Multivariate classification techniques for vibrational spectroscopy in biological samples

Camilo L. M. Morais,[1*] Kássio M. G. Lima,[2] Maneesh Singh,[3] Francis L. Martin[1*]

[1]*School of Pharmacy and Biomedical Sciences, University of Central Lancashire, Preston, United Kingdom*

[2]*Institute of Chemistry, Biological Chemistry and Chemometrics, Federal University of Rio Grande do Norte, Natal, Brazil*

[3]*Biocel Analytics, Hull, United Kingdom*

* cdlmedeiros-de-morai@uclan.ac.uk / flmartin@uclan.ac.uk

Abstract

The use of vibrational spectroscopy techniques, such as Fourier-transform infrared (FTIR) and Raman spectroscopy, has been a successful method to study the interaction of light with biological materials and facilitate novel cell biology analysis. Disease screening and diagnosis, microbiological studies, forensic and environmental investigations make use of spectrochemical analysis very attractive due to its low cost, minimal sample preparation, non-destructive nature and substantially accurate results. However, there is now an urgent need for multivariate classification protocols allowing one to analyse biological-derived spectrochemical data in order to obtain accurate and reliable results. This is stimulated by the fact that applications of deep-learning algorithms of complex datasets are being increasingly recognized as critical towards extracting important information and visualizing it in a readily interpretable form. Hereby, we have constructed a protocol for multivariate classification analysis of vibrational spectroscopy data [FTIR, Raman and near-infrared (NIR)] highlighting a series of critical steps, such as pre-processing, data selection, feature extraction, classification and model validation. This is an essential aspect towards the construction of a practical spectrochemical analysis model for biological analysis in real-world applications, where fast, accurate and reliable classification models are fundamental.

Introduction

Vibrational spectroscopy comprises techniques related to electronic changes in the internal vibrational energy levels of molecules. Biomolecules that contain chemical bonds that vibrate generating a change in the dipole moment as a result of the transition are IR active[1,2]. Infrared (IR) and Raman spectroscopy are the main spectroscopic techniques used to assess vibrational molecular modes, where the first is based on molecular dipole changes and the latter on molecular polarizability changes. IR spectroscopy is dived into near-IR (NIR), mid-IR (MIR) and far-IR (FIR) spectroscopy depending on the incident light frequency. MIR is the main technique used to analyse biological materials since it covers fundamental vibrational modes of important biomolecules. Vibrational spectroscopy can provide rapid, label-free, and objective analysis for the clinical domain. The fingerprint region, between 1800-900 cm^{-1}, include important absorptions of lipids (C=O symmetric stretching at ~1750 cm^{-1}, CH_2 bending at ~1470 cm^{-1}), proteins (Amide I at ~1650 cm^{-1}, Amide II at ~1550 cm^{-1}, Amide III at ~1260 cm^{-1}), carbohydrates (CO-O-C symmetric stretching at ~1155 cm^{-1}), nucleic acid (asymmetric phosphate stretching at ~1225 cm^{-1}, symmetric phosphate stretching at ~1080 cm^{-1}), glycogen (C-O stretching at ~1030 cm^{-1}), and protein phosphorylation (~970 cm^{-1})[3-5]. The high-region, between 3700–2800 cm^{-1}, can also be used for analysis, where information of water (-OH stretching at ~3275 cm^{-1}), protein (symmetric -NH stretching at ~3132 cm^{-1}), fatty acids and lipids (=C-H asymmetric stretching at 3005 cm^{-1}, CH_3 asymmetric stretching at ~2970 cm^{-1}, CH_2 asymmetric stretching at ~2942 cm^{-1}, CH_2 symmetric stretching at ~2855 cm^{-1}) can be obtained[6]. NIR spectroscopy can also be applied as a biospectroscopy tool. This technique is mainly composed of MIR overtones, hence, the signal is very complex containing many overlapping features. Therefore, biomarkers identification using NIR is harder and more ambiguous, although this technique is a powerful tool for quantification and classification applications[7]. Raman spectroscopy is based on an inelastic scattering effect. Most of the photons absorbed by a molecule suffers elastic scattering; only a small portion of them (<1%) suffer inelastic scattering, where the released radiation has lower or higher energy than the initial incoming absorbed radiation[2]. Inelastic scattering can be Stokes (photons with lower energy are emitted) or anti-Stokes (photons with higher energy are emitted), and both correspond to the Raman signal. Due to the small probability of molecules in an initial high energy state at room temperature, the anti-Stokes signal is not so strong, and the Stokes signal is usually recorded as the final Raman spectrum.

Since both IR and Raman are non-destructive and sensitive techniques with a relative low-cost, passive of automation, and translatable to portable devices, their use to investigate biological samples are of great interest[3,8]. Biofluids offer an ideal diagnostic medium due to their ease and low cost of collection and daily use in clinical biology. Applications using IR and Raman spectroscopy to investigate biological samples for food[9-14], plant[15-18], microorganism[19-28] and clinical analysis[29-33] are many. These previously mentioned advantages associated with the application of multivariate statistical methods of data analysis make these techniques every day more attractive for routine application. Previous protocols for IR[3] and Raman[8] spectroscopy to analyse biological samples have been already published, but there is still a lack of good practical procedures on how to analyse the acquired data for classification applications where, for example, the spectral data can be used to determine if a given sample is healthy or disease, or if it belongs or not to a given group. This is critical since the results obtained by these studies are directly dependent on the data analysis methodology being used.

Bio-spectral data analysis is a science that requires multidisciplinary knowledge, where to obtain reliable and chemically-meaningful results, the application of chemometric techniques is

fundamental. Chemometrics is defined as "the science of relating measurements made on a chemical system or process to the state of the system *via* application of mathematical or statistical methods"[34]. The use of statistical methods to solve chemical problems trace back centuries, though in 1949, the first report of least squares regression, design of experiments and analysis of variance (ANOVA) appear in analytical chemistry, by Mandel[35]. In the early 1960's, multivariate methods were first reported in a modern physical-chemistry context to determine the number of components in spectral mixtures as theoretical chemistry approaches[36,37]. Practical implementation into experimental analysis started with the influence of statistical approaches by Pearson and Fisher, whose published work in multivariate analysis in the 1920's and 1930's, acted as inspiration to apply ideas such as principal component analysis (PCA), factor analysis and discriminant modelling in a chemical context[38]. During late 1960's and the 1970's, advancements in computer power and availability, the development of artificial intelligence algorithms, and the work done by Bruce Kowalski in the US and Svante Wold in Sweden enabled the introduction of multivariate methods in analytical chemistry in a modern fashion, and the word "chemometrics" was defined[38].

Multivariate methods in a chemical context can be seen as an expansion of the Lambert-Beer's law in a multi-component approach, where the absorbance (spectral response) is a linear combination of concentration time coefficients[39]. For the notation, generally, bold uppercase characters (*e.g.*, **X**) represent matrices, bold lowercase characters (*e.g.*, **x**) represent vectors, and italic characters (*e.g.*, *n*) represent scalars. The concentrations are related to sample differences, thus being used to assess the real chemical concentration or to find similarities/dissimilarities between samples, while the coefficients represent the weight of each variable (*e.g.*, wavenumber) in the linear combination, hence, being used to find possible spectral markers. In classification applications, most of the algorithms employed to discriminate spectral data are a combination of feature extraction or feature selection methods followed by discriminant or class modelling techniques, which are mostly distance-based; a classical example is the partial least squares discriminant analysis (PLS-DA) algorithm[40].

There are several steps to process biospectroscopic data towards classification applications. Firstly, before analysing the data, one must think if the experiment was performed correctly, if the number of sample is representative to solve their problem, and if the data are bilinear, that is, if the product of spectrum times concentration is a constant. If design of experiments (DoE)[41] are required to acquire representative data, this should be performed. If the data are not bilinear, then non-linear data analysis approaches should be investigated. After data acquisition, the first step is to visualise the data. Anomalous spectral behaviours should be investigated and, depending on the application of interest, removed from the dataset. Outlier detection is a powerful tool to systematically investigate anomalous spectral profiles, though one must always visualise the data during this procedure. Then, pre-processing, data selection, model construction and validation are the essential steps to obtain reliable results. These steps are summarized in Figure 1. We must stress that these steps are iterative and entwined and the user can go back and change them until convergence to a good model is achieved. For this reason, red arrows going backwards are shown in Figure 1; this means that in order to optimise the model the user must test different data selection (including outlier detection), pre-processing and model construction techniques in an interconnected way since there is no single route to validation.

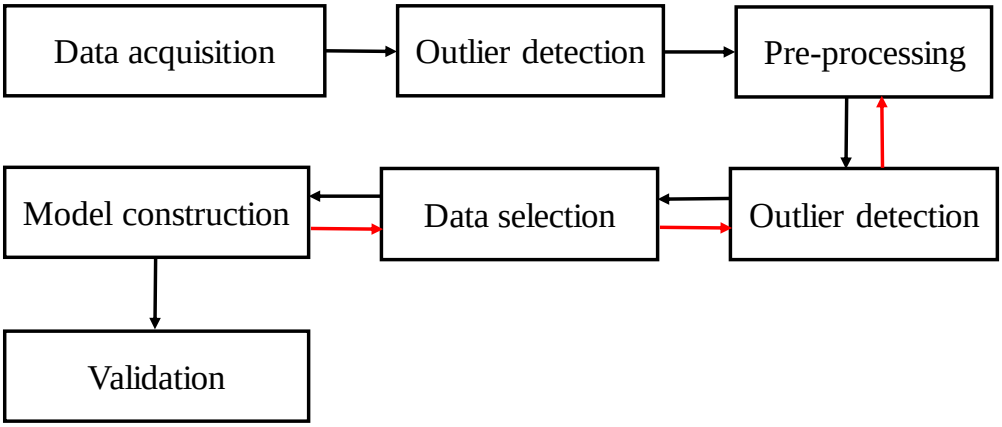

Figure 1. Spectral data analysis flowchart.

Experimental design

More important than the data analysis itself is the experimental setup used to acquire the spectral data. Previous protocols demonstrate all the materials and steps needed for spectral data acquisition of biological-derived samples using both IR and Raman spectroscopy[1,3,8,42]. Therefore, in this protocol, we will focus solely on the spectral data analysis aspect.

Minimum dataset requirements

Before carrying on with the experiments, the number of samples must be defined. For pilot studies, power tests are recommend, where a power of 80% can be used as the minimum number of samples for the dataset[43]. Normally, 5 to 25 point spectra are collected per sample[42], and 10 point spectra have been suggested in a previous protocol for ATR-FTIR[3]. By increasing the number of spectra replicates, the standard-deviation between measurements is reduced, since the standard-deviation is proportional to $1/\sqrt{n}$, where n is the number of spectra replicates. Extra caution should be taken when analysing heterogeneously distributed samples (*e.g.*, tissues), where spectra replicates should be acquired in a way that covers the sample surface as uniformly as possible. Sample replicates are also recommended. For precision estimation, at least six replicates at three levels should be performed[42]. When patient variability is being measured, *i.e.*, when the classification model is performed in a sample-basis, the spectral replicates per sample can be averaged so each resultant spectral response corresponds to a different patient. In this way, the chemometric model is modelled per patient rather than by spectral replicate. However, this requires a larger number of samples and might be difficult to be implemented in small pilot studies. In larger studies, especially before routine implementation, thousand of samples are necessary. This number is defined by the analyst experience and the classification rigour needed. The analyst while designing the experiment must think about confounding factors and the sources of variability that needs to be contemplated in the experiment. If needed, standardisation procedures should be performed to make sure that systematic variations due to environmental, instrumental or analyst changes do not affect the spectral response[42].

Also, the classes' sizes must be taken into consideration. Ideally, classes should have equal size; however, in real clinical scenarios it is unlikely this will occur. For example, for general screening applications, it is very common in clinical settings to have more healthy patients than disease; while, when investigating a specific type of disease, it is more likely that the patients being recruited contain the disease of interest whilst the control group is reduced. When both situations are present, the analyst must take extra care in the data analysis to avoid overfitting the model towards the biggest class size. Some solutions are the application of prior-probability terms based on the classes' size, the use of non-parametric methods, or by increasing the number of samples for each class to ensure that the calibration model covers enough sources of variation for each classes. As well, according to the central limit theorem (CLT), by increasing the number of samples the data will tend to a normal distribution, which will make parametric classification methods more efficient.

Before pre-processing, the data can be evaluated visually and through some statistical methods in order to identify anomalous behaviours or biased patterns. This is first performed by visual inspection (*e.g.*, plotting the data to identify anomalous spectral features), followed by Hotelling's T^2 *versus* Q residuals charts using only the mean-centred raw spectra. Principal component analysis (PCA) residuals can be explored to identify experimental bias, in which heteroscedastic distributions indicate biased experimental measurements, whereas homoscedastic distributions are associated with good sampling[39]. The signal-to-noise ratio (SNR) can be estimated by dividing the signal power ($\mathrm{P_{signal}}$) by the power of the noise ($\mathrm{P_{noise}}$), that is, $\mathrm{SNR} = \mathrm{P_{signal}}/\mathrm{P_{noise}} = \left(\mathrm{A_{signal}}/\mathrm{A_{noise}}\right)^2$, where A is the amplitude; or by the inverse of the coefficient of variation, when only non-negative variables are measured[42]. Collinearity can be evaluated by calculating the condition number, which shows how sensitive the result is to perturbations in the spectral data and to roundoff errors made during the solution process (this value is naturally elevated for spectral data, which indicates high collinearity)[42]. Visualising the data by plotting them throughout all the data analysis steps summarized in Figure 1 is essential. Prior scientific knowledge of the problem being addressed is also important. The data must be meaningful and the analyst has to make decisions based on how the spectral data look. All data analysis algorithms generate numbers based on the input data, thus if the data are not meaningful (*i.e.*, the signal of interest is absent) the model will generate untrue values. Thus, adequate instrumental techniques allied with good chemometric practices is fundamental.

