RoBoost-PLS2-R: An extension of RoBoost-PLSR method for multi-response

Maxime Metz, Maxime Ryckewaert, Sílvia Mas Garcia, Ryad Bendoula, Pierre Dardenne, Matthieu Lesnoff, Jean Michel Roger

RoBoost-PLS2-R : An extension of RoBoost-PLSR method for multi-response

Maxime Metz[a,b], Maxime Ryckewaert[a,b], Silvia Mas Garcia[a,b], Ryad Bendoula[a,b], Pierre Dardenne[d,b], Matthieu Lesnoff[c,b], Jean-Michel Roger[a,b]

[a] *ITAP Univ Montpellier INRAE Institut Agro Montpellier France*
[b] *ChemHouse Research Group Montpellier France*
[c] *SELMET Univ Montpellier CIRAD INRAE Montpellier SupAgro Montpellier France*
[d] *Wallon Agricultural Research Centre Gembloux*

Abstract

Recently, a novel robust PLSR method was developed to address the problem of outliers in the data. In this paper, an extension of this method, called RoBoost-PLS2-R is proposed to predict multi-response variables. Robustness and efficiency of this new approach have been validated on two simulated data sets and one real data set containing different outlier scenarios. Its performance was also compared with reference methods (PLS2-R and RSIMPLS) for predicting multi-response variables. Results confirm that RoBoost-PLS2-R greatly reduces prediction errors when data contain outliers. Prediction performances of RoBoost-PLS2-R are close to the optimal model (PLS2-R) calibrated without outliers and also to RSIMPLS method. This method seems to be a reliable and a competitive robust regression tool for predicting multi-response variables.

Keywords: Robust regression methods, outliers, multi-response, multivariate data analysis

---

1. Introduction

Partial Least Square Regression (PLSR) [1] is a common data analysis method and a well-established tool in chemometrics. PLSR calculates a linear relationship between explanatory variables ($\mathbf{X}$) and response variables ($\mathbf{Y}$). PLSR can be used to predict one response (PLS1) or several responses (PLS2). PLSR is particularly useful for processing high-dimensional data, especially when the number of explanatory variables exceeds the number of samples. This method is widely used in analytical chemistry for predicting constituent concentrations of a sample based on its spectrum obtained by spectroscopic techniques, such as near-infrared (NIR) spectroscopy, Fluorescence spectroscopy and ultraviolet (UV) spectroscopy. The PLSR model is known to be affected by the presence of atypical observations (outliers) in the data set. Outliers can negatively affect the calibration of PLSR models. To deal with outliers, several robust PLSR methods were proposed in the literature [2–12]. These methods were particularly developed to deal with outliers when the response matrix is uni-dimensional [13] (PLS1-R). However, robust methods that address the case of multi-responses (PLS2) are few. Among them, RSIMPLS is one of the most used method [14]. RSIMPLS proposes to robustly estimate the cross-covariance matrix $\mathbf{C_{xy}}$ and the empirical covariance matrix $\mathbf{C_x}$ used in SIMPLS algorithm. For this, a robust principal component analysis (ROBPCA) is performed on the concatenated data matrix of $\mathbf{X}$ and $\mathbf{Y}$. RSIMPLS uses additional information from the previous ROBPCA step to perform a reweighted multiple linear regression.

Recently, a new robust method called RoBoost-PLSR has been developed [15]. RoBoost-PLSR aims at determining the measure of relevance of the samples for PLSR model calibration. Indeed, in practical cases, the samples of a database are not defined as outliers, i.e. not relevant for the calibration of a PLSR model. RoBoost-PLSR proposes to calculate a weight on each latent variable to define the relevance of the samples. The relevance measurement is defined according to three criteria calculated for each latent variable (X-residuals, Y-residuals, leverage). This method has proven to be effective for outliers in both $\mathbf{Y}$ and $\mathbf{X}$. However, this algorithm was only developed for a one-dimensional PLSR response variable (PLS1). This paper contributes to the RoBoost-PLSR method which will be able to manage outliers in a multiple response context.

