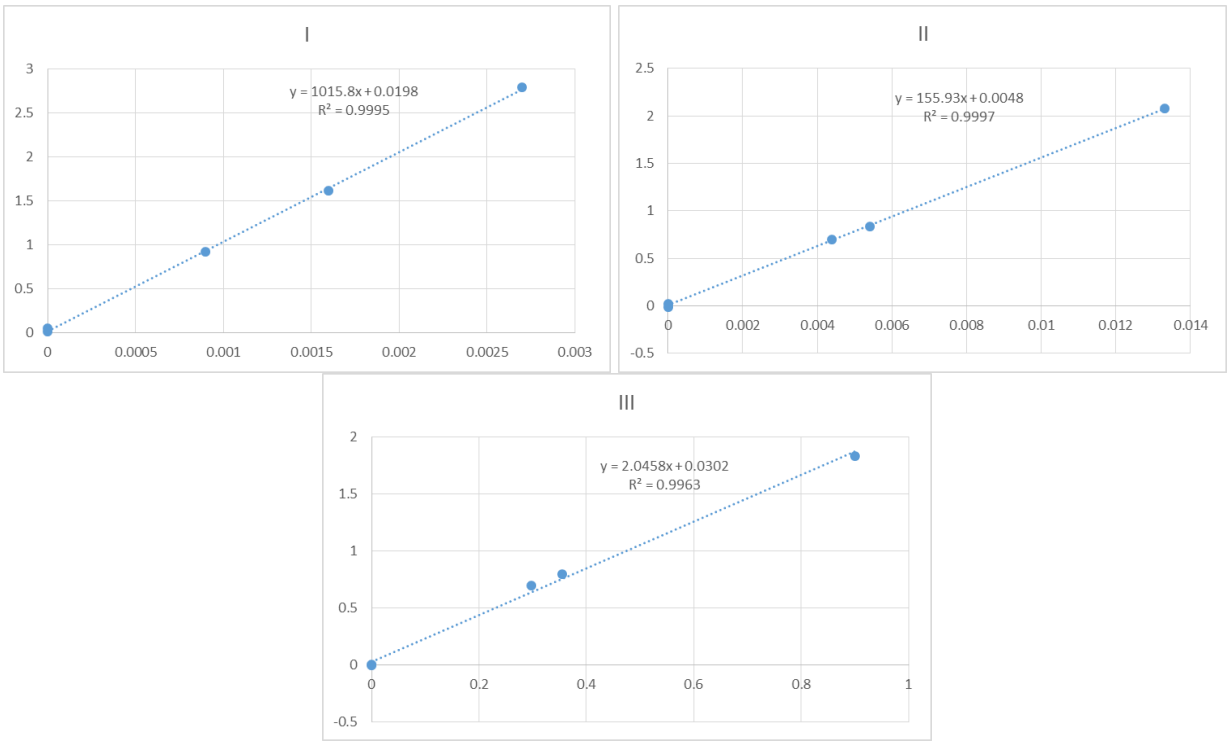


Fig. 8.20. Plots of the calculated (Mode #1) scores versus analytical concentrations for three components.

Correlations are very good, the smallest determination coefficient of 0.9963 is found for the component III. Using regression equations the concentrations might be auto-predicted; they are shown in Table 8.13. Comparing second order and regression calibrations reveals that the standard deviations of all components are smaller.

Table 8.13. Concentrations of three components auto-predicted using linear regression and their standard deviations.

Component	I	II	III
	0.002724	-0.000107	-0.017
	-0.000006	0.013312	-0.014
	0.000029	0.000121	0.883
	0.001569	0.005348	0.376
	0.000884	0.004430	0.325
std	0.000026	0.000086	0.022
std%	2.5%	1.9%	7.2%

8.7 N-way PLS

The program libraries of Spectroscopy and Chemometrics Section of the University of Copenhagen[30,37] contain useful toolbox N-way (nway) (and few other)[28]. There is also a program

for multi-way PLS: npls.m. It has been applied to determine concentrations in Exercise 8.3. Program PLSm.m opens data file claus.m and performs three-way PLS on the data **X** and concentrations **y**. The self-predicted concentrations are displayed in Table 8.14. It is interesting to note that the standard deviations of these concentrations are smaller than those obtained using second order calibration or linear regression shown above.

Table 8.14. Concentrations of three components auto-predicted using three-way PLS and their standard deviations.

Component	I	II	III
	0.002689	0.000019	-0.014
	0.000014	0.013312	-0.013
	0.000015	0.000012	0.886
	0.001549	0.005396	0.363
	0.000874	0.004418	0.325
std	0.000031	0.000016	0.019
std%	3.0%	0.34%	6.0%

8.8 Effect of noise

To study the effect of the random noise, normally distributed noise was added to all the spectra in **X** in Exercise 8.3. The standard deviation was 50% of the largest value of fluorescence of the samples, s_X = 470. Effect of adding such a noise on the spectrum of tryptophan is shown in Fig. 8.21 and the noisy spectra of all the samples in Fig. 8.22.

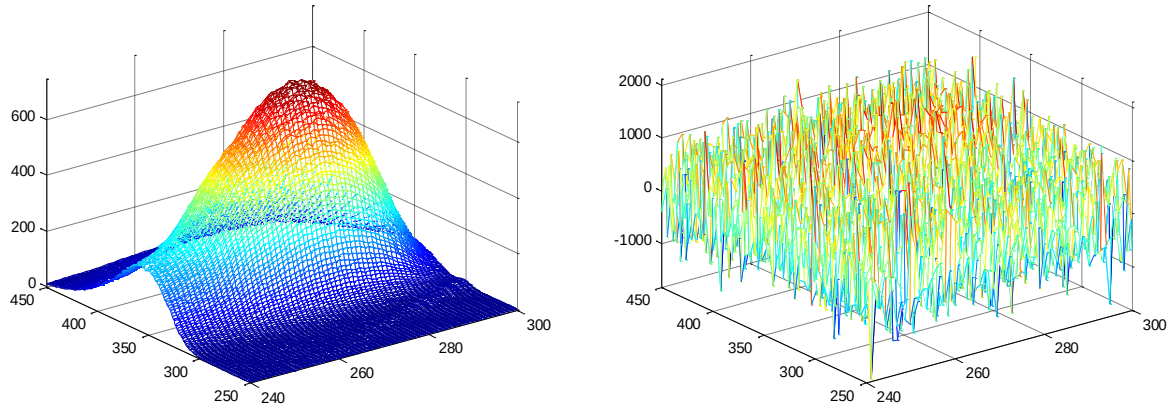


Fig. 8.21. Tryptophan spectrum (sample 1) before (left) and after (right) adding normal noise with standard deviation of 470.

