The applicability of vibrational spectroscopy and multivariate analysis for the characterization of animal feed where the reference values do not follow a normal distribution: A new chemometric challenge posed at the 'Chimiométrie 2019' congress

J. A. Fernández Pierna[1], A. Laborde[2], L. Lakhal[3], M. Lesnoff[4], M. Martin[5], Y. Roggo[6], P. Dardenne[7]

[1] Walloon Agricultural Research Centre, Valorisation of Agricultural Products Department, Food and Feed Quality Unit, Gembloux, Belgium.

[2] UMR914 PNCA INRA/AgroParisTech/Université Paris-Saclay, 75231 Paris, France

[3] Institut Polytechnique UniSalle, 19 rue Pierre Waguet - BP 30313 - F-60026 Beauvais France

[4] CIRAD Campus de Baillarguet, TA C-112 / A (Bât. A), 34398 Montpellier, France

[5] Institut de statistique, biostatistique et sciences actuarielles, ISBA - LIDAM - UCLouvain Voie du Roman Pays 20 bte L1.04.01 B-1348 Louvain-la-Neuve, Belgium

[6] y_roggo@hotmail.com

[7] NIRS & Chemometrics Consultant, 7, rue du Sacré Cœur, 5590- Leignon, Belgium.

Abstract

A chemometric challenge was posed at the annual 'Chimiométrie' congress organized by the French Chemometrics Society in February 2019. The congress was held in Montpellier and the data relating to the challenge are available on the congress website (https://chemom2019.sciencesconf.org/). The aim of the challenge was to test the ability of congress participants for the characterization of animal feed by NIR when the reference values do not follow a normal distribution for three different ingredients. This paper summarizes the five best approaches put forward by participants.

1 Introduction

As in previous years [1-5], a challenge was posed at the 'Chimiometrie' congress held in Montpellier in February 2019, concerning the applicability of spectroscopy and multivariate analysis for the characterization of animal feed where the reference values do not follow a

normal distribution. In particular, part of a dataset provided by the University of Córdoba (UCO), Spain, has been proposed as a challenge for the Chimiométrie 2019 conference participants [6].

When dealing with multivariate analysis, it is quite often the case that the distribution of reference values does not follow a standard distribution and then a classical regression model fails. For those situations, Fearn et al. proposed a Bayesian approach where the aim was to solve such situations [7]. For this, a model explaining the dependence of spectral data on reference values is combined with a prior distribution representing beliefs about the composition of the sample to be predicted. In their work, and using the same data as [6], they have proved that a Bayesian approach could be a relevant technique to predict the percentage of ingredients in a complete feed.

The aim of the challenge at the 'Chimiometrie' congress was to predict three different ingredients used for the production of animal feed through multivariate analysis; and then to predict the blind spectra of an independent test set.

This article presents the five best approaches among the seven solutions received.

2 Dataset and Challenge

The spectra are from feed (farm animal feed) samples with known composition and measured in reflection mode. No chemical composition is given, but the percentages of the different ingredients are given. As the choice of ingredients is very large in feed plants, only three products were used: soya oil (y1), lucern (y2) and barley (y3) contents. The participants obtained the NIR spectra and the reference values for the calibration set and only the NIR spectra for the test set. They were not informed about the products nor the wavelength range. The spectra had been reduced to 550 data points (1300-2398, /2nm). To make data more

challenging, the spectra of the test set were modified by shifting the wavelength axis by half a nanometer.

The test set used in the Fearn's paper [7] was only 100 spectra with a calibration set of 7523 spectra. In this challenge, the test set was included in the calibration set and a new test set of 600 spectra was randomly selected. Eight spectra with high Mahalanobis distances were removed before selecting the test set. Then, the remaining 6915 spectra constituted the final calibration set.