The first section introduces the extension of RoBoost-PLSR named RoBoost-PLS2-R and the associated algorithm. The following section presents the data and the methods used to evaluate and compare the predictive ability of RoBoost-PLS2-R. Finally, the prediction performance of RoBoost-PLS2-R and its comparison with standard methods are shown in the last section.

2. Notations

Capital bold characters will be used for matrices, *e.g.* $\mathbf{X}$; small bold characters for column vectors, *e.g.* $\mathbf{x}_j$ will denote the j^{th} column of $\mathbf{X}$; row vectors will be denoted by the transpose notation, *e.g.* $\mathbf{x}_i^{\mathrm{T}}$ will denote the i^{th} row of $\mathbf{X}$; italic characters will be used for scalars, *e.g.* matrix elements x_{ij} or indices i. Constant scalars will be denoted with italicised characters, *e.g.* number of samples n. $\mathbb{1}$ will represent a column vector of ones, of proper dimension. *med* defines the median. $\mathbf{X}$ and $\mathbf{Y}$ are the spectral and the responses matrices. g is the weight function. $\mathbf{D}$ is the matrix of sample weights where the diagonal of the matrix is the sample weight and the other terms are zero.

3. RoBoost-PLSR extension for multi-responses

3.1. Algorithm

The new algorithm allowing an extension in a multi-response context is the following :

Algorithm RoBoost-PLSR for K LV

For a definite number of K latent variables, the algorithm proceeds as described below :

1: Initialisation step

$$k = 1$$

$$\mathbf{D} = diag(d_1, d_2, ..., d_n) \text{ with } d_i = \frac{1}{n}$$

2: Center the data :

$$\mathbf{X}_k = \mathbf{X} - \mathbb{1}\mathbb{1}^{\mathrm{T}}\mathbf{D}\mathbf{X}$$

$$\mathbf{Y}_k = \mathbf{Y} - \mathbb{1}\mathbb{1}^{\mathrm{T}}\mathbf{D}\mathbf{Y}$$

3: Define $\mathbf{u}_k$ as an arbitrary column of $\mathbf{Y}$

4: Calculate one weighted latent variable NIPALS :

$$\mathbf{w}_k = \frac{\mathbf{X}_k^{\mathrm{T}}\mathbf{D}\mathbf{u}_k}{||\mathbf{X}_k^{\mathrm{T}}\mathbf{D}\mathbf{u}_k||}$$

$$\mathbf{t}_k = \mathbf{X}_k\mathbf{w}_k$$

$$\mathbf{p}_k = \frac{\mathbf{X}_k^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}{\mathbf{t}_k^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}$$

$$\mathbf{q}_k = \frac{\mathbf{Y}_k^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}{\mathbf{t}_\mathbf{k}^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}$$

$$c_k = \frac{\mathbf{u}_k^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}{\mathbf{t}_\mathbf{k}^{\mathrm{T}}\mathbf{D}\mathbf{t}_k}$$

5: Derive ($\mathbf{F}$), ($\mathbf{E}$), ($\mathbf{l}$) :

$$\mathbf{E} = \mathbf{X}_k - \mathbf{t}_k\mathbf{p}_k^{\mathrm{T}}$$

$$\mathbf{F} = \mathbf{Y}_k - \mathbf{t}_k\mathbf{q}_k^{\mathrm{T}}$$

$$\mathbf{l} = \mathbf{t}_k$$

6: Update the weights for each $i \in [1, n]$ sample :

$$d_i = \frac{1}{n} \times g(||\mathbf{e}_i||, \alpha) \times \prod_{j=1}^{m} g(f_{ij}, \beta), \times g(l_i, \gamma)$$

7: Return to (step (2) for k = 1, otherwise return to step (4)) until convergence of successive c's.