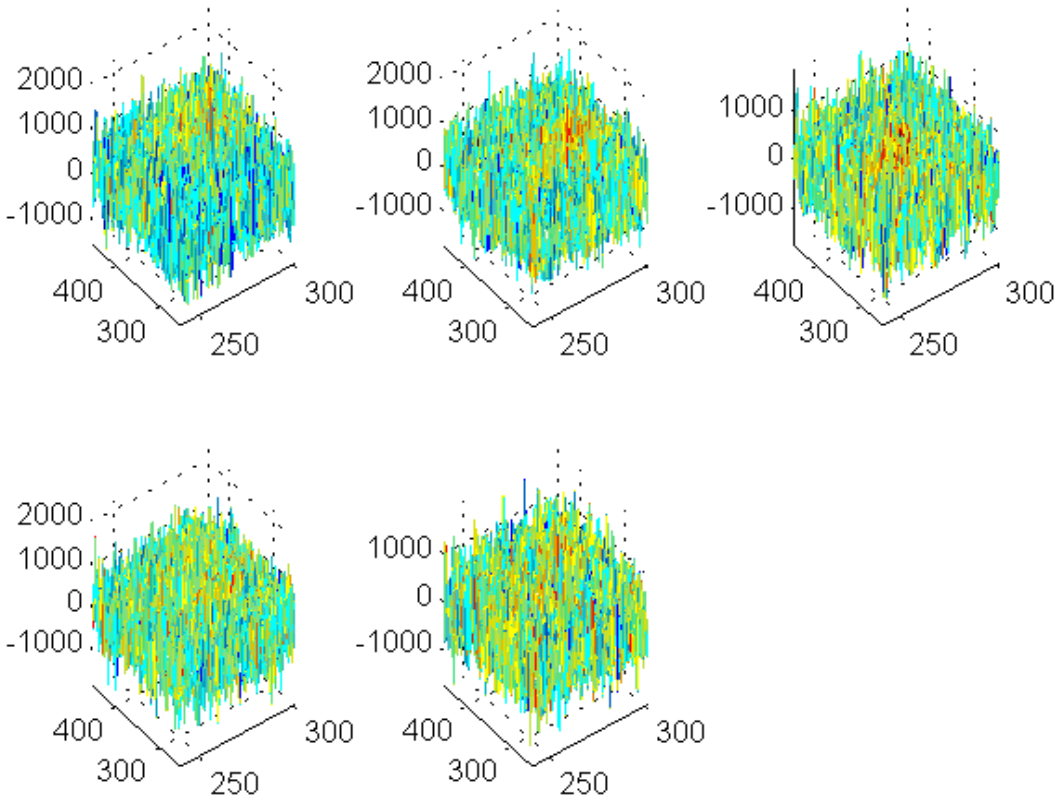


Fig. 8.22. Spectra of four samples in Exercise 8.3 after adding normal; noise with standard deviation of 470.

After adding so large noise spectra of the individual components are hardly visible. Results of the PARAFAC analysis of these data are displayed in Fig. 8.23 and 8.24.

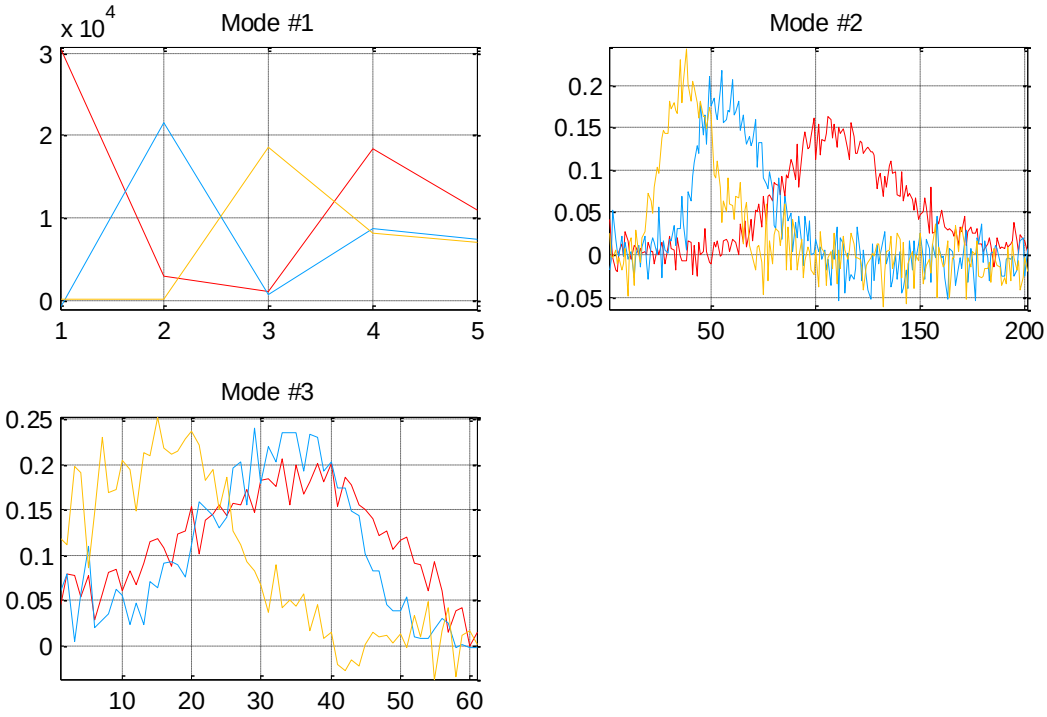


Fig. 8.23. PARAFAC analysis of the noisy data in Fig. 8.22 using three components.

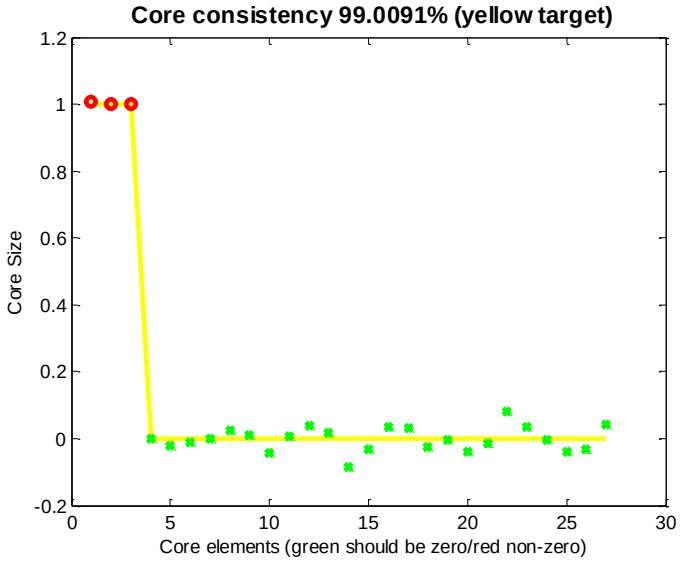


Fig. 8.24. Core consistency for the noisy data in Fig. 8.22.