Pre-processing

Pre-processing is applied to the spectral data in order to remove or reduce the contribution of signals which are not related to the analyte or target property, or to the sample discrimination (which depends on the chemical composition). Pre-processing of the raw data reduces chemically irrelevant variations with the goal of improving accuracy and precision of qualitative and quantitative analyses. The primary role of pre-processing is to transform the spectrum to the best fit condition and to ensure that optimum performance can be achieved in later steps. This process is essential to correct for physical interferences, such as light scattering due to different particle sizes, different sample thickness or different optical paths; and random instrumental noise. However, pre-processing techniques also carry the risk of generating correlations in the noise structure, which would impact negatively on the quality of the multivariate model; thus one should use pre-processing techniques with caution and not overuse them.

In biological applications, the first pre-processing usually consist of truncating the biofingerprint region: 1800-900 cm^{-1} for IR data[5], 2000 to 500 cm^{-1} for Raman[5], and 900 to 2600 nm for NIR[44]. This removes spectral artefacts such as water and CO_2 absorptions present in other parts of the IR spectrum, and additional baseline distortions that may be present in the spectrum[42]. The high-region associated mainly to lipids (3700 to 2800 cm^{-1}) can also been used for IR[6] and Raman[32], however this region is highly affected by water absorption in IR (free *v*(O-H) at 3600-3650 cm^{-1}; hydrogen-bonded *v*(O-H) at 3300-3400 cm^{-1})[45] and Raman (fully hydrogen-bonded *v*(O-H) at 3250 cm^{-1}; partly hydrogen-bonded *v*(O-H) at 3300-3630 cm^{-1})[46]. Usually, the model performance in the fingerprint region is better than in the high-region due to less water interference and the presence of more complex chemical features[6,47].

Figure 2 depicts the effect of each pre-processing for a given spectral dataset; and Figure 3 shows a flowchart to define which pre-processing technique to use after removal of substrate contributions. Pre-processing techniques should be used in the most parsimonious way[48]. The order in which the pre-processing steps are performed is fundamental; they must be performed in a logical order so that the next pre-processing step does not mask the signal of interest highlighted with the previous pre-processing[42]. Each pre-processing has its advantage, disadvantage and optimization step, which will be discussed hereafter.

Digital removal of substrate contributions. Sometimes substrate contributions originating from components such as glass or wax are present in the spectral data. These effects can be mitigated or eliminated through digital filters, such as digital de-waxing[49,50]. For example, the extended multiplicative signal correction (EMSC) algorithm has been reported to neutralise variability caused by paraffin signal and allow selection of unique spectral features related to the sample composition in vibrational spectroscopy[49,50]; independent component analysis (ICA) and non-negatively constrained least squares (NCLS) are also common methods of digital de-waxing for vibrational spectroscopy[49,51,52]. Glass contributions are reduced in the high-wavenumber region of the mid-IR spectrum, thus allowing spectral data analysis within the region between ~2500–3800 cm^{-1} [53]; and can be reduced in NIR spectroscopy by subtracting the glass spectrum from the sample spectrum and by working in the wavelength range of 1850–2150 nm[44].

Smoothing. Smoothing is made by spectral filters that remove random noise while preserving useful spectral information. The most used smoothing technique is the Savitzky-Golay (SG) algorithm[54]. It is based on a polynomial equation fitted in a least squares sense within a pre-defined interval of spectral points, where the central point from the interval is removed and used as a fitting criteria. This interval is then displaced to the next point of the spectrum and the fitting procedure is repeated. SG smoothing is excellent to remove large instrumental noise and can be applied to any type of vibrational spectroscopy technique. Its major disadvantage is that the polynomial order and the window size used in the polynomial fitting affect the result, so one must use an polynomial order similar to the spectral shape features (*e.g.*, 2nd order polynomial for vibrational spectroscopy data), and the window size must be an odd number not too small (which keeps the noise) nor too large (changing the spectral shape)[42]. In addition, moving-window mathematical pre-processing techniques introduce correlations in the noise structure, and this may complicate the use of chemometric models assuming that the noise is identically and independently distributed (iid)[55].

Light scattering correction. Light scattering is present when particles with different sizes, especially smaller than the spectral electromagnetic wavelength, is present on the material being analysed. This shifts the absorbance or spectral intensity in a systematic fashion, affecting the *y*-axis. Light scattering effects are also generated by different probe pressures when portable spectrometers are used to analyse solid samples, or by different lengths of optical path. Light scattering is very common

in NIR spectroscopy, and some techniques such as multiplicative scatter correction (MSC)[56] and standard normal variate (SNV)[57] can be used to correct this problem. MSC corrects light scattering (Mie scattering) maintaining the original spectral shape and the same spectral scale. As main disadvantage, it needs a reference spectrum representative of all measurements. Usually, this reference spectrum is not available, and then it is substituted by the average spectrum across all training samples[42]. SNV also corrects light scattering (Mie scattering) maintaining the original spectral shape with no need of a reference spectrum, but it creates an artificial absorbance scale with negative values since the data are centralized to zero in the *y*-scale[42]. Resonant Mie scattering is also a frequent issue in IR spectroscopy of biological materials[58], where a dispersion artefact occurs through a light scattering when there is simultaneous absorption. This can be often observed by a severe baseline distortion followed by a systematic shift in the *y*-axis[58,59]. Bassan et al.[58,59] have proposed a modified version of the EMSC algorithm to correct for resonant Mie scattering, named the RMieS-EMSC algorithm. Additionally, Mie scattering (elastic scattering) is also present in Raman spectroscopy[60], contributing to baseline distortions which are often mistakenly assigned to a fluorescence background. This can be also corrected by applying a modified version of the EMSC algorithm through the addition of polynomial extensions to the basic EMSC algorithm in order to correct for fluctuating baseline features[61].

Baseline correction. Baseline correction techniques remove background absorptions interferences. Baseline distortions are commonly present in all types of vibrational spectroscopy techniques. For NIR, the baseline distortions are mainly a result of light scattering, which can be corrected by MSC or SNV; however, for IR and Raman spectroscopy, this effect is more apparent, especially in the latter due to fluorescence interferences. There are several techniques of baseline correction, most of them already included in spectrometers' software, in which the main ones are the rubber-band baseline correction, polynomial baseline correction, asymmetric least squares (ALS), automatic weighted least squares (AWLS), and Whittaker filter. The baseline correction technique being chosen affect the final result, therefore, they must be kept consistent throughout all the data analysis, especially if new samples are added after the model is developed.

Spectral differentiation. First and second derivatives can be applied to the spectral data in order to correct both light scattering and baseline distortions. Also, these techniques highlight smaller spectral differences between the samples' spectra, which can be critical to find distinctive spectral features amongst complex samples. It can also be coupled to SG smoothing in a single routine, making these procedures computationally easier. However, derivatives have great disadvantages. Spectral differentiation is not indicated to correct for resonant Mie scattering since it does not correctly deal with these spectral distortions. The order of the derivative function must be chosen carefully to avoid increasing the noise level too much. In addition, derivatives using moving-window procedures also carry the same risk of introducing correlations in the noise structure discussed to smoothing techniques, which affects the use of chemometric models assuming that the noise is iid. Also, derivatives change the spectral scale (*y*-axis scale) to mathematical coefficients instead of absorbance, thus the spectral intensity of derivative bands cannot be used for direct correlation with chemical concentrations; and spectral markers (biomarkers) identification needs to be performed carefully, since derivatives shift the spectral band positions in $i \times d$ wavenumbers, where i is the derivative order and d is the data spacing resolution. Some software automatically correct this spectral shift by deleting the first i wavenumber position(s), thus matching the size of the spectral response (derivative result) with the reference wavenumber vector; but some software do not offer this correction.

Normalisation. Spectral normalisation techniques are commonly employed in IR and Raman spectroscopy to correct for different sample thickness and concentration, hence, avoiding the influence of non-desired spectral signatures among the samples. However, this procedure must be performed only when needed and with care, since the normalisation might hide important spectral bands that could be discriminant features among the samples, such as amide I and amide II absorptions; and it also may introduce non-linearities to the data[42]. Amide I and vector normalisation are the commonest type of normalisation for IR data; the first can be used when the amide I band is not a distinguishing feature between the classes; and the latter when this information is unknown but different sample thickness or concentration correction is needed.

Raman spectrometers using CCD detectors also surfers from cosmic rays interferences, which create spikes in the spectral data compressing important Raman spectral signatures. Spikes removal is an essential step when analysing Raman data and must be performed before any data pre-processing. Most Raman spectrometers' software have spikes removal routines. Finally, scaling methods (also referred as "standardization" by Hastie et al.[62]) are fundamental when dealing with multivariate methods, in particular PCA and partial least squares (PLS). Mean-centring is a very reasonable approach to use with spectral data before modelling, after which all variables in the dataset will have zero mean. When data contain information represented by different scales (*e.g.*, after data fusion using both IR and Raman spectra), block-scaling should be used. In this case, each block of data (*i.e.*, data for each instrumental technique) would have the same sum of squares (normally after mean-centring)[42]. For discrete data with different scales, autoscaling is recommend.

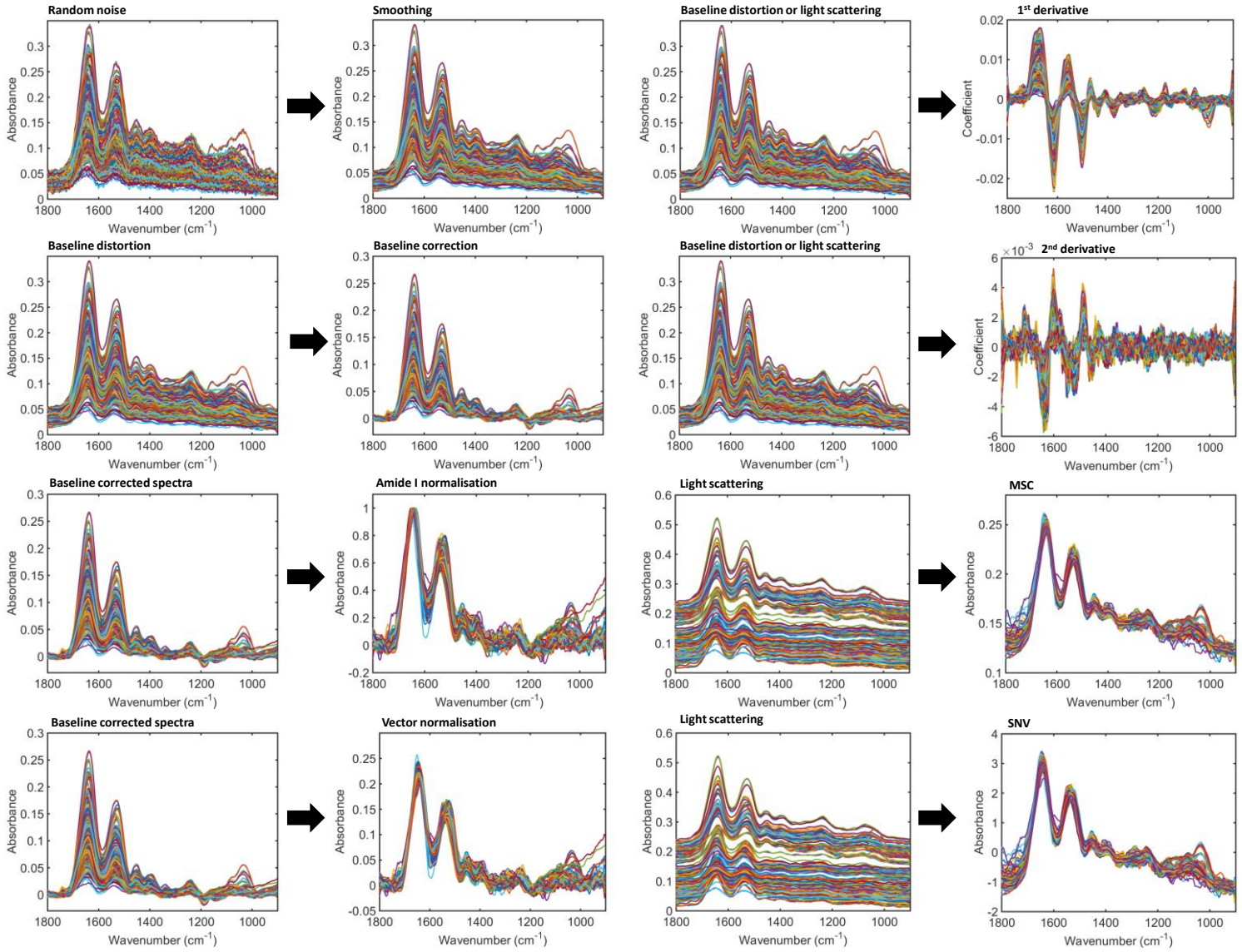

Figure 2. Effect of different pre-processing applied to an IR dataset. MSC: multiplicative scatter correction; SNV: standard normal variate.