8: Deflation step

$$\mathbf{X}_{k+1} = \mathbf{X}_k - \mathbf{t}_k \mathbf{p}_k^{\mathrm{T}}$$

$$\mathbf{Y}_{k+1} = \mathbf{Y}_k - \mathbf{t}_k \mathbf{q}_k^{\mathrm{T}}$$

$$\mathbf{u}_{k+1} = \mathbf{Y}_k \mathbf{q}_k$$

set $k = k + 1 \rightarrow$ then go to step (4)

The regression coefficients resulting for K latent variables are estimated as follows :

$$\mathbf{B} = \mathbf{R}\mathbf{c}^{\mathrm{T}}$$

With $\mathbf{R}$:

$$\mathbf{R} = \mathbf{W}(\mathbf{P}^{\top}\mathbf{W})^{-1}$$

3.2. *Theoretical discussions*

The algorithm RoBoost-PLS2-R have similar properties to the algorithm proposed in [15], but also new properties :

- The RoBoost-PLS2-R framework is designed foremost to facilitate the leverage measurement. Leverage is defined as the distance to the centre of the model (see step 6 in the algorithm). In usual strategies, to define distances between the model centre and individuals, different metrics can be used. Euclidean or Mahalanobis distances between scores and the model centre are strategies commonly used in chemometrics. However, in the case of a Euclidean distance, the latest LVs could have a minor contribution to the leverage value. This is due to the decreasing magnitude of scores.Nevertheless, the predictive potential of these latest LVs may not be necessarily negligible. In the case of a Mahalanobis distance, contributions of all LVs become equal in the computation of the leverage value. This can be also detrimental, since the predictive potentials of the LVs are most usually uneven. Considering these limitations, RoBoost-PLSR proposes to estimate the sample leverage for each latent variable. This avoids the need to define specific metrics for the leverage calculation. However, the use of this strategy may make it difficult to assign a low weight to individuals with a leverage effect that is only identifiable with a large number of latent variables.

— The proposed method takes into account X-residuals (see step 6 in the algorithm). Usually only Y-residuals are considered in robust PLS approaches. The inclusion of these residuals provides additional information that cannot be expressed by leverage and Y-residuals alone.

— The algorithm proposed in this article provides regression coefficients. This makes the constructed RoBoost-PLSR models more easily interpretable. Contrary to the first algorithm proposed in [15], the rotation matrix $\mathbf{R}$ used to estimate the regression coefficients can be estimated. This is due to data centring which is only done for the first model with a single latent variable. In the previous algorithm, repeated centring of $\mathbf{X}$ and $\mathbf{Y}$ matrices led to a bias which made it impossible to estimate the rotation matrix.

— Like PLSR, RoBoost-PLSR makes it possible to deduce any of the 1 to K LVs models from the calibration of a single K LVs model. This preserves the operability during validation and parameterisation process of the RoBoost-PLSR method. Indeed, from this set of one-variable latent models it is possible to define the rotation matrix $\mathbf{R}$ which enables to compute all previous PLS models.

— The algorithm proposed in [15], determines the convergence with q.

However, $\mathbf{q}$ is multidimensional when $\mathbf{Y}$ is multidimensional. In the new algorithm convergence estimation is facilitated by using c which is a scalar when responses matrix $\mathbf{Y}$ is multidimensional (see step 7 in the algorithm).

— The weights of the sample according to the $\mathbf{Y}$-residuals are the product of the estimated weights for each $\mathbf{Y}$-variable (see step 6 in the algorithm). A specific sample weight for each residual of each $\mathbf{Y}$ variable is calculated and then multiply them to give an overall weight. This strategy enables sample weights to be estimated in a way that is appropriate to the multivariate nature of $\mathbf{Y}$. This strategy takes in consideration the fact that $\mathbf{Y}$ variables may have different variances. If this aspect is not taking into account, some outliers could be considered as inliers by the method. For instance, atypical samples on a specific variable of $\mathbf{Y}$ can mask the outliers of other columns of $\mathbf{Y}$ which present a lower variability. This strategy also allows a fast operation by applying the bisquare function on each column of Y-residuals matrix for each LV according to the β hyperparameter. Finally, the global weights associated with Y-residuals are defined as a product of each weight calculated on the Y-residual. This strategy of combining weights is a commonly used strategy. It is basically used to combine the weights calculated according to the three criteria (X-residuals, Y-residuals, leverage) in RoBoost-PLSR. However, different strategies are possible. Like calculating the Mahalanobis distances on $\mathbf{Y}$ or making a combination of weights different from the product. In particular, it is possible to perform a sum of weights, so that the weighting strategy can eliminate individuals who only have weights at 0 for each criterion.