Despite the fact that the fluorescence spectra are hardly distinguishable from the random noise the fluorescence spectra and the scores are similar to those with the instrumental noise only, Fig. 8.18, although they are more noisy. This is a remarkable achievement in comparison with the two-way methods. The core consistency is still 99% though the residual sum of squares of matrix $\underline{\mathbf{X}}$ is much larger, 1.34×10^{10} (to be compared with 1.45×10^{6} in Table 8.11).

Exercise 8.4.

Five chromatograms of the mixtures of two components were studied by the measurement of 30 UV/VIS spectra as functions of time. They were registered at 28 wavelengths. They are in files X1.m to X5.m. The 3D spectra for different samples are displayed in Fig. 8.25. The concentrations of the components are shown in Table 8.15.

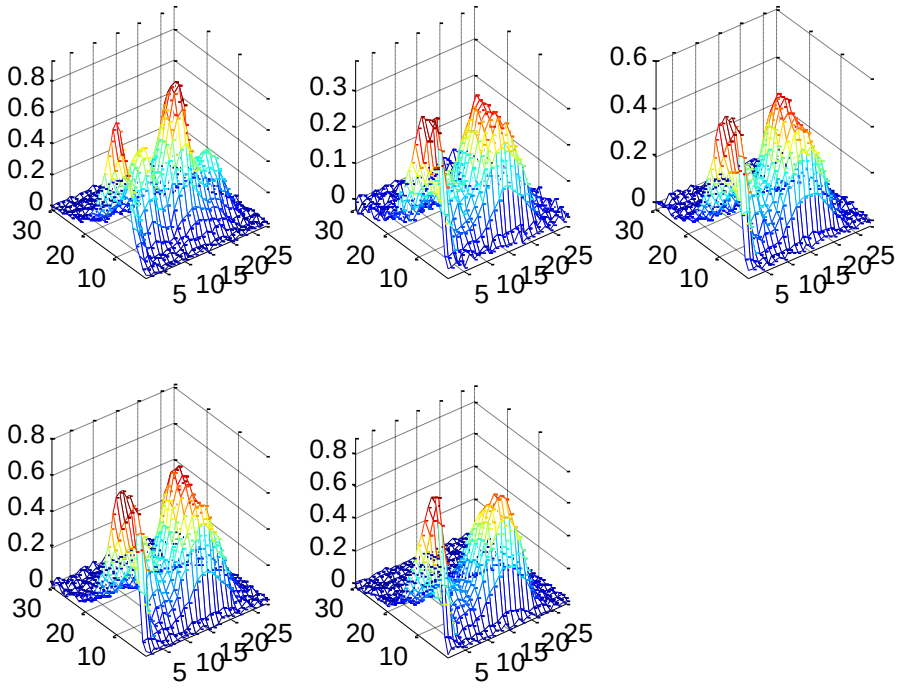


Fig. 8.25. Spectra of the mixtures of two components measured during elution in HPLC

Table 8.15. Total concentrations of the species injected for five chromatograms.

a	b
1.0	**0.2**
0.3	**0.4**
0.5	**0.6**
0.7	**0.8**
0.4	**1.0**

Use PARAFAC model and multiway PLS to calculate self-predicted concentrations. The PARAFAC model (flsc2.m) was executed for one to three factors. The obtained results are shown in Fig. 8.26. 8.28, and in Table 8.16.

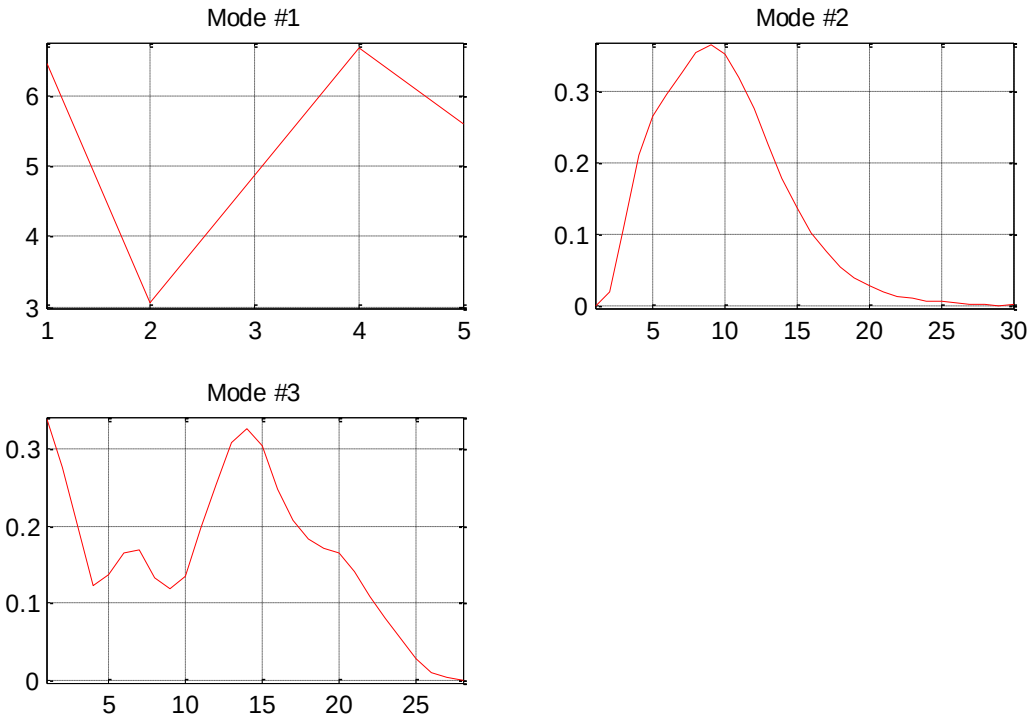
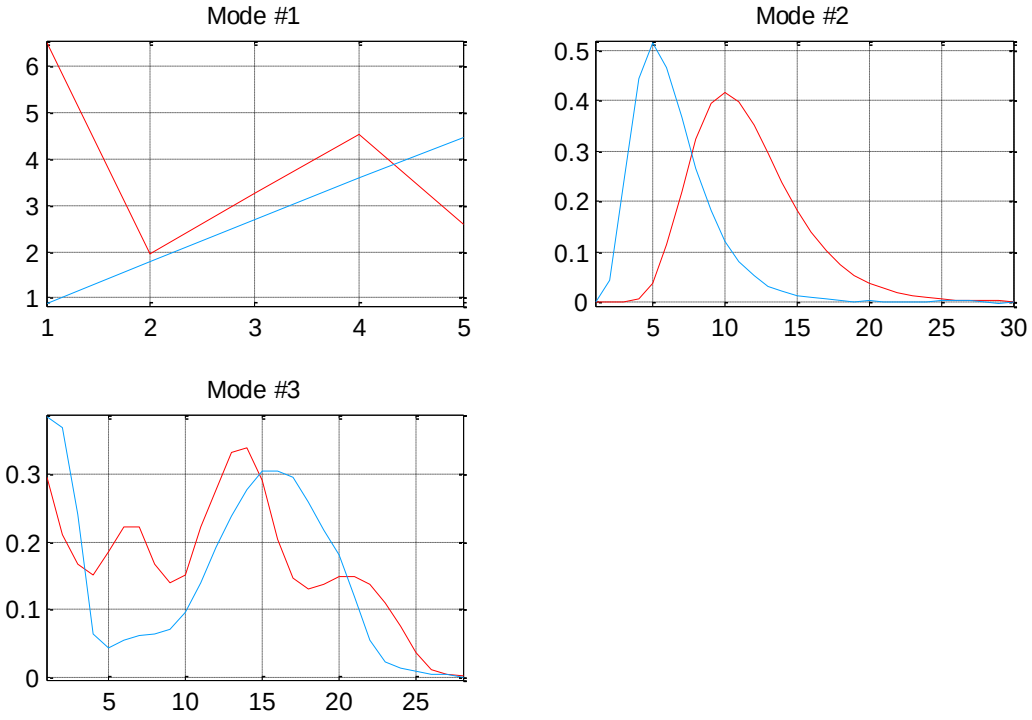


