Microchemical Journal 206 (2024) 111550

Contents lists available at ScienceDirect

Microchemical Journal

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

Differentiating five agrochemicals used in the treatment of intact olives by means of NIR spectroscopy, discriminant analysis and compliant class models

D. Castro-Reigía [a,b], I. García [b], S. Sanllorente [a], L.A. Sarabia [c], M.C. Ortiz [a,*]

[a] Departamento de Química, Facultad de Ciencias, Universidad de Burgos, Plaza Misael Bañuelos s/n, Burgos 09001, Spain
[b] AOTECH (Advanced Optical Technologies S.L.), Edificio Ilgner-BIC Bizkaia Ezkerraldea, Módulo C-7, Avda. Altos Hornos de Vizcaya, 33, 48901, Barakaldo, Bizkaia, Spain
[c] Departamento de Matemáticas y computación, Facultad de Ciencias, Universidad de Burgos, Plaza Misael Bañuelos s/n, Burgos 09001, Spain

ARTICLE INFO

Keywords:
ECOC
PLS-DA
AIHMBU
DMCEN
NIR spectroscopy
Sensitivity
Specificity

ABSTRACT

This paper deals with the application of near infrared spectroscopy (NIR) in a classification problem involving multiple classes in order to differentiate contaminated olives. A total of 452 samples, ripe and unripe, were treated with five different agrochemicals reproducing the traditional fumigation process in the olive tree. The main objective was to differentiate through a classification if the samples were or were not treated, but also, which chemical was used for each olive. Firstly, Partial Least Squares-Discriminant Analysis (PLS-DA) was performed to differentiate between untreated and treated samples. Then, two novel chemometric approaches, a classification one and a modelling one, were applied for ripe and unripe olives, achieving good results and determining with which chemical were the olives sprinkled with. For the classification of the samples in the six different classes (untreated olives, or treated with one of the five agrochemicals), an Automatic Hierarchical Model Builder (AHIMBU) was used, applying sequential binary PLS-DAs. Nevertheless, for the modelling approach, a compliant model, PLS2-CM, also based on PLS, was used with two different codifications for the classes: i) the classic and well-known One Versus All (OVA), and ii) the Error Correction Output Code (ECOC) optimal matrix. The final global results were evaluated using the Diagonal Modified Confusion Entropy (DMCEN) index, which ranges between 0 and 1, and is very sensitive to changes in the sensitivity–specificity matrices (note that the lower the DMCEN, the better the classification is). The best DMCEN value in prediction for unripe olives, 0.4898, was obtained for the PLS2-CM-ECOC, while 0.6937 and 0.7705 DMCEN values were obtained for AHIMBU and PLS2-CM-OVA, respectively. For the case of the ripe samples, the DMCEN values in prediction were better than the ones for the unripe olives: 0.6016, 0.5051, and 0.4166, for AHIMBU, PLS2-CM-OVA and PLS2-CM-ECOC, respectively. In every case, the best DMCEN has been obtained with the PLS2-CM-ECOC procedure.

1. Introduction

Nowadays, modern analytical instruments such as near infrared (NIR) spectroscopy are able to generate huge amounts of information in short time for a large number of samples. Pattern recognition techniques use this information in order to classify samples in different categories or classes [1]. In particular, in this work, it is aimed to assess the feasibility of using NIR spectroscopy to classify olives that have been or have not been sprinkled with different insecticides, fungicides and herbicides.

Olive oil is considered to be a high-quality food due to its nutritional properties and health benefits. Its consumption in Spain rises to more than six liters per capita in 2023 according to the Ministerio de Agricultura, Pesca y alimentación [2]. Therefore, it is essential to ensure its safety and improve the production processes, avoiding possible losses, starting from the olives. Climate change has caused huge harvest losses in the last recent years and that is the reason why the use of fungicides, herbicides and insecticides is on the rise. Despite there are precise regulations to restrict and control these kinds of compounds, they can persist as a residue in the olives, making a health risk. Regarding that, Maximum Residue Limits (MRLs) have been established by the European Union (EU), among other authorities, both for olives and for olive oil. The chemical compounds of interest in this work are diflufenican (DF),

* Corresponding author.
E-mail address: mcortiz@ubu.es (M.C. Ortiz).

https://doi.org/10.1016/j.microc.2024.111550
Received 2 August 2024; Received in revised form 29 August 2024; Accepted 30 August 2024
Available online 6 September 2024

deltamethrin (DM), *lambda*-Cyhalothrin (LC), oxyfluorfen (OF), and Tebuconazole (TB). Traditional methods for detecting residuals of this kind of chemicals are usually based on chromatographic techniques [3,4], but they are time and solvent consuming and furthermore, a sample treatment is required. So, fast and efficient analytical methods that allow in-situ decisions would be of great interest, regarding that, NIR spectroscopy has become a promising alternative. Agrochemical residues have the tendency to be present in food at concentration levels of a very small amount of ppm, nevertheless NIR spectroscopy has been proven to have the ability of detecting low concentrations of analytes [5] and has also been previously used in the intact olives analysis [6,7].

Within this context, the five aforementioned compounds are going to be analyzed using NIR spectroscopy to evaluate their presence in intact olives in this work. The aims were, in a first step, to differentiate between chemically treated and non-treated olives, while in the second step, to evaluate the possibility of differentiating which agrochemical has been used in the treatment of the olives. Regarding this task, there are two possibilities: i) to build a one-class model, for example, a SIMCA model [1] using exclusively the spectra of the untreated olives, or ii) to use also the spectra of the treated ones. In the first case, the specificity of the model (ability to correctly reject treated olives) can not be estimated and considered at the stage of its construction. It can only be evaluated once the model has been built. On the contrary, in the second case, since the spectra of the treated olives are also used, the specificity is partof the figure of merit that is optimized in the training stage.

In that second stage, there are six classes of objects, one corresponding to the untreated olives, and the other five, consisting of treated olives with each one of the five agrochemicals. Consequently, the problem of deciding if an olive has been treated (and in each case, with which agrochemical) or not is a multi-class problem. From a chemometric perspective, the problem can be addressed using a classification method (discriminant analysis, DA) or a modelling one (class modelling, CM). A DA assigns each spectrum to a single class, and all of them must be assigned, that is, it discriminates the spectra. Nevertheless, in a CM, an object can be inside one or more class-models, and even outside all of them. CM methods are called "compliant class-models" [8] if the data from the remaining classes are used to build the model of each class, as is the case in this work.

Partial Least Squares (PLS) is a multivariate regression method widely used in chemometrics [9]. It aims to represent a data set with n samples and p variables in a subspace spanned by a latent variables (LVs), $T = \{t_1, t_2 \ldots, t_a\}$ which are linear combinations of the p original variables and have the greatest variance and a high correlation with the response. PLS has been used both as a discriminant method (PLS-DA) [10,11] and as a modelling method (PLS2-CM) [12,13]. In both cases, n objects in $\mathbf{X}$ (predictor variables) are known to belong to a class of interest (or not), while $\mathbf{y}$ (responses) can be codified in different ways depending on the method that is wanted to be used.

When PLS-DA is used to discriminate k-classes, it is necessary to code the class to which each spectrum belongs using a vector that is the response to be predicted by the PLS regression. One option is coding a vector of k components, one for each class, using a one when the spectrum belongs to the class, and zero in the others. The numerical values are actually irrelevant, -1 is also used instead of zero. In chemometrics, this is the usual coding of classes and is called OVA (one versus all) [14] because each one of the responses is a binary function (binary learner) with which one class is discriminated from the others. This PLS-DA implementation builds a single multi-response (the k binary learners) PLS model.

The class coding issue in chemometrics is not as widespread as in the engineering and telecommunication field, where it has been studied in applications of classification methods [15,16]. Error Correcting Output Coding (ECOC) matrices allow a communication channel (in this case, in a DA or CM method) to reduce errors in the decoding process (when the method assigns a class to a new object). In ref. [17] it has been proposed for the first time the use of ECOC matrices in a CM based on PLS2. With the training set, the number of LV in the PLS and the ECOC matrix are optimized, thus ensuring that the coding/decoding process adapts to the characteristics of the data set under study. PLS2-CM along with ECOC has been also successfully used in the detection of defects in the kneading process of biscuit doughs using NIR spectroscopy [18] and with electromyographic signals regarding hand movements [19].

Sometimes there is a natural order in the classes, as occurs in the classification of loaves of bread into unfermented, fermented, and overfermented [20], but that is not the present case with the olives (where no logical order exists between the agrochemicals used). The number of different possible arrangements of the classes to make a hierarchical model grows exponentially with the number of classes, and this makes the search for an optimal sequence notably difficult. Precisely, the most recent advance in k-class discrimination with PLS-DA is the proposal of an Automatic Hierarchical Classification Model Builder (AHIMBU), which efficiently solves the problem of selecting the order of the classes [21]. AHIMBU was also used to generate a classification structure of the raw materials for the tyre industry [22]. Yet, it still remains the structural problem of the modification of the training set in each stage. Globally, the k-class problem is solved, at least, with k-1 binary PLS-DA models, since in some nodes, AHIMBU can differentiate between two groups of classes.

The performance of any classification method, whether DA or CM, is evaluated by the sensitivity and specificity of each class. Sensitivity is the probability of correctly accepting the objects of the class, while specificity is the probability of correctly rejecting objects that do not belong to the class. In this specific case, it is of great interest to know the specificity of the models of each class regarding the other ones, because a different overlap is expected depending on the agrochemical used. If k-classes are considered, the sensitivity–specificity values, define a square matrix $k \times k$, that includes on the principal diagonal the sensitivities of each class, and outside that diagonal, the specificities of each one regarding the rest of the classes. Knowing that in this problem 36 values are provided (30 specificities and six sensitivity values) it is necessary to use a global index that summarizes them to evaluate the results of the DA or CM method. There are numerous proposals for this purpose, as can be consulted in ref. [23] and ref. [24], but recently, a new index regarding this matter, DMCEN (Diagonally Modified Confusion Entropy), has been proposed [25]. DMCEN analyses the sensitivity and specificity matrix targeting the reduction of the disorder in the assignment of objects to the classes that originates a DA or a CM. This is done by measuring the entropy (amount of information) that the model provides when assigning objects to classes. DMCEN is much more sensitive to changes in the elements of a sensitivity and specificity matrix than the typical indices, and its value ranges from 0 to 1. The performance of a DA or a CM will be as better as the lowest the DMCEN value is. Furthermore, it is consistent with the total efficiency [16], a widely used index in chemometrics to evaluate the performance of these methods.

