POLITECNICO DI TORINO
Repository ISTITUZIONALE

Exploring NIR spectroscopy data: A practical chemometric tutorial for analyzing freeze-dried pharmaceutical formulations

Original
Exploring NIR spectroscopy data: A practical chemometric tutorial for analyzing freeze-dried pharmaceutical formulations / Massei, A., Cavallini, N., Savorani, F., Falco, N., Fissore, D.. - In: CHEMOMETRICS AND INTELLIGENT LABORATORY SYSTEMS. - ISSN 0169-7439. - STAMPA. - 257:(2025). [10.1016/j.chemolab.2024.105291]

Availability:
This version is available at: 11583/2997227 since: 2025-02-06T09:11:59Z

Publisher:
Elsevier

Published
DOI:10.1016/j.chemolab.2024.105291

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

25 November 2026

Chemometrics and Intelligent Laboratory Systems 257 (2025) 105291

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemometrics

Exploring NIR spectroscopy data: A practical chemometric tutorial for analyzing freeze-dried pharmaceutical formulations

Ambra Massei [a,b], Nicola Cavallini [a,*], Francesco Savorani [a], Nunzia Falco [b], Davide Fissore [a]

[a] Dipartimento di Scienza Applicata e Tecnologia, Politecnico di Torino, Corso Duca Degli Abruzzi 25, 10129, Torino, Italy
[b] Global Parenterals Development Department, Merck Serono SpA, Via Luigi Einaudi 11, 00012, Guidonia Montecelio, Roma, Italy

ARTICLE INFO

Keywords:
Chemometrics
Freeze-dry
Near-infrared spectroscopy
Exploratory analysis
Regression analysis
Classification analysis

ABSTRACT

Chemometrics tools are of fundamental importance for data analysis in the pharmaceutical field, especially with the increasingly strong assertion of the Process Analytical Technologies (PAT). In fact, analytical technologies such as Near-Infrared or Raman spectroscopies generate a lot of data, the spectra, that must be analyzed in a proper way. Typically, it is quite difficult to deeply understand the information hidden within the raw data. Therefore, careful, and efficient data exploration is needed to highlight the chemical and physical features of the analyzed samples.

Here, a tutorial on all the fundamental steps and concepts needed to perform a proper data analysis based on a case-study of different freeze-dried formulations in the pharmaceutical field is proposed. The data analysis pipeline begins with the dataset explanation, to better point out the main known differences and similarities among the investigated formulations. After the first step of data preprocessing, Principal Component Analysis (PCA), Partial Least Squares (PLS) for regression, and Partial Least Squares-Discriminant Analysis (PLS-DA) for classification are presented and applied to show how to obtain deep comprehension of the real-case NIR dataset at hand. The experimental results demonstrate that trends related to increasing levels of sucrose and/or arginine, as well as distinct clusters related to the sample type and to the operator who conducted the analysis can be found and modelled in the example data.

The tutorial aims at providing clear practical steps to conduct a robust data analysis, starting from the extraction and organization of the raw data, up to building more advanced predictive models (regression and classification). At each step some key questions are asked and answered to stimulate critical thinking in the reader. Also, commented MATLAB scripts are provided together with the real-case example NIR data, so that anyone could reproduce the whole data analysis in the tutorial, and try first hand to work with the data.

1. Motivation

In the last two decades, the regulation of pharmaceutical manufacturing evolved significantly. In the traditional approach, pharmaceutical operations lead to high costs associated with drug manufacture. For improving pharmaceutical developments and manufacturing, new technologies were encouraged by the regulatory authorities in the last years. Specifically, in September 2004 the US Food and Drug Administration (FDA) published the guidance document: "*PAT – A Framework for Innovative Pharmaceutical Development Manufacturing and Quality Assurance*" [1,2]. This document encourages the introduction of new technologies, called "Process Analytical Technologies" (PAT), aiming to increase the quality of the drugs by following the Quality-by-Design (QbD) approach. A PAT system is defined by FDA as a "*system for designing, analyzing, and controlling manufacturing through timely measurements (i.e., during processing) of critical quality and performance attributes of raw and in-process materials and processes, with the goal of ensuring final product quality*". In this perspective, the quality of a certain product must be embedded in its production process, carefully monitored, and not just tested at the end of the manufacturing. For this purpose, a full understanding and control of the manufacturing process is recommended. In this framework, huge numbers of variables have to be measured every few minutes/seconds, leading to huge amounts of data to be analyzed. Multivariate analysis tools are suitable to face this issue. They are recognized by FDA as powerful tools to facilitate and speed up the understanding for scientific pharmaceutical development

* Corresponding author.
E-mail address: nicola.cavallini@polito.it (N. Cavallini).

https://doi.org/10.1016/j.chemolab.2024.105291
Received 20 August 2024; Received in revised form 28 November 2024; Accepted 28 November 2024
Available online 30 November 2024
0169-7439/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

[3]. Also, the European Council published a guideline on the use of chemometrics methods to process analytical data for quality control and manufacturing in the official European Pharmacopoeia [4]. Multivariate analysis tools belong to the world of Chemometrics, whose methods and tools can be used to extract chemically relevant information from the available data [5]. In fact, the main scope is to reveal hidden patterns in the data, and to highlight possible interactions among the measured parameters and variables [6].

In the pharmaceutical industry, PAT solutions involving spectroscopic techniques such as Near-Infrared (NIR), Raman and Nuclear Magnetic Resonance (NMR) spectroscopies, have been widely implemented in process design and development [6]. These techniques are non-invasive and non-destructive and can be applied at-line, in-line or on-line. These techniques generate large amounts of data, the spectra, which can be easily studied with multivariate tools. The literature about the combination of spectroscopy and chemometrics for modelling, interpreting, and understanding the data is very vast. Zhao et al., for instance, used the Partial Least Squares (PLS) algorithm to quantify the active pharmaceutical ingredient (API) in formulations [7]. Shi et al. built a PLS regression model of a continuous-flow hydrogenation process for the production of an API, using the model in simulation to identify the optimal design space [8]. Tomba et al. applied multivariate analysis for developing a continuous process for the manufacture of paracetamol tablets, using Principal Component Analysis (PCA) to identify the most critical parameters [9]. Lourenço et al. focused on fluid bed granulation, conducting two Design of Experiment (DoE [10]) studies to identify the process parameters having an impact on the granules' quality [11]. Clavaud et al. developed a model able to quantify the residual moisture content in freeze-dried products using both DoE and multivariate data analysis methods [12], while Bobba et al. and Massei et al. applied PCA, PLS and neural networks to quantify the residual moisture content at the end of the freeze-drying process [13,14]. Moreover, Grohganz et al. coupled NIR spectroscopy with multivariate data analysis to analyze the formation of different solid forms of mannitol in freeze-dried formulations, highlighting that β-mannitol form appears in absence of protein [15]. Finally, Ravn et al. visualized the spatial distribution of chemical compounds within a sample by taking and analysing NIR hyperspectral images [16].

Most of the published works used PCA as a tool for exploring the data and then decide if and how to build a subsequent PLS model to predict a certain Critical Quality Attribute (CQA) of the analyzed product. This tutorial paper is based on a case study concerning the data analysis of NIR spectra of different freeze-dried formulations characterized by different residual moisture content, by the presence of different excipients and by the addition of an amino acid to a given excipient. Surrogate solutions, characterized by the typical excipients present in the pharmaceutical formulations, were prepared. Freeze-drying process is a crucial step in drug manufacturing, as it allows removing the water present in the product by sublimation, converting the ice into vapor by operating at low pressure and temperature [17]. At the end of a freeze-drying cycle, products must meet certain CQAs. Among all the residual moisture (RM) is very important since water could promote biological and chemical degradation processes during storage, thus potentially reducing the product's shelf-life [18,19].

Our tutorial is aimed at presenting and highlighting the pros and cons of the multivariate approach and tools for analyzing experimental datasets. The tutorial is organized as a tutorial describing step-by-step the analytical workflow from data generation to classification and prediction of specific product characteristics. At each step MATLAB codes are also provided to stimulate the readers to apply the same simple functions to their own data, possibly with little adaptations in the codes. First, an in-depth description of the dataset will be provided, as knowing the data to model is fundamental to fully develop and understand the data analysis results. Then, the steps of data preprocessing, data exploration, regression and classification modelling are presented and discussed.

In addition to the practical coding tips and descriptions, we want to stress some important take home messages such as the importance of standardizing the data acquisition procedures, and how multivariate methods can reveal any unexpected results due to possible preparation errors, or also instrumental, operator or temporal deviations.

2. Experimental data acquisition

2.1. Specimens' preparation and freeze-drying cycle

Seven types of aqueous solutions were prepared and then freeze-dried into 2 R glass vials in the laboratories of Guidonia Montecelio (Italy) site of Merck Serono S. p.A. (Rome, Italy, an affiliate of Merck KGaA, Darmstadt, Germany) using a lab-scale freeze-dryer (LyoStar3, SP Scientific, Warminster, USA). The filling volume in each vial was 1 mL. The compositions of each solution (starting with the corresponding coded label) are hereafter reported:

- S6: sucrose 6 %w aqueous solution
- S3: sucrose 3 %w aqueous solution
- S9: sucrose 9 %w aqueous solution
- SA05: sucrose 6 %w + arginine 0.5 %w mixture
- SA1: sucrose 6 %w + arginine 1 %w mixture
- SA3: sucrose 3 %w + arginine 3 %w mixture
- T6: trehalose 6 %w aqueous solution.

Sucrose and arginine were supplied by Merck Life Science (Darmstadt, Germany), while trehalose was supplied by Sigma-Aldrich (Saint Louis, USA). Ultra-pure water was obtained by a Millipore water system (IQ 7000, Merck Millipore, Burlington, USA). A honeycomb layout was used as arrangement for the vials, surrounded by metal frames, all in direct contact with the shelves of the freeze-dryer. At the end of the freeze-drying cycle, a manual humidification of the vials, by adding a certain amount of water, was made to get a wider range of residual moisture in the sample. Table 1 summarizes the process conditions used to carry out the freeze-drying cycles. Further details about the conduction of the experimental tests can be found in Bobba et al. [13] and Massei et al. [14].

2.2. Near-infrared (NIR) spectroscopy

Near-infrared (NIR) spectroscopy was used to monitor the freeze-dried products. It is a vibrational spectroscopy operating in the range wavelength range of 700–2500 nm, or, following the notation of the present paper, in the wavenumbers range of 14,300–4000 cm^{-1}. A NIR spectrum can be seen as a collection of values of absorbance, one for each acquired wavelength. NIR radiation can be absorbed only by molecular vibrations resulting in changes of dipole moment: the generated absorbance signal can therefore be related to the chemical composition of the analyzed sample. For an introduction to NIR spectroscopy, the reader can refer to Ref. [20].

2.2.1. Why NIR spectroscopy?

NIR spectroscopy provides a lot of information about molecules containing atomic groups like O–H, N–H, C–H and S–H, since signals like combination bands and overtones are included in its wavelengths range. Therefore, water can be easily studied through NIR spectroscopy thanks to the presence of the O–H groups and its interaction with the C–H groups of the excipients used in the pharmaceutical field [20–23].

Table 1
Process conditions of the freeze-drying cycles.

Variable	Freezing	Annealing	Primary Drying	Secondary Drying
Shelf Temperature [°C]	−45	−15	−25	35
Pressure [Pa]	atm	atm	5	5
Duration [h]	6	2	30	10

For these reasons, NIR spectroscopy has been widely investigated as a powerful alternative technique to the Karl-Fischer (KF) titration for the estimation of residual moisture in freeze-dried products. Its non-destructive nature allows for the analysis of larger numbers of vials in a batch with respect to KF that is, on the contrary, a destructive analysis. Moreover, NIR spectroscopy is fast, since it requires minimal to no sample preparation [24–27]. Finally, NIR spectroscopy is suitable for all molecules having a dipole moment, so it is largely used for measuring water and proteins, which are strong NIR absorbers. In previous works, the focus for determining residual moisture in the final products was placed on the specific water signal, located at 5150 cm^{-1} [27]. It is important to consider that recording and modelling whole spectra makes it possible to detect possible interferences with known signals, like the one of water. When only one wavelength is considered, like in the case of many traditional analytical methods, effects arising from interferent species could be easily missed, leading to unreliable data. This is one of the reasons why multivariate approaches are generally more reliable than univariate ones.

2.2.2. Spectra acquisitions and NIR instrument

All samples were analyzed in diffuse reflectance mode with a Fourier Transform NIR spectrometer (Antaris MX FT-NIR, Thermo Fisher Scientific, Waltham, USA), equipped with an InGaAs detector and a halogen NIR source. The acquisition was done in the wavenumbers range 1000–4000 cm^{-1} with 32 scans for each spectrum. For each sample, the spectrum was obtained by averaging three spectra to reduce the noise of measurements. The NIR beam was pointed to the side of the vials and focused on the freeze-dried cake. The samples were stored at +5 °C to maintain stability during the experimental parts.

2.3. Karl-Fisher titration

The residual moisture content values were obtained by Karl Fischer titration using a coulometric titrator (C30S Mettler Toledo, Columbus, USA) and following the Standard Operative Procedure (SOP) in place at the company.

3. Multivariate algorithms, software, and toolboxes

3.1. Why do we need algorithms? And also, why multivariate?

The need for multivariate algorithms increased in the last decades also due to the rising pressure to use PAT tools, especially when dealing with spectroscopic data. NIR spectra often contain overlapping signals from various molecules (related to the functional groups cited in Section 2.2), making univariate analysis impractical due to its time-consuming nature and limited ability to extract information. In contrast, a multivariate approach processes all data simultaneously, revealing essential correlations between variables and maximizing the extracted information. In multivariate approaches, the original data is often linearly combined to capture the maximum explainable variance, ideally leaving only the noise unmodeled. To do so, algorithms are applied to the data, to extract relevant information through models. When appropriately validated, a multivariate model can substitute for time-consuming and resource-intensive measurements, enabling the prediction of new values [20,28]. A more detailed description on the main techniques used in the tutorial will be provided at the beginning of Sections 6 (exploratory analysis, PCA), 7 (regression, PLS) and 8 (classification, PLS-DA).

In the pharmaceutical industry, software compliance is crucial for ensuring that data analysis processes adhere to stringent regulatory standards. When dealing with data analysis and multivariate methods it is important to consider the European Pharmacopoeia General monograph 5.21 on Chemometric Methods [4]. This document outlines the principles and practices for applying chemometric techniques in pharmaceutical analysis. It emphasizes the importance of data integrity, validation and the use of multivariate methods to enhance the reliability of analytical results. The monograph also provides guidelines for model development and highlights the necessity of proper data preprocessing. By following these guidelines, the whole research and analytical activities can reap benefits such as ensuring accuracy, reliability, and integrity of the data, but also proper and efficient information gathering from experimental runs.