Fig. 8.26. Loadings obtained using one factor.



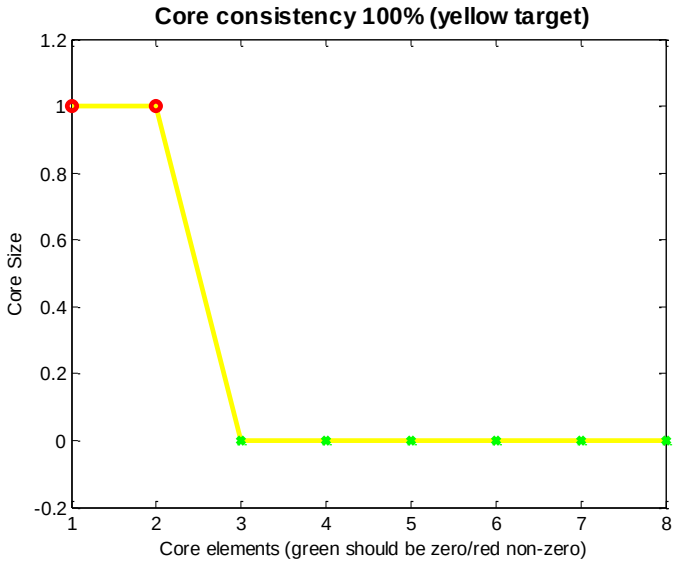
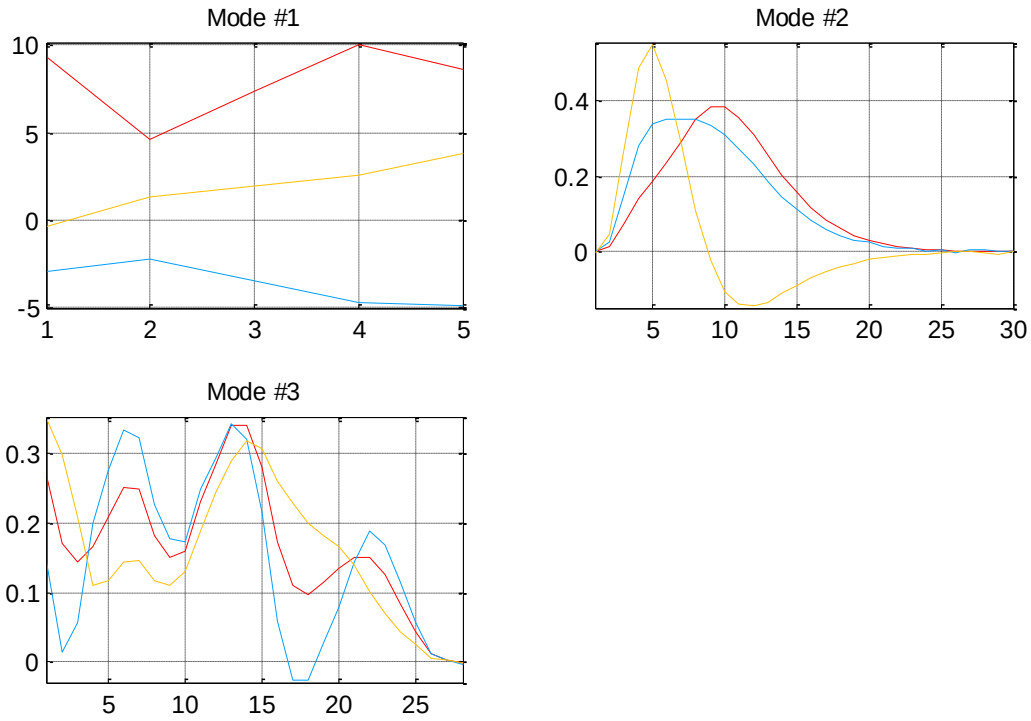
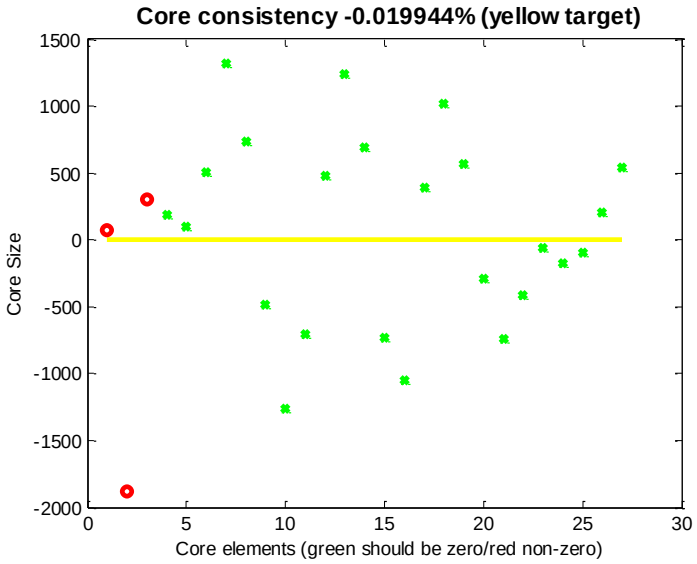