DMCEN allows evaluating the sensitivity and specificity matrices obtained with a DA or a CM method, both in training and in prediction. In fact, this index is the fitting criteria used in the PLS2-CM in this work. Using DMCEN, the optimal number of LV of the PLS2 model and the optimal ECOC matrix for the response are selected.

From a user's perspective, when applying a PLS-DA or a PLS2-CM model to an unknow sample, in both cases the PLS model defines, in the spectral space, a bounded enclosure (PLS-Box) by means of the statistics Q residual and T^2 in such a way that, if the new spectrum is outside this space, the model should not be applied to it. In other words, a spectrum significantly different from those used in the training stage will not be assigned to any class. Note that by construction, the PLS-Box is similar to the SIMCA-Box since it handles latent variables and therefore, the residual regarding the projection space and the T^2 distance in the projection space is calculated. But unlike SIMCA, the PLS latent variables are related to the class to which the spectra belong to. This property of the regression methods that build a subspace of the predictor space makes them very efficient in classification and modelling tasks

compared to others, such as neural networks, which relate the spectra to the class code without projecting onto any subspace of the predictors.

When a new spectrum is similar enough to the training ones, a PLS-DA model will necessarily assign it to one (and only one) of the classes of the training set. On the contrary, a PLS-CM model, can assign it to one of the classes, to several, or to none of them depending on its similarity with the training ones. In these last two cases, an alert is available, for example, if several agrochemicals from the training set had been used, this spectrum should be assigned to several classes, whereas if the agrochemical is different from all of them, it should not be assigned to any. Therefore, the advantage of the PLS-CM compared to the PLS-DA is that it enables to have more information regarding what happens with a future sample.

In this work, after a descriptive analysis through Principal Components Analysis (PCA), the recently developed AHIMBU discrimination and PLS2-CM modelling methods (both with the classical coding of the responses, PLS2-CM-OVA, and coding using the optimal ECOC matrices, PLS2-CM-ECOC) will be applied to the problem of distinguishing the six classes of olives: untreated or treated with one out of the five agrochemicals.

2. Samples and methods

2.1. Olives and sample preparation

Around 1000 samples (intact olives of *Picual* variety) were harvested over two seasons in Spain and were collected to carry out this study, having ripe and unripe olives corresponding to each harvest (see Figure S1 in the supplementary material for the morphological differences between them). Different chemical treatments were applied to both of them, ripe and unripe, by spiking the olives with different level concentrations of five different chemicals (DF, DM, LC, OF and TB, whose chemical structure and further information, like their CAS and their MRL, can be found in Figure S2 of the supplementary material) reproducing the traditional fumigation process in the olive tree. In practice, the concentration of the agrochemical depends on multiple environmental factors which are different for each harvest season. However, it is necessary to comply with the established MRLs when commercializing olives or their derivatives. To do so, five different solutions were prepared individually (without any mixture) in five different concentrations of 1 %, 0.1 %, 0.01 %, 0.0001 % and 0.00001 % (w/v), and were then sprinkled on the olives. For the case of the ripe ones, the most diluted solution was not used. Also, some samples were non-treated in order to carry out the study using them as a reference.

After that, the samples were divided into two groups, one set aside for its analysis by chromatography, and the other one set aside for the study developed in this work. Concretely, a total of 452 samples were used for NIR analysis, 240 of them corresponding to unripe olives, and 212 to ripe ones. Table S1 in the supplementary material shows the untreated number of samples and the number of samples treated with each agrochemical for the analysis, both for ripe and unripe olives. For the case of the unripe olives DF, LC and TB were sprinkled on 50 of them (ten samples at each concentration level), whereas DM and OF were sprinkled on just 40 samples (ten samples at each concentration level, with the exception of the 0.0001 % concentration level). Both for ripe and unripe samples, 10 untreated samples were used in the analysis. On the other hand, for ripe olives, all the chemicals were sprinkled on 40 samples, having ten samples for each concentration level (remember that the most diluted solution was not employed in this case).

2.2. Instrumental. NIR spectrophotometer configuration and measurements

The experiments were carried out at-line in an industry of the olive oil sector in Spain. After the previous chemical treatment of the samples, they were analyzed by NIR spectroscopy. In particular, the experimental procedure was made with the AONIR integrated solution for real-time NIR measurements from AOTECH S.L. [26], including a NIR sensor, a measurement platform, and the precise software to integrate the hardware with the model outcome for real-time measurements.

A total of eight spectra were recorded for each sample, collecting them in four different zones of the olive, and taking two instrumental replicates in each spot. In that way, and considering the explanations given in the previous section, a total of 3618 spectra were registered, 1920 for the unripe olives, and 1698 for the ripe ones (in the case of C_5, two additional spectra were collected for one of the samples). For further details, Table S2 in the supplementary material should be consulted.

The spectrometer was configured manually so that NIR reflectance was measured in a wavelength range from 900 to 1670 nm (125 wavelengths, accounting for a spectral resolution of 6 nm), having 50 readings per spectrum with an integration time of 10.8 ms.

Half of the samples were collected (after the sprinkle procedure) and taken to an external accredited laboratory, to be measured by GC–MS-MS QqQ to ensure that the sprinkled olive procedure was as expected.

2.3. Theoretical aspects

In this work, the previously summarized approaches in the introduction section are employed to achieve the best possible classification of the olive samples according to the chemical treatment used. On the one hand, the binary PLS-DA is used to differentiate between treated and untreated olives, and on the other hand, AHIMBU and PLS2-CM as a multiclass DA and CM, respectively were used to differentiate between the six classes, C_i, corresponding each one of them to samples sprinkled with five different agrochemicals: C_1, DF; C_2, DM; C_3, LC; C_4, OF; C_5, TB; and the last one, C_6, corresponding to untreated olives. In this way, not only the differences between untreated and treated olives are studied, but also, the five chemical treatments used in each case.

To ease the reading of the paper, the elements that are going to be handled are explained below, with the exception of PCA, whose fundamental aspects and applicability are well-known [27]. The main theoretical aspects of the chemometric strategies that are going to be used are described in detail in refs. [11,17,21,25], but the major aspects applied to their use in the present work can be summarized as follows.

2.3.1. Partial least Squares discriminant analysis (two class problem)

PLS-DA is a commonly used strategy, and its application is more widespread regarding AHIMBU or PLS2-CM. As explained in the introduction section, PLS-DA is a discriminant method where a classification rule is made to assign the objects to one class. In PLS-DA, PLS is used to build a model that predicts the class code for each sample. Two procedures can be used to solve this task: i) to build a response variable, y, assigning a 1 to the objects of the C_1 class and a 0 to the objects of the C_2, or ii) to use the OVA matrix with two binary learners, f_1 and f_2, which are the indicator functions of the class C_1 and C_2, respectively [24,28]. The latter procedure uses the fact that PLS is a multivariate regression that deals simultaneously with many responses and tends to build linear combinations with great variance in the **X**-block and **Y**-block, in such a way that the model is prone to the separation of the k-classes in terms of variance. Also, PLS maximizes the correlation between the LV of the **X**-block with those of the **Y**-block, and as a consequence, PLS tends to avoid LV of the **X**-block that do not include information of the **Y**-block (information about the samples classification). Working with just two classes, the OVA matrix has a dimension 2×2, and the (1,0) or (0,1) response vector is assigned to each object depending on the class it belongs to (C_1 or C_2 respectively). Thus, PLS can be performed for this multivariate **Y**.

The fitted PLS model is formed by two binary learners, f_1 and f_2. With the fitted values $\widehat{f}_1(x)$ and $\widehat{f}_2(x)$ for every $x \in X$ a probability distribution is built and a critical value, CV_j, ($j = 1,2$), is established in such a way that the sensitivity and the specificity values of each class are equal.

Section 3 in the supplementary material shows in detail that the sum of the two *CVs* is equal to 1. Furthermore, the normal distributions fitted to each class satisfy certain mathematical relations. As a result, the sensitivity of the C_1 class is the specificity of C_2 class regarding C_1, and the specificity of C_1 regarding C_2 will be the sensitivity of C_2. And more importantly, the assignment of a sample to class C_1 or C_2 is the same whether f_1 or f_2 is used. Therefore, PLS-DA necessarily assigns every object to a single class, with the exception of those belonging to the PLS-Box, to which the model is not applied. Besides, it is concluded that both approaches outlined at the beginning of this section lead to the same result. The PLS-DA goal is actually similar to the linear discriminant analysis (LDA) one and outperforms it in the presence of a high degree of collinearity [11]. However, LDA does not delimit the subspace like the PLS-Box.

To decide the number of LVs in the PLS-DA model, the Classification Error (CE) is considered, which is equal to 1 − (sensitivity + specificity)/2. The CE is the same for C_1 and C_2 because, as shown in the section 3 of the supplementary material, the sensitivity of one class is the specificity of the other and vice versa. CE decreases as the number of LVs increases, and to avoid overfitting, the number of the LVs at which the CE value stabilizes in cross-validation is selected.

2.3.2. Partial Least Squares Discriminant Analysis. Hierarchical approach using AHIMBU

Regarding many class, k, classification problems, it is often challenging to determine which groups can be optimally separated, particularly when the classes are not well-defined. The generalization consists of the OVA coding of the classes, so $\widehat{f}_i(x), i = 1, \cdots, k$ binary learners are obtained and a similar procedure as the one described for two classes is followed. It has been shown that PLS-DA decreases sometimes its performance as the number of classes increases [29], even though it is an open question. For this reason, it has become interesting to address the k-class problem, but with a binary PLS-DA. The alternative, which has started to become widely used in the chemistry field, consist of building a sequence of binary PLS-DA classifiers in a hierarchical manner. The procedure consists of: i) choosing a class C_i and perform a DA versus all the others $\bigcup_{j=1,j\neq i}^{k} C_j$, ii) remove the C_i from the training set, choose another C_h, and build another DA regarding the remaining classes $\bigcup_{j=1,j\neq i,j\neq h}^{k} C_j$, remove C_h and continue applying DA successively. Yet, a hierarchical DA procedure is a sequence of binary discriminants with three main drawbacks: i) the order in which the classes are chosen (that influences the performance of the global classifier); ii) for each binary classifier the training set is different; and, iii) it shares with all Branch and Bound type algorithms the characteristic that in each step, the possibility of using the classes already separated in the previous step, is no longer explored (which could be better discriminated once some others have been removed). Nevertheless, the obvious advantages of this procedure are that the selection of the nodes in the tree is objective and is quite time-saving when having many classes.