3.2. Software specifications

All chemometric elaborations described hereafter were performed under MATLAB environment (R2021a, The Mathworks Inc., Natick, MA, USA) using the toolboxes created by the Milano Chemometrics and QSAR Research Group (Department of Earth and Environmental Sciences, University of Milano-Bicocca). The toolboxes can be freely downloaded from their institutional website (https://michem.unimib.it/download/matlab-toolboxes/, last access October 11, 2024), but the reader can find an organized version of the toolboxes in the Codes Package provided with the tutorial. The instructions on how to correctly set up your MATLAB environment to be able to smoothly use these multivariate toolboxes and the scripts provided in this tutorial are provided in the Supplementary Materials.

3.3. Preparing the data and the tools

An overall description of the data analysis workflow is shown in Fig. 1. This figure summarizes the needed steps to perform a reliable and robust data analysis and exploration.

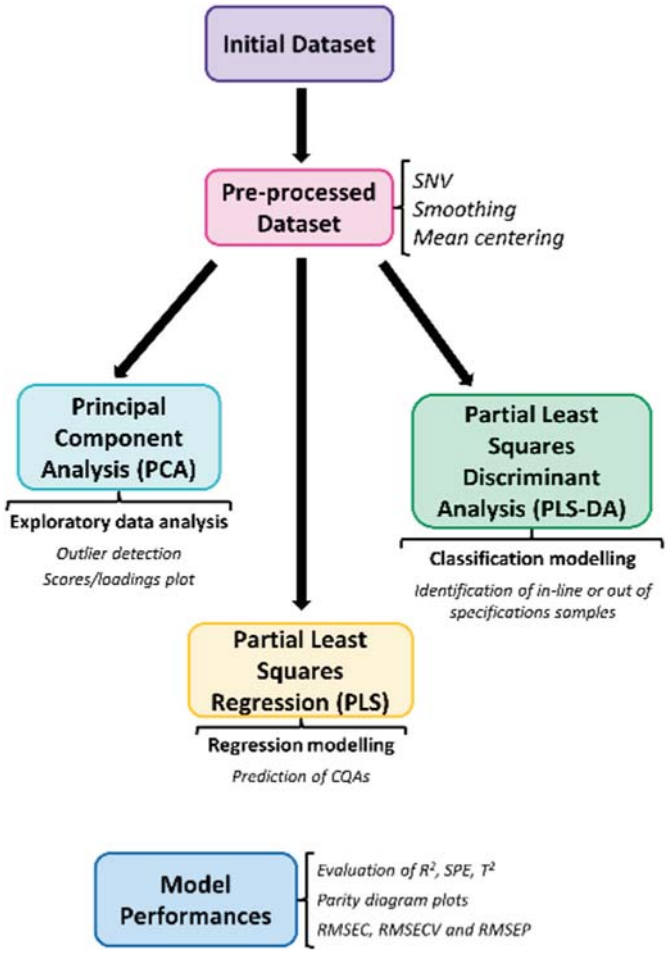


Fig. 1. Data analysis workflow for processing NIR spectra ("Initial Dataset") with annotations about the field of pharmaceutical applications.

The very first step is obtaining the raw data ("Initial Dataset") from the analytical instrument. Then, the preprocessing step is aimed at obtaining clean and useable data from the raw instrumental data. When the data can be considered ready for modelling, it is essential to conduct data exploration, generally using Principal Component Analysis (PCA, more details in Section 6.2), which allows to extract the information and highlight hidden relationships among the samples and the variables in the dataset, in an unsupervised manner. Further analytical steps are generally aimed at building predictive models, using supervised methods. The two main families of predictive modelling are regression and classification, which should be selected based on the following considerations:

- If the aim is to quantify a property, e.g., the content of water or sucrose in the formulation, then a *regression model* is needed. The most widely used method is Partial Least Squares (PLS) regression (more details in Section 7.2);
- If the aim is to classify the samples according to known classes of belonging, then a *classification model* is needed. In this sense, the most used method is the Partial Least Squares-Discriminant Analysis (PLS-DA, more details in Section 8.2).

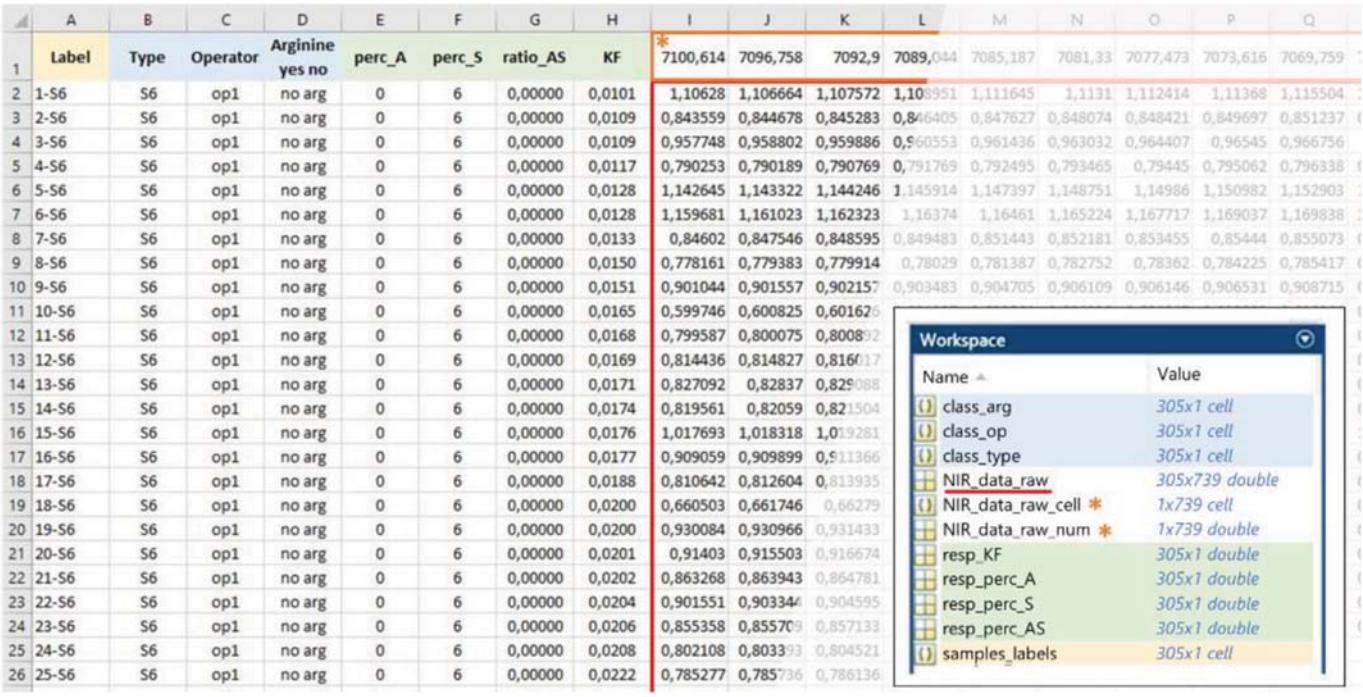
4. Dataset explanation: a case study on freeze-dried formulations

4.1. Data organization and import

The first step to get started inspecting and analysing the data is importing the raw data into MATLAB (or your coding environment of choice). In some cases, the instrument's software will produce an organized Excel file, which is rather easy to process. In any case, a good starting point would be to obtain a table like the one depicted in Fig. 2. In the tutorial's Excel file, all metadata (columns from A to H; more details on "metadata" in Section 4.2) and data (numerical columns from I on) are neatly organized.

From a practical point of view, a NIR spectrum is a collection of many absorbance values, one for each acquired wavelength. A set of NIR spectra can be organized in a data table in which each row represents one sample, and each column represents one descriptor, i.e., a specific wavelength. In the Excel file from column I on, there is a big numerical matrix (highlighted with a red contour): these are the NIR spectra, which we want to process using the tools of Chemometrics. Since each column of the NIR matrix represents one spectral wavelength (or wavenumber), the first row in the example (highlighted with an orange contour) of Fig. 2 contains the wavenumbers. This information will be used for plotting the spectra and to properly interpreting them. Regarding the metadata associated with the spectra, columns from A to H contain the samples' labels (A), the class information (B–D) and other numerical data (the responses, E–H).

The information included in the Excel file can be imported to MATLAB using the "import" command from the program's Home toolbar. This command activates a reader that allows opening, inspecting, and selecting the desired cells from an Excel file. Thus, by properly selecting the different columns or groups of cells, selecting the import format, and renaming the imported data each time, it is possible to obtain the situation depicted in the MATLAB's Workspace screenshot as shown in Fig. 2 (bottom right). Three different classes were imported as cell vectors: the presence of arginine ("class_arg"), the information about the operator who performed the analysis ("class_op"), and the type of freeze-dried solution ("class_type", classes as described in section 2.1). This type of metadata can be used for classification purposes. Regarding the numerical metadata, four responses were imported as a "Numeric Matrix": the percentage of arginine ("resp_perc_A"), the percentage of sucrose ("resp_perc_S"), the ratio between arginine and sucrose ("resp_perc_AS", not analyzed in this tutorial), and finally the residual moisture ("resp_KF"). This type of metadata can be used for building regression models. An example on how to programmatically read the data from Excel files and how to prepare them for the subsequent exploration and chemometric modelling is reported in the Code box 4.1. Run this code box to obtain the same objects in the MATLAB Workspace depicted in Fig. 2.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Label	Type	Operator	Arginine yes no	perc_A	perc_S	ratio_AS	KF	7100,614	7096,758	7092,9	7089,044	7085,187	7081,33	7077,473	7073,616	7069,759
2	1-S6	S6	op1	no arg	0	6	0,00000	0,0101	1,10628	1,106664	1,107572	1,10	1,111645	1,1131	1,112414	1,11368	1,115504
3	2-S6	S6	op1	no arg	0	6	0,00000	0,0109	0,843559	0,844678	0,845283	0,8	0,847627	0,848074	0,848421	0,849697	0,851237
4	3-S6	S6	op1	no arg	0	6	0,00000	0,0109	0,957748	0,958802	0,959886	0,9	0,961436	0,963032	0,964407	0,96545	0,966756
5	4-S6	S6	op1	no arg	0	6	0,00000	0,0117	0,790253	0,790189	0,790769	0,791769	0,792495	0,793465	0,79445	0,795062	0,796338
6	5-S6	S6	op1	no arg	0	6	0,00000	0,0128	1,142645	1,143322	1,144246	1,145914	1,147397	1,148751	1,14986	1,150982	1,152903
7	6-S6	S6	op1	no arg	0	6	0,00000	0,0128	1,159681	1,161023	1,162323	1,16374	1,16461	1,165224	1,167717	1,169037	1,169838
8	7-S6	S6	op1	no arg	0	6	0,00000	0,0133	0,84602	0,847546	0,848595	0,849483	0,851443	0,852181	0,853455	0,85444	0,855073
9	8-S6	S6	op1	no arg	0	6	0,00000	0,0150	0,778161	0,779383	0,779914	0,78029	0,781387	0,782752	0,78362	0,784225	0,785417
10	9-S6	S6	op1	no arg	0	6	0,00000	0,0151	0,901044	0,901557	0,902157	0,903483	0,904705	0,906109	0,906146	0,906531	0,908715
11	10-S6	S6	op1	no arg	0	6	0,00000	0,0165	0,599746	0,600825	0,60162						
12	11-S6	S6	op1	no arg	0	6	0,00000	0,0168	0,799587	0,800075	0,8008						
13	12-S6	S6	op1	no arg	0	6	0,00000	0,0169	0,814436	0,814827	0,816						
14	13-S6	S6	op1	no arg	0	6	0,00000	0,0171	0,827092	0,82837	0,829						
15	14-S6	S6	op1	no arg	0	6	0,00000	0,0174	0,819561	0,82059	0,82						
16	15-S6	S6	op1	no arg	0	6	0,00000	0,0176	1,017693	1,018318	1,0						
17	16-S6	S6	op1	no arg	0	6	0,00000	0,0177	0,909059	0,909899	0,9						
18	17-S6	S6	op1	no arg	0	6	0,00000	0,0188	0,810642	0,812604	0,8						
19	18-S6	S6	op1	no arg	0	6	0,00000	0,0200	0,660503	0,661746	0,66279						
20	19-S6	S6	op1	no arg	0	6	0,00000	0,0200	0,930084	0,930966	0,931433						
21	20-S6	S6	op1	no arg	0	6	0,00000	0,0201	0,91403	0,915503	0,916674						
22	21-S6	S6	op1	no arg	0	6	0,00000	0,0202	0,863268	0,863943	0,864781						
23	22-S6	S6	op1	no arg	0	6	0,00000	0,0204	0,901551	0,90334	0,904595						
24	23-S6	S6	op1	no arg	0	6	0,00000	0,0206	0,855358	0,8557	0,857133						
25	24-S6	S6	op1	no arg	0	6	0,00000	0,0208	0,802108	0,8033	0,804521						
26	25-S6	S6	op1	no arg	0	6	0,00000	0,0222	0,785277	0,785	0,786136						

Fig. 2. Visual representation of the dataset organized as an Excel worksheet (background). A screenshot of the MATLAB workspace is reported in the bottom-right corner, with coloring matching the original Excel information. Code box 4.1 will automatically get the information of the Excel file and import it into MATLAB, reproducing the Workspace depicted in the figure.

The NIR spectra were imported as a numeric matrix, as well as the corresponding wavenumbers vector. The latter was imported also as a cell vector, since the free toolboxes used in the tutorial require the axis scale in such format for graphical reasons.

Code box 4.1
Read data and metadata from the Excel file and define the objects on MATLAB. To directly apply this script, be sure that MATLAB's working folder contains the Excel file that is going to be read by the function "xlsread".

```
%% Read the data from Excel
[~, ~, raw_data] = xlsread('NIR_arginine_raw_data_metadata.xlsx', 'version_OK');

% Define the data objects
NIR_data_raw       = cell2mat(raw_data(2:end, 9:end));   % get raw spectra
NIR_scale_raw_cell = raw_data(1, 9:end);                 % get axis scale (wavenumbers)
NIR_scale_raw_num  = cell2mat(NIR_scale_raw_cell);       % get axis scale as numbers
                                                         % (for plotting)
samples_labels     = raw_data(2:end,1);                  % get samples' labels

% Define the continuous metadata (response vectors)
resp_perc_A   = cell2mat(raw_data(2:end,5));   % get the percentage of arginine
resp_perc_S   = cell2mat(raw_data(2:end,6));   % get the percentage of sucrose
resp_ratio_AS = cell2mat(raw_data(2:end,7));   % get the ratio between arginine and
sucrose
resp_KF       = cell2mat(raw_data(2:end,8));   % get the water content (from Karl-
Fisher method)
% These response vectors are originally stored as numbers in the "raw_data"
% cell array, so they must be extracted from the cells with the "cell2mat"
% function.

% Define the discrete metadata (class vectors)
class_type = raw_data(2:end,2);   % get the class "type"
class_op   = raw_data(2:end,3);   % get the class "operator"
class_arg  = raw_data(2:end,4);   % get the class "arginine yes no"
% These class vectors are strings stored in the "raw_data" cell array, so
% no extraction from the cells is needed, we can directly take a part of
% the complete "raw_data" cell array.

clear raw_data
```

4.2. Initial data analysis and metadata description

The exploration of the initial dataset and any relevant additional information is of fundamental importance for deciding what to model and for interpreting the results. In other words, for any dataset it is important to consider all available additional information (generally referred to as "metadata"), which may be:

- *Categorical information*: this corresponds to *qualitative information*, and it is generally referred to as "classes". For this reason, this type of information can be used in classification analysis. From the exploratory and interpretation point of view, the dataset or the results can be coloured according to the group/class belonging to each sample, allowing for easier and more visual interpretation.
- *Continuous numerical information*: this corresponds to *quantitative information*, and it can be used as "responses". Such information is generally not modelled together with the "data" (in our example, the NIR spectra), but can be used as a response vector in a regression analysis. From the exploratory and interpretation point of view, the dataset or the results can also be coloured based on the content of a specific property/compound in each sample.
- *Discrete numerical information*: this technically corresponds to *quantitative information*, and just like the continuous case it can be used as a response for a regression task, with some more care when interpreting the validity of the results.