The final criterion to order the response was the average of the relative errors (equation 1):

$$RE = \frac{RMSEP\_C1}{Mean\_C1} + \frac{RMSEP\_C2}{Mean\_C2} + \frac{RMSEP\_C3}{Mean\_C3} \quad \text{(Equation 1)}$$

where C1, C2 and C3 correspond to the percentage of Ingredient y1, y2 and y3 respectively and RMSEP is the Root Mean Squared Error of Prediction.

$$RMSEP = \sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{n}} \quad \text{(Equation 2)}$$

where $\boldsymbol{y_i}$ refers to the reference values, $\boldsymbol{\hat{y}_i}$ refers to the NIR estimated values and $\boldsymbol{n}$ the number of validation spectra.

3 Results

3.1 Participant 1

First of all, a series of preprocessing techniques have been applied to the dataset that was then randomly split into training (95%) and test set (5%). The predictive ability was evaluated based on the RMSEP of the test set, after a 10-fold Cross-Validation (CV) training

of PLS-regressions applied to each response. The best strategy, having the minimum sum of RMSEP across all the responses, was to apply the standard normal variate (SNV) followed by linear detrend and standardization (autoscale).

After pre-processing, the calibration set was randomly split into two equal datasets: subset1 and subset2 to independently train the two following steps of the predictive analysis. Within each subset, 95% of the data were used for training and the remaining 5% were used as a test set. For the two steps, several classification and regression methods were trained and their predictive ability compared.

Given the high frequency of 0 values in the responses, the first step was to apply a binary classification over the categorized responses [0, >0]. For each response, the candidate models were trained with the training set of subset1: Partial Least Squares regression (PLS-R – 10-fold CV; Latent Variables = 12(y1), 5(y2), 6(y3)), Support Vector Machines for classification (SVM-C – 10-fold CV; radial kernel, cost = 10), Random Forest for classification (RF-C - mtry = 23, ntree = 500), and k-Nearest Neighbors (k-NN - leave-one-out CV, k = 3). As the objective was to correctly identify the 0-valued responses, based on the test set, the criterion to select the best classifier was the one maximizing the sensitivity (TP/(TP+FN)) and the Positive Predictive Value (PPV = TP/(TP+FP)):

$$Positive\ Predictive\ Value = \frac{TP}{(TP+FN)} + \frac{TP}{(TP+FP)} \qquad \text{(Equation 3)}$$

Where TP, FN and FP represent the True Positive rate, the False Negative rate and the False Positive rate respectively.

The random forest classification performed best for y1 and y2, whereas SVM had the highest classification criterion value for y3. These selected methods were then applied to subset2 to extract the predicted 0 response values.

During the second step of the analysis, the remaining non-0 observations from subset2 were split into training and test sets. Since the responses also had the particularity of having a low number of unique value levels, regression and classification algorithms were both considered as candidate methods and trained in the same way as during the first step: PLS-R (# Latent Variables = 9(y1), 4(y2), 3(y3)), SVM-R (radial kernel, cost=10), SVM-C (radial kernel, cost=10), Random Forest for regression (RF-R - mtry = 366(y1), 23(y2), 92(y3)), ntree = 500), and RF-C (mtry = 23, ntree = 500). The overall method minimizing the RMSEP over the test sets and for all the responses was the SVM-R.

The best strategies were finally applied to the validation set and the remaining negative predicted values were set to 0, as only positive or null values were allowed.

All the analyses have been conducted in R and the applied R functions are here given explicitly for each method: pracma::detrend() for linear detrend, pls::mvr() for PLS, e1071::svm() for SVM and randomForest::randomForest() for RF.