— In this article, the weight function g is the bisquare function :

$$B(z_i) = (1 - z_i^2)^2 \text{ for } |z_i| < 1 \text{ and } B(z_i) = 0 \text{ for } |z_i| > 1$$

with z_i :

$$\frac{x_i}{c \times med(|\mathbf{x}|)}$$

However, any weight function can be considered and tested in order to improve the algorithm to obtain better predictive capacity. In RoBoost-PLS2-R x_i (associated with the bisquare function) is specific according to the chosen statistic. This means that when the weights are calculated according to the residuals of $\mathbf{X}$, x_i corresponds to the norm of the vector $\mathbf{e}_i$ and $\mathbf{x}$ to the norms of the individuals of $\mathbf{E}$. When the residuals $\mathbf{Y}$ are taken into account, x_i is the value of the residual y_{ij} and x is the vector of residuals f_j. Finally, the leverage effect is taken into account, x_i corresponds to the score of a latent variable t_{ik} and $\mathbf{x}$ is the vector of scores $\mathbf{t_k}$ for all samples. Furthermore, the constant c in the bisquares function corresponds to the parameters α, β and γ in step 6 of the algorithm. This constant has to be adjusted according to the type of outlier.

4. Materials and methods

4.1. Simulated Data

To evaluate the performance of RoBoost-PLS2-R in comparison with standard PLS2-R and RSIMPLS, two simulations were performed. The first simulation represents the Y-outlier case and the second simulation the X-outlier case. For each simulation, 1000 samples were generated according to the framework proposed by [16]. Among these samples, 200 outliers were generated. The spectral signatures used for the simulations were the spectral signatures of water, ethanol and glucose estimated in [16]. Using this approach, the matrix of explanatory variables ($\mathbf{X}$) was generated by :

$$\mathbf{X} = \mathbf{t_u}\mathbf{p_u}^t + \mathbf{T_d}\mathbf{P_d}^t + \mathbf{E} \tag{1}$$

And the relationship f between $\mathbf{X}$ and $\mathbf{Y}$ by :

$$\mathbf{Y} = f(\mathbf{t_u}) + \mathbf{F} \tag{2}$$

Where $\mathbf{p_u}$ is the spectral signature in the useful space and $\mathbf{P_d}$ are spectral signatures in the detrimental space. $\mathbf{t_u}$ and $\mathbf{T_d}$ are their associated contributions. The $\mathbf{E}$ and $\mathbf{F}$ matrices are defined as gaussian noises of $\mathbf{X}$ and $\mathbf{Y}$, respectively.

The parameters of the simulations are represented in tables (Table 1 and Table 2) where differences between simulated inliers and outliers were highlighted in bold in the tables. Scripts of the simulations are available at this link : https://github.com/maxmetz/data_simulation

4.1.1. Simulation 1, Y-outliers

The Y-outliers were defined by their relationship f between $\mathbf{X}$ and $\mathbf{Y}$. All other simulation parameters were common between inliers and outliers. The construction of the simulated data set 1 is represented in table 1.