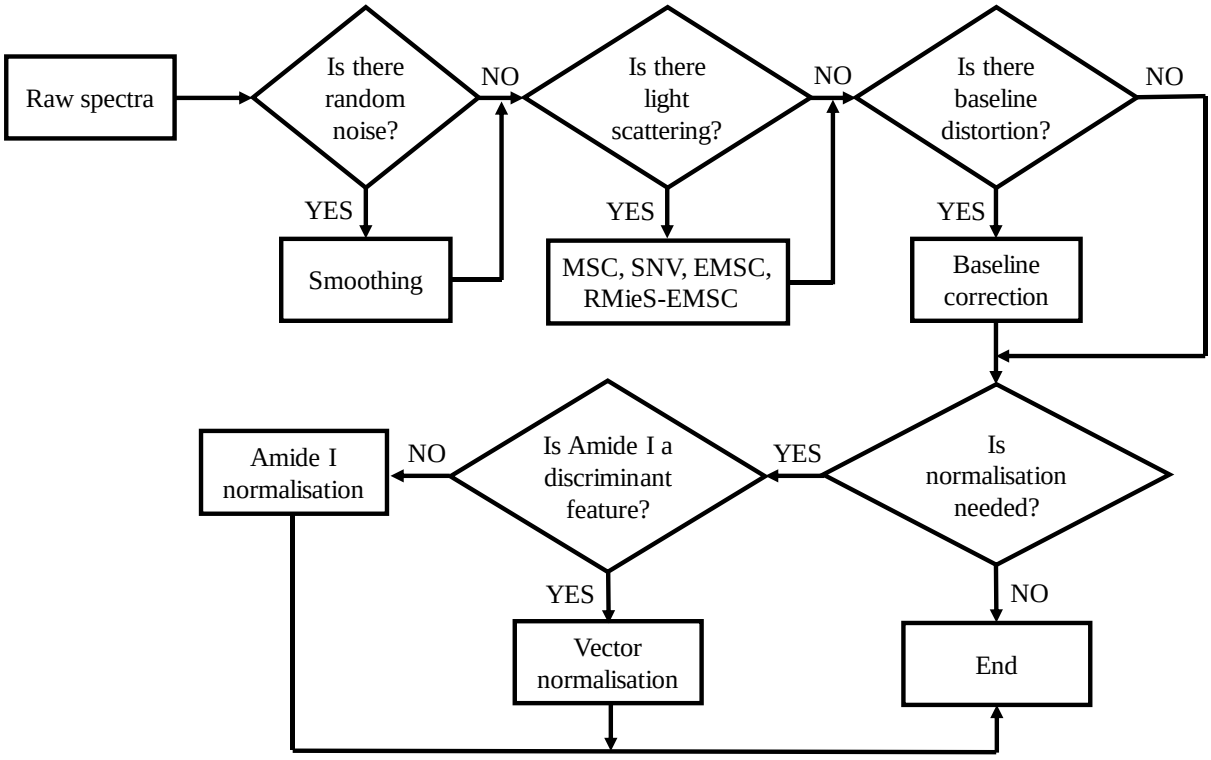

Figure 3. Decision tree to define the pre-processing technique for a spectral dataset. MSC: multiplicative scatter correction; SNV: standard normal variate; EMSC: extended multiplicative signal correction; RMieS-EMSC: resonant Mie scattering - extended multiplicative signal correction.

Outlier detection

When analysing real data, it often occurs that some observations are different from the majority. Such observations are called outliers. The spectral signal for some samples might differ from the spectral signal for the majority of the samples being measured. This can happen either by substantial differences in chemical structure or concentration for these specific samples, or by a measurement error. In the first case, we usually refer to an extreme sample, that is, a sample that belongs to the measurement set but with an extreme property value. This sample is characterized by a high Hotelling's T^2, and usually does not skew the model in a high degree; although it is recommended exclusion of this sample from the dataset before modelling. In the latter case, when the spectral abnormality is caused by a measurement error, this sample is a true outlier, being characterized by a high Q residuals. This sample should be removed from the dataset before analysis. Both extreme samples and outliers should be investigated in order to find possible sources of abnormalities.

There are several techniques for outlier detection, such as the Jack-knife[63], Z-score[64] and K-mode clustering[65]. However, one of the most popular and visually intuitive technique for outlier detection is the Hotelling's T^2 *versus* Q residuals test[66]. In this test, a chart is created using the Hotelling's T^2 values (sum of the normalised squared scores, which is the distance from the multivariate mean to the sample projection onto the PCA principal components (PCs) space) in the *x*-axis and the Q residuals (sum of squares of each sample in the PCA error matrix, representing the residuals between a sample and its projection onto the PCs space) in the *y*-axis, generating a scatter plot[42]. All samples far from the origin of this chart are considered candidates to outliers and should be

investigated and removed. Samples should be removed one at a time, since PCA is highly influenced by the samples that are included in the model. Samples with high values in both Hotelling's T^2 and Q residuals are the outliers with the greatest effect in PCA, while the samples with high values in only one of these parameters are the outliers with the second-greatest effect on the PCA model[42]. Figure 4 illustrates 4 outliers detected amongst a set of 700 IR spectra by a Hotelling's T^2 *versus* Q residuals chart applied to the pre-processed data (AWLS baseline correction and vector normalisation). The outliers were spectra corresponding to background noise measured within the experimental set, most likely by a mistake made by the analyst when placing the samples in the attenuated total reflection (ATR) apparatus.

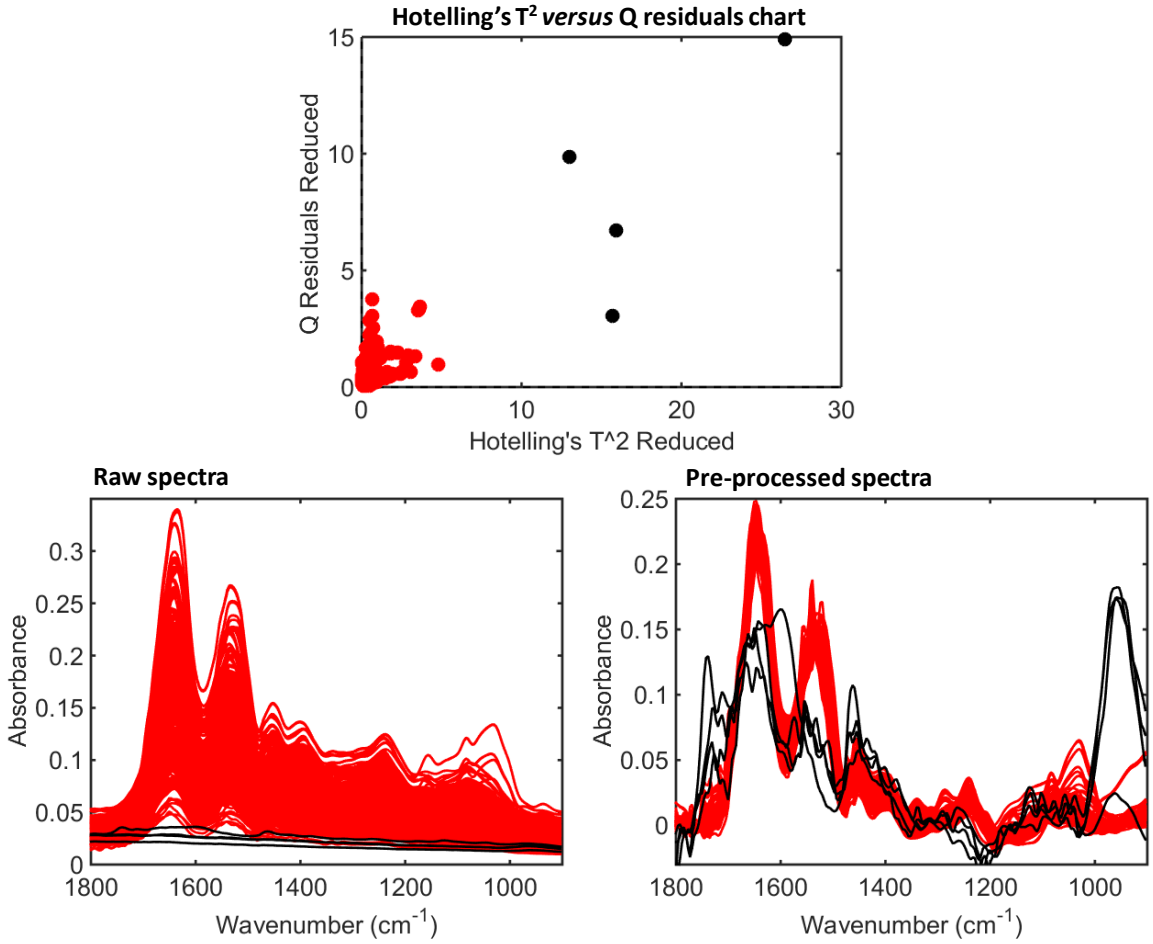

Figure 4. Outlier detection test by a Hotelling's T^2 *versus* Q residuals chart. Pre-processing: AWLS baseline correction and vector normalisation. PCA model built with 8 PCs (94.3% cumulative explained variance). Spectra in black: outliers.

Data selection

A fundamental step towards building predictive chemometric models is data selection, that is, splitting an initial experimental dataset into at least two subsets: training and test. The training set contemplates the major fraction of the samples and is used to build the classifier, whereas the test set includes the remaining fraction of samples and is used to evaluate the model classification performance, since, although they are measured during the same experiment, the test set is

considered external to the model (blind), thus reflecting the expected model behaviour toward new observations[67]. When two subsets are used, cross-validation is recommended to optimize the model parameters. Cross-validation uses samples from the training set to optimize model parameters, such as the number of PCs in PCA-based models or latent variables (LVs) in PLS-based models, in an iterative internal validation process. This is made by first removing a certain number of samples from the training set and then building the model with the remaining samples, where the removed samples are predicted as a temporary validation set[67]. This is performed for a certain number of repetitions usually until all training samples are excluded once from the training set and predicted as an external validation set. One of the most popular cross-validation methods is the leave-one-out cross-validation (also called leave-one-spectrum-out cross-validation). In this case, only one sample spectrum is removed from the training set per each interaction. Although much used, leave-one-out cross-validation is only indicated for small size datasets, usually with no more than 20 samples in the training set[42]. When this number is larger, other cross-validation approaches are recommended, such as venetian blinds or random subset selection. When there are replicate spectra, leave-one-spectrum-out cross-validation should not be used at all, but rather a continuous-block cross-validation (also called leave-one-patient-out cross-validation when the number of replicate spectra is equal for each sample and organised in a sequential way within the spectral matrix **X**), otherwise during the cross-validation procedure the training and temporary validation sets will have spectra from the same sample, hence, giving overoptimistic cross-validation results. In continuous-block or leave-one-patient-out cross-validation, the whole set of replicas for a same sample is transferred to the temporary validation set during cross-validation, thus the training and temporary validation sets will not have spectra for the same sample, hence, giving more realistic results.

When a large number of samples is measured, generally more than 100, it is recommended to split the experimental dataset into three groups: training, validation and test. In this case, an extra validation set that does not contain training samples is used to optimize the model. It is important to stress that the training, validation and/or test sets cannot contain spectra of the same sample distributed among them, *i.e.*, the samples in each set must be independent. Extra caution must be taken when multiple spectral replicates are used to feed the model, to ensure that they do not overlap in different sets.

There are several ways to split the samples into training, validation and/or test sets. Manual splitting is not recommend, since it can introduce bias to the model. Thus, computational-based methodologies are recommended instead. Random-selection and the Kennard-Stone (KS) algorithm[68] are some well-known approaches. Random-selection is a simpler approach where samples are assigned to the training, validation and/or test sets randomly. The KS algorithm works based on a Euclidian distance calculation by first assigning the sample with the maximum distance from all other samples to the training set, and then by selecting the samples that are as far away as possible from the selected sample to this set, until the designed number of selected samples is reached. This ensures that the training model will contain samples that uniformly cover the complete sample space, for which no or minimal extrapolation of the remaining samples is necessary[42]. KS has been proved to be superior than random-selection alone[67], but for biological-derived samples, we have recently proposed a modification to the KS algorithm by adding a small degree of randomness to it, where the model predictive performance increased; this is called the MLM algorithm[67].

Modelling

Exploratory analysis is the first step towards analysing complex spectral data, where the analyst can initially assess the data in order to identify clustering patterns and trends, thus helping them to draw conclusions about the nature of samples, outliers and experimental errors[42]. PCA is the most common method of exploratory analysis, in which the pre-processed spectral data are decomposed into a few number of PCs responsible for the majority of the variance within the original dataset. The PCs are orthogonal to each other and are generated in a decreasing order of explained variance, so that the first PC explain most of the data variance, followed by the second PC and so on[69]. PCA decomposition takes the form:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \quad (3)$$

where $\mathbf{X}$ represents the pre-processed spectral data, $\mathbf{T}$ is the PCA scores, $\mathbf{P}$ the loadings, and $\mathbf{E}$ the residuals.

PCA is then often the first step of the data analysis, followed by classification, cluster analysis, or other multivariate techniques. The PCA scores represent the variance in the sample direction, being used to detect clustering patterns related to chemical similarities/dissimilarities between the samples. The PCA loadings represent the variance in the wavenumber direction, being used to identify spectral variables with high degree of importance for the pattern observed in the scores distribution[42]. The PCA loadings are commonly used for searching spectral markers that distinguish samples from different biological classes. This can be performed by identifying the spectral bands with the highest absolute loadings coefficients (positive or negative) on the discriminant PCs directions[42]. Another strategy proposed by Martin *et al.*[70] is the cluster vector approach, where a median score is calculated for each of the 3 PCs that represent the best samples' clustering in a three-dimensional space and, thereafter, the three loading vectors for these PCs weighted by the median score are summed. As a result, a new loading vector is generated representing the effective loadings profile for the clustering[2,70]. The PCA residuals represent the difference between the decomposed and original pre-processed data, being used to identify experimental errors. Ideally, the PCA residuals should be random and close to zero (homoscedastic distribution); otherwise they indicate experimental bias[42].

PCA is a fast, intuitive and reliable method to identify differences between spectral data, however, it is important to stress that PCA is not a classification technique. PCA is a data reduction and exploratory analysis method, but PCA solely cannot be used to systematically classify samples. For this, supervised classification techniques are needed. Supervised classification methods build computer-based classifiers that can predict future samples based on their training spectral profiles. Therefore, data selection as mentioned previously is fundamental before building supervised classification models.