3.2 Participant 2

The data analysis was performed in two steps. The first part was exploratory: the spectra data were visualized, the cluster in the spectral data set was evaluated as well as the distribution of the reference values in order to have an overview of the data quality. The second step was the selection of the regression method. Several methods have been evaluated: PLS regression, Neural Network and Extreme Gradient Boosting (XGBboost), which is an efficient implementation of the gradient boosting framework from Chen & Guestrin [8]. The best method has been selected according to the RMSEP calculated in a reduced test set selected using the Kennard and Stone algorithm from the calibration set[9]. The method giving the best results was the Xgboost. The procedure performed using the R software included spectral binning in

order to reduce the number of wavelengths. This has been applied during the screening of methods in order to improve the computation speed. Then, classical preprocessing methods were tested and SNV and first derivative Savitzky Golay (p = 3, w = 11, m = 1) was selected. After data visualization using spectra plot, principal component analysis and kmeans, a quantitative prediction was performed. Screening of the PLS, Neural Network and Decision Trees methods were evaluated. The best method was regression tree with gradient boosting using the constraint of non-negativity of Y.

Table 1 shows the different parameters that have been evaluated during the optimization process. The optimization has been performed one parameter at a time. There are in general two ways to control overfitting in XGBoost: 1) by directly controlling model complexity (optimization of max_depth, min_child_weight and gamma.) and 2) by adding randomness to make training robust to noise (optimization of subsample by row and columns and learning rate eta).

Table 1

3.3 Participant 3

Since a main characteristic of the data was the presence (in large numbers) of 0s in the three response variables (y1, y2, y3), a specific two-step sequential modelling (similar to the "hurdle models" approach proposed in econometric studies [10] and subsequently used in many other fields) was applied to each variable y. In the first step (say Step1), the presence (i.e. y > 0) vs. absence (i.e. y = 0) of the ingredient was predicted with a qualitative binary discriminant model. For predictions returning "y > 0" in the Step1, the second step (say Step2) was to predict the positive responses with a quantitative model.

A preliminary data exploration showed some heterogeneity in the spectra, suspected to generate nonlinearity between responses y and the spectra and suggesting using local prediction approaches. K-nearest neighbors locally weighted PLS-DA (KNN-LWPLS-DA) and PLS-R (KNN-LWPLS-R) models were used for Step1 and Step2, respectively. LWPLS-R [11, 12] is a particular case of weighted PLSR (WPLS-R). WPLS-R is a generalization of PLS-R where a statistical weight, different from the standard 1/n, is given to each of the n calibration observations for calculating the PLS scores and loadings, and the predictions. In LWPLS-R, the weights depend on the dissimilarity (defined in the present study by Mahalanobis distances) with the new observation to predict [13]. A simple approach of LWPLS-DA is to implement a LWPLS2-R on the dummy variables matrix created from the discrete variable y and then to select, for each observation to predict, the column of the LWPLS2-R prediction having the highest value.

In the literature, the usual LWPLS strategies often consist of fitting, for each new observation, a LWPLS using the entire set of the n calibration observations. An alternative and faster strategy [13], used in this challenge, is to do a pre-selection of k nearest neighbors of the observation to predict (KNN selection) and then only apply LWPLS to the k neighbors. In addition, a dimension-reduction of the original X-data was performed by a global PLS (before running the KNN-LWPLS described above), as suggested in Shen et al. [14]. The idea is to run the models on these pre-(global)-scores instead of on the original X. This again decreases the calculation time and, in some cases, removes uninformative noise and increases the stability of the results.

Before running the models, the X-data were pre-processed by a SNV transformation followed by a second derivative with a Savistky-Golay filter (polynomial = 3 and window = 21). The model parameters (number of neighbors, number of PLS components, sharpness of the weight function for WPLS) were optimized using an “oriented” validation strategy: a

pseudo-test set (m = 591 observations) was built by selecting in the calibration set CAL the first nearest neighbor of each observation of the validation set VAL (using Mahalanobis distances calculated on 15 PCA scores computed on the un-preprocessed spectra). The principle is to get a validation set (for parameters optimization) as similar as possible as the dataset (VAL) to predict. Although not really demonstrated, this oriented strategy seems more efficient for heterogeneous data than doing naive cross-validation. The results of the optimized models on the pseudo-test are shown in Table 2.