Moving to our real-case NIR dataset, seven different samples subgroups (or classes) can be defined, as reported in Table 2. Specifically, most of the samples are sucrose solutions at different percentages: 6 %, 3 % and 9 %. Then, another consistent part of the dataset is represented by samples of sucrose to which an amount of the amino acid arginine was added. The remaining part of the dataset is represented by samples containing 6 % of trehalose instead of sucrose. The described solutions and concentrations refer to the liquid formulations, which were then subjected to the freeze-drying process. It is important to consider that each sample also contains a certain amount of residual water after the freeze-drying cycle. This content was determined by Karl-Fischer titration, which serves as the analytical reference technique.

The NIR spectra were acquired directly from the powders obtained

Table 2
Number of samples in each dataset and the description of the different formulations.

Dataset name	Liquid formulation	N° of Samples	Samples from Operator 1	Samples from Operator 2
S6	Sucrose 6 $\%_w$	91	1–91	0
S3	Sucrose 3 $\%_w$	59	1–28	29–59
S9	Sucrose 9 $\%_w$	36	1–8	9–36
SA05	Sucrose 6 $\%_w$ + arginine 0.5 $\%_w$	31	1–9	10–31
SA1	Sucrose 6 $\%_w$ + arginine 1 $\%_w$	32	1–8	9–32
SA3	Sucrose 3 $\%_w$ + arginine 3 $\%_w$	31	1–9	10–31
T6	Trehalose 6$\%_w$	25	1–25	0

from the drying process. An important aspect is that the acquisitions were performed by two operators, during different sessions of measurements (as reported in the last two columns of Table 2). This experimental aspect can have an impact on the results, due to the manual skill of the operator or the environment conditions, as shown and discussed in the Results (Section 6.2). In other words, for each dataset it is important to consider all available additional information in the form of "metadata", which in our case study would be:

- *Categorical information*: type of compounds in the formulations, the operator who performed the analysis. Coloring the data or the results according to these classes allows for easily spotting groupings and clusters in the data.
- *Continuous numerical information*: the amount of water contained in each sample. Coloring the data or the results according to the water quantification allows for spotting possible trends among the samples. Otherwise, this information could be used as a response vector in a regression model for estimating the water content (as reported in Refs. [13,14] by Massei et al. and Bobba et al.).
- *Discrete numerical information*: the percentages of arginine, sucrose, and trehalose in the considered product. Coloring the data or the results according to these quantifications is conceptually the same as the continuous numerical information, but the visual result could look like the case of the categorical information (as the discrete nature of these coloring vectors would result in fewer shades of colors).

5. Data preprocessing

5.1. Raw data inspection and preprocessing techniques

Before any modelling steps, it is important to inspect the raw data and choose one or more (if needed or in doubt) preprocessing methods. Prior to conduct multivariate data analysis, the quality of sample response may be evaluated. In fact, in the pipeline of multivariate data analysis, careful attention must be paid to the choice of the preprocessing technique to improve the quality of the analysis [4].

Starting from data inspection, it is always a good idea to simply plot the raw data, as they are. To do so, one can execute the code from Code box 5.1

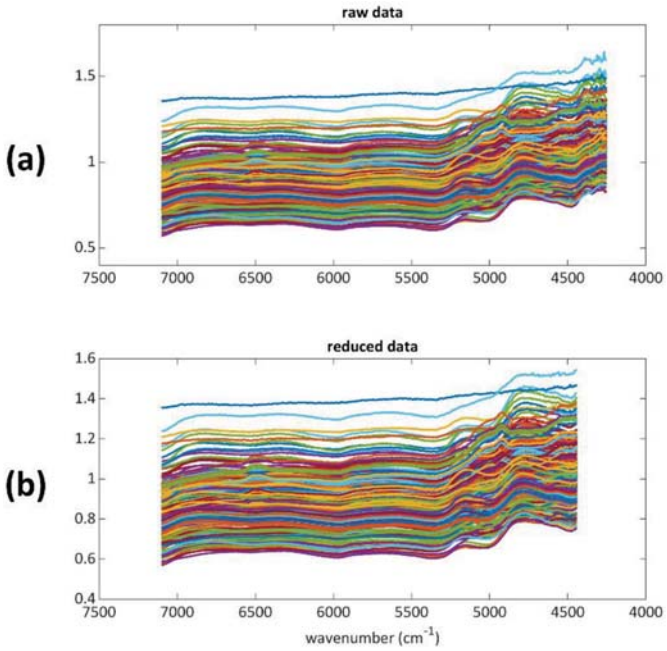


Fig. 3. Raw data (a) and the reduced data (b), i.e., without the noisy part at wavenumbers lower than 4443 cm^{-1}.

, which produces Fig. 3. The plot in Fig. 3a depicts the raw data ("NIR_data_raw"), in which we can notice a rather noisy part on the right end of the spectra: the information contained there could result confused by the noise, so a data reduction step is needed. The reduced version of the dataset ("NIR_data") is depicted in Fig. 3b. This dataset will be used to proceed with the analytical workflow.

To this point, no signal interpretation has been performed: the situation depicted in Fig. 3b allows for a general overview of the data, maybe allowing for spotting potential outliers (like the dark blue spectrum in the top part of the plot), but no more detailed conclusions can be drawn. To start understanding the information content of our data and

Code box 5.1
Data reduction: cut the noisy part and define a new MATLAB object containing the data to be modelled with PCA (and the further analytical steps).

```
%% Preprocessing - transform the raw data
NIR_data       = NIR_data_raw(:,1:690);       % reduce data removing the noisy parts
NIR_scale_num  = NIR_scale_raw_num(1:690);    % remember to cut the axis scale too!
NIR_scale_cell = num2cell(NIR_scale_num);     % this "cell" version of the axis scale
                                              % is needed for the toolboxes used in this
                                              % tutorial

figure                                  % open a new figure

subplot(2,1,1)                          % plot in the upper part of the figure
plot(NIR_scale_raw_num, NIR_data_raw')  % plot the raw data with their scale
title('raw data')                       % add title to the subplot
set(gca,'xdir','rev')                   % revert the x-axis direction

subplot(2,1,2)                  % plot in the lower part of the figure
plot(NIR_scale_num, NIR_data')  % plot the reduced data with their scale
title('reduced data')           % add title to the subplot
set(gca,'xdir','rev')           % revert the x-axis direction

xlabel('wavenumber (cm^-^1)')   % add x-label
```

to get prepared for the modelling steps, the preprocessing step is needed.

The aim of preprocessing is to highlight the information content of the data, mainly by further noise reduction, and in the case of NIR spectra also by removing offsets due to scattering phenomena, which are generally not related to the chemical information. Scattering arises from the interaction of the particle with light, so it can be due to technical limitations of the instrument, to the acquisition conditions or just to the physical nature of the sample [29].

indicated by Sun et al. [30]. However, it is good practice to plot and visually inspect the smoothing results, also by applying different window sizes. In this tutorial we will use a 31-points smoothing window, and a comparison with the raw data and the application of a 9-points window can be inspected in Fig. 4 (as a result of running Code box 5.2). You can play with the smoothing window size by changing the last input in the function "smoothdata" in Code box 5.2 to see how different values produce different results.

Code box 5.2
Data preprocessing: smoothing application and plot of a specific zoom, i.e., in the wavenumber range 4500–5500 cm^{-1}, to better visualize the smoothing effect on the noisy region of the spectra.

```
%% Apply a light smoothing (with Savitzky-Golay method) to the data
NIR_data_sm9  = smoothdata(NIR_data, 2, 'sgolay', 9);    % smooth data (window = 9)
NIR_data_sm31 = smoothdata(NIR_data, 2, 'sgolay', 31);   % smooth data (window = 31)

% Plot and compare the different levels of smoothing
figure

% plot the raw data
subplot(3,1,1), plot(NIR_scale_num, NIR_data')
title('raw data')
set(gca,'xdir','rev', 'XLim',[4500 5500], 'YLim',[0.5 1.8])
%                     'XLim' and 'YLim' set the horizontal and vertical
%                     intervals in of the desired zoom into the plot

% plot the smoothed data with window = 9
subplot(3,1,2), plot(NIR_scale_num, NIR_data_sm9')
title('smoothed data (window = 9 pts)')
set(gca,'xdir','rev', 'XLim',[4500 5500], 'YLim',[0.5 1.8])

% plot the smoothed data with window = 31
subplot(3,1,3), plot(NIR_scale_num, NIR_data_sm31')
title('smoothed data (window = 31 pts)')
set(gca,'xdir','rev', 'XLim',[4500 5500], 'YLim',[0.5 1.8])

xlabel('wavenumber (cm^-^1)')     % add x-label
```

The preprocessing methods can be divided into two families: row-wise methods, in which the mathematical transformation is performed on each sample on its own, and column-wise methods, in which the operation affects all samples. The most common row-wise methods are the normalizations, while the most common column-wise ones generally operate some sort of scaling [30]. In the present study, the two main techniques for NIR spectra (plus mean centering, as described below) were employed:

- **Smoothing** [30]**:** this row-wise preprocessing aims at removing the high-frequency variability, which in the case of spectra is mainly ascribable to the experimental noise. This is generally done by smoothing out the sample's profile using a moving window in which the corrected value of the window's central point is computed combining the information of its neighbouring points. The window for smoothing NIR spectra must have an odd number of points (a central point is needed) and it is generally in the interval 7–11, as
- **Standard Normal Variate (SNV,** [30]**):** this method is a must-have in NIR spectroscopy preprocessing. It allows for better highlighting the actual differences among the samples, while removing at the same time offsets due to scattering effect. Each spectrum is normalized by subtracting its mean value and dividing by its standard deviation. The application of SNV is reported in Code box 5.3.
- **Mean centering:** in this final preprocessing step the "average spectrum" computed from the whole dataset is removed from each sample, allowing for modelling the data directly according to the differences *among* samples. Mean centering is applied to our data in Code box 5.3.

This preprocessing sequence is rather standard, and it is generally the go-to preprocessing to start analysing NIR data. As described in Sun et al. [30], there are many other methods for NIR spectra preprocessing, which can be chosen according to the needs of the analyst.

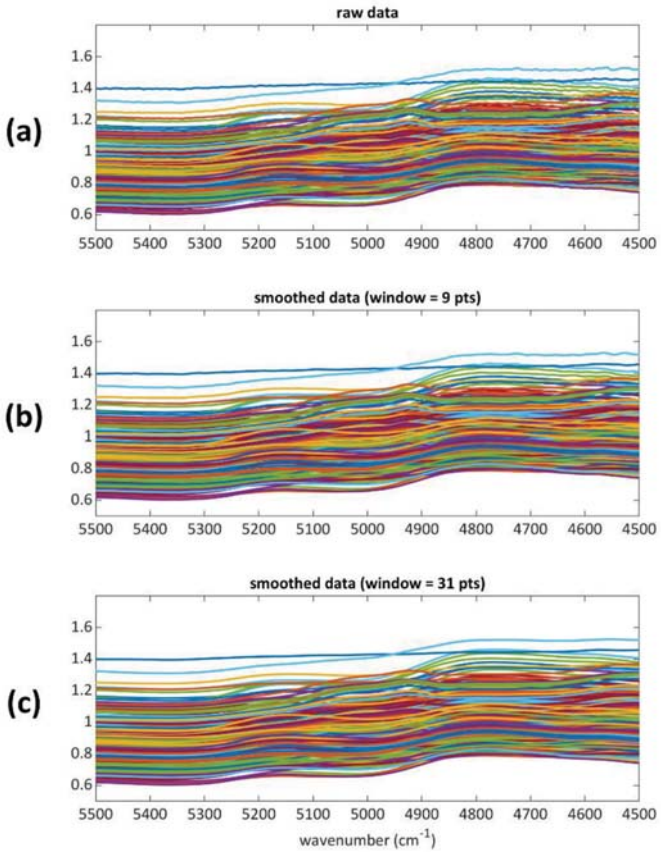


Fig. 4. This very simple plot allows inspecting the effect of different levels of smoothing, as defined in Code box 5.2: (a) the raw non-smoothed data; (b) smoothed data with window = 9; (c) smoothed data with window = 31. It can be noticed that the overall noise affecting the spectra decreases with increasing smoothing window sizes, and this is particularly clear in the right part of the plots. Please note that the code for this plot is written in a very basic manner, so no axis scale is plotted.

Code box 5.3
Data preprocessing: apply SNV correction and mean center to the data, then plot the preprocessing results.

```
%% Apply standard normal variate (SNV) to the smoothed data
NIR_data_sm31_SNV = SNV(NIR_data_sm31);    % apply Standard Normal Variate
                                           % preprocessing using the "SNV" function
                                           % provided in the tutorial's scripts

% Apply mean center to the smoothed and SNV data
NIR_data_sm31_SNV_mc = meancenter(NIR_data_sm31_SNV);    % apply mean center using
                                                         % the "meancenter" function
                                                         % provided in the tutorial's
                                                         % scripts
```

The chosen preprocessing sequence can be inspected step by step by running Code box 5.4, which produces Fig. 5: by moving from (a) to (b) we can see the effect of SNV on the raw data, while by moving from (b) to (c) the effect of mean center becomes clear. Now the preprocessed data exhibit smooth signals, which can be interpreted according to literature assignments of personal knowledge of the acquired data.