Classification methods are separated into two groups: one-class modelling (also called class-modelling) and discriminant models. In one-class modelling, the classification model output does not solely depend on the training classes, thus it can assume values such as "unknown" or that the test sample does not belong to any of the training classes. On the other hand, in discriminant models the model outputs are always referent to one of the training classes. The one-class modelling approach is very useful when only one class is modelled and the model output is whether the sample belongs or not to the reference class. However, one-class approaches requires a large number of samples, since the classes boundaries must include all the sample space as much as possible, and usually

provides worst classification results than discriminant models, since in one-class modelling slightly extreme samples to the reference class could be interpreted as not belonging to the class. These problems are not present in discriminant models, since the model output is always one of the training classes, and the class space is much larger. Discriminant approaches cannot predict samples that do not belong to the training classes, thus building meaningful training sets is fundamental in this type of analysis.

The main algorithm of one-class modelling is the soft independent modelling by class analogy (SIMCA)[71]. In SIMCA, each class is modelled by an independent PCA model of opportune dimensionality. Then, the class space is defined according to some statistically defined outlier detection criterion, which is often the distance-to-the-model criterion[72]. This is made by calculating the probability distributions for the T^2 statistics and Q statistics for the PCA model of each class, where a threshold corresponding to a determined confidence level (usually 95%) is chosen for both statistics to define the class space[72]. Other ways to define the class space are possible, such as the method proposed by Pomerantsev[73], although the T^2 and Q statistics is the most common approach.

There are several discriminant analysis algorithms, most of them based on distance calculations on the real or transformed sample space. The main discriminant analysis algorithms employed in biological-derived spectrochemical applications will be discussed below.

Linear discriminant analysis (LDA). LDA is a discriminant analysis algorithm based on a Mahalanobis distance calculation between the samples for each class[74]. This calculation can be performed with or without Bayesian probability terms, which can be applied when classes have different sizes[74]. LDA uses the pooled variance-covariance matrix in the distance calculation, hence, the distance between a test sample and a given class centroid is weighted according to the overall variance of each spectral variable[74]. This is particularly useful when the classes have similar variance structures or when the sample size is small[75]. However, LDA is highly affected when classes have different variance structures, which often happens in complex biological medium. In addition, LDA is a parametric method that assumes the samples follow a normal distribution and cannot be applied to ill-conditioned data, *e.g.*, when the number of spectral variables is larger than the number of samples[42]. Although spectral data usually do not perfectly follow a normal distribution, LDA is robust enough to handle spectroscopy data and, according to the CLT, this effect can be reduced by increasing the sample size. The issue related to ill-conditioned data can be solved by the application of PCA or variable selection techniques to the pre-processed spectral data prior LDA, such as the principal component analysis linear discriminant analysis (PCA-LDA) algorithm, where LDA is applied to the PCA scores[76].

Quadratic discriminant analysis (QDA). Similarly to LDA, QDA is a discriminant analysis algorithm based on a Mahalanobis distance calculation between the samples for each class, which can use Bayesian probability terms to correct for classes having different sizes[74]. However, differently from LDA, QDA forms a separate variance model for each class, thus using a different variance-covariance matrix for each class[74]. For this reason, QDA outperforms LDA when classes exhibiting different within-category variances are being analysed[77]. Like LDA, QDA is also a parametric method that is highly affected by ill-conditioned data; however, these issues can be solved in the same manner as described for LDA. Often, QDA is applied to the PCA scores in the principal component analysis quadratic discriminant analysis (PCA-QDA) algorithm[76]. Its main disadvantages are that QDA underperforms LDA for small size datasets and it has a higher risk of overfitting than LDA[42,75,77].

Partial least squares discriminant analysis (PLS-DA). PLS-DA[40] is a feature extraction and classification algorithm that usually performs better than PCA followed by LDA, as the scores from PCA do not

necessarily describe the difference between the samples, but the variance in the spectral data[42]. In PLS-DA, a PLS model[78] is applied to the pre-processed spectral data reducing the original spectral variables to a small number of latent variables (LVs), where then a linear discriminant classifier is used for classifying the groups[34]. It is important to stress that there are different ways of performing the PLS model, such as using the SIMPLS[79] or the non-linear iterative partial least squares (NIPALS) algorithm[80], and that the classification rule of PLS-DA vary according to the application or software being used. Linear classifiers based on Euclidian distance to centroids[40], LDA[81] and Bayesian decision rule[82] are some examples that can be used in PLS-DA. In addition, PLS-DA can be adapted to one-class modelling, as described by Pomerantsev and Rodionova[81]. The main disadvantage of PLS-DA is that this algorithm is greatly affected by classes having different sizes and it requires optimization of the number of LVs, which is often performed by cross-validation[42]. Also, it is important to highlight that PLS-DA is a binary classifier, that is, when more than two classes are analysed a PLS2 model is built where the classes are codded in a matrix with size *m* (rows, samples) x *n* (columns, classes) containing zeros when the sample does not belong to the target class and ones when the sample belong to the target class. In PLS-DA, class-coding cannot be made in a sequential manner (*e.g.*, 1, 2, 3, …) since this imply a distance relationship between the samples (*e.g.*, samples from class 1 are farther from class 3 than the samples in class 2). Some softwares allow the input of sequential class-coding, but this information is internally convert into a zeros and ones matrix before model construction.

K-nearest neighbours (KNN). KNN[83] is a local non-parametric classification method where samples are classified based on the "majority vote" approach, that is, a given test sample spectrum is projected in a feature space and based on the calculation of a distance or dissimilarity metric (Manhattan, Euclidian, Minkowski or Mahalanobis distance; or by correlation), depending on the number of nearest surrounding neighbour training samples to this test sample, the sample is classified towards the majority observed class. The main advantage of KNN is that it can be applied to almost all type of data independent of its probability distribution or condition number, and does not require a particular ratio between the number of samples and the number of spectral wavenumbers[72]. KNN main disadvantages are that the model tends to overfit by skewing towards the bigger class size when unequal classes sizes are analysed, and that the model is highly sensitivity towards random spectral noise and to the "curse of dimensionality"[42,562,72]. In addition, KNN requires the optimization of the distance calculation method and the *k* value (number of neighbours), which can be performed through cross-validation[42].

Support vector machines (SVM). SVM is a binary linear classifier with a non-linear step called the kernel transformation[84]. A kernel function transforms the input spectral space into a feature space by applying a mathematical transformation which is often non-linear. Then, a linear decision boundary is fit between the closest samples to the border of each class (called support vectors), thus defining the classification rule. Although being highly accurate to classify spectral data, SVM requires many parameters optimization, such as the type of kernel function and its parameters, and it is highly susceptible to overfitting; besides being a highly time-consuming algorithm[42]. The radial basis function (RBF) kernel is often the best kernel to use in SVM, since it can adapt to different data distribution. To avoid overfitting, cross-validation should be always performed to estimate the best kernel parameters[42]. Multiclass SVMs are possible through the implementation of approaches such as one *vs.* all, one *vs.* one, fuzzy rules and directed acyclic graph trees[85], although the first approach is the most common.

When data complexity increases, for instance, when the spectra data are not following a bilinear rule or when the components complexity are too excessive to be analysed by the previous methods,

"black box" algorithms, *i.e.*, machine learning techniques where the classification rules are hard to interpret, can be applied. Most of these algorithms were developed for applications such as face recognition, where a high degree of non-linearity is observed between the measurements, but they have found their way into spectrochemical applications. Artificial neural networks (ANN)[86], random forests[87] and deep-learning approaches[88] are common classification methods applied in such situations. All these techniques have a non-linear classification nature and higher accuracy in comparison with more simpler methods, however in order for these algorithms work properly with a low-risk of overfitting many parameters need to be optimised, which depends on the analyst skills and usually demands high computation cost[42]. Classification techniques should be used in a parsimonious order[48], in which the simplest algorithms should be performed first before testing more complex algorithms. A suggested order for running these classification algorithms is: LDA > PLS-DA > QDA > KNN > SVM > ANN > Random forests > deep-learning approaches[42].

Apart from these discriminant methods, there are many other discriminant analysis algorithms that are known but not much applied to biological-derived spectral data, such as learning vector quantization (LVQ)[74] and regularized discriminant analysis (RDA)[75]. These algorithms are not much used probably due to the lack of available software containing these routines. The main chemometric softwares for classification applications are shown in Table 1. Apart from these main softwares, there are many open source freely-available options with specific algorithms for classification of spectral data, such as the MultiDA[89] toolbox for MATLAB that contains some classification routines; the Biodata[90] toolbox for MALTAB that contain PCA-LDA routines; the SAISIR[91] toolbox that contains PCA-LDA, PLS-DA and QDA routines; the ParLeS[92] software that contain routines including PCA and PLS; the Raman Processing Program[93] that contains LDA, ANN and SVM routines; the PML[94] toolbox for machine learning; the DD-SIMCA[95] toolbox for MATLAB that contain SIMCA routines; the libPLS[96] library for MATLAB that contain PLS-DA routines; and the LIBSVM[97] library for SVM. Other classification routines can be found in spectrometer softwares or by specific libraries or toolbox available online for MATLAB, Octave, Scilab, R and Python.

Octave and Scilab are open source freely-available platforms with syntax very similar to MATLAB and may interchangeable routines[98], *i.e.*, routines made for MATLAB often work in Octave or Scilab. Freely-available chemometric toolboxes for Octave and Scilab include the SAISIR toolbox[91] (https://www.chimiometrie.fr/saisir_webpage.html) and the FACT (Free Access Chemometrics Toolbox) (https://www.scilab.org/fact-free-access-chemometrics-toolbox). R is another powerful open source statistical platform often used for chemometric applications[99]. Apart from the CAT (Chemometric Agile Tool) toolbox showed in Table 1, there are other freely-available chemometrics packages for R, such as the Chemometrics package (https://www.rdocumentation.org/packages/chemometrics/versions/1.4.2) and the CRAN package Chemometrics[100] (https://cran.r-project.org/web/packages/chemometrics/index.html). Python is a high-level computer programming language which is also becoming popular for chemometric applications. Jarvis *et al.*[101] developed an open source chemometric toolbox named PYCHEM for multivariate analysis of spectral data using Python. PYCHEM is freely-available at http://pychem.sourceforge.net/.

Table 1. Main chemometric softwares for multivariate classification.

Software	Website	Classification algorithms	Availability
IRootLab[102]	http://trevisanj.github.io/irootlab/	LDA, PCA-LDA, ANN, fuzzy classification, KNN, logistic regression, SVM, PCA-SVM, binary decision trees.	Free
Classification Toolbox for MATLAB[103]	http://www.michem.unimib.it/	LDA, QDA, PCA-LDA, PCA-QDA, classification trees (CART), PLS-DA, SIMCA, unequal class models (UNEQ), potential functions, SVM, KNN, backpropagation neural networks.	Free
CAT (Chemometric Agile Tool)	http://gruppochemiometria.it/index.php/software	LDA, QDA, KNN.	Free
PLS_Toolbox	http://www.eigenvector.com/	PLS-DA, SVM, SIMCA, KNN.	Commercial
Statistics and Machine Learning Toolbox for MATLAB	https://mathworks.com/	Binary decision trees, LDA, QDA, naïve Bayes classifier, KNN, SVM, random forest.	Commercial
Unscrambler X	https://www.camo.com/unscrambler/	SIMCA, LDA, PLS-DA, SVM.	Commercial
Pirouette	https://infometrix.com/	KNN, SIMCA, PLS-DA.	Commercial
SIMCA Umetrics	https://umetrics.com/	SIMCA, PLS-DA, orthogonal partial least squares discriminant analysis (OPLS-DA).	Commercial

Feature extraction and selection

The feature extraction stage is responsible for producing a smaller number of variables that are more informative than the original whole set of wavenumber/variables. Feature selection is commonly applied as a stage prior to classification as a means to prevent overfitting and to circumvent the "curse of dimensionality". Feature extraction and selection can be used to reduce data complexity, to reduce redundant information, to speed computation-time, and to aid biomarkers identification. Feature extraction techniques allow one to extract spectral features related to important chemical components within the spectral dataset, and feature selection techniques significantly reduces the pre-processed spectral dataset to a small set of variables responsible for class differentiation. The most straightforward way to identify important spectral features is by plotting the colour-coded mean-centred data. As mentioned before, mean-centring is a key scaling step applied to the data before multivariate analysis. By plotting the mean-centred data, the analyst can see spectral regions where the general data trend between the classes diverge. Spectral regions where one can clearly see a spectral difference are often the most important spectral regions within the dataset. These regions can be selected for model construction; though it is a trying and error procedure. *I.e.*, the analyst needs to evaluate the model validation performance

using the whole spectrum, some selected spectral regions, and spectral variables selected by other methods. Also, knowing the nature of the phenomena being measured can aid and guide the analyst to select important spectrochemical features.

Feature extraction techniques can also be used directly or indirectly to identify important spectral variables. PCA and PLS-DA are two very common feature extraction techniques. PCA loadings and the PLS-DA regression coefficients can be used to identify important spectral features. Some approaches based on PLS such as variable importance in projections (VIP) scores[104], selectivity ratio[104] and interval partial least squares (iPLS)[105] are useful tools to find important spectral variables. VIP scores, selection ratio and iPLS are very different techniques where the first two are methods employing the PLS loadings or the regression coefficients, which carry some risk of misinterpretation, since regression vectors of inverse calibration methods, such as PLS, can exhibit extremely complex behaviour in even the most simplistic circumstances[106]. On the other hand, methods employing the minimum error as guidance, such as iPLS, do not carry this risk and are in some degree more reliable for qualitative spectral interpretation.