Table 2

Other modelling approaches (simple weighted KNN, non-parametric Bayesian predictions such as in Fearn et al 2010 [1], etc.) were also assessed on this pseudo-test but all returned higher RMSE values than KNN-LWPLS and were therefore not considered for the final VAL predictions. All this work was implemented with the R package *rnirs* available at https://github.com/mlesnoff/rnirs.

3.4 Participant 4

Firstly, data were processed with SNV correction followed by a Savitsky-Golay filter to obtain the first derivative of the spectra (2nd order polynomial and 7 points window). A principal component analysis (PCA) was performed on the training dataset so that loadings and scores were extracted from the first to the fifteenth component. The following describes the applied procedure for each test sample: The first fifteen scores of the test spectrum are calculated using the PCA loadings. These scores are used to calculate the Euclidian distance with all the training samples. Then samples are ordered in ascending order of distance and the neighbors with a distance lower than 0.9 (determined by cross-validation with the training set) are chosen. In the case that less than 300 neighbors satisfy this condition, the next closest

neighbors are integrated into the subset until 300 individuals are chosen. If the y values of the resulting dataset contain 50% of zero, the prediction is directly set to zero. Then this dataset is randomly split (sub-train: 2/3, sub-test: 1/3) to perform a PLS regression calibration. The number of latent variables is set as the one that minimizes the RMSEP. This procedure is applied for the three y values to predict. The sample test is predicted using the local partial least square regression model. Finally, all negative predictions were set to 0.

3.5 Participant 5

The Multi-layer Perceptron (MLP) is a non-linear statistical data modeling tool that tries to simulate the functions of biological neural networks. It consists of an interconnected collection of simple processing elements or artificial neurons and processes information in a connectionist approach to computation [15]. MLP is generally considered to be an adaptive system that changes its structure in response to external or internal information that flows through the network during the learning phase.

While establishing the MLP model, all the data were normalized between 0 and 1. For normalization, the following equation was used:

$$y_{nor} = \frac{y - y_{min}}{y_{max} - y_{min}} \quad \text{(Equation 4)}$$

To obtain real values from the normalized values, “y” value was calculated using the same formula.

Normalized data were divided into two datasets for training and test. In the training set, 5532 NIR spectra were used, whereas 1383 NIR spectra were used in the test set.

The choice of model parameters was made on the basis of the minimum error on internal validation set. The optimal model parameters were found to be one hidden layer with *7* neurons, trained for 13 epochs with learning rate $\eta = 0.50$.

The MLP model was trained with a backpropagation learning algorithm based on the Levenberg Marquardt algorithm which minimizes the total error by varying the weights in order to enhance the network performance. Training of the network was continued until the test error reached the determined tolerance value.

The model was tested with test data after training. The model performance was evaluated by the root mean square error (RMSE) and the coefficient of determination (R^2).

4 Debriefing

No one among the participants discovered the X shift. In any case, it was possible to see the shift in several ways. The first one was by projecting the test set on the PC scores calculated for the calibration set. The shift was not visible on the first PC's but was obvious for the 13^{th} to 20^{th} PC's. A simpler and easier way was to see the classical T2 Hotelling vs Q (X residuals) plot (figure 1).

Figure 1

A third way was to plot the spectra for the calibration set together with those of the test set with a zoom. It is surprising that none of the 7 participants noticed this shift.

Before predicting a test set, it is clever to select from the calibration set (when this one is much wider) the spectra which are the most similar to those of the test spectra. This was done by choosing the closest samples based on their standardized Mahalanobis distances. A set of

600 spectra was extracted from the cal set. The 600 selected and the 600 of the test are averaged to produce two spectra which are plotted in Figure 2.

Figure 2

Warping methods could have been used to correct the shift on the wavelength axis but with this dataset a simple homemade method was applied. Firstly, a gap first derivative was applied on the two mean spectra presented in the previous figure. From these two spectra, a certain number of peaks (local maxima - 26) can be extracted from both spectra. Some are identical between calibration and validation, several are shifted to the right by one unit (Table 3).