Code box 5.4
Data preprocessing: plot the preprocessing results.

```
%% Preprocessing - plot the preprocessing steps
figure

% 1) Plot the smoothed data
subplot(3,1,1)
plot(NIR_scale_num, NIR_data_sm31')
title('Preprocessed data: smoothing')
set(gca, 'XGrid','on', 'xdir','rev'), box on      % 'XGrid' enables the vertical grid
on the main x-axis ticks

% 2) Plot the smoothed + SNV data
subplot(3,1,2)
plot(NIR_scale_num, NIR_data_sm31_SNV')
title('Preprocessed data: smoothing + SNV')
set(gca, 'XGrid','on', 'xdir','rev'), box on

% 3) Plot the smoothed + SNV + mean centered data
subplot(3,1,3)
plot(NIR_scale_num, NIR_data_sm31_SNV_mc')
title('Preprocessed data: percentage (%) of sucrose')
set(gca, 'XGrid','on', 'xdir','rev'), box on

xlabel('wavenumber (cm^-^1)')    % add x-label
```

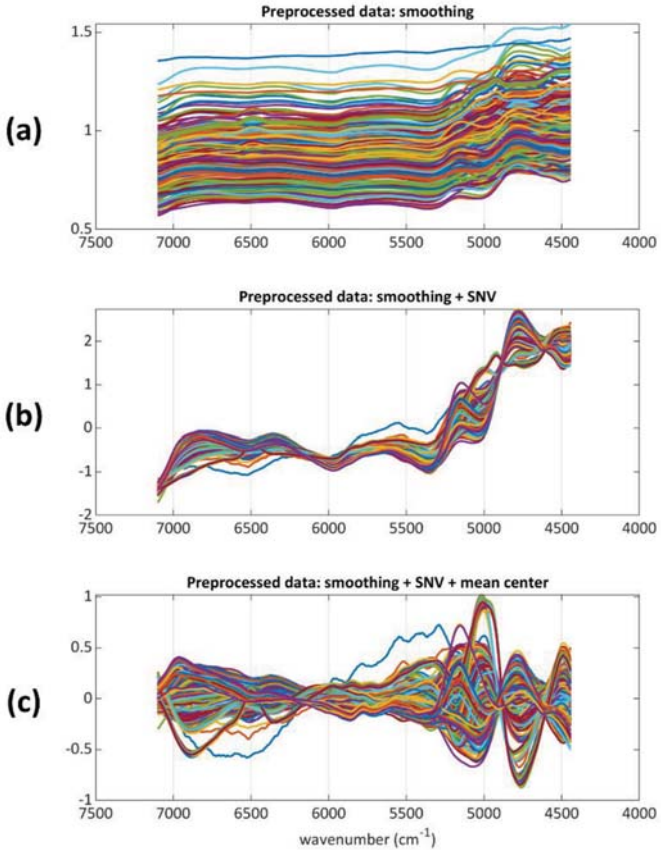


Fig. 5. Data preprocessing application: (a) smoothed data; (b) smoothed and SNV-corrected data; (c) final combination of preprocessing (smoothing + SNV + mean center) which will be used for all subsequent chemometric modelling.

To better inspect the preprocessed data, the spectra can be coloured according to the metadata information. An example using two numerical responses is provided in Code box 5.5. This produces Fig. 6, in which the dataset of Fig. 5c is now coloured according to the percentage of arginine (a), the percentage of sucrose (b), and the water content (c). The colorbars on the right-hand side allow interpreting the colors of the spectra, thus allowing to recognize the most interesting areas of the spectra. To proceed with the interpretation according to specific signals, a preliminary assignment study can be done, as described in the following section.

Code box 5.5
Data preprocessing: plot the chosen preprocessing combination and color according to the metadata (samples type, percentage of arginine, percentage of sucrose) using the function plot_color_resp provided with the tutorial's code package.

```
%% Preprocessing - plot the datasets for inspection
figure

% 1) Plot the smooth + SNV + mean center data, colored according to percentage of
arginine
subplot(2,1,1)
plot_color_resp(NIR_data_sm31_SNV_mc, resp_perc_A, NIR_scale_num, '', 1)
title('Preprocessed data: percentage (%) of arginine')
set(gca, 'XGrid','on', 'xdir','rev'), box on
h = colorbar; h.Location = 'east';   % get the colorbar's handle and move it within the
plot

% 2) Plot the smooth + SNV + mean center data, colored according to percentage of
sucrose
subplot(2,1,2)
plot_color_resp(NIR_data_sm31_SNV_mc, resp_perc_S, NIR_scale_num, '', 1)
title('Preprocessed data: percentage (%) of sucrose')
set(gca, 'XGrid','on', 'xdir','rev'), box on
h = colorbar; h.Location = 'east';   % get the colorbar's handle and move it within the
plot

% 3) Plot the smooth + SNV + mean center data, colored according to the water content
subplot(3,1,3)
plot_color_resp(NIR_data_sm31_SNV_mc, resp_KF, NIR_scale_num, '', 1)
title('Preprocessed data: water content (KF)')
set(gca, 'XGrid','on', 'xdir','rev'), box on
h = colorbar; h.Location = 'east';   % get the colorbar's handle and move it within the
plot

xlabel('wavenumber (cm^-^1)')    % add x-label
clear h                          % clean up a bit
```

5.2. Preliminary signal assignment

Since each chemical component of the powder formulation has its own specific NIR spectral profile, it is possible to try to identify the most characteristic signals. A short list of band assignments is reported in Table 3. As previously underlined, all formulations differ by the contents of arginine, trehalose, and water. In the region between 5000 and 4500 cm^{-1} sucrose shows an absorption band corresponding to the C–H stretching. Regarding water, it is important to focus on the signals arising from the band of O–H stretching and H–O–H bending at around 5150 cm^{-1}. As expected, this band shows the largest variability.

On the other hand, the spectra of the formulations containing arginine appear quite different. A close inspection of Fig. 6a clearly shows that as the concentration of arginine increases (in terms of percentage of the total solid fraction), the absorption bands in the water region tends to get lower and lower. Moreover, another band, specific of pure arginine, appears at around 4900 cm^{-1}. This finding can be better visualized by looking at Fig. 6a, where the samples were coloured according to the percentage of arginine. This is a crucial point for the understanding of the following findings.

6. Exploratory data analysis

6.1. Why do we need to explore the data?

Pharmaceutical products must meet standards set by the regulatory health authorities as far as concerns their quality. The two most important phases to create a new drug are:

1) *Formulation development*: choose the excipients to be mixed with the APIs to obtain certain/desired physical-chemical properties.
2) *Process development*: all the processing parameters, such as temperature, mixing conditions, the duration of each process step, etc, must be monitored since they strongly define the properties of the final product.

In this framework, exploring the NIR data is of fundamental importance to assess them from the points of view of both the "numerical quality" (i.e., the data integrity, to spot obvious wrong acquisitions or import problems like missing commas giving rise to oddly big or small numbers) and the information content (also in relation to the additional information and other measured chemical properties). To this aim, an

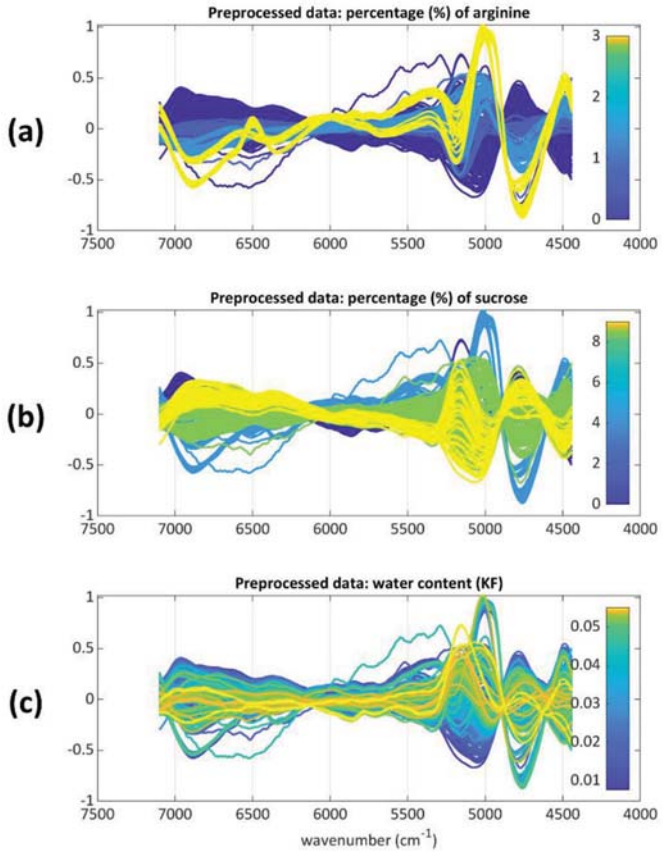


Fig. 6. Preprocessed dataset (smoothing + SNV + mean centering) coloured according to (a) the percentage of arginine, (c) the percentage of sucrose and (c) the water content.

Table 3
Band assignments.

Band assignment	Wavenumber [cm^{-1}]
Water	5150, 6900
C–H stretching	2960, 5920 and 8880
Arginine	4900 and 6500

exploratory data analysis step is needed, and many multivariate tools are available: Principal Component Analysis is usually the go-to method for inspecting the information content of any numerical dataset. The goal of exploratory analysis is to detect correlation patterns among the samples and among the variables, which in our case are represented by the intensities at different wavenumbers of the NIR spectra [31]. By using PCA all relevant information can be extracted and visualized, emphasising both similarities and differences among the samples, but also among the variables. In addition to this, potential outlier samples can be identified, inspected, and possibly removed from the dataset, leading to more reliable and robust quantitative models.

6.2. What is Principal Component Analysis?

Principal Component Analysis (PCA [32]) is a mathematical decomposition tool that aims at extracting and organizing all pieces of information contained in a dataset. It defines the so-called principal components (PCs), which are "new summary variables" describing patterns of correlations among the original variables: each PC describes a phenomenon contained in the data. These newly defined variables allow describing the data within a mathematical space of fewer dimensions, generally in the order of ten PCs: this is particularly useful in the case of spectral data, where hundreds of (correlated) variables, i.e., the wavelengths or the wavenumbers, are recorded for each spectrum. PCA exploits the natural correlations among the variables to describe overall patterns, ordering them from the "most intense" (which mathematically correspond to the largest sources of variance) to the smaller ones. However, in the case of spectra, two or three PCs are often adequate to capture a phenomenon as they describe most of the variance.

Mathematically speaking, PCA is a bilinear decomposition method, which means that the initial data are decomposed into their patterns of correlations in both the samples and variables directions. This is why from a PCA model we get to inspect the relationships among samples using the so-called "scores plots" and the relationships among variables using the so-called "loadings plots". These are scatter plots in which each sample or variable is represented as a point, whose coordinates are the respective scores or loadings values of the PCs that are plotted. Of course, only pairs or at most triplets of PCs can be plotted, so the visual inspection of a PCA model when more than three PCs are modelled can require inspecting different combinations.

It is very important to remind that PCA (and exploratory analysis in general) always represents the first step of any data analysis pipeline [31–33]. In this tutorial, we are going to use PCA to inspect the information content of our NIR data, especially for linking the spectral information with the external additional information about the arginine, sucrose, and water content, in the perspective of building predictive (i.e., regression and classification) models. PCA is not a predictive model as it is an unsupervised approach, i.e., the only drive for building the model is maximizing the variance explained by each PC, so PCA only uses the information of the dataset itself, and no other/external information is included or related to such information. As anticipated in Section 4.2 about the importance and organization of the metadata, the additional information can be used to color the scores plots, allowing to spot tendencies and to possibly explain the natural groupings present in the data: if a match is found between a cluster and the additional information, or a tendency is noted in the PCs space, this could be sign that the modelled data have a relationship with the additional information, so further regression or classification models might be built.

6.3. PCA results

The discussion of PCA results is largely based on visual inspection of the scores and loadings and will be focused on the groupings related to the kind of sample, the operator who performed the experiments and the trend linked to the arginine and the water content. Remember that all these pieces of additional information are not included in the data matrix modelled by PCA, so if any groupings or trends were to be found, they were naturally present in the spectral data.

To determine how many components are needed to properly model the data, one should begin by inspecting the eigenvalues plot (Fig. 7), and then proceed to inspect how the information, e.g., groups and trends, is described across the different scores plots. The eigenvalues plot (in the toolbox: Results > PCA Results > plot eigenvalues) represents the information described by each component. The three subplots of Fig. 7 basically represent the same information, but from different points of view. The "eigenvalues" and the explained variance ("exp var (%)") both describe the how much of the total variance is associated with each component: the components are naturally ordered from the largest to the smallest. The smaller the share of variance explained by each component, the more likely it is that the component does not carry relevant information. For example, PC1 clearly captures a significant and potentially valuable portion of the information. PC2 follows with a much smaller percentage, and PC3 with even less. While PC4 may be similar to PC3, it is likely already capturing variance unrelated to meaningful information, indicating that at this point, the model is at least partially describing the noise.

This profile in the eigenvalues and explained variance plots is very

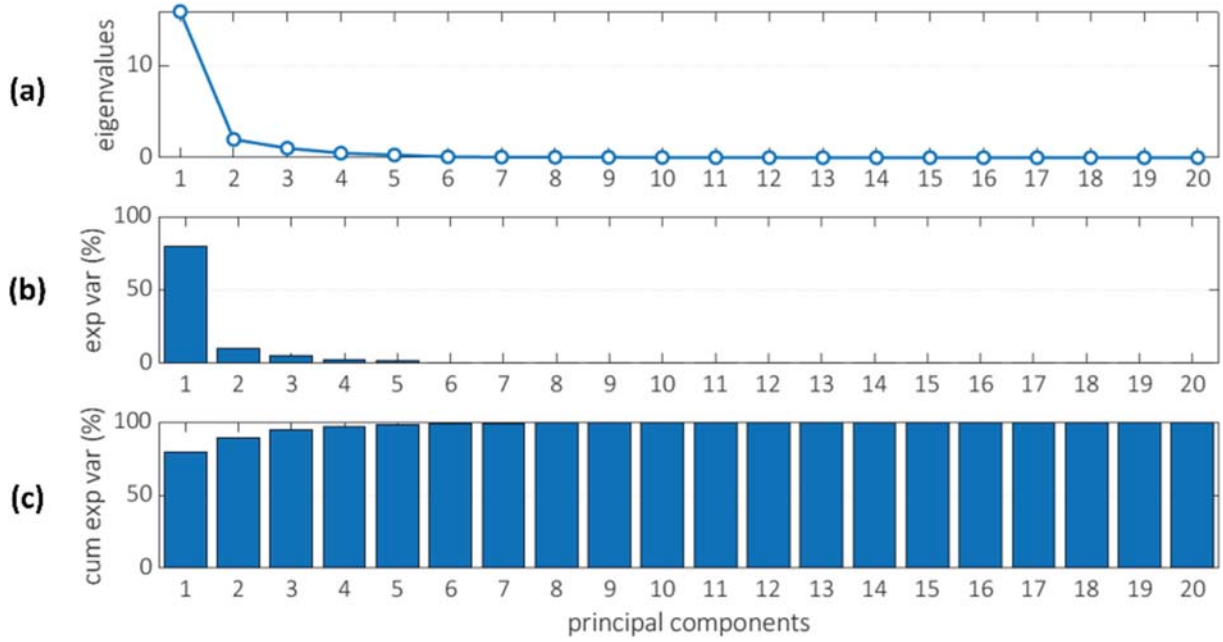


Fig. 7. PCA eigenvalues plot: (a) the eigenvalues obtained by mathematical decomposition; (b) the variance explained by each component; (c) the cumulative explained variance associated to each component.

common with spectral data, as there often is a very large PC1, followed by few smaller but relevant components, quickly fading into noise components. The third subplot "cum exp var (%)" is obtained by adding the component's explained variance to the sum of the previous ones.

As a complementary inspection the scores plot can help understanding where the information starts to fade into unstructured variability, i.e., into the noise. In our case study the first three PCs resulted to be the most meaningful and are reported and discussed in Fig. 7. For the sake of brevity, the components from PC4 on will not be shown nor discussed. Also, the discussed PCA model was fitted with three principal components: this is crucial to discuss the outliers inspection, as the visualization to spot them changes according to how many components are included in the model.