The problem of considering all possible hierarchical trees to evaluate them and choose the best one is an NP complete problem (nondeterministic polynomial-time complete). Therefore, it requires an approximate solution using heuristic methods and approximation algorithms. Given a node from an AHIMBU, that is, when h classes have already been removed from the k originals, with the remaining ones, $\bigcup_{j=1,}^{k-h} C_j$, the proposed AHIMBU procedure builds a binary PLS-DA with the $P = \binom{k-h}{2}$ different pairs of classes, estimating in each one of them, the non-error rate, defined as: class error = 1 – (sensitivity + specificity)/2. Remember that as a binary classifying, the specificity of one class is the sensitivity of the other. With the $P(P\text{-}1)$ values, the class or classes with the lowest misclassification error are peeled off and separated from the others and a binary PLS-DA is performed between the discriminated class and the union of the k-h-1 remaining ones. If all classes can be perfectly separated after that within a comprehensive classification model, or if only two classes remain, then a classification model should be constructed to separate these. Otherwise, the procedure must be done again from the beginning. This procedure should be repeated for classified groups that comprised combined classes. As the assignment of the objects to separate a class in each node is the same as if it is considered the complementary class to set the critical value, it is obvious that the sensitivities and specificities that the hierarchical method achieves globally are not those of each individual stage. Another remark is that AHIMBU necessarily assigns each object to a single class, because the classification is done by performing one PLS-DA model at each node. The number of LVs of the PLS-DA model in each stage is calculated based on the CE, as explained in section 2.3.1. In this work, these sensitivity and specificity values and its calculation for comparative purposes with the other approaches used, is done by applying the hierarchical model to the training set and the prediction set.

2.3.3. PLS2-CM

Unlike a discriminant analysis like PLS-DA, where the response is binary, PLS2-CM adapts to the characteristics of each data set thanks to its coding of the classes through an ECOC matrix.

The codification ECOC matrix applied to the vector of labels is a $\boldsymbol{M}_{k\times c} = (m_{ij})$, $m_{ij} \in \{-1, 1\}$, matrix where k is the number of classes and c the length of the codeword. Each row of the ECOC matrix is the codeword of the class, and each column is the binary learner. The number of binary learners will vary depending on the length of the codeword used in each case. The binary learners will be the + 1 and −1 values of a binary function that divides all classes into two groups or super classes $\mathbb{A}$ and $\mathbb{B}$, defined by the union of classes which have a −1 or a + 1 respectively assigned. Given a length of codeword, c, there are $q = \binom{2^{k-1}-1}{c}$ different ECOC matrices.

When the number of classes is $k = 3$, there is just one possible ECOC matrix, whereas for the k greater than 3 case, the ref. [17] proposes a procedure to build some of them with optimal properties since there are many criteria to evaluate their performance (given that their enumeration is also a NP complete problem). For each ECOC matrix, a PLS2-CM model is built, ranging from a minimum to a maximum number of LV. This involves considering a large number of different PLS2 models that are multivariate multi-response regressions between a spectrum and the class code of each object, and allows a coding of the classes adapted to the data set to be modelled.

The decoding of the response is obtained with PLS2 (code of class of each object) through critical values at a preestablished significant level without using any probability distribution predefined for the PLS2 predicted values. The critical values of the statistics Q and T^2 are also used (that define whether an object is valid or not to apply the PLS2 regression to it). Concretely, for a given binary learner, f, and the super classes $\mathbb{A}$ and $\mathbb{B}$ that it defines, the density function for the values, $\widehat{f}$, is computed using a univariate Kernel density [30]. In Fig. 1 a sketch of these two density functions (in blue for the superclass $\mathbb{A}$, and in green for class $\mathbb{B}$) for the decoding of the responses that have been obtained with the PLS2 model and the optimal ECOC matrix for one of the studied cases (ripe olives, length of codeword, ls = 4 and binary learner f_2) that will be presented in detail in the section of results.

By fixing the γ_i y δ_i probabilities (that do not have to be equal to every binary learner), the critical values CV_i ($\mathbb{A}$) and CV_i ($\mathbb{B}$) are computed with the previously fitted distribution so that Eq. (1) is fulfilled,

$$\mathrm{P}\{\hat{\mathrm{y}} \in \widehat{f}_i(\mathbb{A}) \,|\, \hat{\mathrm{y}} \leq CV_\mathrm{i}(\mathbb{A})\} = \gamma_\mathrm{i}$$

$$\mathrm{P}\{\hat{\mathrm{y}} \in \widehat{f}_i(\mathbb{B}) \,|\, \hat{\mathrm{y}} \leq CV_\mathrm{i}(\mathbb{B})\} = \delta_\mathrm{i} \qquad (1)$$

where P stands for probability, and $\hat{\mathrm{y}}$ is the value of the i-th binary learner calculated by the PLS2 model for a spectrum from the training

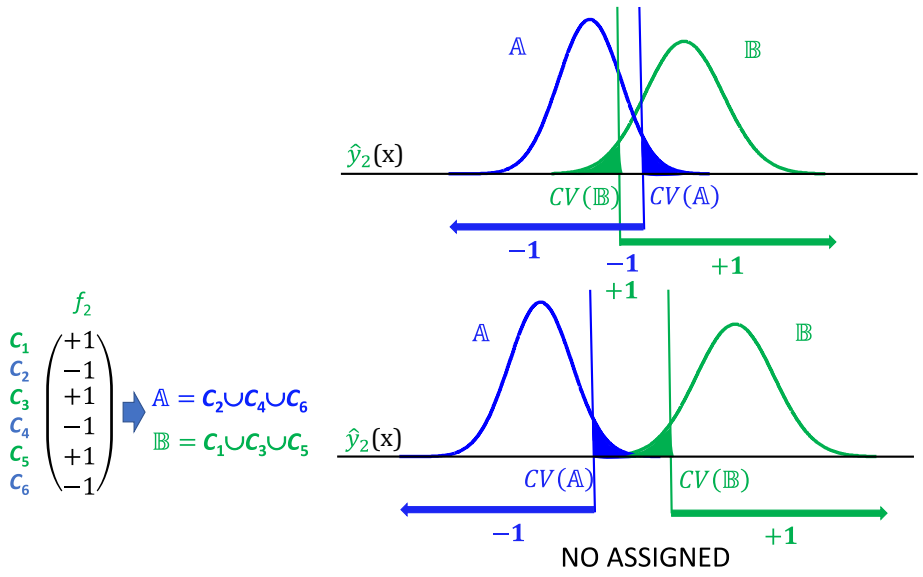

Fig. 1. Density functions for decoding f_2 for the ripe olives. a) when CV_i ($\mathbb{A}$) is greater than CV_i ($\mathbb{B}$), and b) when CV_i ($\mathbb{B}$) is greater than CV_i ($\mathbb{A}$). In blue for the superclass $\mathbb{A}$ and in green for the superclass $\mathbb{B}$. See in Table S4 the ECOC matrix for ripe olives (length of codeword, ls = 4 and binary learner f_2). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

set or the validation one. Notice that the definitions in Eq. (1) involves that γ_i would be a large value close to one, whereas δ_i would be close to zero.

Both critical values CV_i ($\mathbb{A}$) and CV_i ($\mathbb{B}$) define the + 1 or −1 assignment to each spectrum, that is, they assign the spectrum to one of the super classes. There are two possible situations: i) that CV_i ($\mathbb{A}$) is greater than CV_i ($\mathbb{B}$); and ii) that CV_i ($\mathbb{B}$) is greater than CV_i ($\mathbb{A}$). In the first representation of Fig. 1, when a $\hat{y}$ value belongs to the interval marked in green, a + 1 ($\mathbf{x} \in \mathbb{B}$) is assigned to the corresponding spectrum **x**, while if it belongs to the interval marked in blue, a −1 ($x \in \mathbb{A}$) is assigned. Finally, if it belongs to the intersection of the two intervals, $\mathbf{x} \in \mathbb{A} \bigcap \mathbb{B}$, the two values, +1 and −1 are assigned. Opposite to that, if the relative position of both critical values is that of the second representation of the same figure (Fig. 1), the assignment to the super classes is as explained in first instance, but there is the possibility that the value of $\hat{y}$ is external to both intervals. In that particular case, the value assigned would be a 0 (this indicates that the corresponding spectrum x does not belong to either of the two super classes).

Once the response of the PLS2 model has been decoded, each sample is assigned to the class whose code matches the one in the ECOC matrix, thus obtaining a sensitivity–specificity matrix. For the internal validation a subset of the training test (that has not intervened in the construction of each PLS2-CM-ECOC model) has been used. After, this matrix is evaluated using DMCEN to establish in the training stage the PLS2-CM model with the highest prediction performance, which is the one with the lowest DMCEN value, as will be explained in the following section. To do that, the original training set is split into two, one to build the model and another one for the internal validation for its optimization (in this work, the partition has been done with the Kennard Stone algorithm [31], splitting the 30 % of the data as an internal validation set). In this way, the procedure optimizes simultaneously the ECOC matrix and the number of LV of the PLS2. Finally, the optimal PLS2-CM model is applied to all the elements of the original training set to obtain the DMCEN in training and to the prediction set to obtain the DMCEN in prediction.

The implemented procedure in PLS2-CM (decoding; assignment of objects to one, several, or no classes; and the selection of the number of LV that minimizes DMCEN in prediction) is also applied to the OVA matrix in this work, which is the common one in chemometrics (see the OVA matrix in Table S3 in the supplementary material).

Clearly, in this case, only the complexity of the PLS2 is optimized but not the coding matrix.

2.3.4. DMCEN

It has been mentioned in the introduction section that for *k* classes, the majority of the index used as a figure of merit in DA or CM are the average (arithmetic or geometric average) of the sensibilities and specificities of the models of the *k* classes. This is equivalent to the idea that the global performance of the method is described by the average of the performance for each class. When a DA or CM model is applied to a data set, the disorder in it is reduced by assigning each object to one class, several, or none of them. In this sense, the *k*-class-model decreases the entropy of the data set. Inspired by Shanon's concept of entropy, the initial development of this idea consisted of proposing Confusion Entropy (CEN) as a measure of the uncertainty generated by a *k*-class-model [32]. Some problems raised by this measure are solved in ref. [33], where the Modified Confusion Entropy (MCEN) is introduced. To determine the Shannon entropy, it is necessary to calculate the quotient of the frequency of objects of one class, C_j, that are mistakenly assigned to another C_m, divided by the sum of the frequencies of all possible erroneous assignments of both objects of C_j and C_m. Finally, in ref. [25] Diagonal Modified Confusion Entropy (DMCEN), that generalizes MCEN considering the correct assignments to each class, is proposed. In addition, an analysis of the performance of DMCEN is made regarding the total efficiency, which is the most used figure of merit in chemometrics.