TABLE 1 – The different choices in the simulation 1

<table>
<tr><th></th><th>Inliers</th><th>Outliers</th></tr>
<tr><td>$\mathbf{p}_u$</td><td colspan="2">Pure spectrum of glucose</td></tr>
<tr><td>$\mathbf{t}_u$</td><td colspan="2">Folded-normal distribution</td></tr>
<tr><td>$\mathbf{P}_d$</td><td colspan="2">Pure spectrum of water
Pure spectrum of ethanol
Spectrum of water-ethanol Interaction
10 Artificial spectra</td></tr>
<tr><td>$\mathbf{T}_d$</td><td colspan="2">Folded-normal distribution
Folded-normal distribution
Product between T_{water} and $T_{ethanol}$
Folded-normal distribution</td></tr>
<tr><td>$\mathbf{E}$</td><td colspan="2">Gaussian distribution</td></tr>
<tr><td>f</td><td>$Y_1 = 10 * T_{ethanol}$
$Y_2 = 10 * T_{glucose}$
$Y_3 = 10 * T_{water}$</td><td>$Y_1 = 10 * T_{ethanol}$
$\mathbf{Y_2 = -10 * T_{glucose}}$
$Y_3 = 10 * T_{water}$</td></tr>
<tr><td>$\mathbf{F}$</td><td colspan="2">Gaussian distribution</td></tr>
</table>

4.1.2. Simulation 2, X-outliers

The X-outliers were defined by others artificial spectral signatures. These signatures correspond to minority compounds. All other simulation parameters were common between inliers and outliers. The simulation is represented in table 2.

TABLE 2 – The different choices in the simulation 2

	Inliers	Outliers
$\mathbf{p}_u$	Pure spectrum of glucose	
$\mathbf{t}_u$	Folded-normal distribution	
$\mathbf{P}_d$	Pure spectrum of water Pure spectrum of ethanol Spectrum of water-ethanol Interaction 10 Artificial spectra	Pure spectrum of water Pure spectrum of ethanol Spectrum of water-ethanol Interaction 10 Artificial spectra **10 Artificial spectra**
$\mathbf{T}_d$	Folded-normal distribution Folded-normal distribution Product between T_{water} and $T_{ethanol}$ Folded-normal distribution	Folded-normal distribution Folded-normal distribution Product between T_{water} and $T_{ethanol}$ Folded-normal distribution **Folded-normal distribution**
$\mathbf{E}$	Gaussian distribution	
f	$Y_1 = 10 * T_{ethanol}$ $Y_2 = 10 * T_{glucose}$ $Y_3 = 10 * T_{water}$	$Y_1 = 10 * T_{ethanol}$ $Y_2 = 10 * T_{glucose}$ $Y_3 = 10 * T_{water}$
$\mathbf{F}$	Gaussian distribution	

4.2. Real data set

The real data set was formed by 261 spectra of raw cow milk collected from farms in Wallonia in 2014 and 2015. Spectra were recorded over a spectral range 397-4000 cm-1 with a resolution of 4 cm-1 by using a FTIR spectrometer (Delta LactoScope, PerkinElmer). For each sample, chemical measurements were performed to obtain two-responses variable : fat content and protein content. Fat and Protein content were determined in accordance with reference methods "ISO 1211 :2010 [IDF 1 :2010]" and "ISO 8968-1 :2014 [IDF 20-1 :2014]", respectively. This database is particularly interesting because it contains missing data whose values have been replaced by 0.

4.2.1. Evaluation strategies

RoBoost-PLS2-R was evaluated and compared with two standard regression algorithms : PLS2-R and RSIMPLS.

In the case of the simulations, the 1000 samples were divided into two groups : 800 for calibration and 200 for validation. The reference method in terms of prediction performance was PLS2-R calibrated without outliers. For the real data set, calibration set was composed of 209 samples. The validation was conducted on 52 samples. These samples were selected from a study of the data in order to represent the samples as well as possible without containing potential outliers. The reference method in terms of prediction performance was RSIMPLS.

The method performance was evaluated according to the validation sets and Root Mean Square Error of Prediction (RMSEP) as a figure of merit. Only the results achieved using the optimal parameters (*i.e.* the parameters that provide the minimum value of the RMSEP) of RoBoost-PLS2-R and RSIMPLS were presented.