Another feature extraction technique particularly useful for hyperspectral imaging data is the multivariate curve resolution alternating least squares (MCR-ALS) algorithm[107,108]. MCR-ALS can be applied to reduce the spectral dataset into important spectral features and aid biomarker identification through the interpretation of the concentration and spectral profiles containing the purest components in the experimental spectral matrix. MCR-ALS allows the introduction of chemical constraints in the factor analysis model, improving the resolution of spectral mixtures through adjusting chemically-meaningful parameters; and allow the construction of concentration distribution maps based on the recovered MCR-ALS concentration profiles for hyperspectral images datasets[109]. For first-order spectral data, MCR-ALS can also be applied to solve mixtures and to recover concentration and pure spectral profiles towards kinetic, quantitative or qualitative applications. However, MCR-ALS is limited by the fact that its bilinear decomposition may show a large degree of rotational ambiguity, precluding the obtainment of reliable results.

Feature selection techniques allows the identification of specific spectral wavenumbers within the original spectral space responsible for maximizing the class differences. Some examples of feature selection algorithms include the minimum redundancy maximum relevance (mRMR) algorithm[110], in which the variable selection process is based on the maximization of the relevance of extracted features and simultaneous minimization of redundancy between them[42]; the successive projections algorithm (SPA)[111], which is an iterative forward feature selection method operating by solving co-linearity problems within the spectral dataset, thus selecting wavenumbers whose information content is minimally redundant[112]; and the genetic algorithm (GA)[113], which is a iterative combinational algorithm inspired by Mendelian genetics where a set of initial variables undergo selection, cross-over combinations and mutations until the fittest selected variables are found[112]. The variables selected by these techniques can be used as input for the classification methods described previously, which is important since these techniques reduce the data size and collinearity, hence, improving the model accuracy and analysis time. Adaptations of PCA, PLS, SPA and GA (as feature extraction/selection techniques) to LDA, QDA, SVM, KNN and ANN (as classifiers) are well known[114-116].

Model validation

The performance of any classification model must be validated by calculating some quality metrics, or figures of merit, for a validation or test set. The training or cross-validation performance reflects the model fitting but it does not reflect the model predictive performance towards unknown samples. For this reason, figures of merit such as accuracy (AC), sensitivity (SENS), specificity (SPEC), positive predictive value (PPV), negative predictive value (NPV), positive likelihood ratio (LR+), negative likelihood ratio (LR-), F-Score and G-Score are often calculated for external validation or test sets[117,118]. The equations to calculate these parameters are depicted in Table 2.

Table 2. Quality parameters to evaluate the model classification performance. TP stands for true positives, FP for false positives, TN for true negatives, and FN for false negatives.

Parameter	Equation	Meaning
Accuracy (AC) / %	$\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \times 100$	Number of samples correctly classified considering true and false negatives. Optimal value: 100%.
Sensitivity (SENS) / %	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \times 100$	Proportion of positive samples (*e.g.*, disease) that are correctly classified. Optimal value: 100%.
Specificity (SPEC) / %	$\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} \times 100$	Proportion of negative samples (*e.g.*, healthy controls) that are correctly classified. Optimal value: 100%.
Positive predictive value (PPV) / %	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \times 100$	Number of test positives that are true positives. Optimal value: 100%.
Negative predictive value (NPV) / %	$\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FN}} \times 100$	Number of test negatives that are true negatives. Optimal value: 100%
Positive likelihood ratio (LR+)	$\frac{\mathrm{SENS}}{1-\mathrm{SPEC}}$	Ratio between the probability of predicting a sample as positive when it is truly positive and the probability of predicting a sample as positive when it is actually negative. SENS and SPEC are not in percentage. Optimal value: infinite.
Negative likelihood ratio (LR-)	$\frac{\mathrm{SPEC}}{1-\mathrm{SENS}}$	Ratio between the probability of predicting a sample as negative when it is actually positive and the probability of predicting a sample as negative when it is truly negative. SENS and SPEC are not in percentage. Optimal value: 0.
F-Score	$\frac{2 \times \mathrm{SENS} \times \mathrm{SPEC}}{\mathrm{SENS}+\mathrm{SPEC}}$	Model performance considering imbalanced

G-Score	$\sqrt{\text{SENS} \times \text{SPEC}}$	classes. Optimal value: 100%. Model performance not accounting for the classes size. Optimal value: 100%.

For binary models, *i.e.*, models containing two classes, these parameters are calculated only once, where the positive class is the class of interest (*e.g.*, disease) and the negative class is the control class (*e.g.*, healthy controls). When more than two classes are modelled, then these parameters must be calculated individually per class. Often, receiver operating characteristic (ROC) curves, including the area under the curve (AUC) value; and confusion matrices containing the predicted number of samples or predicted classification rate per class are reported in a form of a table or graphically to aid the reader evaluating the model classification performance. Also, based on the confusion matrix, the Cohen's kappa coefficient (κ)[119] can be calculated, which is a weighted average of the model performance. Other parameters, such as model uncertainty, can be calculated. Uncertainty is related to the probability of misclassification and model robustness[120], and can be calculated for LDA, QDA and SVM models[120]; PLS-DA[121,122]; and ANN[123]. The number of quality metrics to report depends on the application and rigor of the study. We recommend reporting at least the accuracy, sensitivity and specificity for small studies, while all the metrics in Table 2 can be reported for bigger studies.

Procedure

Loading the data ● Timing 5 min – 2 d, depending on the size of the dataset

1. The spectra data must be loaded into the software for data analysis. Usually, the spectral data need to be converted to suitable .txt or .csv files within the spectrometer software, or saved in an extension format readable by the software used for data analysis. MATLAB commands such as 'csvread' and 'importdata', or the option right click > 'Import Data…' over the file in the "Current Folder" window of MATLAB, can be used to load standard .csv or .txt files. IRootLab[86] toolbox for MATLAB contain an interface called "mergertool" to load different spectral formats: .DAT files, .csv files, OPUS binary files from Fourier transform infrared (FTIR) Bruker® spectrometers, and .txt and Wire files from Renishaw® Raman spectrometers. We strongly suggest saving the spectral files in .csv or .txt file formats since these are universal formats most chemometric software read regardless the instrument manufacturer brand.

▲ **CRITICAL** Experimental procedures for sample preparation and Raman and FTIR spectral acquisition are demonstrated in other protocols[1,3,8,42].

▲ **CRITICAL** The routine to load the spectral data depends on the file format, spectrometer manufacturer, and software being used to analyse the data.

? TROUBLESHOOTING

In case of fail to load directly the spectral data into the software for data analysis, export the spectral data into readable .txt, .csv or .xls formats within the spectrometer software and load them

into Microsoft® Excel or any spreadsheet software. Then, copy and paste the spectral data from the spreadsheet to the data analysis software.

■ **PAUSE POINT** Save the spectral dataset in the data analysis software format (*e.g.*, .mat for MATLAB) into a known folder for further analysis.

Data quality evaluation ● Timing 40 min – 4 h, depending on the size of the dataset

2. Evaluate the raw spectral data by plotting them and by performing quality tests to identify anomalous spectra or biased patterns before applying processing. This can be done by visual inspection of the spectral profiles, followed by plotting Hotelling's T^2 *versus* Q residuals charts using only the mean-centred data, and the analysis of the PCA residuals. Samples far from the origin of the Hotelling's T^2 *versus* Q residuals chart should be investigated and removed. Outliers must be removed one at the time from the PCA model. PCA residuals should be random and close to zero. Further instructions about data quality evaluation can be found in 'Minimum dataset requirements' and 'Outlier detection' in the 'Experimental design' section. Hotelling's T^2 *versus* Q residuals charts can be built using the automatic Outlier Detection algorithm[42] for MATLAB at https://figshare.com/articles/Outlier_Detection/7066613/2.

Data pre-processing ● Timing 15 min – 4 h, depending on the size of the dataset

▲ **CRITICAL** Steps 3 – 8 below can be modified depending on the nature of the dataset. Pre-processing effects are depicted in Fig. 2 and a pre-processing decision flowchart is shown in Fig. 3. Further details about pre-processing techniques can be found in 'Pre-processing' in the 'Experimental design' section.

3. *Selecting the biofingerprint region*. Truncate the spectra dataset to the biofingerprint region to eliminate atmospheric interference present in other regions of the spectra. FTIR: 1800 – 900 cm^{-1}; Raman: 2000 – 500 cm^{-1}; NIR: 900 – 2600 nm.
4. *Savitzky-Golay (SG) smoothing for removing spectral noise.* When random noise is present, SG smoothing should be applied. Window size varies according to the spectral dataset resolution and size. The window size must be an odd number, since a central data point is required for the smoothing process. Try different window sizes from 3 to 21 and observe how the spectra change (in shape) and how the noise is reduced. Use the smallest window size that removes a considerable amount of the noise while maintaining the original spectral shape. *E.g.*, using a spectral resolution of 4 cm^{-1}, the IR biofingerprint region (900-1800 cm^{-1}) usually contains 235 wavenumbers; in this case, a window size of 5 points should be used. The polynomial order of the SG fitting should be second order for IR, Raman and NIR datasets due to the quadratic band shape of the spectrum.
5. *Light scattering correction using either MSC, SNV or second derivative.* First, try using MSC or SNV, as MSC maintains the spectral scale and both methods maintain the original spectral shape. If the results are not satisfactory (*e.g.*, classification accuracy < 75% in the validation set), try using the second-derivative spectra.

▲ **CRITICAL** If resonant Mie scattering[58] is present in the spectra (often detected by a severe baseline distortion followed by a systematic shift in the *y*-axis), then the RMieS-EMSC algorithm[59] should be used for spectral correction instead of MSC, SNV, second derivative or other baseline correction technique.

6. *Perform baseline correction using AWLS or rubber-band baseline correction*. If EMSC or spectral differentiation is applied as the light scattering correction method, baseline correction is not necessary.
7. *Normalisation*. Normalise the spectrum to the Amide I or Amide II peak, or perform a vector normalisation (2-norm, length = 1) to correct different scales across spectra (*e.g.*, due to different sample thickness when using FTIR in transmission mode).
8. *Scaling*. Mean-centre the spectral dataset. In case of data fusion, block-scaling should be used.

▲ **CRITICAL** Plot the spectral data throughout all the pre-processing steps to identify anomalous behaviours. For parsimonious reason, only use the pre-processing methods that are needed for the dataset (see Fig. 3 and 'Pre-processing' in the 'Experimental design' section).

■ **PAUSE POINT** Save the pre-processed spectral dataset in the data analysis software format (*e.g.*, .mat for MATLAB) into a known folder for further analysis.

Exploratory analysis ● Timing 1 h – 4 d, depending on the size of the dataset

9. Perform a PCA model with the mean-centred pre-processed spectral data to identify clustering patterns, trends and outliers within the dataset. Determine the number of PCs by plotting the number of PCs *versus* the model explained variance, where the selected number of PCs should be the one that contains the majority of the cumulative explained variance before a constant trend is observed in the next following PCs. Usually the number of PCs should not exceed 10 PCs, since this can add random noise to the model.
10. Plot the PCA scores on PC1 *versus* PC2, and investigate other combinations of PCA scores plot on different PCs according to the number of selected PCs to identify possible clustering patterns or trends. Colour-code the samples to facilitate visualisation. If a clear segregation pattern between the classes is observed on the PCA scores space, this is an indication that PCA-based discriminant models, such as PCA-LDA and PCA-QDA, might work well with the dataset.
11. Plot the Hotelling's T^2 *versus* Q residuals chart for the PCA model built in order to identify possible outliers still within the spectral dataset. The outliers should be removed from the dataset before proceeding to the next steps.

▲ **CRITICAL** The pre-processed spectral data must be mean-centred before PCA.

▲ **CRITICAL** PCA is not a classification method, thus the PCA scores plot is not the final classification model.

? TROUBLESHOOTING

If no segregation trend is observed in the PCA scores plot, this is an indicative of the dataset complexity. The visualisation of the PCA scores is limited to 3-demenions (3D) plots, hence, no apparent segregation trend does not mean that the dataset cannot be discriminated in the PCA scores space. Therefore, PCA-based discriminant models can still be built by using 4 to 10 PCs, or more.

Data selection ● Timing 10 min – 4 h, depending on the size of the dataset

12. Separate the samples that will be used for the training and test sets. Data selection should be performed before model construction. The samples can be split into training (70%) and test (30%) sets, using a cross-validated model; or they can be split into training (70%), validation (15%) and test (15%) sets without using cross-validation. To maintain consistency and account for well-balanced training models, the KS or MLM algorithms are suggested to separate the samples into the sub-sets. The KS algorithm is freely available at https://doi.org/10.6084/m9.figshare.7607420.v1; and the MLM algorithm is freely available at https://doi.org/10.6084/m9.figshare.7393517.v2.

▲ **CRITICAL** Spectrum replicas for a same sample cannot be present in more than one sub-set; that is, spectral replicates cannot be distributed amongst the training, validation and/or test sets.

? TROUBLESHOOTING

When spectral replicates are present in the dataset, the data selection algorithm can be applied in a way to keep the spectrum replicas together, or by averaging the spectral replicates before applying the data selection algorithm.

? TROUBLESHOOTING

If the percentages of samples (70%, 30% or 15%) for each sub-set generate numbers with decimal places, round them to the closest integer values.

■ **PAUSE POINT** Save the training, validation and/or test sets in the data analysis software format (*e.g.*, .mat for MATLAB) into a known folder for further analysis.

Model construction ● Timing 1 h – 4 d, depending on the size of the dataset

▲ **CRITICAL** Feature extraction (*e.g.*, by means of PCA) or feature selection (*e.g.*, by means of SPA or GA) should be used to reduce data collinearity and speed up data processing and analysis time. PLS-DA is already a feature extraction method; thus performing a feature extraction technique prior PLS-DA is not necessary. KNN, SVM and ANN algorithms can be applied either without or after feature extraction/selection techniques. The classification technique being tested must follow a parsimony order: LDA > PLS-DA > QDA > KNN > SVM > ANN > random forests > deep-learning approaches.