Table 3

A polynomial function of second degree fits five points at each peak and the first derivative of the polynomial gives the exact position of the peak which is calculated at –b/2c (first derivative of y=a+bx+cx2 at the crossing point with y=0). The median of the 26 differences of the exact peak positions was calculated to be equal to 0.2496 and rounded to 0.25. Then, the validation spectra were interpolated from 1.00 to 550.25 by a step of 0.25. The first column was removed and then the absorbance at 1.25 became the ones at point 1. Then subsequent columns were selected every 4 (1.25, 2.25, 3.25...) to obtain 550 variables. After correction, the M distances for the calibration and validation sets show, then, the same distributions.

After such shift correction, and based on the 600 spectra selected, the local algorithm of the WinISI package (Foss, Hilerod, Dk) was optimized to find the best combination of the

number of samples, the number of PLS factors to be ignored and the maximum number of factors for each PLS model. Table 4 gives the results of this optimization and within parentheses the stats on the validation set.

Table 4

Table 5 shows the results for all participants. The first three columns represent the RMSEP for the three available y (Soy oil, lucern and barley) and the fourth column represents the relative error as calculated in equation 1 for the data as received from the participants, i.e. without the shift correction.

Table 5

In a later stage, after the conference, the participants received the data corrected for the X shift and the same procedure was applied again. The results are indicated also in the second right part of Table 5. It can be observed that the impact of changing half a nanometer in the spectra has a large influence in the final results, with RMSEP of almost double the value. Most of the differences of the RMSEP are in biases: the random errors (SEPC) remain similar on corrected and uncorrected data. Moreover a 2-by-2 t student comparison has been done. Table 6 shows the results indicating whether the proposed procedures are significantly different or not.

Table 6

It is important to point out that the shift obviously had a big impact on all the methods using the distances between the observations, and in particular KNN-LWPLS that gave very good results on the un-shifted data.

Conclusion

Dealing with a large dataset, the 2019 challenge demonstrated the efficiency of discriminant analyses and local regressions or nonlinear regressions respectively to detect the presence of the ingredients and quantify them. As in previous editions, the aim of the paper was, not to compare different techniques, nor to indicate whether a procedure is better than another one, but just to show different alternatives for the same problem. The differences between the methods are relatively small in practice and they are ranked in a different order regarding whether the data are shifted or not. The main surprise is that no participant noticed the shift and this shows again that very good chemometricians may forget a simple principle that has been repeated for a long time: when using NIRS, look at the spectra.

Acknowledgments

We thank the University of Córdoba (UCO) in Spain for supplying the spectra included in this paper.

We would also like to thank all the participants who spent time analyzing the data and presenting their results.

,

References

[1] Dardenne P., Fernández Pierna J.A. (2006) ‘A NIR data set is the object of a chemometric contest at 'Chimiométrie 2004'’. Chemometrics and Intelligent Laboratory Systems 80, 236-242.

[2] Fernández Pierna J.A, Dardenne P. (2007) ‘Chemometric contest at 'Chimiométrie 2005': a discrimination study’. Chemometrics and Intelligent Laboratory Systems 86, 219-223.

[3] Fernández Pierna J.A., Dardenne P. (2008) ‘Soil parameter quantification by NIRS as a chemometric challenge at 'Chimiométrie 2006'’. Chemometrics and Intelligent Laboratory Systems 91, 94-98.

[4] Fernández Pierna J.A., Chauchard F., Preys S., Roger J.M., Galtier O., Baeten V., Dardenne P. (2011) ‘How to build a robust model with only a few reference values: a chemometric challenge at 'Chimiométrie 2007'’. Chemometrics and Intelligent Laboratory Systems 106 (2), 152-159.