The potential outliers are generally identified by inspecting the so-called "residuals plot". To obtain the plot using the toolbox open the "scores" window (Results > PCA results > scores), and then select "Hotelling T2" to be plotted on the x axis, and "Q residuals" to be plotted on the y axis. A plot like the one reported on Fig. 8 should be obtained.

This scatter plot combines two important and complementary measures about the samples: the Q residuals measure how much unmodelled information is still associated with each sample, while the Hotelling T^2 measures how extreme each sample is within the model. The red horizontal and vertical dashed lines are statistically computed limits that can be used as a reference to estimate whether a sample exceeding them should be considered an outlier or not. The situation of Fig. 8 describes a nice compact dataset: most of the samples are densely grouped within the statistical limits, few of them exceed lightly one or both limits while remaining close to the large part of the rest of the samples. Just one sample exceeds both thresholds in a significant way: sample 128-S3 (spectrum in row number 148 of the data table). This sample was already noticed in Section 5.1 ("like the dark blue spectrum in the top

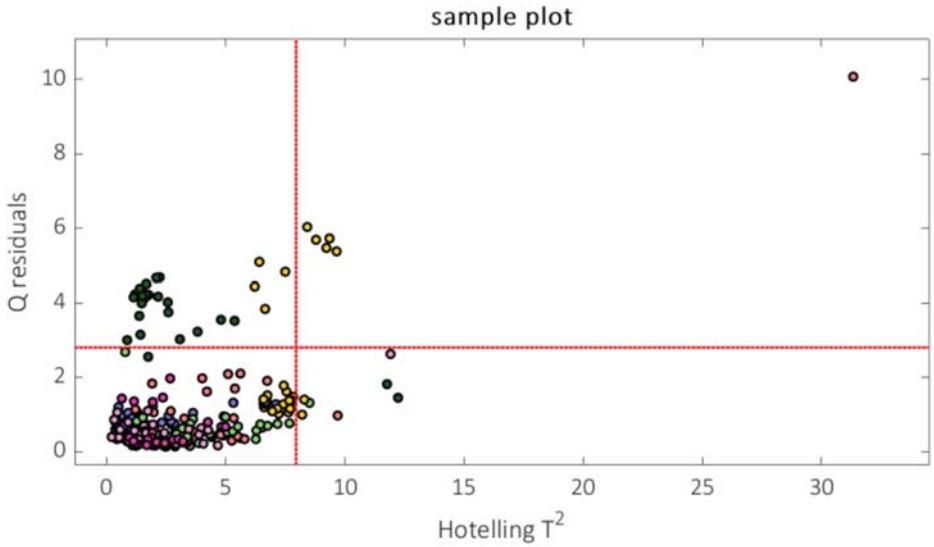


Fig. 8. PCA residuals plot.

part of the plot" of Fig. 3), and it should be inspected before considering removing it from the dataset. By running Code box 6.1, one can obtain Fig. 9, a visual representation of the whole spectral dataset (in grey) with the potential outlier sample (128-S3) plotted in red over it.

Code box 6.1
Plot specific samples over the whole dataset (represented in grey). Fig. 8 will be obtained.

```
%% Plot potential outlier against the whole dataset

figure   % open new figure

% plot the whole dataset in grey
plot(NIR_scale_num, NIR_data_sm31_SNV', 'Color',[0.7 0.7 0.7])
                                          % [0.7 0.7 0.7] = RGB values for a light
shade of grey

% plot the potential outlier in red
hold on   % keep what was already plotted and overimpose the following plotting command
plot(NIR_scale_num, NIR_data_sm31_SNV(148,:)', 'Color','r')   % 'r' means color in red

set(gca, 'XGrid','on', 'xdir','rev'), box on
xlabel('wavenumber (cm^-^1)')
```

The potential outlier generally follows the shape of the other spectra but exhibits higher noise and different offsets across the spectral window. With the sample's name and number, the analyst could trace its history and potentially explain the unusual appearance of its spectrum. However, to avoid producing even more datasets and make the tutorial's workflow more complex, sample 128-S3 will be kept in the dataset. The results of the PCA and further modelling are going to be basically the same. Please refer to Code box S1 in the Supplementary Materials to get an example of how to remove an outlier from the data table in MATLAB.

The scores plot of PC1 vs PC3 are reported in Fig. 10a: the samples result grouped according to their type (refer to the legends for interpreting the groups). These PCs provide a clear grouping trend related to the presence of arginine. In fact, the sucrose-based formulations and the trehalose ones are almost overlapping in the central area of the plot (Fig. 10a). The three different arginine-based formulations are located on the diagonal of the plot as the arginine concentration increases, as confirmed by Fig. 10d where the same PC1-PC3 scores plot is coloured according to arginine content in percentage. The NIR signals responsible for these groupings and trends are described by the loadings plots, which are reported in Fig. 10e (PC1) and 6.4f (PC3). These plots are used as a link with the original spectral data and allow us to interpret the information provided by the scores according to actual signals. In particular, a close inspection of Fig. 10e allows detecting the specific signals at 4900 cm^{-1} and 6500 cm^{-1} that were only found in the dataset at the highest concentration of arginine, i.e., samples of type SA3. In fact, the yellow samples in Fig. 10d result isolated from the others due to their specific absorption bands, which have positive values on both loadings plots: indeed, the SA3 samples are located in the positive directions of both PCs (first quadrant).

Moving to the content of sucrose, in Fig. 10c the same PC1-PC3

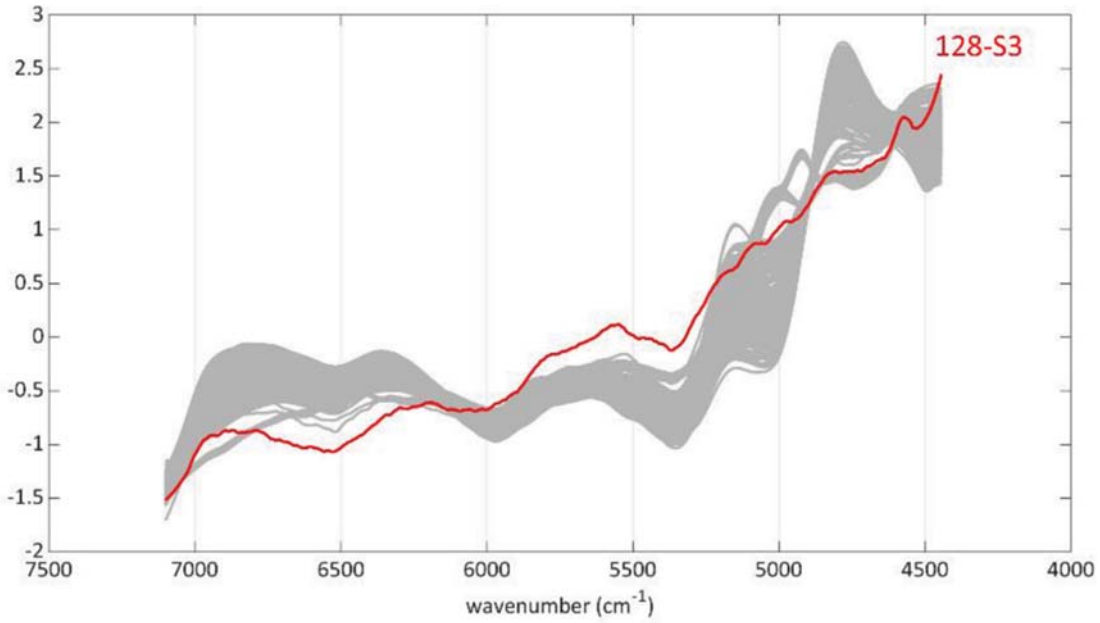


Fig. 9. The potential outlier 128-S3 plotted against the whole dataset in grey (code in Code box 6.1).

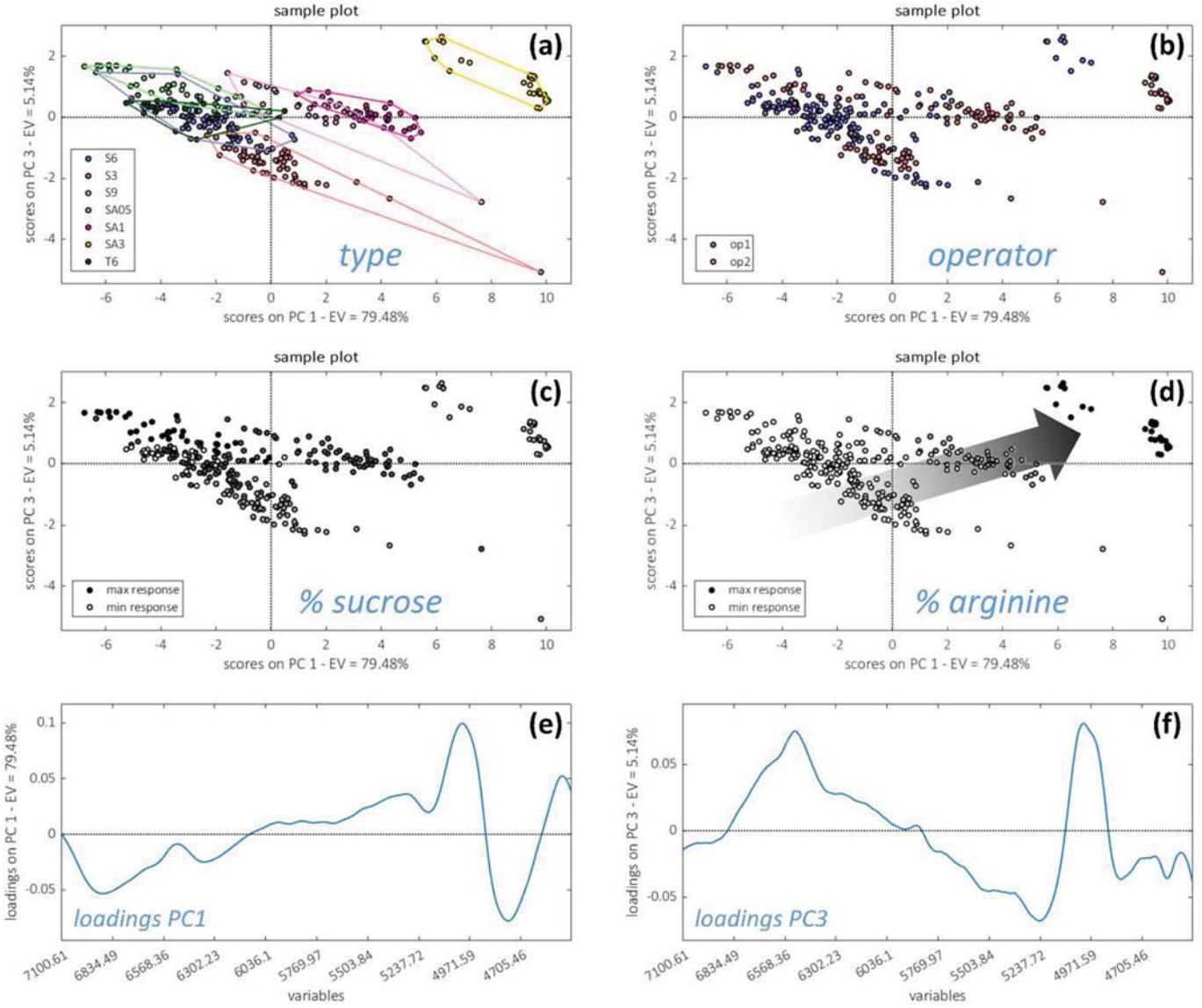


Fig. 10. PCA results, PC1-PC3. In the first four plots, the scores are: (a) coloured using the type class information "class_type"; (b) coloured using the operator class information "class_op"; (c) coloured according to the percentage of sucrose ("resp_perc_S″); (d) coloured according to the percentage of arginine ("resp_perc_A″). In (e) and (f) the PC1 and PC3 loadings are reported, respectively.

scores plot is reported and coloured according to the percentage of sucrose in the formulation. Some confirmation of the previously found information can be obtained:

- the samples at the highest percentage of sucrose (9 %) are located in the central cluster for positive values of PC3;
- the samples with the lowest percentage of sucrose (3 %) are located in the central cluster for negative values of PC3;
- the samples at 6 % of sucrose and the trehalose formulation are located in the central cluster around the zero of PC3 and they are almost overlapping.

It can be concluded that the samples containing sucrose show an increasing trend from the bottom to the top moving from negative to positive PC3 values.

A closer inspection of Fig. 10a–*c*-d allows identifying two distinct subgroups within the SA3 samples. By coloring the samples according to the operator information (Fig. 10b) an explanation for this separation could be found: the two groups could be related to the two different operators. By inspecting the preprocessed spectra of these samples (Fig. 11c), it can be noticed that the spectra corresponding to operator 1 (in red) differs from the spectra of operator 2 (blue) due to the absence of the absorption band typical of pure arginine powder, around 4900 cm^{-1}. The loadings of PC3, reported in Fig. 10e, confirm the importance of this arginine signal. Therefore, the samples produced by operator 1 could be considered unreliable, due to possible issues during the measurements or even their preparation, as there is an apparent lack of arginine. The ability of PCA to provide clear information about the effect of the operator on the measurements is potentially an extremely useful achievement, since it allows to better investigate on the possible issues during the experimental tests, finally leading to data of higher quality and, in turn, to more reliable models.