DMCEN can take values from zero to one, and the lower its value is, the better the performance of the model is. This index will be the one used in this work for comparison purposes.

2.4. Software

The AONIR platform developed by AOTECH [26] was used to record the spectra, while PLS-Toolbox [28] working under MATLAB version 9.9.0 (R2020b) [34] was employed for fitting PCA, PLS-DA and AHIMBU [21]. PLS2-CM has been programmed also in MATLAB, and DMCEN was calculated using an ad-hoc MATLAB code available in ref. [35], that calculates DMCEN given a sensitivity–specificity matrix.

3. Results and discussion

The main objective of the study is, not only to differentiate between treated and non-treated olives, but also to differentiate between the

different chemical treatments applied with different chemical compounds. To do that, after the spectra collection, PLS-DA models were carried out to separate the treated olives from the untreated ones, whereas AHIMBU and PLS2-CM were performed considering six different classes, C_i, corresponding each one of them to samples sprinkled with five different agrochemicals: C_1, DF; C_2, DM; C_3, LC; C_4, OF; C_5, TB; and, the last one, C_6, corresponding to untreated olives. The analysis were made separately for ripe and unripe olives.

3.1. NIR spectra and preprocessing

As explained before, a total of 3618 spectra were registered, 1920 for the unripe olives, and 1698 for the ripe ones. Fig. 2 shows the representation of these spectra for each type of olive and for each one of the classes: in Fig. 2a, it can be seen the differences between ripe and unripe olives, while in Fig. 2b, the averaged spectra can be observed. On the other hand, Fig. 2c shows the spectra from the unripe samples (but colored according to each one of the six classes), and Fig. 2d, those same spectra, but preprocessed. On another note, Fig. 2e and Fig. 2f represent in the same way the spectra corresponding to the ripe samples. Fig. 2g shows the averaged preprocessed spectra of each of the six classes for the unripe olives and Fig. 2h for the ripe ones.

The application of smoothing, baseline corrections, and scatter corrections to rectify artefacts in analytical signals is highly dependent on the data and the analysis objectives [36,37] and can be implemented in any order [38]. In this case, the spectra were preprocessed by applying the Standard Normal Variate (SNV) followed by Savitzky-Golay (SG). In the latter case, a window width of 7 points using a second-degree polynomial and a second derivative were used. This data preprocessing was the same with every chemometric strategy applied in this work in order to evaluate the performance of the methods under the same conditions.

3.2. Unsupervised analysis (PCA)

Before presenting the discriminating and modelling results, a preliminary exploratory analysis (Principal Component Analysis, PCA) has been performed separately for ripe and unripe olives. Three PC were selected in both cases (explaining an 84.9 % of the variance for the unripe olives and an 86.9 % for the ripe ones), and as it can be seen in Fig. 3, all the classes are totally overlapped and at first sight it seems quite difficult to separate them. Not even the untreated samples (coloured in red) are possible to be differentiated from the rest.

3.3. Treated versus nontreated olives (PLS-DA)

For each one of the following analysis (not just the PLS-DA, but every performed analysis in this work), it is necessary to point out that the data set were split in two sets according to the Kennard Stone [31] method, having the same data sets in training, TRT, and in prediction, TEST. The proportion of the data between TRT and TEST has been maintained for each class. This data distribution of the registered spectra can be consulted in Table S2 in the supplementary material. The original data were split into two sets, the training one (TRT set, of dimensions 1536 x 125 for the unripe olives and 1362 x 125 for the ripe ones) and the prediction one (TEST set, of dimensions 384 x 125 for the unripe olives and 336 x 125 for the ripe ones). Also, in the three chemometric approaches, the critical values for the statistics Q and T^2 for the outlier detection in the PLS models have been the set at a confidence level of 0.99. In addition, for the PLS2-CM model, the CV_i of the distributions of the binary learners, f_i, have been obtained establishing the probabilities $\gamma_i = 0.99$ and $\delta_i = 0.01$ regarding Eq. (1).

Despite the PCA analysis, good results were achieved when trying to differentiate between treated and untreated samples by performing a PLS-DA study (supervised procedure). As previously explained, two different analyses were carried out for ripe and unripe olives respectively. In this case, the main objective is to separate the untreated samples (assigned with a 0) from the treated ones (assigned with a 1), regardless of which chemical treatment was used.

In the two built models, which needed 5 and 6 LV according to the cross-validation, for the unripe and ripe olives, respectively, the sensitivities and specificities were evaluated. The threshold (*CV* value) selected by the PLS-DA algorithm in each case can be observed in Fig. 4 as a dashed red line: for the case of unripe olives, 0.14 and 0.86 for the untreated (grey circles) and treated olives (red diamonds), respectively, while 0.24 and 0.76 for the untreated and treated olives, respectively, for the case of the ripe ones. The position of each calculated y value regarding these thresholds can also be seen. Note that, from the sample 1536 on, the samples correspond to the prediction data sets for the unripe olives (Fig. 4a and Fig. 4b) and from the sample 1362 on, the samples correspond to the prediction data sets for the ripe ones (Fig. 4c and Fig. 4d).

The samples classification can be explained, for instance, for Fig. 4b, as follows: most of the y predicted values (≈ 91.30 %, 4 red diamonds) are greater than 0.86 for the treated olives (unripe ones). That means they are classified as correctly chemically treated. Moreover, a 7.02 % of the non-treated samples (represented as grey circles), are also classified mistakenly as chemically treated.

Given that it is just a two classes problem, the sensitivity and specificity values of one class, will also be the specificity and sensitivity ones of the other class, respectively. For the case of unripe olives, 0.9298 will be the sensitivity of the untreated olives, and 0.9130 is its specificity regarding the treated class, while 0.9130 and 0.9298 will be the sensitivity and the specificity values for the treated samples respectively. For the ripe olives, the sensitivity of the untreated olives class is 1.0000 and 0.9760 its specificity, while 0.9760 and 1.0000 will be the sensitivity and the specificity values for treated samples class (see Table S4 in the supplementary material). Yet, a new problem arises when trying to know which agrochemical has been used in the chemical treatment of the olives. For that matter, new discriminant and modelling strategies were applied since we are dealing now with a six-class classification problem.

3.4. Hierarchical multiclass classification using AHIMBU

It has been explained beforehand the reasons why it would be suitable to use the AHIMBU approach to solve a multiclass problem. In this case, two models were performed, one for the unripe olives and another for the ripe ones. For these models, the TRT and TEST set are the same as for the PLS-DA ones (*section 3.3*) but using the distribution of the objects in the six classes according to Table S2 in the supplementary material. The training set was employed to build the PLS-DA of each node using cross-validation in order to select the appropriate number of LV for each one of them. In the view of the fact that we are dealing with six classes, the AHIMBU will perform five different individual PLS-DA models (represented as nodes or rules) to separate one class at a time according to Figure S3 in the supplementary material. At each node, once the separated classes in the previous nodes have been eliminated, it is decided which class will be separated from the remaining ones. This is done by calculating the misclassification error in cross-validation between all pairs of classes. After that, the class (or classes) with the lowest misclassification error, *C*, are separated from the others, *D*, and a binary PLS-DA is applied to discriminate *C* from *D*. For a DA between two categories, the misclassification error can be defined as 1 – (sensitivity + specificity) / 2.

In ref. [21] the performance of the AHIMBU will be given as the average, over all classes, of true positive ratio (TPR), the ratio between the number of samples rightly categorized as true and the total number of samples of a class. TPR is therefore the sensitivity of each class, and globally, the average of all of them. When evaluating a method (DA or CM) without considering the specificity values, is poorly descriptive of its performance. AHIMBU is a discriminant procedure that, once it has