13. Apply the feature extraction or selection technique. The optimization of the number of PCs during PCA-based methods or LVs during PLS-DA can be performed using an external validation set (15% of the original dataset) or using cross-validation (leave-one-out for small

datasets (≤20 samples); venetian blinds or random subsets with 10 data splits can be used for large datasets (>20 samples); or continuous-block (*i.e.*, leave-one-patient-out cross-validation) when replicate spectra are present). GA should be performed three times, starting from different initial populations, and the best result using an external validation set (15% of the original dataset) should be used. Cross-over probability should be set to 40% and mutation probability should be set to 1–10%, according to the dataset size.

14. The classification method should be used optimized by an external validation set or by using cross-validation, especially for selecting the number of LVs of PLS-DA, and the kernel parameters for SVM. The kernel function for SVM should be the radial basis function (RBF) kernel, due to its adaptation to different data distribution. To avoid overfitting, cross-validation should be always performed during model construction to estimate the best RBF parameters in SVM.

▲ **CRITICAL** The final classification model must be built with the optimum classifier parameters.

■ **PAUSE POINT** Save the training model parameters for further analysis.

Model validation ● Timing 1 h – 8 h, depending on the size of the dataset

15. After model construction using the training set, the model must be blindly validated by an external test set. The samples in the test set cannot be present in the training set; and the model output for the test set must be statistically compared with reference known values.
16. Based on the model output for the training and test sets, calculate the accuracy, sensitivity and specificity for each set. The metrics for the training set are used to assess the fitting of the model, but they do not reflect the true model behaviour towards unknown samples. The metrics for the test set are the true expected results representing the predictive classification ability of the model.

Troubleshooting

Loading the data

The file format in which the spectral data is saved must be readable by the data analysis software. Check the importing data routines in the data analysis software beforehand to save the experimental files in a suitable file format.

Data pre-processing

Data pre-processing techniques should be used in a parsimonious way, and they cannot mask the signal of interest. Testing different pre-processing techniques is recommend to find the best solution in terms of cross-validation or validation performance.

Model construction

In case of unsatisfactory classification results, the complexity of the model being tested should increase in the following order: LDA > PLS-DA > QDA > KNN > SVM > ANN > random forests > deep-learning approaches. Changing the type of classifier, the feature extraction/selection technique and the type of pre-processing are ways to narrow down the classification results and find the best classification model. The performance testing of candidates models can be made by cross-validation or by using an external validation test. The final performance of the classification model must be calculated using an external test set containing independent samples (samples not present in the training set).

Timing

Loading the data

Step 1, importing the data to the data analysis software: 5 min – 2 d, depending on the size of the dataset.

Data quality evaluation

Step 2(A), plotting and inspecting the spectral profiles: 15 min – 1 h, depending on the size of the dataset.

Step 2(B), inspecting Hotelling's T^2 *versus* Q residuals charts for the mean-centred raw data: 15 min – 2 h, depending on the size of the dataset.

Step 2(C), analysis of PCA residuals: 10 min.

Data pre-processing

Step 3, selecting the biofingerprint region: 10 min.

Step 4, Savitzky-Golay (SG) smoothing for removing spectral noise: 2 min – 20 min, depending on the size of the dataset.

Step 5, light scattering correction using either MSC, SNV, second derivative or RMieS-EMSC: 2 min – 20 min, depending on the size of the dataset.

Step 6, performing baseline correction using AWLS or rubber-band baseline correction: 2 min – 40 min, depending on the size of the dataset.

Step 7, normalisation: 1 – 10 min, depending on the size of the dataset.

Step 8, scaling: 1 – 10 min, depending on the size of the dataset.

Exploratory analysis

Step 9, building a PCA model with the mean-centred pre-processed spectral data to identify clustering patterns: 10 min – 4 h, depending on the size of the dataset.

Step 10, plot the PCA scores on PC1 *versus* PC2, and investigate other combinations of PCA scores plot on different PCs to identify possible clustering patterns or trends: 30 min – 12 h, depending on the number of selected PCs.

Step 11, checking the Hotelling's T^2 *versus* Q residuals chart to identify outliers: 10 min – 2 h, depending on the size of the dataset.

Data selection

Step 12, sample splitting: 10 min – 4 h, depending on the size of the dataset.

Model construction

Step 13, application of feature extraction or selection techniques: 15 min – 8 h, depending on the size of the dataset and the feature selection technique being used.

Step 14, construction of the classification model: 15 min – 4 d, depending on the size of the dataset, type of cross-validation and type of classifier.

Model validation

Step 15, obtaining the model outputs for the test set: 15 min – 1 h, depending on the size of the test set.

Step 16, calculation of model performance metrics: 30 min – 8 h, depending on the number of metrics being calculated and the number of classes in the dataset.

Anticipated results

To illustrate how this protocol can be used to analyse spectral data, we will conduct some classification models for 3 real datasets: (1) Syrian hamster embryo (SHE) cells dataset composed of

FTIR spectra, free available as part of IRootLab toolbox (http://trevisanj.github.io/irootlab/); (2) Raman spectra of blood plasma to detect ovarian cancer, free available at https://doi.org/10.6084/m9.figshare.6744206.v1; and (3) NIR spectra of corn samples, free available at http://www.eigenvector.com/data/Corn/index.html.

Dataset 1 (SHE cells) was originally published by Trevisan *et al.*[124] and is composed of originally 10 classes, containing attenuated total reflection Fourier-transform infrared (ATR-FTIR) spectra of cells exposed to 5 contaminants in two levels (non-transformed and transformed). In this example, only two classes are being used for analysis: class 1 (cells exposed to benzo[a]pyrene (B[a]P) non-transformed, n = 59) and class 2 (cells exposed to B[a]P transformed, n = 62). The spectra at the fingerprint region (1800-900 cm^{-1}) was pre-processed by 2nd differentiation. The step-by-step analysis of this dataset using PCA-LDA/QDA is demonstrated in the Supplementary Method. Dataset 2, originally published by Paraskevaidi *et al.*[125], is also composed of two classes: class 1 containing 182 Raman spectra of blood plasma from healthy individuals, and class 2 containing 189 Raman spectra of blood plasma from ovarian cancer patients. The raw spectra were pre-processed by cutting the Raman fingerprint region (2000-500 cm^{-1}), followed by Savitzky-Golay 2nd differentiation (window of 21 points, 2nd order polynomial fit) and vector normalisation. Dataset 3 consists of 80 corn samples measured on 3 different NIR spectrometers. The spectra were acquired in the range between 1100 – 2498 nm at 2 nm intervals. The classification models for this dataset were based on the spectra collected by instrument 5 ('m5spec') where class 1 was defined as the samples with protein content ≤ 8.5 (n = 36), and class 2 the samples with protein content > 8.5 (n = 44). The spectra for this dataset were pre-processed by SNV. The raw spectra for datasets 1–3 are show in the Supplementary Information, Figure S1.

A PCA was applied for exploratory analysis of the pre-processed spectral data followed by an outlier detection algorithm. The PCA scores plot and Hotelling's T^2 *versus* Q residuals charts for datasets 1–3 are show in the Supplementary Information, Figure S2. For dataset 1, it is possible to identify a segregation trend between the samples from class 1 and 2 along PC1, where the samples from class 1 are mostly distributed on the left-side, and the samples from class 2 on the right-side of the PCA scores plot (Figure S2a). The Hotelling's T^2 *versus* Q residuals chart (Figure S2b) does not show any sample significantly far from the origin, thus no outlier is present in this dataset. In dataset 2, the PCA scores plot on PC1 *versus* PC2 do not show any clear segregation between the classes (Figure S2c); and the Hotelling's T^2 *versus* Q residuals chart (Figure S2d) indicates the presence of 4 outliers in class 2. These 4 spectra were removed from the dataset before further analysis. The PCA scores plot for dataset 3 (Figure S2e) show a separation trend along PC2, where the samples from class 1 are mostly on the positive side of the scores on PC2, and the samples from class 2 are mostly in the

negative side of the scores on PC2. The Hotelling's T^2 *versus* Q residuals chart for this dataset (Figure S2f) does not indicate the presence of outliers.

After data selection using the MLM algorithm (70% of samples for training and 30% of the samples for test), the mean-centred pre-processed spectra were used to build PCA-LDA, PCA-QDA and PLS-DA classification models. The training and cross-validation accuracies for datasets 1–3 using PCA-LDA and PLS-DA are show in Table 3.

Table 3. Training accuracies for PCA-LDA and PLS-DA algorithms applied to datasets 1–3. Cross-validation using venetian-blinds with 10 data splits. PCs stands for principal components; LVs stands for latent variables; and EV stands for cumulative explained variance.

Dataset	Algorithm	Number of factors	Training accuracy	Cross-validation accuracy
1	PCA-LDA	10 PCs (97% EV)	93%	90%
	PLS-DA	5 LVs (86% EV)	95%	92%
2	PCA-LDA	9 PCs (27% EV)	61%	61%
	PLS-DA	2 LVs (6% EV)	89%	72%
3	PCA-LDA	8 PCs (100% EV)	91%	84%
	PLS-DA	6 LVs (98% EV)	93%	88%

The optimal number of factors was selected by cross-validation (Figures S3–S5). Overall, the PLS-DA models are superior to the PCA-LDA models, which often happens since the PLS decomposition takes into consideration the reference classes labels for the training set in a way that the latent variables maximize the covariance between the samples, which emphasize the differences between the classes; while PCA decomposition only describe the variance in the data, which might not be totally related to class differences[42].

The performance of the discriminant analysis models in Table 3 applied to an external test set is depicted in Table 4. PLS-DA models show superior predictive performance, where higher accuracies, sensitivities and specificities are observed in the test set in comparison to PCA-LDA. The mean pre-processed spectrum and discriminant function plot for PLS-DA applied in datasets 1–3 are show in Figure 5. The PLS-DA regression coefficients and ROC curves are show in the Supplementary Information, Figures S6–S9.

Table 4. Test performance of PCA-LDA and PLS-DA models applied to datasets 1–3.

Dataset	Algorithm	Accuracy	Sensitivity	Specificity
1	PCA-LDA	86%	95%	78%
	PLS-DA	89%	95%	83%
2	PCA-LDA	67%	59%	75%
	PLS-DA	80%	75%	85%
3	PCA-LDA	83%	69%	100%
	PLS-DA	88%	77%	100%

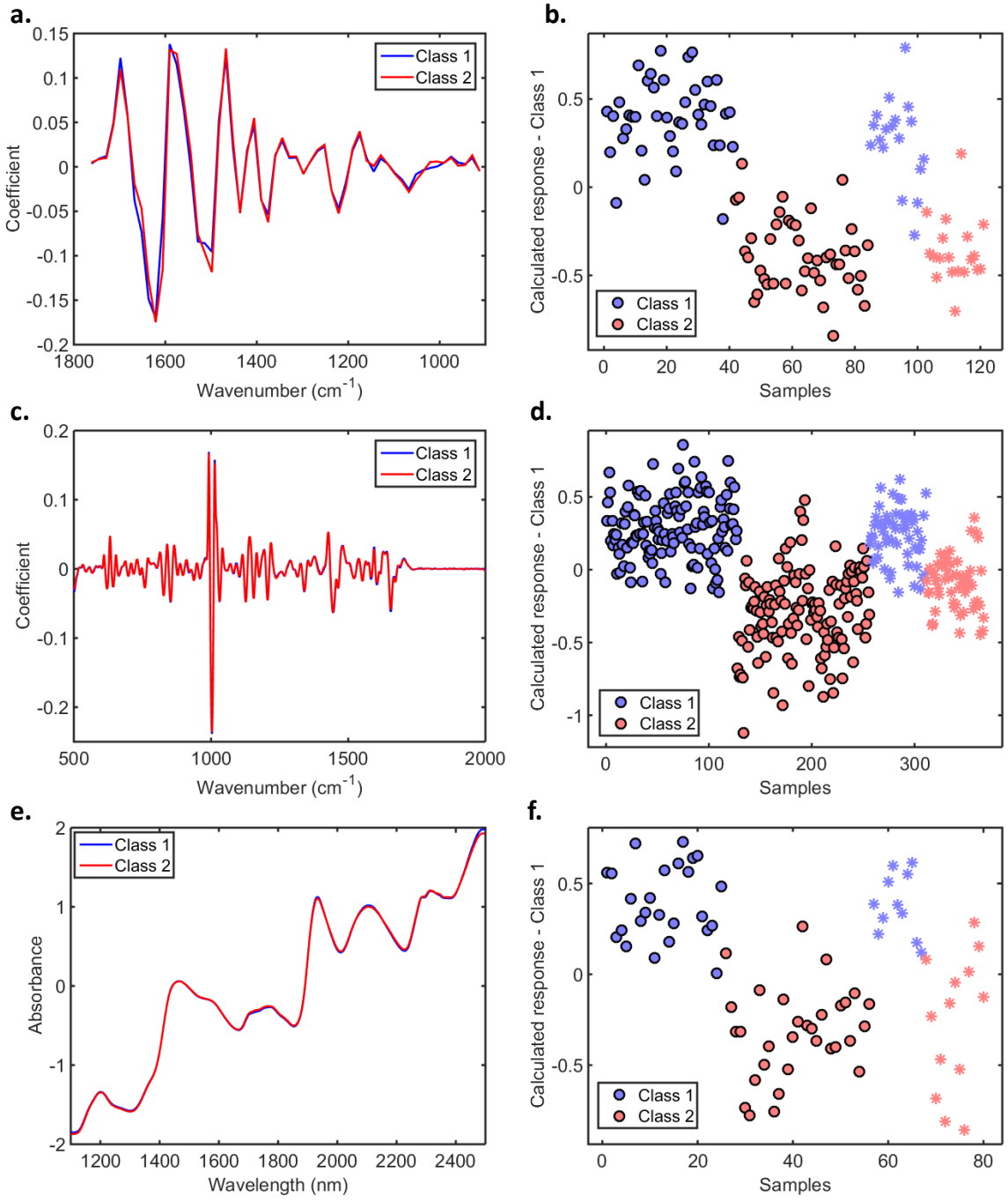

Figure 5. Results for PLS-DA models in datasets 1–3. (a) Mean pre-processed FTIR spectra (2nd derivative) for dataset 1; (b) calculated PLS-DA response for dataset 1, where o = training samples and * = test samples; (c) mean pre-processed Raman spectra (2nd Savitzky-Golay derivative (window of 21 points, 2nd order polynomial function) and vector normalisation) for dataset 2; (d) calculated PLS-DA response for dataset 2, where o = training samples and * = test samples; (e) mean pre-processed NIR spectra (SNV) for dataset 3; (f) calculated PLS-DA response for dataset 3, where o = training samples and * = test samples.