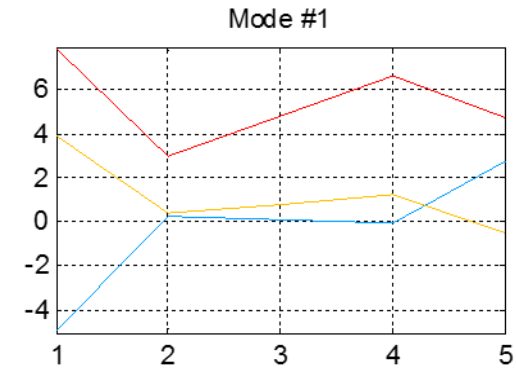
Fig. 8.27. Loadings and core consistency for two factors.

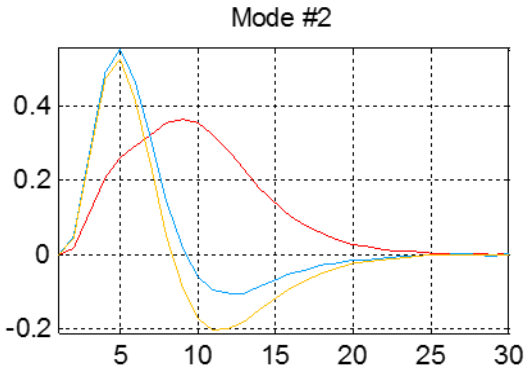
first run

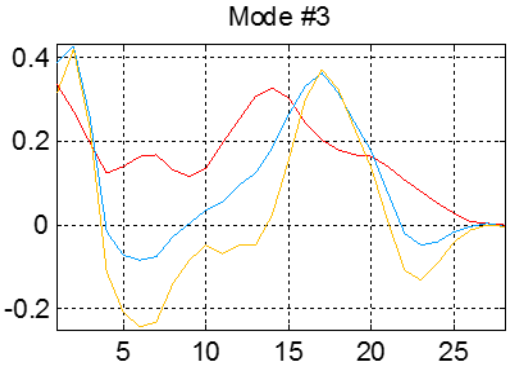



Core consistency -0.019944% (yellow target)
Core Size
Core elements (green should be zero/red non-zero)
1500
1000
500
0
-500
-1000
-1500
-2000
0
5
10
15
20
25
30

second run


Mode #1
6
4
2
0
-2
-4
1
2
3
4
5


Mode #2
0.4
0.2
0
-0.2
5
10
15
20
25
30


Mode #3
0.4
0.2
0
-0.2
5
10
15
20
25

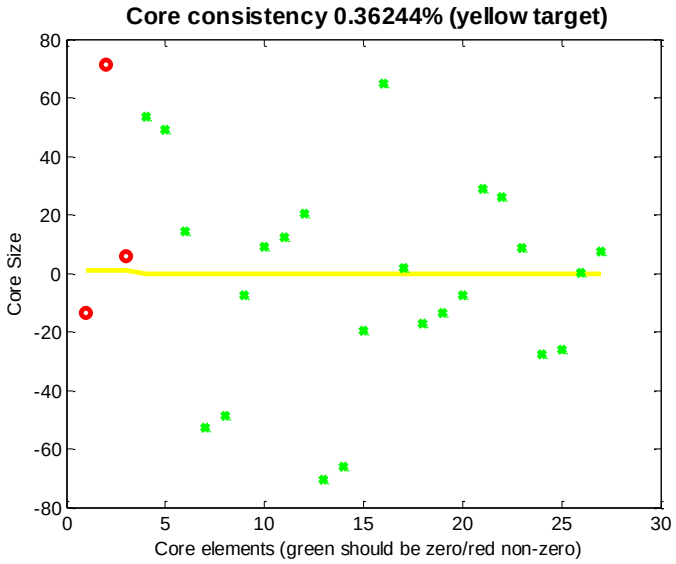


Fig. 8.28. Examples of two sets of solutions found using three factors.

Table 8.16. Numerical results of the application of the PARAFAC model to the chromatographic data for one to three factors.

factors	1	2	3	3
iterations	4	30	10	7
SSQ	12.326	0.07859515	0.078146	0.076204
corcondia	100	99.9999958	0.362439	-0.01994

Using one factor in the PARAFAC model shows strange elution profile (Mode #2) with two overlapping peaks. By increasing number of factors to two dramatic decrease of the residual sum of squares is observed and core consistency stays close to 100%. However, when the number of components is increased to three no important changes in SSQ are observed, however repetition of the model gives each time different sets of loadings and the loadings are often negative. This is accompanied by very low value of core consistency close to zero. These results indicate that two factors should be used, in agreement with the two components analyzed.

Comparison of the scores (Mode #1) with the concentrations indicates that the first column of scores is proportional to the concentrations of component b and the second is proportional to the concentrations of a.

Table 8.17. Values of scores in Mode #1 that is elements of matrix **A**. The first column corresponds to concentrations of component b and the second of component a.

Scores A (Mode #1)	
b	a
6.515549	0.85472
1.956963	1.783488
3.26237	2.68788
4.566108	3.571523
2.614333	4.466589

Relations between scores and concentrations are displayed in Fig. 8.29. The correlations are excellent. One should notice that scaling factors for different components are different. Theoretical elution profiles (Mode #2) and spectra of the individual components (Mode #3) were also compared with the theoretical values. They are displayed in Fig. 8.30-8.31. The correlations are excellent.