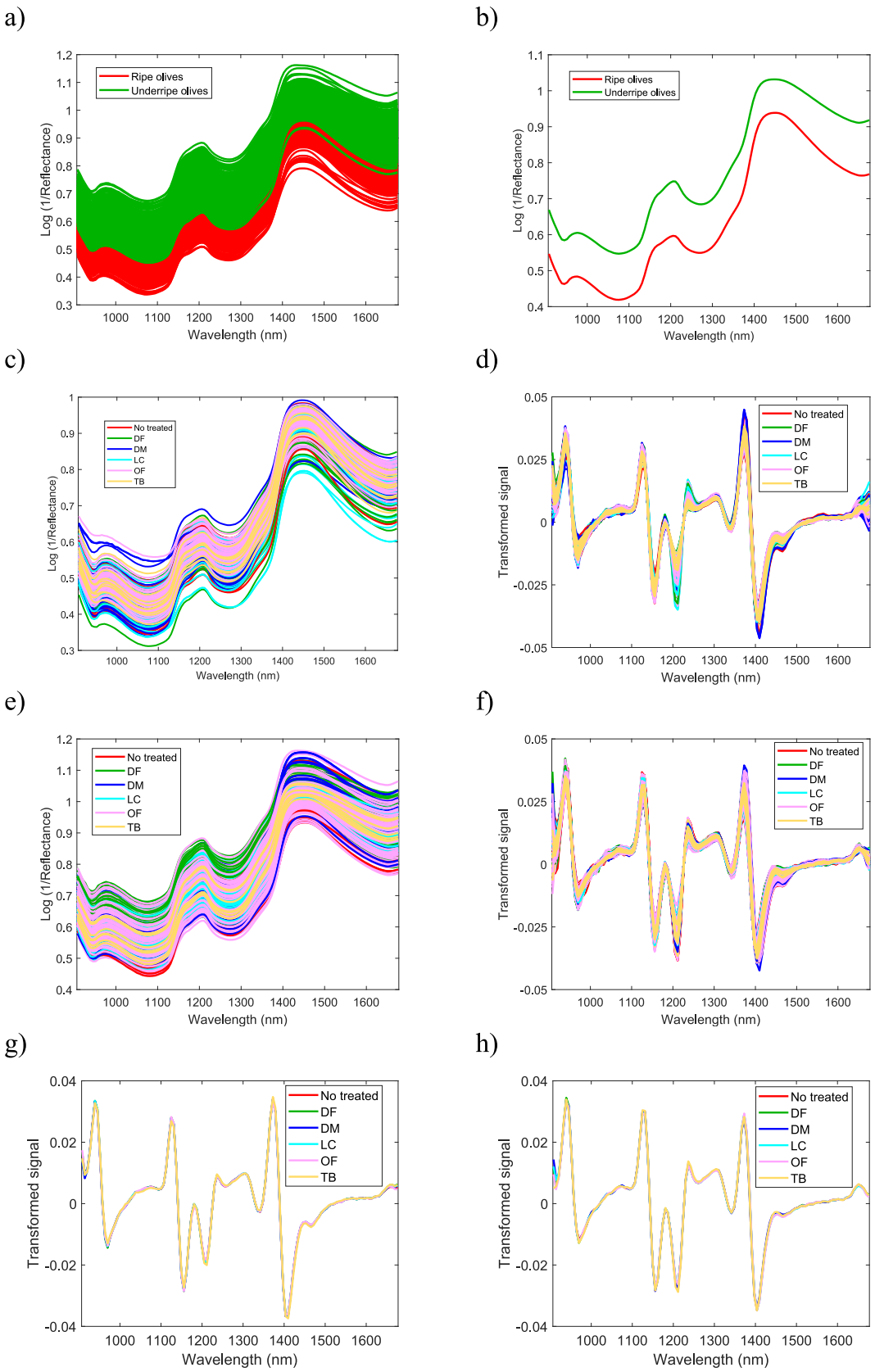

Fig. 2. NIR spectra recorded for the olive samples. a) Ripe olives in red and unripe ones in green, b) averaged spectra, c) Unripe olive spectra, d) Preprocessed spectra of unripe olives, e) Ripe olive spectra, and f) Preprocessed spectra of ripe olives g) averaged preprocessed spectra for each class for the unripe olives h) averaged preprocessed spectra for each class for the ripe olives. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

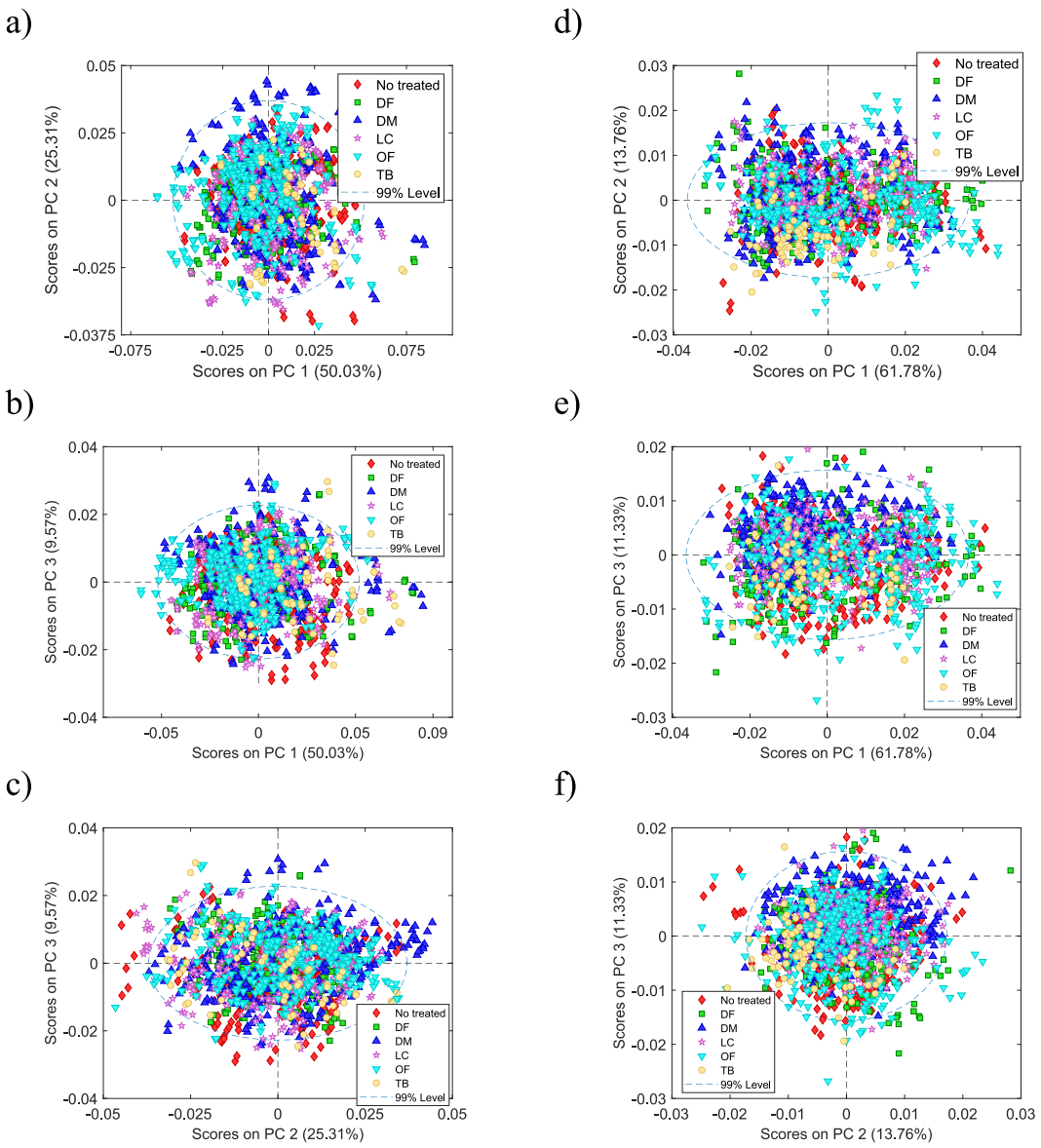

Fig. 3. PCA analysis for a), b) and c) unripe olives and for d), e) and f) ripe olives.

been built, can be applied to the objects in the training set, TRT, and those in the prediction one, TEST, to obtain the matrix of sensitivities and specificities in training and in prediction. In this way, it is comparable with the matrix of sensitivities and specificities provided by any other method, in this work, with PLS2-CM.

As explained in the introduction section, DMCEN is a useful index to compare sensitivity–specificity matrices. It is observed that the prediction value for this index is a little bit worse than the training one, and it is better in the case of the ripe olives (Table 2a) than for the unripe ones (Table 1a).

The most remarkable aspect in the sensitivity–specificity matrices is the low sensitivity values obtained with AHIMBU. For the unripe olives, the values were 0.8875, 0.5469, 0.1250, 0.4063, 0.3250 and 0.2500 for C_1, C_2, C_3, C_4, C_5, and C_6 respectively, whereas 0.625, 0.4175, 0.2813, 0.4375, 0.8594 and 0.6250 for the ripe olives.

On the one hand, the model of class C_6 (untreated olives), for the unripe olives has specificity values that range from 0.7813 to 0.9875 in prediction. These values are better in the ripe olives case, where they range from 0.8438 to 1.000. On the other hand, as explained beforehand, the sensitivity of the AHIMBU (for C_6) is 0.2500 and 0.6250 for unripe and ripe olives, respectively. Therefore, in spite of the good specificity values, the sensitivity ones are quite low, both in training and in prediction. That means that the AHIMBU prioritize the rejection of the objects that do not belong to the class, rather than classify accurately an object in its actual class. In general, when a discriminant reduces the sensitivity of a class, the specificity increases. DMCEN values for AHIMBU, 0.6937 and 0.6016, for unripe and ripe olives respectively, emphasizes the substandard of the model, which is worse for the unripe olives.

For all that has been globally seen, the model built with AHIMBU seems like is not effective for the discrimination of the six classes, particularly because the model sensitivities are relatively low. The model does not achieve the sensitivity obtained with PLS-DA to discriminate between the olives treated with the five agrochemicals and the untreated ones.

3.5. *Multiclass modelling (PLS2-CM)*

Regarding PLS2-CM models, two models were built for the case of the unripe olives and for the ripe ones, what makes a total of four models, two of them, were built using the classical codification (OVA) of the classes to apply the PLS2 regression. The other two models were built