[5] Fernández Pierna J.A., Duval H., Valderrama P., Rutledge D.N., Baeten V., Dardenne P. (2011) ‘A case study of extrapolation in NIR modelling – A chemometric challenge at ‘Chimiométrie 2009’’. Chemometrics and Intelligent Laboratory Systems 106 (2), 205-209.

[6] Pérez-Marín, D., Fearn, T., Guerrero, J.E., Garrido-Varo, A. (2012) ‘Improving NIRS predictions of ingredient composition in compound feedingstuffs using Bayesian non-parametric calibrations’. Chemometrics and Intelligent Laboratory Systems 110, 108–112. https://doi.org/10.1016/j.chemolab.2011.10.007.

[7] Fearn T., Pérez Marín D., Garrido Varo, A., Guerrero Ginel, J.E. (2010) ‘Inverse, classical, empirical and non-parametric calibrations in a Bayesian framework’, J. Near Infrared Spectrosc. 18, 27-38.

[8] Chen T., Guestrin C. (2016) 'XGBoost: A Scalable Tree Boosting System', Proceedings of the 22nd ACM SIGKDD International Conference of Knoxledge Discovery and Data Mining, San Francisco, California USA, 785-794.

[9] Kennard R.W., Stone L.A. (1969) 'Computer aided design of experiments', Technometrics, 11, 137-148.

[10] Cragg, J.G., (1971) 'Some Statistical Models for Limited Dependent Variables with Application to the Demand for Durable Goods'. Econometrica 39, 829–844.

[11] Schaal S., Atkeson C.G., Vijayakumar, S. (2002) 'Scalable Techniques from Nonparametric Statistics for Real Time Robot Learning'. Appl. Intell. 17, 49–60.

[12] Sicard E., Sabatier R. (2006) 'Theoretical framework for local PLS1 regression, and application to a rainfall data set'. Comput. Stat. Data Anal. 51, 1393–1410.

[13] Lesnoff, M., Metz, M., Roger, J.M. (2019) 'Comparison of locally weighted PLS strategies for regression and discrimination on agronomic NIR data'. Submitted for publication to Journal of Chemometrics.

[14] Shen G., Lesnoff M., Baeten V., Dardenne P., Davrieux F., Ceballos H., Belalcazar J., Dufour D., Yang Z., Han L., Fernández Pierna J.A. (2019) 'Local partial least squares based on global PLS scores'. J. of Chemom. 33 (5) e3117.

[15] Han, J., Kamber, M. (2001) *Data Mining Concepts and Techniques*, Academic Press, ISBN 81-7867-023-2.

Figure 1 - Influence plot T2 vs Q (circle = cal set, diamond = val set)