A visual representation of the data like the one of Fig. 11c can be very useful for interpreting the groupings identified in the PCA scores. The

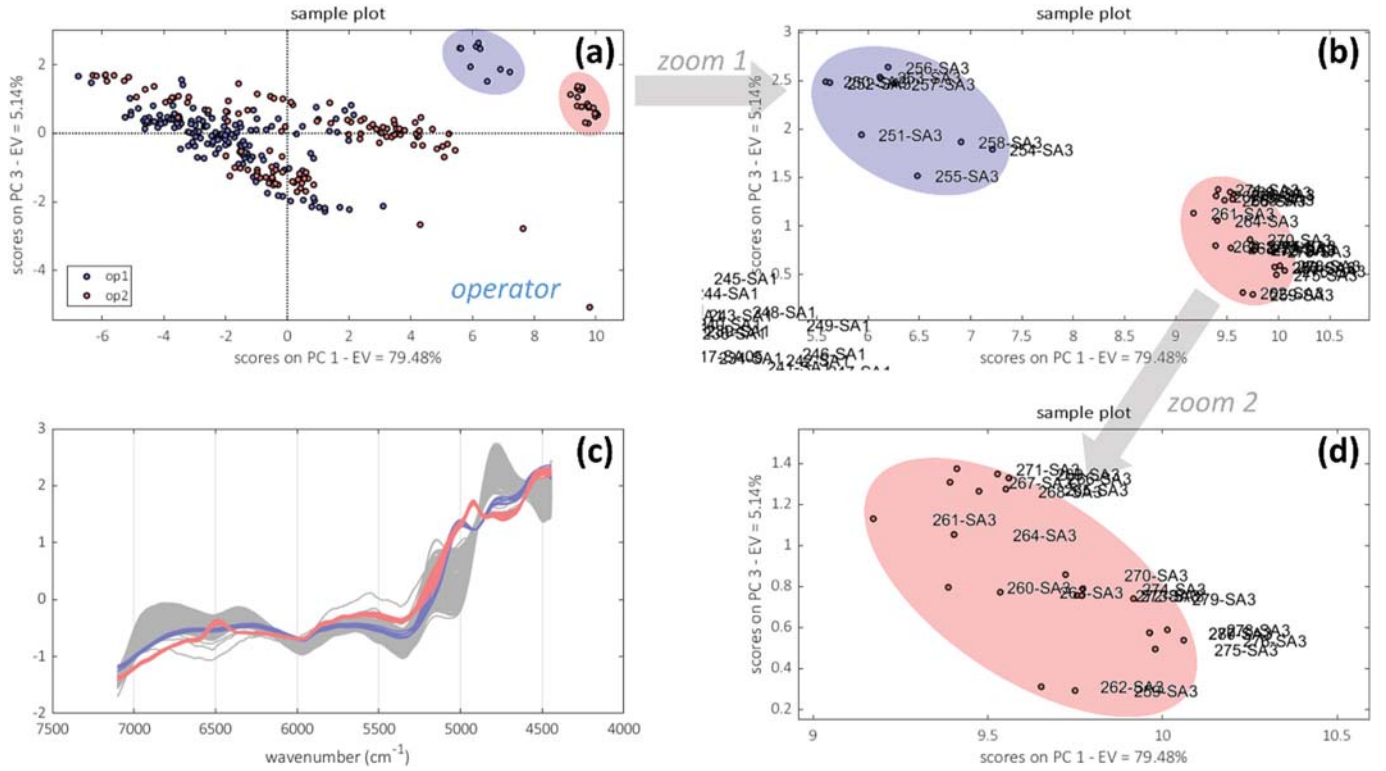


Fig. 11. Inspection of two groups of samples related to the operator effect: (a) scores plot coloured according to the operator, and (c) the with identified samples plotted against the whole dataset in grey (code in Code box 6.2). The plots in (b) and (d) are zooms of the exported scores plot with the labels: different zooms allow unravelling the overlapped labels to find out which samples correspond to the two clusters.

code for setting two groups of samples, in this case the two SA3 sub-groups corresponding to the operators (Fig. 11a), is provided in Code box 6.2. The samples numbers are needed in lines 3 and 4 of the Code box, and two ways for identifying them can be: 1) zoom in into the scores plot (Fig. 11c and d) and add the labels to the plot (this can be easily done within the toolbox) to get the "identity" of the samples; 2) inspect the metadata table to locate these SA3 samples and then inspect the operator information.

PC1 vs PC2 is another interesting combination to inspect. The PC1 vs PC2 scores plot depicted in Fig. 12a provides information about the content of water. Here the samples are coloured according to the water content determined through the Karl Fisher method. It is possible to observe a diagonal trend moving from top-left to bottom-right, with some exceptions. By analyzing the spectra of these exceptions, no evident anomalies were found. The loadings plot in Figure 12c-d confirm that the most important signal is related to water, and it corresponds to

Code box 6.2
Plot specific samples over the whole dataset (represented in grey). Fig. 11c will be obtained.

```
%% Plot weird samples against the whole dataset
% define the positions of the spectra to inspect...
weird_op1 = [250:258];    % ...from operator 1
weird_op2 = [259:280];    % ...from operator 2

figure

% plot the whole dataset in grey
plot(NIR_scale_num, NIR_data_sm31_SNV', 'Color',[0.7 0.7 0.7])
                                          % [0.7 0.7 0.7] = RGB values for a light
shade of grey
% plot samples from operator 1
hold on   % keep what was already plotted and overimpose the following plotting command
plot(NIR_scale_num, NIR_data_sm31_SNV(weird_op1,:)', 'Color',[128 128 255]/255)

% plot samples from operator 2
hold on
plot(NIR_scale_num, NIR_data_sm31_SNV(weird_op2,:)', 'Color',[255 128 128]/255)

set(gca, 'XGrid','on', 'xdir','rev'), box on
xlabel('wavenumber (cm^-^1)')
```

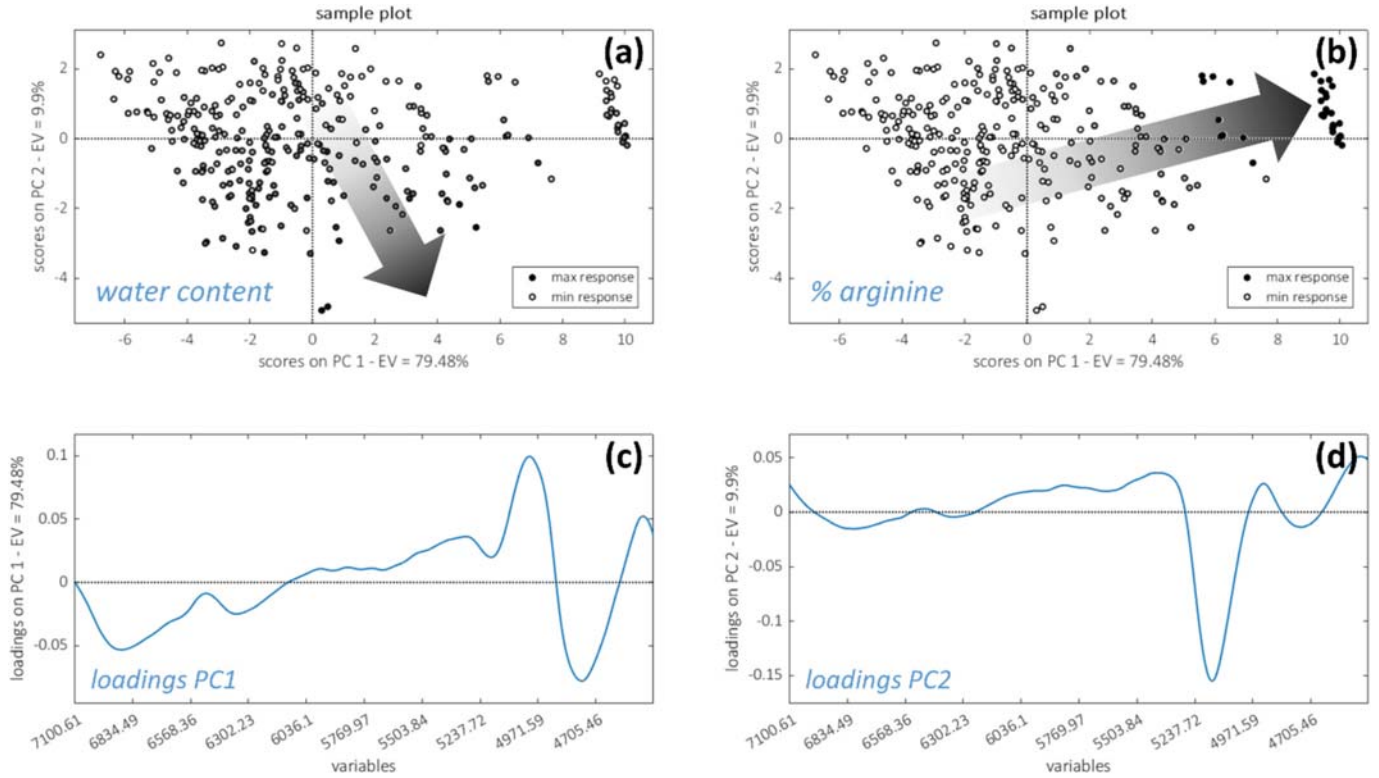


Fig. 12. PCA results, PC1-PC2. In the first two plots, the scores are: (a) according to the water content determined through the Karl Fisher method ("resp_KF"); (b) coloured according to the percentage of arginine ("resp_perc_A"). In (c) and (d) the PC1 and PC2 loadings are reported, respectively.

the band around 5150 cm^{-1} for both components, and for PC2 it is also the most influential signal. Notice that the orientation of the water content trend (top-left to bottom-right) is determined by the fact that the signals related to water have positive values in the PC1 scores (which are plotted horizontally in Fig. 12a) and negative in the PC2 scores (which are plotted vertically in Fig. 12a). However, the most influential PC for the water content is PC2, as the trend develops mainly vertically. PC1 also describes the signals of arginine, and a trend related to the content of arginine can be clearly seen in Fig. 12b: its direction is mainly horizontal, as highlighted by the shaded arrow.

Having performed a rather deep exploratory data analysis, the reader should now be able to remove the outliers or select parts of the data on which it could be more important/interesting to focus the analysis. These decisions are strongly related to the purpose of the study and should be carefully considered before proceeding with more advanced modelling steps.

It should be clear now how exploratory analysis done by PCA allows to obtain a large amount of information, both expected (different contents of arginine, sucrose, and water) and unexpected (the operator effect). Together with the information about the samples exhibiting different features, the most influent spectral signals and bands were identified, thanks to the information represented by the loadings. This also allows focusing, if it makes sense, on specific spectral areas, both for a deeper exploratory analysis and for considering more refined regression and classification models. Similar representations of how the original variables influence the models and their performances will also be present in the next sections: the logic behind the loadings' interpretation can be easily transferred to the upcoming sections.

7. Regression analysis

7.1. Why do we need regression analysis?

In the pharmaceutical field, the techniques used for the quantitative analysis of CQAs are often expensive, time-consuming and, in some cases, even destructive. When the attributes to measure are quantitative, regression analysis is needed. In general, an analytical technique is used to acquire data about the product, and then the data are processed to estimate the value of the CQA of interest, as this is usually not directly measurable, or the measurement with other techniques is expensive or time-consuming. In the perspective of the PAT approach, Partial Least Squares regression (PLS) is often used to model spectroscopic data to predict quantitatively the parameters of interest, allowing for a reduction of experimental effort and time. Once the regression model has been developed, many advantages can be obtained:

- Extract information from large and complex datasets, such as the value of CQAs;
- Extract information (chemical, physical, quality parameters) from data in real-time;
- Make decisions on processes;
- Increase the process knowledge and speed up the development phase;
- Replace expensive and time-consuming analytical methods, leading to speed-up the testing phase of CQAs.

Of course, several critical factors must be taken into consideration to ensure successful implementation and compliance:

- Developing robust models is essential for accurately interpreting data and making informed decisions during the manufacturing processes. This involves not only selecting the appropriate variables, but also ensuring that the models are representative of the underlying processes.
- It is essential to validate the models to confirm their predictive capabilities and reliability. This ensures that the models can consistently produce accurate results across different batches and conditions, which is crucial for maintaining product quality. The models also need

maintenance over time, which is generally done by performing scheduled validation measures, but also when the PAT tools signal possible deviations from the normal operative conditions.
- As pharmaceutical companies operate in a highly regulated environment, it is vital that any PAT and PLS methods comply with regulatory standards. Engaging with regulatory bodies early in the development process can facilitate smoother approvals and ensure that the methodologies align with industry guidelines.
- When transferring methods between laboratories or production sites, it is essential to ensure that the models maintain their accuracy and reliability. This requires thorough documentation and training to ensure consistency in application across different settings.

7.2. What is Partial Least Squares (PLS) regression?

Partial Least Squares regression (PLS [34]) aims at finding a relationship between a data matrix **X**, (in our case the NIR spectra) and a response vector **Y** (e.g., residual moisture, sucrose or arginine content). This technique is based on a decomposition of the matrix **X** and the response vector **Y** in a new space described by the so-called latent variables (LVs), which are conceptually similar to the principal components in PCA, in the sense that in both cases they determine the model's "dimensions". However, while PCA deals with describing the information contained in a data matrix **X** alone, PLS looks for the information contained in **X** that better correlates with the response vector **Y** [34]. Speaking in terms of spectral data, PLS models the signals and their patterns of correlations to find which pieces of information could be correlated with the response vector **Y**. PLS is a supervised approach since the definition of the model also involves the use of additional information (the response vector **Y**), so the learning phase is "supervised" by the reference information of **Y**.

In our case study the water content could easily be used as a response vector **Y**. However, since the water content calibration for these data was extensively addressed in previous works (as cited before [13,14]), in this tutorial two further chemical components were modelled by PLS, to provide to the reader with additional and different examples. One model was built to predict the percentage of arginine and another one to predict the percentage of sucrose.

Both models serve as the "calibration step" in regression, enabling the acquisition of crucial yet constrained information regarding the predictive capability of the model. A complete and validated model would also require splitting the data into a calibration and a test set, or the production of a new external test set. The test set is a dataset such as the one used for calibration, but it must not be included or used otherwise for model computation: its purpose is to test the model with "fresh" information that the model has not encountered before. There are several methods for splitting data in calibration and test sets, but for this tutorial all data were used for the calibration step, therefore, no test set was defined. Please refer to the works by Amigo [29] and Westad and Marini [35] for more details on model validation strategies and test set selection.

For both models, the smoothed data ("NIR_data_sm31") were loaded into the regression toolbox ("select_toolbox ("regr")") and SNV and mean center preprocessings were applied from within the toolbox. The number of LVs was determined using the built-in "Optimal LV" function of the toolbox: the resulting plot for choosing how many LVs to model is reported in Fig. 13a and 13a for arginine and sucrose, respectively. The resulting models' dimensions and performances are reported in Table 4 and discussed in the following section.