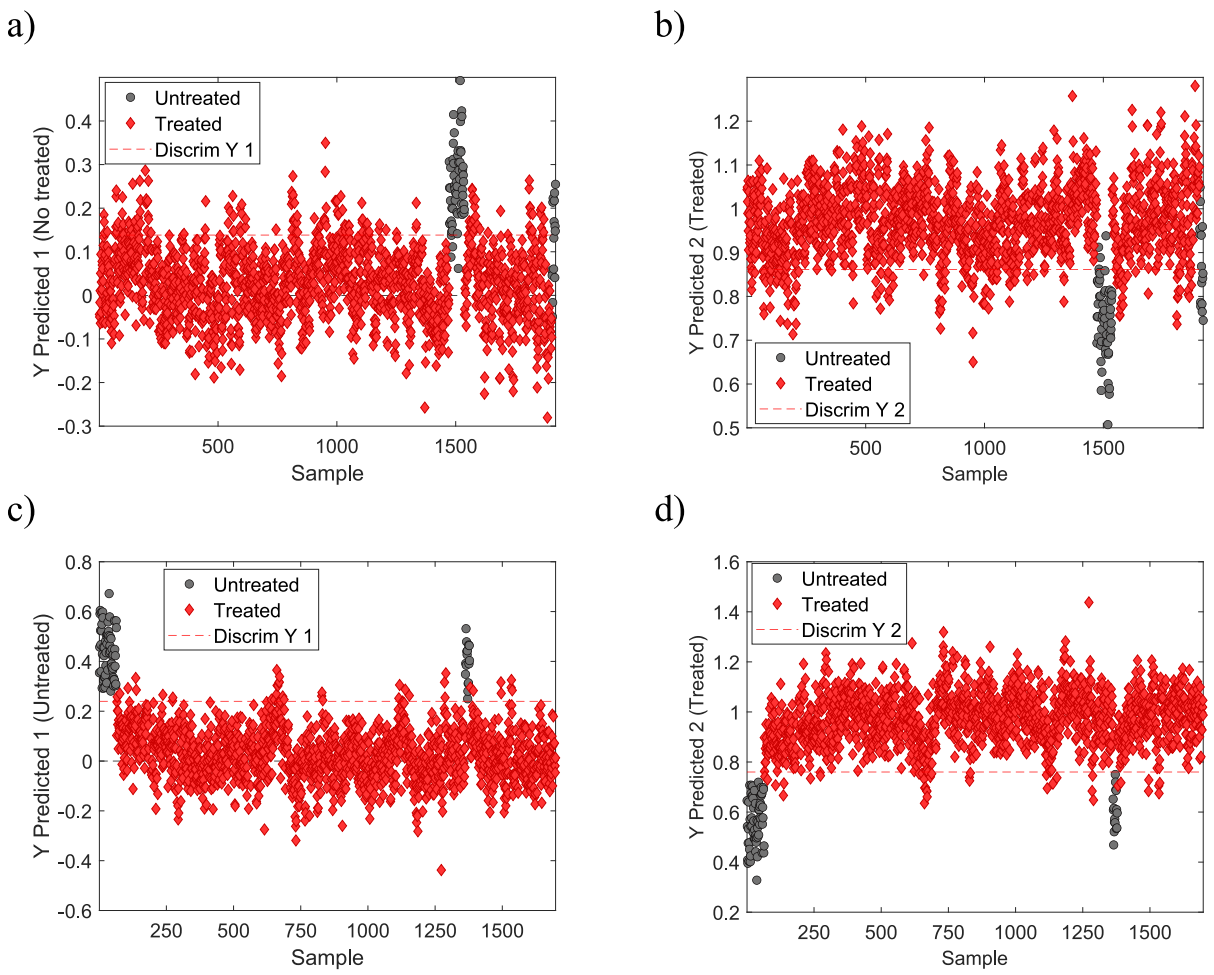

Fig. 4. PLS-DA graphical results for a) and b) unripe olives (from the sample 1536 on, the samples correspond to the prediction data sets), and c) and d) for ripe olives (from the sample 1362 on, the samples correspond to the prediction data sets).

using the optimal ECOC matrix that gave the best DMCEN results. The OVA codification of the classes was obviously the same for both cases (unripe and ripe olives, Table S3 in supplementary material), whereas the ECOC matrix of the PLS2-CM is optimized during the training stage, and therefore, it is different for the unripe and ripe olives (despite using in both cases a length of codeword of 4). This information and the matrices used for the codification of the responses can be consulted in detail in Table S3 in the supplementary material.

It also must be considered that for these models, the training set, TRT, and the prediction set, TEST, are the same as for the PLS-DA and AHIMBU models. However, in order to build the PLS2-CM selecting the correct number of LV and the optimal ECOC matrix in each case, the calibration set TRT was split again by means of the Kennard Stone algorithm [31] in 70 % of the data, TR1, to build the model, and the 30 %, EV set, to obtain the optimized PLS2-CM model (this is the model with the best DMCEN value in the EV set). After that, the final model is applied to the TRT set and to the TEST set in order to evaluate its performance in training and in prediction.

3.5.1. *Classical codification (PLS2-CM-OVA)*

It has already been proven with other data sets that systematically better PLS2-CM are obtained by optimizing the coding matrix, than when using the OVA coding [14]. In fact, OVA is a particular case of ECOC matrix, and it could be possible that for a given data set it would be the optimal.

Using the OVA matrix codification in **Table S3a** in the supplementary material for the PLS2-CM models construction, 7 LV were needed both for the ripe and unripe olives to optimize DMCEN using the EV set. Remember that the codes are + 1 for the spectra of one class, and −1 for the rest, for example, in the second row of that table (−1, +1, −1, −1, −1, −1) codes the objects of class C_2.

The corresponding sensitivity and specificity matrices for the unripe olives are shown both for the TRT set and for the TEST set in Table 1b. The performance in this case is precisely the opposite of the AHIMBU, and this also happens with the ripe olives (Table 2b). In other words, good sensitivity values are obtained, while specificity is quite poor. Furthermore, the sensitivity of the untreated samples (C_6) in prediction for the unripe olives is also bad (0.1250). As it can be seen, there is also quite a lot of discrepancy between the training results and the prediction ones, what means that the model is not stable.

On the other hand, as for the ripe olives, the DMCEN value in training is 0.4646, while in prediction is 0.5051. The corresponding sensitivity and specificity matrices are shown in Table 2b. In this case, the same conclusions as for the unripe olives model can be achieved, with the exception of the sensitivity value for the untreated samples (C_6), that in this case is more much better (0.9375). Despite that, the PLS2-CM-OVA models cannot provide good results either because although they can correctly classify the objects correctly in their category, it is practically impossible for them to reject the objects that do not belong to the actual class (which highlights the need to always consider the specificity of the models).

Globally, PLS2-CM-OVA has a better DMCEN in training (0.4925 and 0.4646 for unripe and ripe olives, respectively) regarding AHIMBU, but it shows instability in prediction for the case of the unripe olives, being

Table 1
Unripe olives. Sensitivity and specificity matrices in training and in prediction with a test set, of the 6-class model using the NIR spectra for the three employed approaches.

a) AHIMBU procedure

Training (DMCEN=0.6415)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.8875	0.9437	0.9969	0.9594	0.9906	0.9969
	C_2	0.9180	0.7070	0.9844	0.9180	0.9531	0.9531
	C_3	0.7719	0.7562	0.1031	0.8000	0.8906	0.8844
	C_4	0.9648	0.8086	0.9570	0.4375	0.9297	0.7930
	C_5	0.8781	0.8531	0.9906	0.8063	0.3750	0.8562
	C_6	0.9219	0.9531	0.9688	0.8438	0.9688	0.6406

Prediction (DMCEN=0.6937)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.8875	0.9500	1.0000	0.9750	1.0000	0.9875
	C_2	0.9063	0.5469	0.9844	0.8438	0.8750	0.9375
	C_3	0.7375	0.8250	0.1250	0.8625	0.8875	0.8375
	C_4	0.9844	0.7813	0.9219	0.4063	0.9375	0.7813
	C_5	0.8750	0.7875	0.9750	0.8500	0.3250	0.8375
	C_6	0.8750	1.0000	0.9375	0.9375	0.5000	0.2500

b) PLS2-CM procedure. OVA codification

Training (DMCEN=0.4925)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.9344	0.4844	0.1875	0.3719	0.6094	0.8094
	C_2	0.3320	0.9414	0.3438	0.5820	0.5313	0.7852
	C_3	0.2469	0.6094	0.9000	0.5563	0.3750	0.7594
	C_4	0.4922	0.5898	0.4297	0.8945	0.5156	0.7773
	C_5	0.3313	0.6625	0.2906	0.5750	0.9875	0.8375
	C_6	0.3438	0.5938	0.2500	0.4063	0.5313	0.8594

Prediction (DMCEN=0.7705)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.8875	0.4375	0.0625	0.6000	0.2875	0.6875
	C_2	0.2813	0.9375	0.2813	0.4375	0.5156	0.9063
	C_3	0.2250	0.5500	0.8875	0.5750	0.3000	0.9500
	C_4	0.4531	0.5313	0.3438	0.9688	0.5469	0.6719
	C_5	0.2375	0.6500	0.2125	0.3875	0.9750	0.8875
	C_6	0.3750	0.6875	0.0625	0.1875	0.2500	0.1250

c) PLS2-CM procedure. ECOC codification

Training (DMCEN=0.4402))

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.9437	0.3344	0.0844	0.4906	0.2687	0.9531
	C_2	0.4727	0.9883	0.3867	0.4141	0.6836	0.9414
	C_3	0.0750	0.2938	0.9437	0.4656	0.3312	0.9281
	C_4	0.4727	0.2148	0.4883	0.9258	0.4102	0.8438
	C_5	0.1969	0.4938	0.3344	0.3031	0.9781	0.9500
	C_6	0.9688	0.9531	0.9688	0.9531	0.9688	0.9844

Prediction (DMCEN=0.4898)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.9250	0.3750	0.1250	0.4625	0.1750	0.8875
	C_2	0.4844	1.0000	0.4219	0.3906	0.6563	0.9844
	C_3	0.0750	0.3750	0.9125	0.5375	0.3000	0.9750
	C_4	0.4375	0.1719	0.4375	0.8906	0.4063	0.7969
	C_5	0.2500	0.4625	0.3500	0.2750	0.9500	0.9250
	C_6	0.8125	0.8750	0.8750	0.6875	0.6875	0.8125

Abbreviations: C_1, DF; C_2, DM; C_3, LC; C_4, OF; C_5, TB; and C_6, untreated.

the DMCEN values 0.7705 and 0.5051 for the unripe and ripe olives, respectively.

3.5.2. *Optimal ECOC codification (PLS2-CM-ECOC)*

To obtain a good compliant class model, the optimal ECOC matrix for build the PLS2-CM models were used, where 9 LV were needed for the ripe olives and 7 for the unripe ones. These ECOC matrices can be found in **Table S3 b-c** in the supplementary material. In both cases the PLS2 model that provides the lowest DMCEN value uses a length of the codeword of 4, nevertheless, the matrices are different. The codification of each class is the corresponding row of the matrix, for instance, C_6 has the same code (+1, −1, −1, −1) both for the models of unripe and ripe olives. By observing the columns of these matrices (binary learners), it can be seen that there are two shared binary learners between both matrices, the f_4 in both cases. The binary learner f_3 for the unripe olives and the f_2 for the ripe ones also coincide. The f_4, models the difference between the untreated samples (C_6) and the remaining classes, whereas f_2 and f_3 model the classes C_1 (DF), C_3 (LC) and C_5 (TB) versus the C_2 (DM), C_4 (OF) and C_6 (untreated samples) classes. The structure of these matrices adapts the coding to the current training set, making the modeling using PLS2 much more efficient.

Moving forward to the figures of merit, regarding the unripe olives, the calibration DMCEN is 0.4402, while in prediction is 0.4898, what implies a reduction of 23.6 % and 29.4 % regarding the values obtained with the AHIMBU. The corresponding sensitivity and specificity matrices are shown in Table 1c where it can be seen that for the unripe olives, the obtained values in prediction for the sensitivity are 0.9250, 1.0000, 0.9125, 0.8906, 0.9500 and 0.8125 for classes C_1, C_2, C_3, C_4, C_5, and C_6 respectively. Just as in the classification, these values are better for the ripe olives, which are 0.9688, 0.9219, 0.9688, 0.9375 and 0.9375 for the same classes. It is quite evident the improvements in every sensitivity value regarding the AHIMBU, and particularly, for the C_6 class. Furthermore, the instability of the OVA modelling no longer manifests.