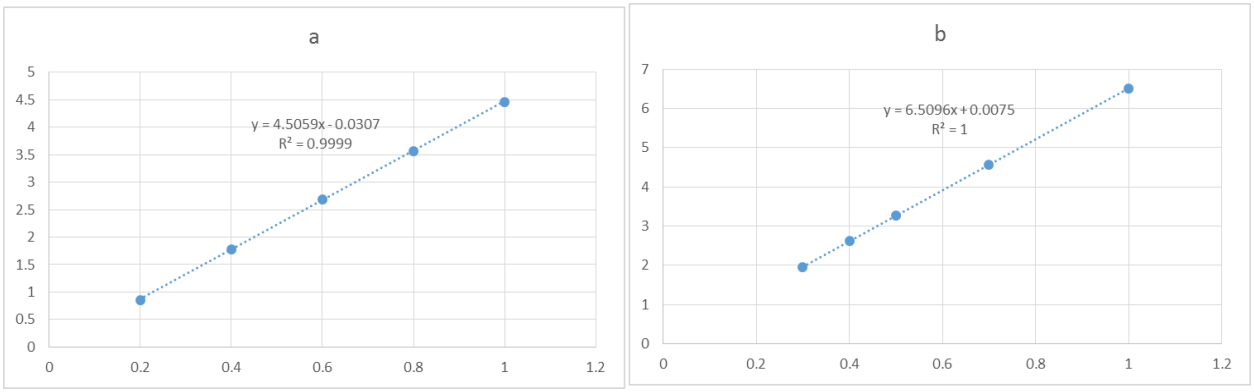


Fig. 8.29. Dependence of the scores on concentrations for two components.

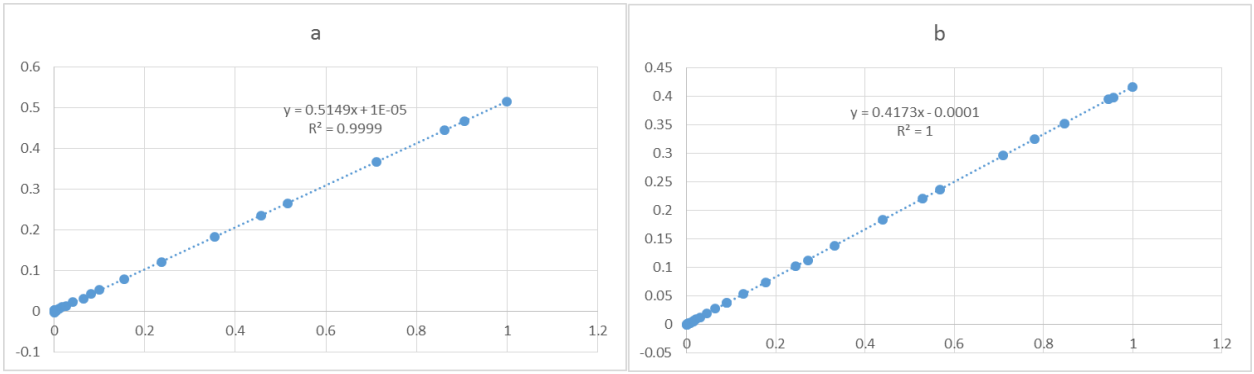


Fig. 8.30. Comparison of the calculated loadings, matrix **B** (Mode #2) with the theoretical elution profiles of two components.

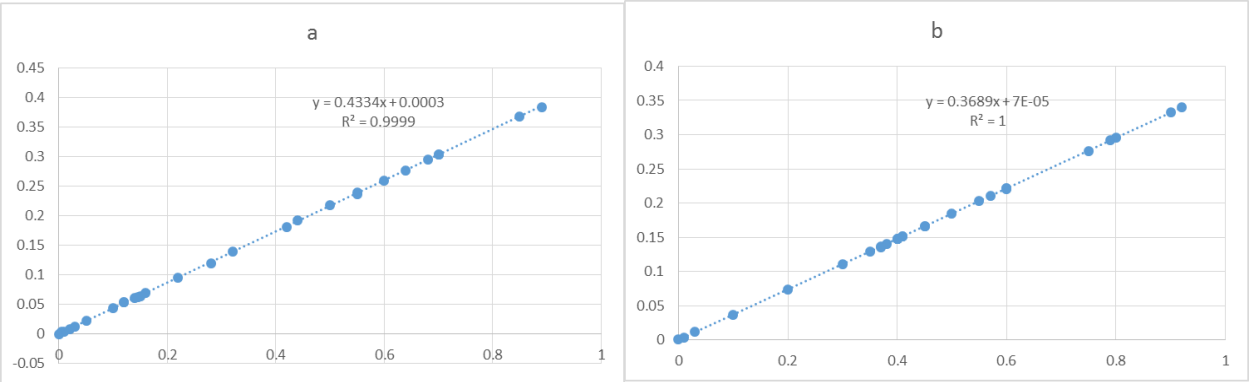


Fig. 8.31. Comparison of the calculated loadings, matrix **C** (Mode #3) with the theoretical spectra of two compounds.

To predict concentrations first the second order calibration was also used. As the standard values first the sample #1 was used and later the same analysis was repeated assuming that the concentrations of sample #4 are known. The results are shown in Table 8.18.

Table 8.18. Concentrations calculated using second order calibration and sample #1 or #4 as a standard value and the standard deviations obtained.

	from sample #1		from sample #4	
Component	a	b	a	b
	0.200	**1.00000**	0.1915	0.99890
	0.417	0.30034	0.3995	0.30001
	0.629	0.50068	0.6021	0.50013
	0.836	0.70077	**0.8000**	**0.70000**
	1.045	0.40120	1.0005	0.40076
std	0.038	0.00094	0.0051	0.00078
std%	6.1%	0.16%	0.51%	0.19%

It is interesting to note that using sample #1 as the standard the standard deviation of component “a” is much larger than that of component “b”. However, when sample #4 is used as the standard, the standard deviation of the component “a” is over 10 times smaller. This behavior must be related to the relative concentrations in the samples. In sample #1 concentration of “a” is five times smaller than that of “b” while in sample #4 these concentrations are similar.
Next, the multi-way PLS method (PLSm.m) was used to auto-predict the concentrations. The results are displayed in Table 8.19. This method gives the smallest standard deviations of concentrations.

Table 8.19. Self-predicted concentrations using multi-way PLS method.

Component	a	b
	0.2004	0.99988
	0.3997	0.30009
	0.5996	0.50013
	0.7994	0.70017
	1.0007	0.39977
std	0.0006	0.00017
std%	0.10%	0.03%

9 Exercises and programs

All the exercises were solved using Matlab programs developed by Brereton[3] in the site of his book. They were modified and corrected for this book. The data for Exercises were either taken from Brereton[3] and Pomerantsev[6] books but mainly they were simulated using only concentration and/or spectra from these books with added 1-2% Gaussian noise. For multiway models see below),

The solutions are shown in Excel files located in the Exercise folders. The readers should use these data and check if the received the same results.

The Matlab programs are located in three folders PCA, PCR, and PLS. The results might be simply transferred from Matlab to Excel using Copy and Paste or by saving them on the disk (see later).