7.3. PLS regression results

Different aspects of a PLS regression model must be inspected for correctly interpreting it. The two models (% of arginine and % of sucrose) will be presented separately, but the same inspection approach will be used. Just like any regression model (from simple linear to multivariate regression), the prediction plot is very important, and it is

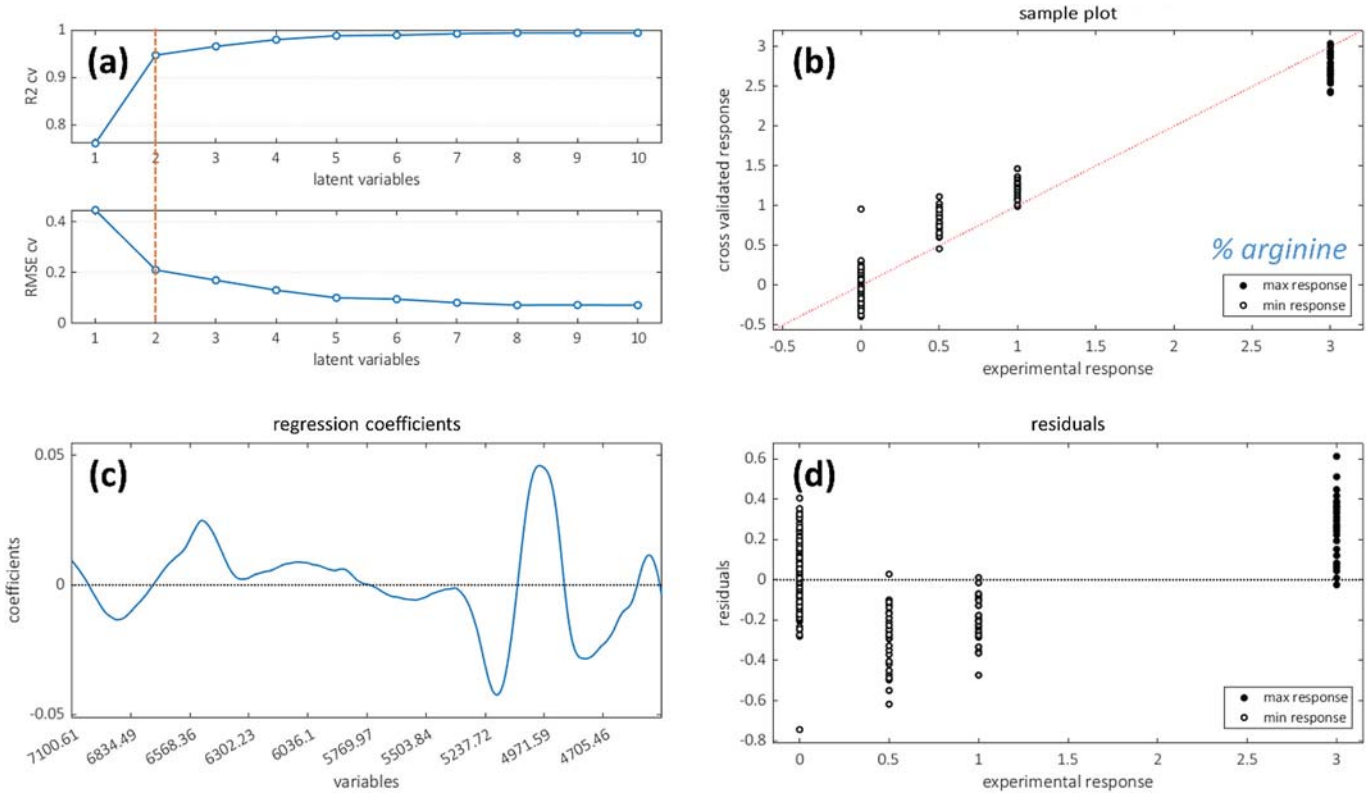


Fig. 13. Arginine PLS regression results: (a) the cross-validation R2 and RMSE results spanning 10 LVs, (b) the cross-validated computed response values for arginine, (c) the regression coefficients plot and (d) the residuals of the response prediction.

Table 4
Regression performance measures of the PLS models of arginine and sucrose.

Model	LVs	R^2	RMSE	R^2_{CV}	$RMSE_{CV}$
% of arginine	2	0.949	0.2062	0.947	0.2103
% of sucrose	5	0.874	0.8405	0.865	0.8708

generally the starting point for evaluating any models. Then, the model's performances (Table 4) must be considered, since the R^2 and the root mean squared error (RMSE) values provide information about how well the data are modelled and about the error associated with the prediction of the response.

Starting from the model of arginine, the prediction plot of Fig. 13b provides for each sample the predicted value of the response under examination. The samples are coloured according to the content of arginine, for ease of interpretation. From the prediction plot and the regression performances (high R^2 and low RMSE) we can state that the model is performing well. However, even if the model generally works well, we can also spot that at lower concentrations of arginine the model slightly overestimates the prediction, while it slightly underestimates it at higher concentrations. This is also confirmed by the residuals plot of Fig. 13d, which is complementary to the information provided by Fig. 13b. Finally, by inspecting the regression coefficients it is possible to interpret which variables are the most influent on the model, i.e., for predicting the response. For this model, the regression coefficients are reported in Figure 13c, and the characteristic absorption band of arginine (also found in PCA, Fig. 12c) appears evident at 4900 cm^{-1}, together with some other minor signals across the spectral width.

Regarding the PLS model of sucrose a similar situation is found, even if the model shows slightly worse performances. The predictions of the different concentrations of sucrose (Fig. 14b) appear more dispersed than what was found with the arginine model (described in Fig. 14b – remember that these are two different PLS models). No strong under/overestimation effects seem to be present across the concentration range of sucrose, as confirmed also by the residuals plot of Fig. 14d. However, the dispersion of the predicted values at the different % of sucrose is larger than the arginine model, and for this reason a lower R^2 and a higher RMSE were obtained in this case. It is important to consider that modelling a response vector that is "quantized" is somehow risky, as it is generally assumed that the precise content of sucrose is either 0, 3, 6 or 9 %, with no variability in between. The model's output is however not constrained to yield whole numbers, so the estimated responses will likely be dispersed around the "quantized steps" that were used as an input. Of course, this is also valid for evaluating the arginine model.

The interpretation of the regression coefficients is in this case more difficult, as many signals are contributing to the sucrose content prediction, with the strange influence of the most extreme signals on the sides of the recorded spectral range. For this reason, further investigation and more detailed signal assignment might be needed to proceed to the possible subsequent steps of model deployment for process control. If predicting the content of arginine or sucrose was the practical aim, the next step would be to quantify the actual content of these substances, and then build new PLS models: more "precise" variability in the response vector generally leads to better models, especially considering that the response could be linked to specific spectral signals. This is very clear in the arginine case (Fig. 13c), while in the sucrose case it is more problematic (Fig. 14c).

It is important to notice that the same dataset that was explored via PCA is used here for predicting the content of arginine (Fig. 13) and sucrose (Fig. 14): the multivariate approach allows for fully exploiting the information content of the spectral dataset which is often a mixture of signals that might correlate with different properties of the objects under examination. Following this remark, it becomes clear that in both PLS models the sucrose-arginine mixtures are considered together, without evident effects on the models' performances (Table 4). This is

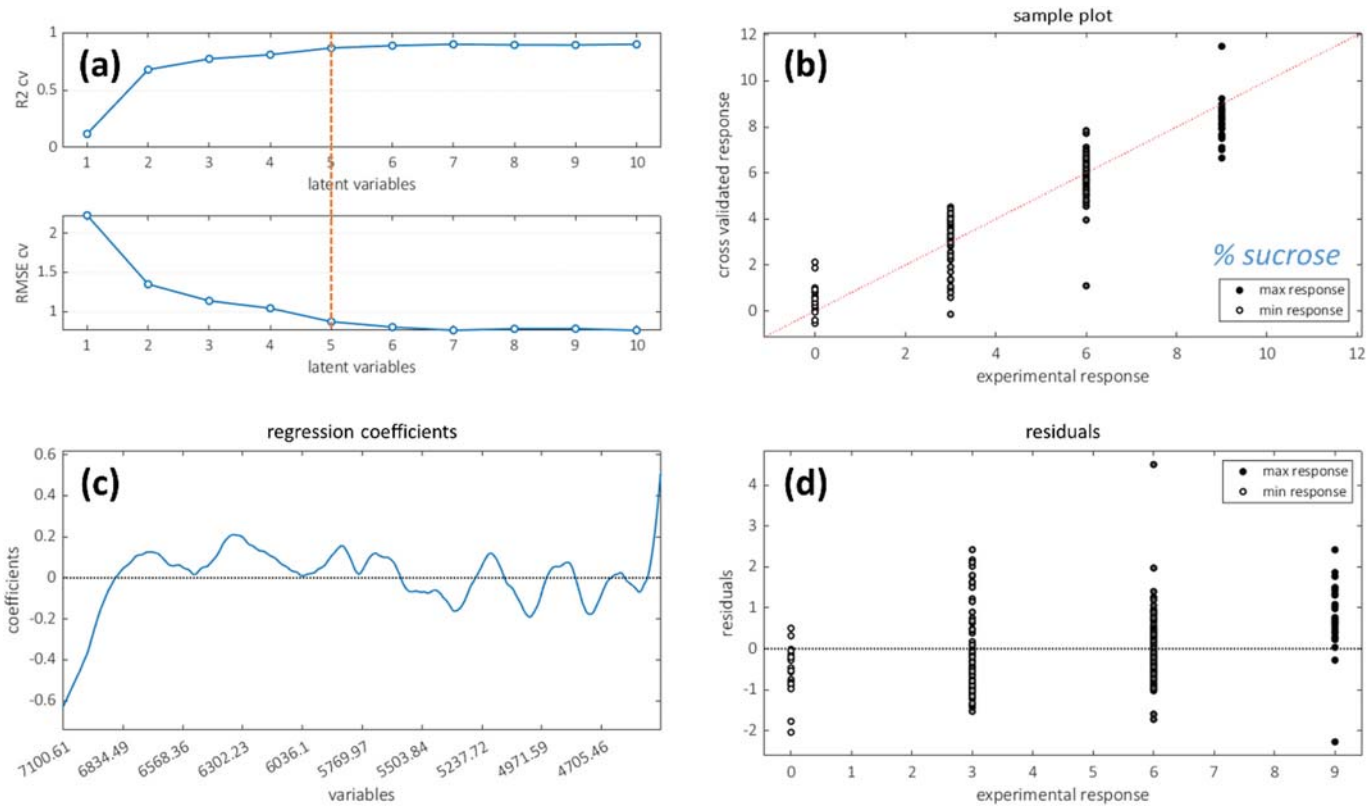


Fig. 14. Sucrose PLS regression results: (a) the cross-validation R2 and RMSE results spanning 10 LVs, (b) the cross-validated computed response values for arginine, (c) the regression coefficients plot and (d) the residuals of the response prediction.

mostly because of how PLS works, as the algorithm looks for specific pieces of information in the spectral dataset that best correlate with the modelled response, so multiple responses could be calibrated from the same set of data.

8. Classification analysis

8.1. Why do we need classification analysis?

Classification analysis is needed when the analyzed samples must be categorized into predefined groups or classes, based on certain characteristics. Specifically, classification models can be built to detect out-of-specification samples with respect to a threshold defined according to normal operative conditions. This type of modelling is helpful also for statistical quality control applications. As an example, a classification model could be useful to detect in-line and out-of-specification samples (as a support to the procedural visual inspection activities) to exclude samples characterized by a water content higher than the specification established by the company.

In this tutorial, the classification example concerns the ability of discriminating between samples containing arginine or not. A PLS-DA model with 2 components was built to distinguish between these two classes. Even if the case at hand might result rather trivial, the procedure to build and inspect the model can be easily adapted for different cases.

8.2. What is Partial Least Squares-Discriminant Analysis (PLS-DA)?

Partial Least Squares-Discriminant Analysis (PLS-DA [36]) is a particular use of PLS, as this regression method can be also used for determining class membership. As discussed in Section 7.2, PLS needs a response vector to perform the regression computations, and in this case the response would be the class information. To do so, the class information is converted into a "dummy matrix" **Y**, a binary matrix with as many rows as the samples and as many columns as the numbers of classes, minus one. For each sample, a "1" is placed in the column representing the class it belongs to, while all other positions contain zeros. Keep in mind that almost any toolbox for multivariate modelling would not require inputting the dummy matrix, since it would generate it directly from the class information vector. The PLS algorithm looks for the information contained in **X** that better correlates with the dummy matrix **Y**, thus providing an output that replicates the class belonging. Then, a discriminant analysis (DA) step is applied to the PLS output. The calibrated model can then be used for assigning new unknown samples to their most probable class of belonging, automatically.

To interpret a PLS-DA model it is important to inspect the classification plot (Fig. 15a), but also the classification performances (Tables 5

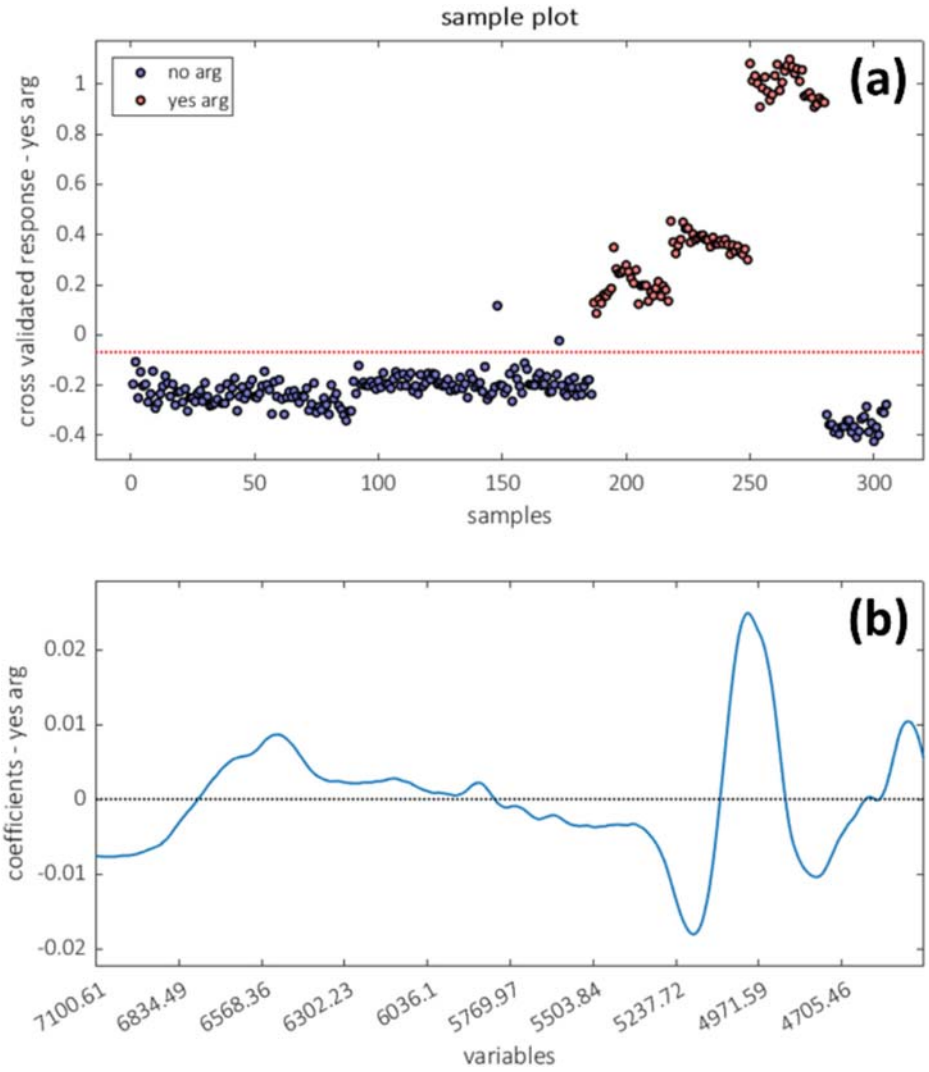


Fig. 15. Arginine presence PLS-DA results: in (a) the cross-validated classification plot from the point of view of the presence of arginine ("yes arg"), in (b) the regression coefficients.

Table 5
Classification performances in cross-validation of the PLS-DA model to predict the presence of arginine (two classes, no arginine/yes arginine).

2 LVs	sensitivity	specificity	precision
no arginine (nA)	0.99	1.00	1.00
yes arginine (A)	1.00	0.99	0.98

Table 6
Confusion Matrices of the PLS-DA model to predict the presence of arginine.

		Calibration		Cross-validation	
		pred. class		pred. class	
		nA	A	nA	A
actual class	nA	209	2	209	2
	A	0	94	0	94

and 6). Many classification measures can be computed and interpreted [37], but in this tutorial we will stick to the three directly provided by the toolbox:

- *Sensitivity*: it is the model ability to correctly recognize samples belonging to a specific class; this measure is used to assess the false positives rate.
- *Specificity*: it is the ability of the model to reject the samples of all the other classes but the considered one; this measure is used to assess the false negatives rate.
- *Precision*: it represents the capability of a model not to include samples of other classes in the considered class.