Another aspect of the C_6 class model is that, in prediction, its specificity ranges from 0.7969 to 0.9844 in the case of unripe olives and from 0.9375 to 1.0000 for the ripe ones. Moreover, the specificity of the other classes with respect to C_6 is acceptable in unripe olives and quite good in the ripe ones. That is, for unripe olives, the C_6 model can not only classify the objects of its class but can also correctly rejects the ones that do not belong to its class, which was actually one of the main objectives of this work. Inversely, each one of the models for classes C_1 to C_5 can correctly reject the C_6 objects in comparison with the AHIMBU model results.

For the ripe olives case, the results are even better, not only can be well classified the C_6 objects (or correctly rejected by the models of the other classes, as for the unripe olives), but also the confusion between different classes can be better studied (and therefore, the confusion between the chemical treatment used), since the specificity values were improved. For example, for the ripe olives, the specificity of the C_3 class (LC) regarding the C_2 class (DM), 0.2813, is the lowest of all specificities of the C_3 class, and also, the lowest specificity of the C_2 class regarding the C_3 class (0.3750). That is to say, both classes have the highest degree of confusion between them, which actually makes sense knowing that they are both insecticides and belong to the same family of chemical compounds (pyrethrins). Also, its chemical structure is more similar between them regarding the rest of the agrochemicals. The specificity of the agrochemical classes is less clear and less systematic for the unripe olives than for the ripe ones, in fact, in the unripe ones, DMCEN is worse. This different performance was also observed in the PLS-DA analysis (when applied with only two classes). In that case, it was found that the discrimination between treated and untreated olives was also worse in the unripe olives.

Table 2
Ripe olives. Sensitivity and specificity matrices in training and in prediction with a test set, of the 6-class model using the NIR spectra for the three employed approaches.

a) AHIMBU procedure

Training (DMCEN=0.5552)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.7852	0.9531	0.9844	0.9609	0.9023	0.9844
	C_2	0.9766	0.4453	0.8789	0.8164	0.9219	0.8633
	C_3	0.8485	0.9053	0.4242	0.8220	0.9432	0.9053
	C_4	0.8828	0.9258	0.9727	0.6094	0.9063	0.9219
	C_5	0.9887	0.9925	1.0000	0.9962	0.9624	0.9850
	C_6	0.5469	0.9844	0.9688	0.9531	0.9219	0.3750

Prediction (DMCEN=0.6016)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.6250	0.9219	0.9375	0.8594	0.9375	0.9688
	C_2	0.8906	0.4375	0.8906	0.8750	0.9375	0.8438
	C_3	0.7969	0.8906	0.2813	0.6875	0.9844	0.9219
	C_4	0.8125	0.8750	0.9688	0.4375	0.8281	0.9531
	C_5	0.9844	0.9375	0.9844	0.9531	0.8594	1.0000
	C_6	0.8750	1.0000	0.8750	0.8750	1.0000	0.6250

b) PLS2-CM procedure. OVA codification

Training (DMCEN=0.4646)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.8672	0.3750	0.5703	0.4727	0.5977	0.5664
	C_2	0.4727	0.5781	0.5664	0.4648	0.7891	0.8750
	C_3	0.4015	0.3674	0.9432	0.4129	0.8182	0.8712
	C_4	0.4180	0.4297	0.6055	0.9609	0.5508	0.7656
	C_5	0.4060	0.5526	0.8534	0.4248	0.9474	0.7707
	C_6	0.7500	0.8906	0.9063	0.7500	0.8750	0.9219

Prediction (DMCEN=0.5051)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.8906	0.2656	0.5313	0.2656	0.4531	0.5781
	C_2	0.2813	0.9531	0.4219	0.4375	0.6563	0.9531
	C_3	0.4531	0.3125	0.7656	0.3594	0.8438	0.8438
	C_4	0.2969	0.3281	0.5938	0.9688	0.3125	0.7813
	C_5	0.2656	0.4844	0.8438	0.4375	0.9375	0.7969
	C_6	0.5000	0.8750	0.6875	0.5000	0.6875	0.9375

c) PLS2-CM procedure. ECOC codification

Training (DMCEN=0.3885)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.9688	0.7227	0.5820	0.6250	0.3359	0.9570
	C_2	0.9219	0.9570	0.3438	0.7148	0.8086	0.9453
	C_3	0.7652	0.4697	0.9735	0.8750	0.6288	0.9621
	C_4	0.5508	0.3516	0.5469	0.9570	0.3164	0.9453
	C_5	0.4436	0.8759	0.6917	0.6241	0.9361	0.9737
	C_6	1.0000	0.9844	0.9844	1.0000	1.0000	0.9844

Prediction (DMCEN=0.4166)

		Predicted class					
		C_1	C_2	C_3	C_4	C_5	C_6
True class	C_1	0.9688	0.6250	0.4063	0.3750	0.0469	1.0000
	C_2	0.8906	0.9219	0.2813	0.6250	0.7656	0.9531
	C_3	0.7344	0.3750	0.9688	0.7813	0.5469	0.9375
	C_4	0.5469	0.4531	0.6563	0.9688	0.3281	0.9844
	C_5	0.2969	0.7500	0.5938	0.5000	0.9375	0.9844
	C_6	0.9375	0.9375	0.9375	0.9375	0.9375	0.9375

Abbreviations: C_1, DF; C_2, DM; C_3, LC; C_4, OF; C_5, TB; and C_6, untreated.

3.6. Comparative analysis based on DMCEN

Finally, as a summary, Fig. 5 represents the DMCEN values in training (TRT set) and in prediction (TEST set) for each case (unripe and ripe olives) and each chemometric approach. Considering that this is a problem of multiple classes, DMCEN is an appropriate option because it considers the capability of the PLS-DA or the PLS2-CM to assign each spectrum to its class (sensitivity), and also evaluates the specificity of all classes regarding one, and a specificity regarding the rest. When the sensitivity and specificity values decrease, the entropy increases, and the classification and modelling procedures have a lower performance. This approach, which is explained more in detail in ref. [25], is more sensitive to changes in the elements of the sensitivity–specificity matrix than the arithmetic or geometric averages of these elements, and of course, more much informative than the averaged TPR for each class.

Observing Fig. 5, it can be concluded that the strategy that presented the best performance is the PLS2-CM using the optimal ECOC matrix (the one whose DMCEN value was the lowest). Studying more in detail this figure and pointing out again the differences on the sensitivity–-specificity values between the approaches (the AHIMBU offered good specificities but unsatisfactory sensitivities and the PLS2-CM-ECOC improves every sensitivity making some specificity values lower), the reason why the use of DMCEN is essential, because otherwise, it would be quite difficult to evaluate which results are the best ones. PLS2-CM with the optimal ECOC both in training and in prediction provide a better DMCEN for the ripe olives.

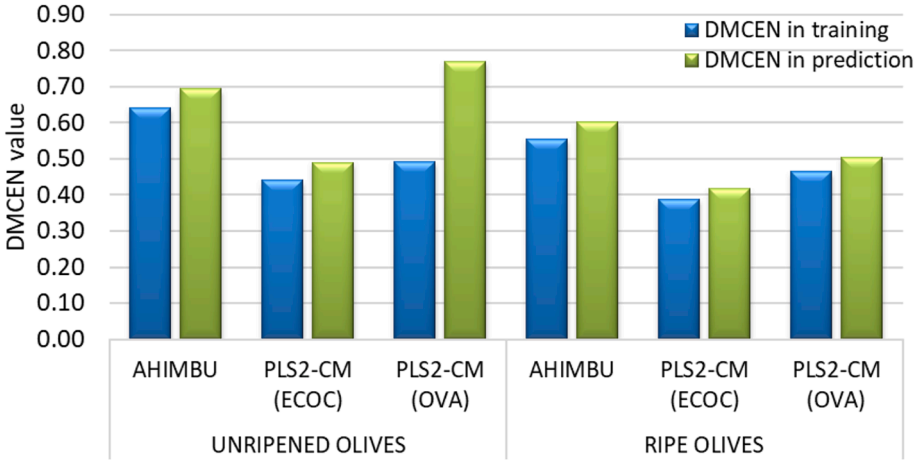

Fig. 5. Summarized results for DMCEN values for each chemometric approach used.

Considering that a *k*-class-model is more informative if its performance is better than one that assigns the objects randomly, it is possible to define a Benchmark threshold. This idea has been proposed in ref. [24] and applied to DMCEN [25]. Admitting that a *k*-class-model assigns objects randomly when all sensitivity and specificity values are equal to 0.5, the Benchmark threshold value for DMCEN for six classes is 0.7234. That is, it can be stated that a model for six classes with a DMCEN greater than 0.7234 has an analogous performance as a random class-model. For unripe olives, particularly in prediction, the DMCEN value for AHIMBU is quite close this value (0.7234), while the PLS2-CM-OVA one exceeds it. On the other hand, the DMCEN value obtained with PLS2-CM-ECOC is lower, and this approach is distant from being a random class-model, especially for ripe olives.

4. Conclusions

The results in this paper bring to the surface the underlying importance of some chemical factors, like the maturation of food samples, or some elements related with multiclass problems, such as the usefulness of using compliant class models instead of discriminants, and the necessity of using a global index to evaluate the final results of the sensitivity–specificity matrices. Also, the importance that this index should consider the specificity values and not just the sensitivity ones to define a compliant class model.

The main objective in this work, which was to differentiate the untreated samples, was successfully achieved by means of PLS-DA models, but also, it was possible to study the different results provided by two novel chemometric approaches compared with a classical one. The AHIMBU prioritizes the specificities, whereas the PLS2-CM (with the OVA codification or the ECOC one) prioritizes the sensitivities. Finally, it was concluded that PLS2-CM using the optimal ECOC matrix, provides the best performance for DMCEN for all models built, either in training or in prediction working with a many-class classification problem.

In summary, the feasibility of using NIR spectroscopy along with PLS2-CM has been proven to be an effective strategy to differentiate between different contaminants in intact olives, at least, in terms of differentiating the ones that can be already ingested (or used for olive oil manufacture). The great advantage of using a unique model for the six classes such as PLS2-CM-ECOC is that allows to increase the information about the failures of the model, as could be seen with DMCEN.

Funding

This work has been funded by the Ministerio de Industria, Comercio y Turismo under Project ESPECTROLIVE (AEI-010500–2023-232).