The classification performances of the PLS-DA models in Table 4 are satisfactory. Usually, in clinical applications, the minimum threshold for accuracy, sensitivity or specificity is 75%, the level often found in routine clinical procedures. The AUC values for the PLS-DA models in Table 4 were 0.99 (dataset 1), 0.96 (dataset 2) and 0.99 (dataset 3), indicating excellent predictive performance. Nevertheless, the classification performance of these models might improve by changing the type of pre-processing or by increasing the degree of complexity of the classification technique, such as by using feature selection techniques (*e.g.*, SPA and GA) or non-linear classifiers (*e.g.*, SVM and ANN). For sake of simplicity, herein only the results obtained by PCA-LDA and PLS-DA, which are the most common classification algorithms, are reported.

Acknowledgements

C.L.M.M. thanks Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Brazil (grant 88881.128982/2016-01) for financial support.

Author contributions

F.L.M. is the principal investigator who conceived and developed the idea for the article; C.L.M.M. performed the data analysis and wrote the manuscript; K.M.G.L and M.S. contributed with recommendations and provided feedback and changes to the manuscript; and C.L.M.M. and F.L.M. brought together the text and finalized the manuscript.

Competing interests

The authors declare no competing financial interest.

Data Availability

The datasets generated during and/or analysed during the current study are available in the IRootLab toolbox, http://trevisanj.github.io/irootlab/; in the Figshare repository, https://doi.org/10.6084/m9.figshare.6744206.v1; and in the Eigenvector Research repository, http://www.eigenvector.com/data/Corn/index.html.

Code Availability

The MATLAB code and instructions on how to process the data are present in the Supplementary Method.

References

1. Martin, F. L. et al. Distinguishing cell types or populations based on the computational analysis of their infrared spectra. *Nat. Protoc.* **5**, 1748–1760 (2010).
2. Santos, M. C. D., Morais, C. L. M., Nascimento, Y. M., Araujo, J. M. G. & Lima, K. M. G. Spectroscopy with computational analysis in virological studies: A decade (2006–2016). *Trends Anal. Chem.* **97**, 244–256 (2017).
3. Baker, M. J. et al. Using Fourier transform IR spectroscopy to analyze biological materials. *Nat. Protoc.* **9**, 1771–1791 (2014).
4. Movasaghi, Z., Rehman, S. & ur Rehman I. Fourier Transform Infrared (FTIR) Spectroscopy of Biological Tissues. *Appl. Spectrosc. Rev.* **43**, 134–179 (2008).
5. Kelly, J. G. et al. Biospectroscopy to metabolically profile biomolecular structure: a multistage approach linking computational analysis with biomarkers. *J. Proteome Res.* **10**, 1437–1448 (2011).
6. Paraskevaidi, M. et al. Differential diagnosis of Alzheimer's disease using spectrochemical analysis of blood. *Proc. Natl. Acad. Sci. U. S. A.* **114**, E7929–E7938 (2017).
7. Pasquini, C. Near infrared spectroscopy: A mature analytical technique with new perspectives – A review. *Anal. Chim. Acta* **1016**, 8–36 (2018).
8. Butler, H. J. et al. Using Raman spectroscopy to characterize biological materials. *Nat. Protoc.* **11**, 664–687 (2016).
9. Qu, J. H. et al. Applications of Near-infrared Spectroscopy in Food Safety Evaluation and Control: A Review of Recent Research Advances. *Crit. Rev. Food Sci. Nutr.* **55**, 1939–1954 (2015).
10. Scotter, C. Use of near infrared spectroscopy in the food industry with particular reference to its applications to on/in-line food processes. *Food Control* **1**, 142–149 (1990).

11. Prieto, N., Pawluczyk, O., Dugan, M. E. R. & Aalhus, J. L. A Review of the Principles and Applications of Near-Infrared Spectroscopy to Characterize Meat, Fat, and Meat Products. *Appl. Spectrosc.* **71**, 1403–1426 (2017).
12. Karoui, R., Downey, G. & Blecker, C. Mid-Infrared Spectroscopy Coupled with Chemometrics: A Tool for the Analysis of Intact Food Systems and the Exploration of Their Molecular Structure−Quality Relationships − A Review. *Chem. Rev.* **110**, 6144-6168 (2010).
13. Li-Chan, E. C. Y. The applications of Raman spectroscopy in food science. *Trends Food Sci. Tech.* **7**, 361–370 (1996).
14. Jin, H., Lu, Q., Chen, X., Ding, H., Gao, H. & Jin, S. The use of Raman spectroscopy in food processes: A review. *Appl. Spectrosc. Rev.* **51**, 12–22 (2015).
15. Bittner, L. K., Schonbichler, S. A., Bonn, G. K. & Huck, C. W. Near Infrared Spectroscopy (NIRS) as a Tool to Analyze Phenolic Compounds in Plants. *Curr. Anal. Chem.* **9**, 417–423 (2013).
16. Cozzolino, D. Use of Infrared Spectroscopy for In-Field Measurement and Phenotyping of Plant Properties: Instrumentation, Data Analysis, and Examples. *Appl. Spectrosc. Rev.* **49**, 564–584 (2014).
17. Buitrago, M. F., Skidmore, A. K., Groen, T. A., Hecker, C. A. Connecting infrared spectra with plant traits to identify species. *ISPRS J. Photogramm. Remote Sens.* **139**, 183–200 (2018).
18. Baranska, M., Roman, M., Dobrowlski, J. C., Schulz, H. & Baranski, R. Recent Advances in Raman Analysis of Plants: Alkaloids, Carotenoids, and Polyacetylenes. *Curr. Anal. Chem.* **9**, 108–127 (2013).
19. Quintelas, C., Mesquita, D. P., Lopes, J. A., Ferreira, E. C. & Sousa, C. Near-infrared spectroscopy for the detection and quantification of bacterial contaminations in pharmaceutical products. *Int. J. Pharm.* **492**, 199–206 (2015).
20. Naumann, D., Helm, D. & Labischinski, H. Microbiological characterizations by FT-IR spectroscopy. *Nature* **351**, 81–82 (1991).
21. Schmitt, J. & Flemming, H. C. FTIR-spectroscopy in microbial and material analysis. *Int. Biodeterior. Biodegratation* **41**, 1–11 (1998).
22. Rodriguez-Saona, L. E., Khambaty, F. M., Fry, F. S. & Calvey, E. M. Rapid Detection and Identification of Bacterial Strains By Fourier Transform Near-Infrared Spectroscopy. *J. Agric. Food Chem.* **49**, 574–579 (2001).
23. Zarnowiec, P., Lechowicz, Ł., Czerwonka, G. & Kaca, W. Fourier Transform Infrared Spectroscopy (FTIR) as a Tool for the Identification and Differentiation of Pathogenic Bacteria. *Curr. Med. Chem.* **22**, 1710–1718 (2015).
24. Jarvis, R. M. & Goodacre, R. Discrimination of Bacteria Using Surface-Enhanced Raman Spectroscopy. *Anal. Chem.* **76**, 40–47 (2004).
25. Stöckel, S., Kirchhoff, J., Neugebauer, U., Rösch, P. & Popp, J. The application of Raman spectroscopy for the detection and identification of microorganisms. *J. Raman Spectrosc.* **47**, 89–109 (2016).
26. Strola, S. A. et al. Single bacteria identification by Raman spectroscopy. *J. Biomed. Opt.* **19**, 111610 (2014).
27. Weiss, R. et al. Surface-enhanced Raman spectroscopy of microorganisms: limitations and applicability on the single-cell level. *Analyst* **144**, 943–953 (2019).
28. Lorenz, B., Wichmann, C., Stöckel, S., Rösch, P. & Popp, J. Cultivation-Free Raman Spectroscopic Investigations of Bacteria. *Trends Microbiol.* **25**, 413–424 (2017).
29. Sakudo, A. Near-infrared spectroscopy for medical applications: Current status and future perspectives. *Clin. Chim. Acta* **455**, 181–188 (2016).

30. De Bruyne, S., Speeckaert, M. M. & Delanghe, J. R. Applications of mid-infrared spectroscopy in the clinical laboratory setting. *Crit. Rev. Clin. Lab. Sci.* **55**, 1–20 (2018).
31. Bunaciu, A. A., Aboul-Enein, H. Y. & Fleschin, Ş. Vibrational Spectroscopy in Clinical Analysiserban Fleschin. *Appl. Spectrosc. Rev.* **50**, 176–191 (2014).
32. Pence, I. & Mahadevan-Jansen, A. Clinical instrumentation and applications of Raman spectroscopy. *Chem. Soc. Rev.* **45**, 1958–1979 (2016).
33. Baker, M. J. et al. Clinical applications of infrared and Raman spectroscopy: state of play and future challenges. *Analyst* **143**, 1735–1757 (2018).
34. Hibbert, D. B. Vocabulary of concepts and terms in chemometrics (IUPAC Recommendations 2016). *Pure Appl. Chem.* **88**, 407–443 (2016).
35. Mandel, J. Statistical methods in analytical chemistry. *J. Chem. Educ.* **26**, 534 (1949).
36. Wallace, R. M. Analysis of Absorption Spectra of Multicomponent Systems. *J. Phys. Chem.* **64**, 899–901 (1960).
37. Weber, G. Enumeration of Components in Complex Systems by Fluorescence Spectrophotometry. *Nature* **190**, 27–29 (1961).
38. Brereton, R. G. et al. Chemometrics in analytical chemistry—part I: history, experimental design and data analysis tools. *Anal. Bioanal. Chem.* **409**, 5891–5899 (2017).
39. Beebe, K. R., Pell, R. J. & Seasholtz, *Chemometrics: A Practical Guide* Vol. 4 (Wiley, New York, 1998).
40. Brereton, R. G. & Lloyd, G. R. Partial least squares discriminant analysis: taking the magic away. *J. Chemometr.* **28**, 213–225 (2014).
41. Jacyna, J., Kordalewska, M. & Markuszewski, M. J. Design of Experiments in metabolomics-related studies: An overview. *J. Pharm. Biomed. Anal.* **164**, 598–606 (2019).
42. Morais, C. L. M. et al. Standardization of complex biologically derived spectrochemical datasets. *Nat. Protoc.* **14**, 1546–1577 (2019).
43. Jones, S., Carley, S. & Harrison, M. An introduction to power and sample size estimation. *Emerg. Med. J.* **20**, 453–458 (2003).
44. Paraskevaidi, M. et al. Blood-based near-infrared spectroscopy for the rapid low-cost detection of Alzheimer's disease. *Analyst* **143**, 5959–5964 (2018).
45. Pavia, D. L., Lampman, G. M., Kriz, G. S. & Vyvyan, J. A. *Introduction to Spectroscopy* (Cengage Learning, Belmont, CA, 2008).
46. Hu, Q., Lü, X., Lu, W., Chen, Y. & Liu H. An extensive study on Raman spectra of water from 253 to 753 K at 30 MPa: A new insight into structure of water. *J. Mol. Spectrosc.* **292**, 23–27 (2013).
47. Callery, E. L. et al. New approach to investigate Common Variable Immunodeficiency patients using spectrochemical analysis of blood. *Sci. Rep.* **9**, 7239 (2019).
48. Seasholtz, M. B. & Kowalski, B. The parsimony principle applied to multivariate calibration. *Anal. Chim. Acta* **277**, 165–177 (1993).
49. Tfayli, A., Gobinet, C., Vrabie, V., Huez, R., Manfait, M. & Piot, O. Digital Dewaxing of Raman Signals: Discrimination between Nevi and Melanoma Spectra Obtained from Paraffin-Embedded Skin Biopsies. *Appl. Spectrosc.* **63**, 564–570 (2009).
50. de Lima, F. A. et al. Digital de-waxing on FTIR images. *Analyst* **142**, 1358–1370 (2017).
51. Ibrahim, O. et al. Improved protocols for pre-processing Raman spectra of formalin fixed paraffin preserved tissue sections. *Anal. Methods* **9**, 4709–4717 (2017).
52. Meksiarun, P. et al. Comparison of multivariate analysis methods for extracting the paraffin component from the paraffin-embedded cancer tissue spectra for Raman imaging. *Sci. Rep.* **7**, 44890 (2017).