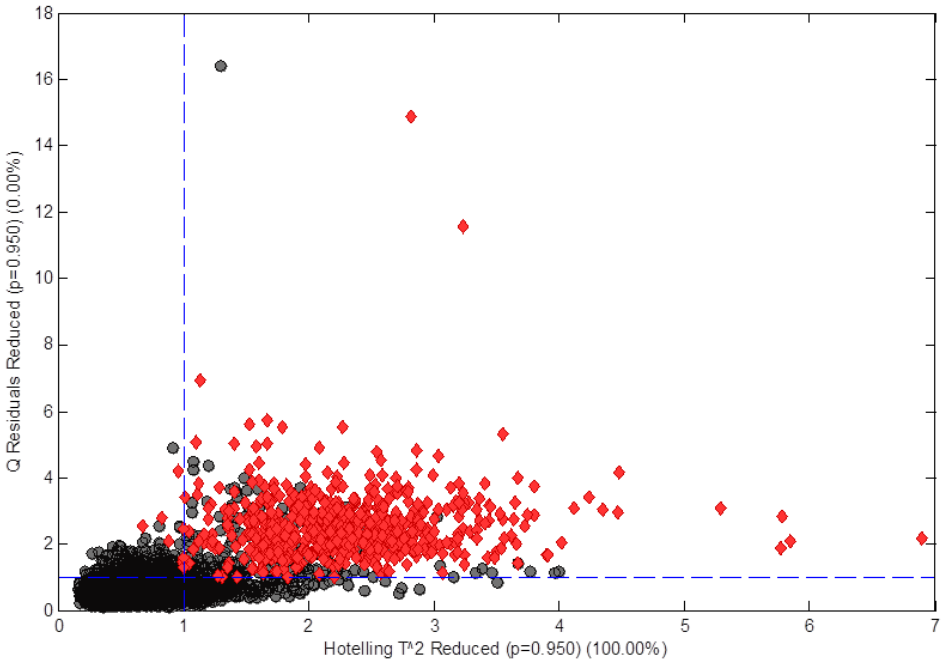

Figure 2 - Mean spectra of the calibration and test sets

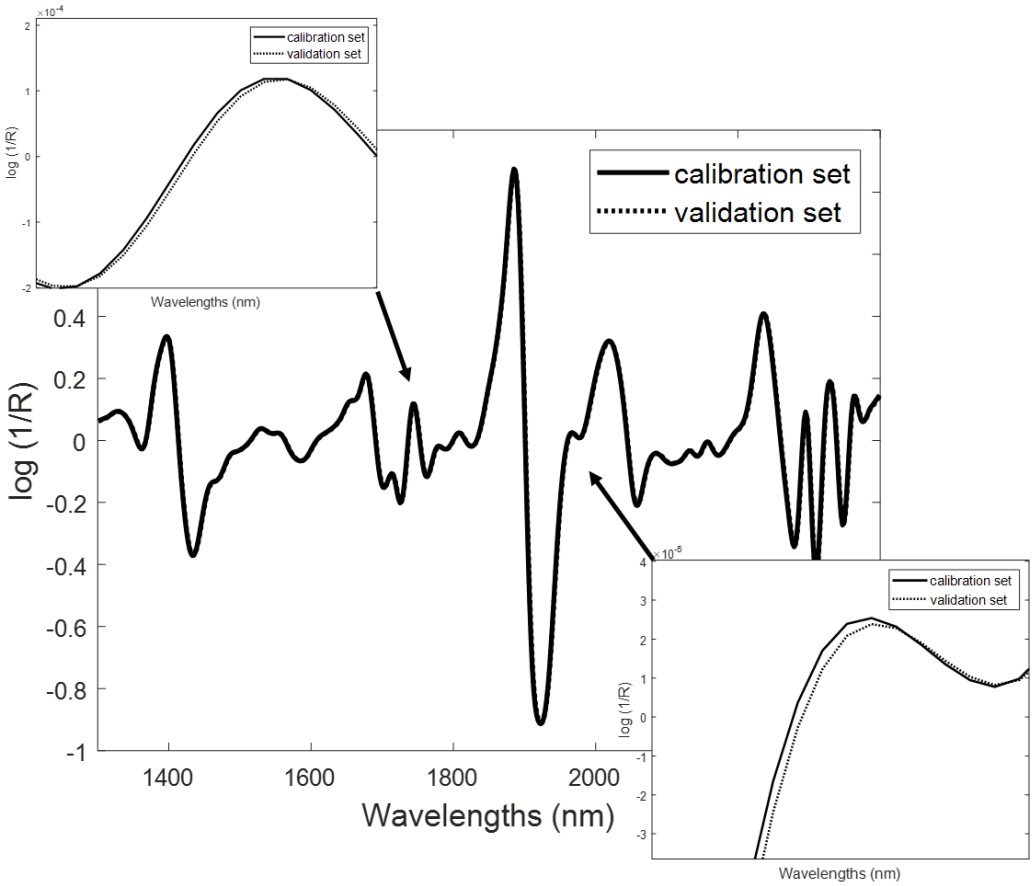

Table 1 – Parameter optimization for Participant 2

Parameters and definition	Tested range during the optimization	Selected values
Learning rate (eta) Step size shrinkage used in update to prevents overfitting. Eta shrinks the feature weights to make the boosting process more conservative.(*)	Between 0.01 and 0.3	0.1
Maximum depth of a tree (max depth) Increasing this value will make the model more complex and more likely to overfit. (*)	Between 2 and 10	3 for variables V1 and V3 2 for V2
Gamma Minimum loss reduction required to make a further partition on a leaf node of the tree. The larger gamma is, the more conservative the algorithm will be(*)	Between 0.001 and 1000	0.01 for variables V1 and V2 10 for V3
Lambda L2 regularization term on weights - Increasing this value will make model more conservative (*)	Between 0 and 100	10
Minimum child weight this simply corresponds to minimum number of instances needed to be in each node. The larger min_child_weight is, the more conservative the algorithm will be (*)	Between 0 and 100	50
Subsample (i.e. Sample selection) Setting it to 0.5 means that XGBoost would randomly sample half of the training data prior to growing trees. and this will prevent overfitting. Subsampling will occur once in every boosting iteration.(*)	Between 0.25 and 1	0.75
Subsampling of columns (i.e Variable selection) (*)	Between 0.25 and 1	0.75

Table 2 - Results of calibrations on the pseudo-test for participant 3

	Response variable		
	***y* 1**	***y* 2**	***y* 3**
BINARY % Err	1.7	0	3.9
QUANT. RMSEP	0.28	1.74	2.87