CRediT authorship contribution statement

D. Castro-Reigía: Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **I. García:** Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **S. Sanllorente:** Writing – review & editing, Supervision, Conceptualization. **L. A. Sarabia:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Formal analysis, Data curation, Conceptualization. **M.C. Ortiz:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Formal analysis, Data curation.

Declaration of competing interest

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

Data availability

The authors do not have permission to share data.

Acknowledgements

The authors thank Fundación CITOLIVA/INOLEO for allowing the measurements in their facilities and the entities for financial support.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.microc.2024.111550.

References

[1] M. Forina, P. Oliveri, S. Lanteri, M. Casale, Class-modeling techniques, classic and new, for old and new problems, Chemom. Intel. Lab. Syst. 93 (2) (2008) 132–148, https://doi.org/10.1016/j.chemolab.2008.05.003.
[2] Informe de consumo de España en 2023. Ministerio de agricultura, pesca y alimentación. https://www.mapa.gob.es/es/alimentacion/temas/consumo-tendencias/panel-de-consumo-alimentario/ultimos-datos/ (accessed 24 June 2024).
[3] A. Stachniuk, E. Fornal, Liquid chromatography-mass spectrometry in the analysis of pesticide residues in food, Food Anal. Methods 9 (2016) 1654–1665, https://doi.org/10.1007/s12161-015-0342-0.
[4] Y. Chen, Y. Yu, X. Liu, Y. Yang, P. Lu, D. Hu, Development and validation of a liquid chromatography–tandem mass spectrometry method for multiresidue determination of 25 herbicides in soil and tobacco, Chromatographia 83 (2020) 229–239, https://doi.org/10.1007/s10337-019-03834-6.
[5] M. Blanco, M. Castillo, A. Peinado, R. Beneyto, Determination of low analyte concentrations by near-infrared spectroscopy: effect of spectral pretreatments and estimation of multivariate detection limits, Anal. Chim. Acta 581 (2) (2007) 318–323, https://doi.org/10.1016/j.aca.2006.08.018.
[6] L. Salguero-Chaparro, A.J. Gaitán-Jurado, V. Ortiz-Somovilla, F. Peña-Rodríguez, Feasibility of using NIR spectroscopy to detect herbicide residues in intact olives, Food Control 30 (2) (2013) 504–509, https://doi.org/10.1016/j.foodcont.2012.07.045.
[7] A.J. Márquez, G.B. Maza, 'In Situ' olive ripening monitoritation by low-cost handheld NIR, Smart Agricultural Technology 5 (2023) 100233, https://doi.org/10.1016/j.atech.2023.100233.
[8] O.Y. Rodionova, P. Oliveri, A.L. Pomerantsev, Rigorous and compliant approaches to one-class classification, Chemom. Intel. Lab. Syst. 159 (2016) 89–96, https://doi.org/10.1016/j.chemolab.2016.10.002.
[9] S. Wold, M. Sjöström, L. Eriksson, PLS-regression: a basic tool of chemometrics, Chemom. Intel. Lab. Syst. 58 (2) (2001) 109–130, https://doi.org/10.1016/S0169-7439(01)00155-1.
[10] L. Stahle, S. Wold, J. Chemometrics 1 (1987) 185, https://doi.org/10.1002/cem.1180010306.
[11] M. Barker, W. Rayens, Partial least squares for discrimination, Journal of Chemometrics: A Journal of the Chemometrics Society 17 (3) (2003) 166–173, https://doi.org/10.1002/cem.785.
[12] M.C. Ortiz, L.A. Sarabia, R. García-Rey, M.D. Luque de Castro, Sensitivity and specificity of PLS-class modelling for five sensory characteristics of dry-cured ham using visible and near infrared spectroscopy, Anal. Chim. Acta 558 (2006) 125, https://doi.org/10.1016/j.aca.2005.11.038.
[13] M.S. Sánchez, M.C. Ortiz, L.A. Sarabia, V. Busto, Class-modelling techniques that optimize the probabilities of false noncompliance and false compliance, Chemom. Intel. Lab. Syst. 103 (2010) 25–42, https://doi.org/10.1016/j.chemolab.2010.05.007.
[14] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes, Pattern Recogn. 44 (8) (2011) 1761–1776, https://doi.org/10.1016/j.patcog.2011.01.017.
[15] E.L. Allwein, R.E. Schapire, Y. Singer, Reducing multiclass to binary: A unifying approach for margin classifiers, J. Mach. Learn. Res. (2000) 113–141.
[16] T.G. Dietterich, G. Bakiri, Solving multiclass learning problems via error-correcting output codes, J. Artif. Intell. Res. 2 (1995) 263–286, https://doi.org/10.1613/jair.105.
[17] O. Valencia, M.C. Ortiz, S. Ruiz, M.S. Sánchez, L.A. Sarabia, Simultaneous class-modelling in chemometrics: A generalization of Partial Least Squares class modelling for more than two classes by using error correcting output code matrices, Chemom. Intel. Lab. Syst. 227 (2022) 104614, https://doi.org/10.1016/j.chemolab.2022.104614.
[18] D. Castro-Reigía, M.C. Ortiz, S. Sanllorente, I. García, L.A. Sarabia, PLS class modelling using error correction output code matrices, entropy and NIR spectroscopy to detect deficiencies in pastry doughs, Chemom. Intel. Lab. Syst. 105092 (2024), https://doi.org/10.1016/j.chemolab.2024.105092.
[19] P. Sarabia, A. Araujo, L.A. Sarabia, M.D.L.C. Ortiz, Electromyography Gesture Model Classifier for Fault-Tolerant-Embedded Devices by Means of Partial Least Square Class Modelling Error Correcting Output Codes (PLS-ECOC), Algorithms 16 (3) (2023) 149, https://doi.org/10.3390/a16030149.

[20] D. Castro-Reigía, I. García, S. Sanllorente, L.A. Sarabia, J.M. Amigo, M.C. Ortiz, Bread fermentation monitoring through NIR spectroscopy and PLS-DA. Determining the optimal fermentation point in bread doughs, J. Food Eng. 361 (2024) 111738, https://doi.org/10.1016/j.jfoodeng.2023.111738.
[21] L. Marchi, I. Krylov, R.T. Roginski, B. Wise, F. Di Donato, S. Nieto-Ortega, J.F. Q. Pereira, R. Bro, Automatic hierarchical model builder, J. Chemom. 36 (12) (2022) e3455.
[22] R. Voccio, C. Malegori, P. Oliveri, F. Branduani, M. Arimondi, A. Bernardi, G. Luciano, M. Cettolin, Combining PLS-DA and SIMCA on NIR data for classifying raw materials for tyre industry: A hierarchical classification model, Chemom. Intel. Lab. Syst. 250 (2024) 105150, https://doi.org/10.1016/j.chemolab.2024.105150.
[23] L. Cuadros-Rodríguez, E. Perez-Castano, C. Ruiz-Samblas, Quality performance metrics in multivariate classification methods for qualitative analysis, Trends Anal. Chem. 80 (2016) 612–624, https://doi.org/10.1016/j.trac.2016.04.021.
[24] D. Ballabio, F. Grisoni, R. Todeschini, Multivariate comparison of classification performance measures, Chemom. Intel. Lab. Syst. 174 (2018) 33–44, https://doi.org/10.1016/j.chemolab.2017.12.004.
[25] O. Valencia, M.C. Ortiz, M.S. Sánchez, L.A. Sarabia, A modified entropy-based performance criterion for class-modelling with multiple classes, Chemom. Intel. Lab. Syst. 217 (2021) 104423, https://doi.org/10.1016/j.chemolab.2021.104423.
[26] AOTECH. Advanced Optical Technologies. https://www.aotech.es/ (accessed 25 June 2024).
[27] I.T. Jolliffe, Principal Component Analysis, Second Edition, Springer Verlag, New York, 2002.
[28] B.M. Wise, N.B. Gallagher, R. Bro, J.M. Shaver, W. Winding, R.S. Koch, PLS Toolbox 9.1.1, Eigenvector Research Inc., Wenatchee, WA, USA, 2023.
[29] T. Singh, N.M. Garg, S.R.S. Iyengar, Nondestructive identification of barley seeds variety using near-infrared hyperspectral imaging coupled with convolutional neural network, J. Food Process Eng 44 (10) (2021) e13821.
[30] Wand, M.P., & Jones, M.C. Kernel Smoothing, Monographs on Statistical an Applied Probability, 60; Springer-Science-Business Media: New York, NY, USA, 1995.
[31] R.W. Kennard, L.A. Stone, Computer aided design of experiments, Technometrics 11 (1969) 137–148, https://doi.org/10.1080/00401706.1969.10490666.
[32] J.M. Wei, X.Y. Yuan, Q.H. Hu, S.Q. Wang, A novel measure for evaluating classifiers, Expert Syst. Appl. 37 (2010) 3799–3809, https://doi.org/10.1016/j.eswa.2009.11.040.
[33] R. Delgado, J.D. Núñez-González, Enhancing Confusion Entropy (CEN) for binary and multiclass classification, PLoS One 14 (2019) e0210264.
[34] MATLAB, MATLAB Version 9.9.0 (R2020b), The Mathworks Inc., Natick, Massachusetts, 2022. https://www.mathworks.com.
[35] Sánchez M.S., Valencia O., Ruiz S., Ortiz M.C., Sarabia L.A. DMCEN a MATLAB Function to Evaluate the Entropy Improvement provided by a Multivariate K-Class-Model. Available online: dmcen - File Exchange - MATLAB Central (mathworks.com). (accessed 24 June 2024).
[36] M. Schoot, C. Kapper, G.H. van Kollenburg, G.J. Postma, G. van Kessel, L. M. Buydens, J.J. Jansen, Investigating the need for preprocessing of near-infrared spectroscopic data as a function of sample size, Chemom. Intel. Lab. Syst. 204 (2020) 104105, https://doi.org/10.1016/j.chemolab.2020.104105.
[37] C. Mas, L. Rubio, L. Valverde-Som, L.A. Sarabia, M.C. Ortiz, Impact of the pretreatment of ATR-FTIR signals on the figures of merit when PLS is used, Chemom. Intel. Lab. Syst. 201 (2020) 104006, https://doi.org/10.1016/j.chemolab.2020.104006.
[38] J. Engel, J. Gerretzen, E. Szymańska, J.J. Jansen, G. Downey, L. Blanchet, L. M. Buydens, Breaking with trends in pre-processing? Trends Anal. Chem. 50 (2013) 96–106, https://doi.org/10.1016/j.trac.2013.04.015.