PCA and PLS analysis is also included in Origin although options are limited (standardized data were used) and e.g. in PCA the obtained scores were 1.054 times smaller.

There are of course many commercial programs for example PLS_Toolbox/Solo from Eigenvector Research Inc.[38] or The Unscrambler (CAMCO),[39] with many useful tools and plotting tools. They also organize courses on aspects of chemometrics.

In each folder of each Exercise there are working programs, all the necessary subroutines, and data, and they might be directly executed in Maple.

9.1 Brief description of programs

In all cases the functions centre.m and scale.m are used to center and standardize the **X** and **C** data matrices.

9.1.1 PCA

There are two programs to carry out PCA analysis:

1) PCAtest.m: It performs PCA on data in file Xdata.m (**X**). It needs "maxrank" that is the rank, *r*, of the **X** matrix and "preoption" which is 1 for the raw data, 2 for the centered data, and 3 for the standardized data. It can display scores, **T**, loadings, **P**, calculated value of $\hat{\mathbf{X}}$, and RSS. It uses subroutine pca.m
2) PCAcross.m: It performs cross-validation on Xdata.m (**X**) on preprocessed data, according to the value of "preoption". It produces RSS and PRESS parameters for each PC up to "maxrank". It uses subroutine mpcacross.m.

Mahalanobis distances were calculated using program maha.m and subroutine mahaldist.m[3]

9.1.2 PCR

There are

1) PCRtest.m: It performs PCR on data in Xdata.m (**X**) and Cdata.m (**C**). It needs input "maxrank" and "preoption", as above. It uses subroutine pcr1.m. It produces:
   a) Matrix of scores **T**,
   b) Matrix of loadings **P**,

c) Vector **s** containing eigenvalues for "maxrank" principal components
d) Rotation matrix **R**,
e) "Pred" containing self-predicted concentrations,
f) Matrix "RMS" containing RMS_{sp} values
g) Transposed matrix Xc containing calculated spectra, $\hat{\mathbf{X}}$
h) Vector RRR containing root mean-square of deviations between the calculated and experimental spectra:

$$\mathrm{RRR}_{\mathrm{X}}(r) = \sqrt{\frac{\sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{i,j} - \hat{x}_{i,j}\right)^2}{I - r - f}} \tag{9.1}$$

where I is the number of spectra, r number of principal components (concentrations) and f is the loss of the degree of freedom due to preprocessing, see Eq. (5.9).

2) PCRcross.m: It performs cross-validation of data in in Xdata.m (**X**) and Cdata.m (**C**). It needs input "maxrank" and "preoption", as above. It uses subroutines: pcrcross1.m, pcr1.m, and pcrpred1.m. It produces:
   a) "PRESS" (transposed), Eq. (5.15),
   b) "RMS" that is RMS_{cv}, Eq. (5.16).
3) PCRpred.m: It calculates predicted concentrations for the validation/test data XVtest.m (**Xtest**) using regression information from the training data in Xdata.m (**X**) and Cdata.m (**C**). When the concentrations for the test data are not known **Ctest** should be filled with zeros. In such a case the calculated residuals and RMS_{test} do not have any meaning. When the data file CVtest.m (**Ctest**) is known validation of the method is performed. It uses subroutines pcrpredtest1.m, pcr1.m, and pcrpred1.m. The program produces:
   a) "Pred" predicted concentrations for the test/validation spectra
   b) "Resi" residuals $\sum_{i=1}^{L}\left(c_{i,k} - \hat{c}_{i,k}\right)$ (see Eq. (5.17) for definitions)
   c) "RMS" RMS_{test}, Eq. (5.17).

9.1.3 PLS

There are programs:

1) PLS1.m to perform PLS1 on Xdata.m (**X**) and a vector of concentrations, e.g. CA.m (**c**). It uses subroutine pls3.m. It produces:
   a) "Pred" vector of self-predicted concentrations,
   b) "RMS" one RMS_{sp} value of root mean square error for one concentration (CA.m).
2) PLS2.m: It performs PLS2 analysis on matrices Xdata.m (**X**) and Cdata.m (**C**). It uses subroutine pls3.m and produces:
   a) "Pred" matrix of self-predicted concentrations,
   b) "RMS" a vector of RMS_{sp} for each component.
3) PLS1pred.m: It performs PLS1 analysis on Xdata.m (**X**) and a vector of concentrations, e.g. CA.m (**c**). It predicts a vector of concentrations for the test/validation data XVtest.m (**Xtest**). In the case when the concentrations for the vector of test data CAp.m (**Ctest**) are unknown they must be filled with zeros. In such a case the calculated residuals and RMS_{test}

do not have any meaning. It uses subroutines plspredtest1.m and pls3.m and plspred.m. It produces:
   a) "Pred" vector of predicted concentrations for test/validation data.
   b) "RMS" one RMS_{test} value of root mean square error for one concentration (CAp.m).

4) PLS2pred.m: It performs PLS2 analysis on matrices Xdata.m (**X**) and Cdata.m (**C**), then predicts concentrations for XVtest.m (**Xtest**). When the concentration matrix CVtest.m (**Ctest**) is unknown matrix CVtest.m (**Ctest**) must be filled with zeros and the calculated residuals and RMS_{test} do not have any meaning. When the concentration matrix CVtest.m (**Ctest**) is known validation of the method is performed. It uses subroutine plspredtest1.m and pls3.m and plspred.m. It produces:
   a) "Pred" matrix of predicted concentrations for test/validation data,
   b) "RMS" that is RMS_{test} when concentrations CVtest.m (**Ctest**) are known.
4) PLScross.m: It performs cross-validation of data in in Xdata.m (**X**) and Cdata.m (**C**). It needs input "maxrank" and "preoption", as above. It uses subroutines: pcrcross1.m, pcr1.m, and pcrpred1.m. It produces:
   a) "PRESS" (transposed), Eq. (5.15),
      "RMS" that is RMS_{cv}, Eq. (5.16).
5) The use of PLS1 might be carried out supplying concentration column separately for each component but it can be automatized using PLS1anal.m where program choses sequentially all the columns for all the concentrations.

The output is displayed on the screen and might be copied (Copy and Paste) to Excel. It might also be saved using simple Matlab instructions, for example to save matrix **C**:

```
data=[C];
save output data -ascii –tabs
```

and the matrix of concentrations is saved to the file "output", separated by tabs.