53. Bassan, P., Mellor, J., Shapiro, J., Williams, K. J., Lisanti, M. P. & Gardner, P. Transmission FT-IR Chemical Imaging on Glass Substrates: Applications in Infrared Spectral Histopathology. *Anal. Chem.* **86**, 1648–1653 (2014).
54. Savitzky, A. & Golay, M. J. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.* **36**, 1627–1639 (1964).
55. Brown, C. D. & Wentzell, P. D. Hazards of digital smoothing filters as a preprocessing tool in multivariate calibration. *J. Chemometr.* **13**, 133–152 (1999).
56. Geladi, P., MacDougall, D. & Martens, H. Linearization and scatter-correction for near-infrared reflectance spectra of meat. *Appl. Spectrosc.* **39**, 491–500 (1985).
57. Barnes, R., Dhanoa, M. S. & Lister, S. J. Standard normal variate transformation and de-trending of near-infrared diffuse reflectance spectra. *Appl. Spectrosc.* **43**, 772–777 (1989).
58. Bassan, P., Byrne, H. J., Bonnier, F., Lee, J., Dumas, P. & Gardner, P. Resonant Mie scattering in infrared spectroscopy of biological materials – understanding the 'dispersion artefact'. *Analyst* **134**, 1586–1593 (2009).
59. Bassan, P. et al. Resonant Mie Scattering (RMieS) correction of infrared spectra from highly scattering biological samples. *Analyst* **135**, 268–277 (2010).
60. Kiefer, W. et al. Raman-Mie scattering from single laser trapped microdroplets. *J. Mol. Struct.* **408–409**, 113–120 (1997).
61. Liland, K. H., Kohler, A. & Afseth, N. K. Model-based pre-processing in Raman spectroscopy of biological samples. *J. Raman Spectrosc.* **47**, 643–650 (2016).
62. Hastie, T., Tibshinari, R. & Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* 2nd edn (Springer, New York, 2009).
63. Martens, H. & Martens, M. Modified Jack-knife estimation of parameter uncertainty in bilinear modelling by partial least squares regression (PLSR). *Food Qual. Prefer.* **11**, 5–16 (2000).
64. Rousseeuw, P. J. & Hubert, M. Robust statistics for outlier detection. *Wiley Interdiscip. Rev. Data Min.* **1**, 73–79 (2011).
65. Jiang, F., Liu, G., Du, J. & Sui, Y. Initialization of K-modes clustering using outlier detection techniques. *Inf. Sci.* **332**, 167–183 (2016).
66. Bakeev, K. A. *Process Analytical Technology: Spectroscopic Tools and Implementation Strategies for the Chemical and Pharmaceutical Industries* 2nd edn (John Wiley & Sons, Chichester, UK, 2010).
67. Morais, C. L. M., Santos, M. C. D., Lima, K. M. G. & Martin, F. L. Improving data splitting for classification applications in spectrochemical analyses employing a random-mutation Kennard-Stone algorithm approach. *Bioinformatics* btz421 (2019). Doi: 10.1093/bioinformatics/btz421
68. Kennard, R. W. & Stone, L. A. Computer Aided Design of Experiments. *Technometrics* **11**, 137–148 (1969).
69. Bro, R. & Smilde, A. K. Principal component analysis. *Anal. Methods* **6**, 2812–2831 (2014).
70. Martin, F. L. et al. Identifying variables responsible for clustering in discriminant analysis of data from infrared microspectroscopy of a biological sample. *J. Comput. Biol.* **14**, 1176–1184 (2007).
71. Wold, S. & Sjöström, M. SIMCA: A Method for Analyzing Chemical Data in Terms of Similarity and Analogy. in *Chemometrics: Theory and Application* (ed. Kowalski, B. R.) 243–282 (American Chemical Society, Washington, 1977).
72. Marini, F. Classification Methods in Chemometrics. *Curr. Anal. Chem.* **6**, 72–79 (2010).
73. Pomerantsev, A. L. Acceptance areas for multivariate classification derived by projection methods. *J. Chemometr.* **22**, 601–609 (2008).

74. Dixon, S. J. & Brereton, R. G. Comparison of performance of five common classifiers represented as boundary methods: Euclidean Distance to Centroids, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Learning Vector Quantization and Support Vector Machines, as dependent on data structure. *Chemom. Intell. Lab. Syst.* **95**, 1–17 (2009).
75. Wu, W. et al. Comparison of regularized discriminant analysis linear discriminant analysis and quadratic discriminant analysis applied to NIR data. *Anal. Chim. Acta* **329**, 257–265 (1996).
76. Morais, C. L. M. & Lima, K. M. G. Principal Component Analysis with Linear and Quadratic Discriminant Analysis for Identification of Cancer Samples Based on Mass Spectrometry. *J. Braz. Chem. Soc.* **29**, 472–481 (2018).
77. Morais, C. L. M., Lima, K. M. G. & Martin, F. L. TTWD-DA: A MATLAB toolbox for discriminant analysis based on trilinear three-way data. *Chemom. Intell. Lab. Syst.* **188**, 46–53 (2019).
78. Geladi, P. & Kowalski, B. R. Partial least-squares regression: a tutorial. *Anal. Chim. Acta* **185**, 1–17 (1986).
79. de Jong, S. SIMPLS: An alternative approach to partial least squares regression. *Chemom. Intell. Lab. Syst.* **18**, 251–263 (1993).
80. Wold, S., Sjöström, M. & Eriksson, L. PLS-regression: a basic tool of chemometrics. *Chemom. Intell. Lab. Syst.* **58**, 109–130 (2001).
81. Pomerantsev, A. L. & Rodionova, O. Ye. Multiclass partial least squares discriminant analysis: Taking the right way—A critical tutorial. *J. Chemometr.* **32**, e3030 (2018).
82. Pérez, N. F., Ferré, J. & Boqué, R. Calculation of the reliability of classification in discriminant partial least-squaresbinary classification. *Chemom. Intell. Lab. Syst.* **95**, 122–128 (2009).
83. Cover, T. & Hart, P. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**, 21–27 (1967).
84. Cortes, C. & Vapnik, V. Support-Vector Networks. *Mach. Learn.* **20**, 273–297 (1995).
85. Brereton, R. G. & Lloyd, G. R. Support Vector Machines for classification and regression. *Analyst* **135**, 230–267 (2010).
86. Marini, F., Bucci, R., Magrì, A. L. & Magrì, A. D. Artificial neural networks in chemometrics: History, examples and perspectives. *Microchem. J.* **88**, 178–185 (2008).
87. Fawagreh, K., Gaber, M. M. & Elyan, R. Random forests: from early developments to recent advancements. *Syst. Sci. Control Eng.* **2**, 602–609 (2014).
88. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
89. Yang, Q., Zhang, L., Wang, L. & Xiao, H. MultiDA: Chemometric software for multivariate data analysis based on Matlab. *Chemom. Intell. Lab. Syst.* **116**, 1–8 (2012).
90. De Gussem, K., De Gelder, J., Vandenabeele, P. & Moens, L. The Biodata toolbox for MATLAB. *Chemom. Intell. Lab. Syst.* **95**, 49–52 (2009).
91. Cordella, C. B. Y. & Bertrand, D. SAISIR: A new general chemometric toolbox. *Trends Anal. Chem.* **54**, 75–82 (2014).
92. Rossel, R. A. V. ParLeS: Software for chemometric analysis of spectroscopic data. *Chemom. Intell. Lab. Syst.* **90**, 72–83 (2008).
93. Reisner, L. A., Cao, A. & Pandya, A. K. An integrated software system for processing, analyzing, and classifying Raman spectra. *Chemom. Intell. Lab. Syst.* **105**, 83–90 (2011).
94. Jing, R., Sun, J., Wang, Y., Li, M. & Pu, X. PML: A parallel machine learning toolbox for data classification and regression. *Chemom. Intell. Lab. Syst.* **138**, 1–6 (2014).
95. Zontov, Y. V., Rodionova, O. Ye., Kucheryavskiy, S. V. & Pomerantsev, A. L. DD-SIMCA – A MATLAB GUI tool for data driven SIMCA approach. *Chemom. Intell. Lab. Syst.* **167**, 23–28 (2017).

96. Li, H. D., Xu, Q. S. & Liang, Y. Z. libPLS: An integrated library for partial least squares regression and linear discriminant analysis. *Chemom. Intell. Lab. Syst.* **176**, 34–43 (2018).
97. Chang, C. C. & Lin, C. J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**, 27:1–27:27 (2011).
98. Alsberg, B. K. & Hagen, O. J. How octave can replace Matlab in chemometrics. *Chemom. Intell. Lab. Syst.* **84**, 195–200 (2006).
99. Wehrens, R. *Chemometrics with R: Multivariate Data Analysis in the Natural Sciences and Life Sciences* (Springer, New York, 2011).
100. Varmuza, K. & Filzmoser, P. *Introduction to Multivariate Statistical Analysis in Chemometrics* 1st ed. (CRC Press, Boca Raton, 2009).
101. Jarvis, R. M., Broadhurst, D., Johnson, H., O'Boyle, N. M. & Goodacre, R. PYCHEM: a multivariate analysis package for python. *Bioinformatics* **22**, 2565–2566 (2006).
102. Trevisan, J. et al. IRootLab: a free and open-source MATLAB toolbox for vibrational biospectroscopy data analysis. *Bioinformatics* **29**, 1095–1097 (2013).
103. Ballabio, D. & Consonni, V. Classification tools in chemistry. Part 1: linear models. PLS-DA. *Anal. Methods* **5**, 3790–3798 (2013).
104. Ferrés, M., Platikanov, S., Tsakovski, S. & Tauler, R. Comparison of the variable importance in projection (VIP) and of the selectivity ratio (SR) methods for variable selection and interpretation. *J. Chemometr.* **29**, 528–536 (2015).
105. Nørgaard, L. et al. Interval Partial Least-Squares Regression (iPLS): A Comparative Chemometric Study with an Example from Near-Infrared Spectroscopy. *Appl. Spectrosc.* **54**, 413–419 (2000).
106. Brown, C. D. & Green, R. L. Critical factors limiting the interpretation of regression vectors in multivariate calibration. *Trends Anal. Chem.* **28**, 506–514 (2009).
107. de Juan, A. & Tauler, R. Multivariate Curve Resolution (MCR) from 2000: Progress in Concepts and Applications. *Crit. Rev. Anal. Chem.* **36**, 163–176 (2006).
108. Jaumot, J., de Juan, A. & Tauler, R. MCR-ALS GUI 2.0: New features and applications. *Chemom. Intell. Lab. Syst.* **140**, 1–12 (2015).
109. de Juan, A. et al. Spectroscopic imaging andchemometrics: a powerfulcombination for global andlocal sample analysis. *Trends Anal. Chem.* **23**, 70–79 (2004).
110. Radovic, M., Ghalwash, M., Filipovic, N. & Obradovic, Z. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC Bioinformatics* **18**, 9 (2017).
111. Soares, S. F. C., Gomes, A. A., Araujo, M. C. U., Galvão Filho, A. R. & Galvão, R. K. H. The successive projections algorithm. *Trends Anal. Chem.* **42**, 84–98 (2013).
112. Theophilou, G. et al. Synchrotron- and focal plane array-based Fourier-transform infrared spectroscopy differentiates the basalis and functionalis epithelial endometrial regions and identifies putative stem cell regions of human endometrial glands. *Anal. Bioanal. Chem.* **410**, 4541–4554 (2018).
113. McCall, J. Genetic algorithms for modelling and optimisation. *J. Comput. Appl. Math.* **184**, 205–222 (2005).
114. Siqueira, L. F. S. & Lima, K. M. G. MIR-biospectroscopy coupled with chemometrics in cancer studies. *Analyst* **141**, 4833–4847 (2016).
115. Siqueira, L. F. S. & Lima, K. M. G. A decade (2004 – 2014) of FTIR prostate cancer spectroscopy studies: An overview of recent advancements. *Trends Anal. Chem.* **82**, 208–221 (2016).

116. Siqueira, L. F. S., Morais, C. L. M., Araújo Júnior, R. F., de Araújo, A. A. & Lima, K. M. G. SVM for FT-MIR prostate cancer classification: An alternative to the traditional methods. *J. Chemometr.* **32**, e3075 (2018).
117. Morais, C. L. M. & Lima, K. M. G. Comparing unfolded and two-dimensional discriminant analysis and support vector machines for classification of EEM data. *Chemom. Intell. Lab. Syst.* **170**, 1–12 (2017).
118. Siqueira, L. F. S., Araújo Júnior, R. F., de Araújo, A. A., Morais, C. L. M. & Lima, K. M. G. LDA vs. QDA for FT-MIR prostate cancer tissue classification. *Chemom. Intell. Lab. Syst.* **162**, 123–129 (2017).
119. Warrens, M. J. Cohen's kappa is a weighted average. *Stat. Methodol.* **8**, 473–484 (2011).
120. Morais, C. L. M., Lima, K. M. G. & Martin, F. L. Uncertainty estimation and misclassification probability for classification models based on discriminant analysis and support vector machines. *Anal. Chim. Acta* **1063**, 40–46 (2019).
121. Rocha, W. F. C. & Sheen, D. A. Classification of biodegradable materials using QSAR modelling with uncertainty estimation. *SAR QSAR Environ. Res.* **27**, 799–811 (2016).
122. de Almeida, M. R., Correa, D. N., Rocha, W. F. C., Scafi, F. J. O. & Poppi, R. J. Discrimination between authentic and counterfeit banknotes using Raman spectroscopy and PLS-DA with uncertainty estimation. *Microchem. J.* **109**, 170–177 (2013).
123. Allegrini, F. & Olivieri, A. C. Sensitivity, Prediction Uncertainty, and Detection Limit for Artificial Neural Network Calibrations. *Anal. Chem.* **88**, 7807–7812 (2016).
124. Trevisan, J. et al. Syrian hamster embryo (SHE) assay (pH 6.7) coupled with infrared spectroscopy and chemometrics towards toxicological assessment. *Analyst* **135**, 3266–3272 (2010).
125. Paraskevaidi, M. et al. Raman spectroscopic techniques to detect ovarian cancer biomarkers in blood plasma. *Talanta* **189**, 281–288 (2018).