For more detailed definitions and explanations on the classification measures, please refer to Ballabio et al. [37].

In this tutorial PLS-DA will be used to predict the presence of arginine using the information contained in the spectra, i.e., signals specific for arginine or somehow correlated to its presence. The concept is similar to what was done in PLS for predicting the amount of arginine, so it can be expected that the classification analysis will work based on the same arginine signals that were identified in the previous sections (with both PCA and PLS). Also in this case, we are going to build a model and interpret its results limiting our analysis to the calibration step: a proper validation of the model would require splitting the data into a calibration and a test set, or the acquisition of an external test set. The strategies for obtaining a test set by splitting the initial dataset were described in Section 7.2 and still hold valid in this case.

Regarding the tutorial's example, the class information ("class_arg" in MATLAB) is going to be a simple vector containing "yes arg" for samples containing arginine, while all the others are marked with "no arg". The dummy matrix will be composed by the toolbox, but with only two classes we can imagine it as a simple vector of ones ("yes arg") and zeros ("no arg").

8.3. PLS-DA classification results

Table 5 summarizes the performances of the 2-components PLS-DA model to predict the presence of arginine ("yes arg"). Table 6 reports the confusion matrices. Fig. 15a reports the prediction plot of the model, from the point of view of the classification of samples containing arginine: since this is a two-class problem, the same plot from the point of view of predicting the samples that do not contain arginine would look exactly the same, but vertically mirrored. The plot has a horizontal dashed red line separating the two classes: this line is the result of applying the discriminant analysis to the PLS output, and it is computed so that the classification errors are minimized, based on a Bayesian criterion. Only two samples appear to lie on the wrong side, and they both belong to the "no arg" class: they are misclassified, i.e., they were mistakenly assigned to a class they do not belong to. This is the reason why the classification performances of Table 5 of the model are not perfect, even if very high. The number of misclassified samples can be deduced by looking at the prediction plot (Fig. 15a) or more easily by inspecting the confusion matrix (Table 6). The confusion matrix provides the details of the model's errors, since it clearly represents how many samples were correctly classified (the diagonal of the matrix) and how many were classified as belonging to other classes (the off-diagonal elements). Coherently with Fig. 15a, two samples belonging to the "no arg" class were classified as if they contained arginine.

Since also this model is multivariate (it is indeed based on PLS regression), it is possible to interpret the model's functioning in relation to the original variables, so in terms of spectral signals. By looking at the coefficients plot in Fig. 15b, it can be noticed that the signals are basically the same described by the PLS regression models, confirming that the water and the arginine-specific signals are the most important ones. This is not unexpected and can be used as further confirmation that the model not only is working well, but it is also modelling actual signals arising from arginine, the target compound of the classification problem. Being able to explain why a model works is an important validation step, and in this case the possibility of interpreting the regression coefficients as linked to the signals of the compound defining our classes gives a "chemical meaning" to the model, thus contributing to its validation.

9. Conclusions

In this tutorial we illustrated the potential of using a multivariate approach in the pharmaceutical field, especially from the point of view of the information that can be extracted from a set of NIR spectra. We believe that the real case dataset provided many insights for all data analysis steps, from the exploratory to the more applicative/predictive regression and classification ones. All the fundamental steps needed to carry out a proper data analysis were exposed and explained with practical hints and suggestions, which we tried to design as straightforward as possible.

We expect that after reading and hopefully practicing this tutorial (with the provided data or even better with the reader's own data) it became clear that also a lot of different types of data can be explored using these tools, to understand what has been measured, but also to find out possible problems or unexpected effects, such as the influence of the operator or the session of measurements. We invite the readers to try using the material provided in this tutorial directly on their own data, also with little changes to the lines in the Codes boxes: our aim was to provide a thorough yet robust and clear data analysis workflow, to help anyone dig into their own data.

CRediT authorship contribution statement

Ambra Massei: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Nicola Cavallini:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Francesco Savorani:** Writing – review & editing, Validation, Supervision, Methodology. **Nunzia Falco:** Writing – review & editing, Supervision, Resources, Funding acquisition. **Davide Fissore:** Writing – review & editing, Validation, Supervision, Project administration, Methodology, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Davide Ballabio and the Milano Chemometrics and QSAR Research Group are gratefully acknowledged for allowing the inclusion of their PCA, Regression, and Classification toolboxes in the Codes Package provided with this tutorial. The link to the Download page of the research group is reported in Section 3.2 “Software specifications”. The authors also thankfully acknowledge Merck Serono S. p.A for the contribution and financial support. The support in the experimental activities of Adamo Sulpizi (Researcher), Caterina Sapienza (Associate Researcher), Daniele Mari (Senior Laboratory Technician), and Michele Dimattia (Associate Researcher) from the Global Parenteral Development Department, Merck Serono S.p.A., Guidonia Montecelio, Roma, is also gratefully appreciated.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.chemolab.2024.105291.

Data availability

The data andthe full code package can be downloaded from: https://github.com/NiKemometrics/NIR_chemometrics_tutorial.

References

[1] Y. Wang, F.J. Muzzio, B.J. Glasser, Using multivariate analysis for pharmaceutical drug product development. https://doi.org/10.7282/T39Z9770, 2016.
[2] E.K. Read, J.T. Park, R.B. Shah, B.S. Riley, K.A. Brorson, A.S. Rathore, Process analytical technology (PAT) for biopharmaceutical products: Part I. concepts and applications, Biotechnol. Bioeng. 105 (2010) 276–284, https://doi.org/10.1002/BIT.22528.
[3] A.P. Ferreira, M. Tobyn, Multivariate analysis in the pharmaceutical industry: enabling process understanding and improvement in the PAT and QbD era, Pharmaceut. Dev. Technol. 20 (2015) 513–527, https://doi.org/10.3109/10837450.2014.898656.
[4] Ph Eur. 11.1, 5.21 Chemometric Methods Applied to Analytical Data, 4545-4569 (04/2023), (n.d.).
[5] S. Wold, Chemometrics; what do we mean with it, and what do we want from it? Chemometr. Intell. Lab. Syst. 30 (1995) 109–115, https://doi.org/10.1016/0169-7439(95)00042-9.
[6] G. Gerzon, Y. Sheng, M. Kirkitadze, Process Analytical Technologies – advances in bioprocess integration and future perspectives, J. Pharm. Biomed. Anal. 207 (2022) 114379, https://doi.org/10.1016/J.JPBA.2021.114379.
[7] J. Zhao, G. Tian, Y. Qiu, H. Qu, Rapid quantification of active pharmaceutical ingredient for sugar-free Yangwei granules in commercial production using FT-NIR spectroscopy based on machine learning techniques, Spectrochim. Acta Part A Mol. Biomol. Spectrosc. 245 (2021) 118878, https://doi.org/10.1016/J.SAA.2020.118878.
[8] Z. Shi, N. Zaborenko, D.E. Reed, Latent variables-based process modeling of a continuous hydrogenation reaction in API synthesis of small molecules, J. Pharm. Innov. 8 (2013) 1–10.
[9] E. Tomba, M. De Martin, P. Facco, J. Robertson, S. Zomer, F. Bezzo, M. Barolo, General procedure to aid the development of continuous pharmaceutical processes using multivariate statistical modeling – an industrial case study, Int. J. Pharm. 444 (2013) 25–39, https://doi.org/10.1016/J.IJPHARM.2013.01.018.
[10] R. Leardi, Experimental design in chemistry: a tutorial, Anal. Chim. Acta 652 (2009) 161–172, https://doi.org/10.1016/J.ACA.2009.06.015.
[11] V. Lourenço, D. Lochmann, G. Reich, J.C. Menezes, T. Herdling, J. Schewitz, A quality by design study applied to an industrial pharmaceutical fluid bed granulation, Eur. J. Pharm. Biopharm. 81 (2012) 438–447, https://doi.org/10.1016/J.EJPB.2012.03.003.
[12] M. Clavaud, Y. Roggo, K. Dégardin, P.Y. Sacré, P. Hubert, E. Ziemons, Global regression model for moisture content determination using near-infrared spectroscopy, Eur. J. Pharm. Biopharm. 119 (2017) 343–352, https://doi.org/10.1016/J.EJPB.2017.07.007.
[13] S. Bobba, N. Zinfollino, D. Fissore, Application of Near-Infrared Spectroscopy to statistical control in freeze-drying processes, Eur. J. Pharm. Biopharm. 168 (2021) 26–37, https://doi.org/10.1016/J.EJPB.2021.08.009.
[14] A. Massei, N. Falco, D. Fissore, Use of machine learning tools and NIR spectra to estimate residual moisture in freeze-dried products, Spectrochim. Acta Part A Mol. Biomol. Spectrosc. 293 (2023) 122485, https://doi.org/10.1016/J.SAA.2023.122485.
[15] H. Grohganz, D. Gildemyn, E. Skibsted, J.M. Flink, J. Rantanen, Rapid solid-state analysis of freeze-dried protein formulations using NIR and Raman spectroscopies, J. Pharmaceut. Sci. 100 (2011) 2871–2875, https://doi.org/10.1002/JPS.22490.
[16] C. Ravn, E. Skibsted, R. Bro, Near-infrared chemical imaging (NIR-CI) on pharmaceutical solid dosage forms—comparing common calibration approaches, J. Pharm. Biomed. Anal. 48 (2008) 554–561, https://doi.org/10.1016/J.JPBA.2008.07.019.
[17] D. Fissore, Freeze drying of pharmaceuticals, in: Encycl. Pharm. Sci. Technol., fourth ed., CRC Press, 2013, pp. 1723–1737, https://doi.org/10.1081/E-EPT4-120050278.
[18] I. Oddone, R. Pisano, R. Bullich, P. Stewart, Vacuum-induced nucleation as a method for freeze-drying cycle optimization, Ind. Eng. Chem. Res. 53 (2014) 18236–18244.
[19] T. De Beer, A. Burggraeve, M. Fonteyne, L. Saerens, J.P. Remon, C. Vervaet, Near infrared and Raman spectroscopy for the in-process monitoring of pharmaceutical production processes, Int. J. Pharm. 417 (2011) 32–47, https://doi.org/10.1016/J.IJPHARM.2010.12.012.
[20] G. Reich, Near-infrared spectroscopy and imaging: basic principles and pharmaceutical applications, Adv. Drug Deliv. Rev. 57 (2005) 1109–1143, https://doi.org/10.1016/J.ADDR.2005.01.020.
[21] M. Blanco, J. Coello, H. Iturriaga, S. Maspoch, C. De La Pezuela, Near-infrared spectroscopy in the pharmaceutical industry. Critical Review, Analyst. 123 (1998) 135R–150R, https://doi.org/10.1039/A802531B.
[22] E.W. Ciurczak, B. Igne, Pharmaceutical and Medical Applications of Near-Infrared Spectroscopy, second ed., CRC Press, 2019. https://www.routledge.com/Pharmaceutical-and-Medical-Applications-of-Near-Infrared-Spectroscopy/Ciurczak-Igne/p/book/9780367377977. (Accessed 19 September 2023).
[23] M. Jamrógiewicz, Application of the near-infrared spectroscopy in the pharmaceutical technology, J. Pharm. Biomed. Anal. 66 (2012) 1–10, https://doi.org/10.1016/J.JPBA.2012.03.009.
[24] J. Luypaert, D.L. Massart, Y. Vander Heyden, Near-infrared spectroscopy applications in pharmaceutical analysis, Talanta 72 (2007) 865–883, https://doi.org/10.1016/J.TALANTA.2006.12.023.
[25] M.A. Lakeh, S.K. Karimvand, M.R. Khoshayand, H. Abdollahi, Analysis of residual moisture in a freeze-dried sample drug using a multivariate fitting regression model, Microchem. J. 154 (2020) 104516, https://doi.org/10.1016/J.MICROC.2019.104516.
[26] G. Clua-Palau, E. Jo, S. Nikolic, J. Coello, S. Maspoch, Finding a reliable limit of detection in the NIR determination of residual moisture in a freeze-dried drug product, J. Pharm. Biomed. Anal. 183 (2020) 113163, https://doi.org/10.1016/J.JPBA.2020.113163.
[27] M.W.J. Derksen, P.J.M. Van De Oetelaar, F.A. Maris, The use of near-infrared spectroscopy in the efficient prediction of a specification for the residual moisture content of a freeze-dried product, J. Pharm. Biomed. Anal. 17 (1998) 473–480, https://doi.org/10.1016/S0731-7085(97)00216-1.
[28] A. Giraudo, S. Grassi, F. Savorani, G. Gavoci, E. Casiraghi, F. Geobaldo, Determination of the geographical origin of green coffee beans using NIR spectroscopy and multivariate data analysis, Food Control 99 (2019) 137–145, https://doi.org/10.1016/j.foodcont.2018.12.033.
[29] J.M. Amigo, Data mining, machine learning, deep learning, chemometrics: definitions, common points and trends (Spoiler Alert: VALIDATE your models!), Brazilian J. Anal. Chem. 8 (2021) 22–38, https://doi.org/10.30744/brjac.2179-3425.AR-38-2021.
[30] D.-W. Sun, Å. Rinnan, L. Nørgaard, F. van den Berg, J. Thygesen, R. Bro, S. B. Engelsen, Data pre-processing, in: Da-Wen Sun (Ed.), Infrared Spectrosc. Food Qual. Anal. Control, Elsevier, 2009, pp. 29–50, https://doi.org/10.1016/B978-0-12-374136-3.00002-X.
[31] M. Li Vigni, C. Durante, M. Cocchi, Exploratory data analysis, in: Data Handl. Sci. Technol., Elsevier, 2013, pp. 55–126, https://doi.org/10.1016/B978-0-444-59528-7.00003-X.
[32] R. Bro, A.K. Smilde, Principal component analysis, Anal. Methods 6 (2014) 2812–2831, https://doi.org/10.1039/C3AY41907J.
[33] D. Colucci, J.M. Prats-Montalbán, D. Fissore, A. Ferrer, Application of multivariate image analysis for on-line monitoring of a freeze-drying process for pharmaceutical products in vials, Chemometr. Intell. Lab. Syst. 187 (2019) 19–27, https://doi.org/10.1016/J.CHEMOLAB.2019.02.004.
[34] P. Geladi, B.R. Kowalski, Partial least-squares regression: a tutorial, Anal. Chim. Acta 185 (1986) 1–17, https://doi.org/10.1016/0003-2670(86)80028-9.
[35] F. Westad, F. Marini, Validation of chemometric models – a tutorial, Anal. Chim. Acta 893 (2015) 14–24, https://doi.org/10.1016/J.ACA.2015.06.056.
[36] D. Ballabio, V. Consonni, Classification tools in chemistry. Part 1: linear models. PLS-DA, Anal. Methods 5 (2013) 3790–3798, https://doi.org/10.1039/c3ay40582f.
[37] D. Ballabio, F. Grisoni, R. Todeschini, Multivariate comparison of classification performance measures, Chemometr. Intell. Lab. Syst. 174 (2018) 33–44, https://doi.org/10.1016/j.chemolab.2017.12.004.