9.1.4 PARAFAC model

Matlab PARAFAC model and some tools are available form http://www.models.life.ku.dk/algorithms, toolbox nway and others. The use of this model is illustrated in Exercises. The programs presented there (fac2.m, flsc1.m, flsc2, PLSm.m) display first the 3D plots of the data and the results are in shown the graphical form (loadings, core-consistency: corecond) and the numerical values of the scores and loadings A, B, and C. PARAFAC program contains many explanations.

The simplest use is to run it as:

```
[Factors]=parafac(X,Fac)
```

where X is the data box and Fac is the number of factors = components.

The full syntax is:

```
[Factors,it,err,corcondia]=parafac(X,Fac,Options,const,OldLoad,FixMode,Weights)
```

PARAFAC can be controlled by Options; Options(2) controls the initialization method. The default value is zero. The analysis results can be confirmed by running with random

orthogonalized initialization Option(2)=2; when each run produces the same results and the sum of squares “err” the solution seems correct. Option(3)=1 produces useful graphical results.

Parameter const controls type of constrains with const = 2 for non-negativity of results. Other parameters are usually omitted and will not be discussed here.

The obtained parameters are in Factors containing three loadings. They might be translated into numbers by running program:

[A,B,C] = fac2let(Factors).

Other results are it – number of iterations, err – residual sum of squares, and corcondia – core consistency factor.

PLS on multi-way data is performed using PLSm.m which calls npls.m subroutine. As input the following parameters must be supplied: the measurements array **X**, concentration matrix **y**, number of factors Fac, and control parameter show=1. The output ypred contains calculated concentrations.

10 References

[1] M. Otto, Chemometrics, Wiley-VCH, Weinheim, 2007.

[2] Comprehensive Chemometrics, Chemical and Biochemical Data Analysis, S.D. Brown, R. Tauler, B. Walczak, Edts., Elsevier, 2009: J.H. Kalivas, Calibration Methodologies, Chapter 3.01.

[3] R. G. Brereton, Chemometrics Data Analysis for the Laboratory and Chemical Plant, Wiley, Chichester, 2003.

[4] S. Wold, Chemometr. Intell. Lab. Syst., 30 (1995) 109.

[5] S.D. Brown, R.S. Bear, Jr., Crit. Rev. Anal. Chem., 24 (1993) 99.

[6] A.L. Pomerantsev, Chemometrics in Excel, Wiley, 2014.

[7] R.G. Brereton. Chemometrics. Application of Mathematics and Statistics to Laboratory Systems, Ellis Horwood, New York, 1990.

[8] K.H. Esbensen, D. Guyot, F. Westad, L.P. Houmoller, Multivariate Data Analysis - in Practice, CAMO Process AS, 2002.

[9] R.G. Brereton, Applied Chemometrics for Scientists, Wiley, Chichester, 2007.

[10] R.G. Brereton, Chemometrics for Pattern Recognition, Wiley, Chichester, 2009.

[11] R.G. Brereton, Chemometrics. Data Driven Extraction for Science, Wiley, Hoboken NJ, 2018.

[12] D.L. Massart, B.G.M. Vandeginste, S.N. Deming, Y. Michotte, L. Kaufman, Chemometrics:a Textbook, Elsevier, Amsterdam, 1988.

[13] D.L. Massart, B.G.M. Vandeginste, L.M.C. Buydens, S. De Jong, P.J. Lewi and J. Smeyers-Verbeke, Handbook of Chemometrics and Qualimetrics Part A, Elsevier, Amsterdam, 1997.

[14] B.G.M. Vandeginste, D.L. Massart. L.M. C. Buydens, S. De Jong, P.J. Lewi and J. Smeyers-Verbeke, Handbook of Chemometrics and Qualimetrics Part B, Elsevier, Amsterdam, 1998.

[15] K.R. Beebe, R.J. Pell and M.B. Seasholtz, Chemometrics: a Practical Guide, John Wiley & Sons, Inc., New York, 1998.

[16] R. Kramer, Chemometrics Techniques for Quantitative Analysis, Marcel Dekker, New York, 1998.

[17] P.J. Gemperline (editor), Practical Guide to Chemometrics, 2nd Edn, CRC, Boca Raton, FL, 2006.

[18] M.A. Sharaf, D.L. Illman and B.R. Kowalski, Chemometrics, John Wiley & Sons, Inc., New York, 1986.

[19] K.H. Esbensen, Multivariate Data Analysis in Practice, CAMO, Oslo, 2010.

[20] M.J. Adams. Chemometrics in Analytical Spectroscopy, RSC, Cambridge, UK, 1995.

[21] J.W. Einax, H.W. Zweinziger, S. Geiß. Chemometrics in Environmental Analysis, Wiley-VCH, Weinheim, 1997.

[22] H. Mark, J. Workman, Chemometrics in Spectroscopy, Elsevier, Amsterdam, 2007.

[23] Comprehensive Chemometrics. Chemical and Biochemical Data Analysis, vol. 1-4, S.D. Brown, R. Tauler, B. Walczak, Edts., Elsevier, 2009.

[24] S. Wold, Pattern Recogn., 8 (1976) 127.

[25] P. Thy, K. Esbensen, J. Geophys. Res. Solid Earth, 98, B7 (1993) 11799.

[26] P. Geladi, B.R. Kowalski, Anal. Chim. Acta, 185 (1986) 1.

[27] K. Roy, S. Kar, R.N. Das, Understanding the Basics of QSAR for applications in pharmaceutical sciences and risk assessment, Elsevier-AP, 2015.
[28] A. Smilde, R. Bro, P. Geladi, Multi-way analysis with applications in the chemical sciences, J. Wiley, The Atrium, Southern Gate, Chichester, England, 2004.
[29] P. M. Kroonenberg, Three-mode principal component analysis: Theory and applications, DSWO Press, Leiden, 1983.
[30] http://www.models.life.ku.dk/
[31] R. Bro, PARAFAC. Tutorial and applications, Chemom. Intell. Lab. Syst., 38 (1997) 149.
[32] R. Bro, Youtube, Multi-way analysis. Parts 21-27.
[33] R.A. Harshman, Foundations of the PARAFAC procedure: Models and conditions for an 'explanatory' multi-modal factor analysis, UCLA Working Papers in Phonetics, 16 (1970) 1-84.
[34] E. Sanchez, B.R. Kowalski, J. Chemometr., 2 (1988) 265.
[35] R. Bro, PhD Thesis, University of Amsterdam, 1998.
[36] R. Bro, H.A.L. Kiers, J. Chemometr., 17 (2003) 274.
[37] http://www.models.life.ku.dk/algorithms
[38] http://www.eigenvector.com/software/index.htm
[39] http://www.camo.com/rt/Products/Unscrambler/unscrambler.html