TOTAL RMSEP	0 .14	0 .87	2.76

Table 3 – Peaks (local maxima) extracted from mean cal and val spectra

PEAKS CAL	28	35	57	108	132	154	196	218	227	250	259	267	301	330	338	340	373	427	430	445	480	496	501	510	520	528
PEAKS VAL	28	35	57	109	133	154	196	219	227	251	259	268	301	330	339	341	373	427	431	445	481	496	501	510	521	529

Table 4 - Optimization results using Local Winisi – Foss (in parentheses with the stats on the corrected validation set).

	Validation set corrected				
	range	**mean CAL**	**RMSEP**	**R2**	**RMSEP (%)**
SoyOil	0 - 5.5%	0.38	0.34 (0.37)	0.83	0.89
Lucern	0 - 40%	8.71	1.11 (1.15)	0.99	0.13
Barley	0 - 52%	14.3	3.44 (4.07)	0.94	0.24

Table 5 – Final results for all participants.

	Validation set uncorrected				Validation set corrected				
	RMSEP			RE (in %)	RMSEP			RE (in %)	Method used
	SoyOil	Lucern	Barley		SoyOil	Lucern	Barley		
Participant 1	0.54	2.03	6.46	1.15	0.51	1.72	4.80	0.96	RF,SVM
Participant 2	0.54	2.75	6.08	1.25	0.45	2.17	5.72	1.06	XGBboost
Participant 3	0.55	2.65	7.01	1.29	0.26	0.99	2.67	0.52	KNN-LWPLS(DA/R)
Participant 4	0.58	2.53	7.06	1.30	0.46	1.66	4.15	0.87	Local PLS
Participant 5	0.60	3.38	9.01	1.59	0.37	1.31	4.43	0.77	ANN

Table 6 – 2-by-2 t-student tests comparison

SoyOil	Participant 1	Participant 2	Participant 3	Participant 4
Participant 2	NS			
Participant 3	NS	NS		
Participant 4	S	S	NS	
Participant 5	S	S	S	NS
Lucerne	Participant 1	Participant 2	Participant 3	Participant 4
Participant 2	S			
Participant 3	S	NS		
Participant 4	S	S	NS	
Participant 5	S	NS	S	S
Barley	Participant 1	Participant 2	Participant 3	Participant 4
Participant 2	NS			
Participant 3	S	S		
Participant 4	S	S	NS	
Participant 5	S	S	S	S

S: Significantly different

NS: Not significantly different