The evaluation strategy also aimed at assessing the weights attributed to each sample. Indeed, the RoBoost-PLS2-R method allows the visualisation of the weight given to each sample for each LV. In this work, the parameters of the methods RoBoost-PLS2-R and RSIMPLS such as the constants used in the weight functions were adjusted to obtain the minimum RMSEP.

4.3. Software

PLS2-R was performed with "rnirs" and RoBoost-PLS2-R is available RoBoost-PLSR functions available in R. RSIMPLS was performed using the function of the LIBRA package available in MALTLAB.

5. Results and discussions

5.1. Simulation set 1

5.1.1. Data visualisation

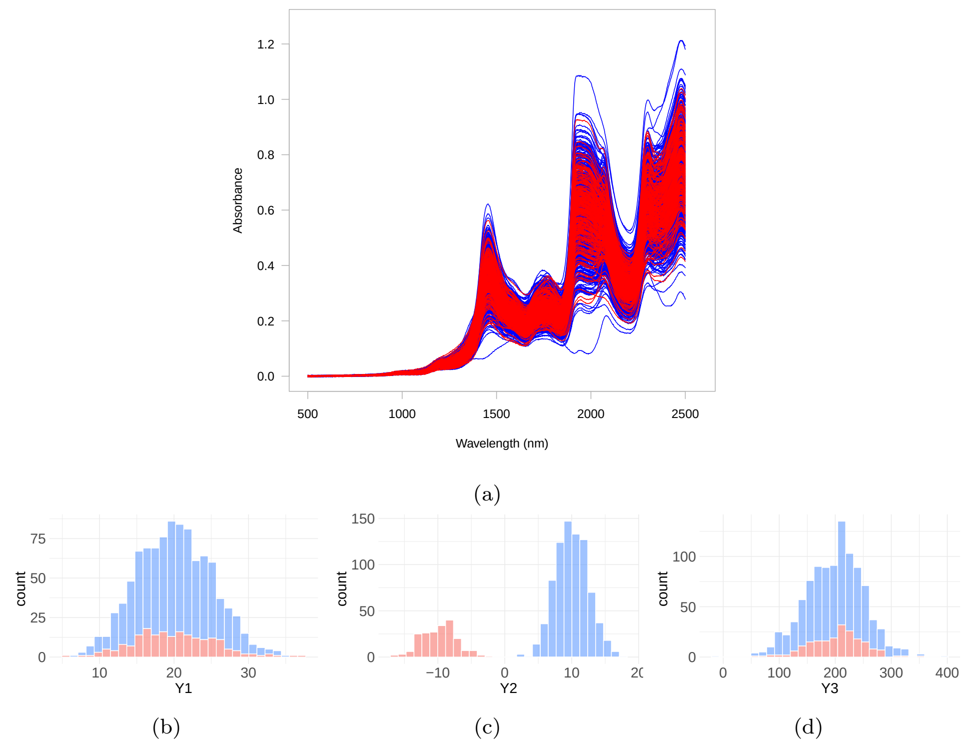


FIGURE 1 – Graphical representation of simulation 1 : (a) spectral data (b) value distribution of Y1 response variable (c) value distribution of Y2 response variable (d) value distribution of Y3 response variable. Outliers are shown in red and inliers in blue.

Figure 1 shows the graphical representation of simulation 1. From the spectra plot (Figure 1a), it can be seen that is difficult to identify outliers (in red) from a simple visual inspection. In this case, the outliers were defined by a distinct relation f on one of the response variables (see Table 1). Therefore, no spectral difference between the two groups is expected.
From the plot of value distributions of the response variables (see Figure 1b,c,d) it can be observed that Y1 and Y3 variables present the same distribution for both outliers and inliers. However, different distribution for

these two groups is presented in Y2 variable. Moreover, the variances of Y1 are smaller than the variance of Y3.

5.1.2. *Method evaluation*

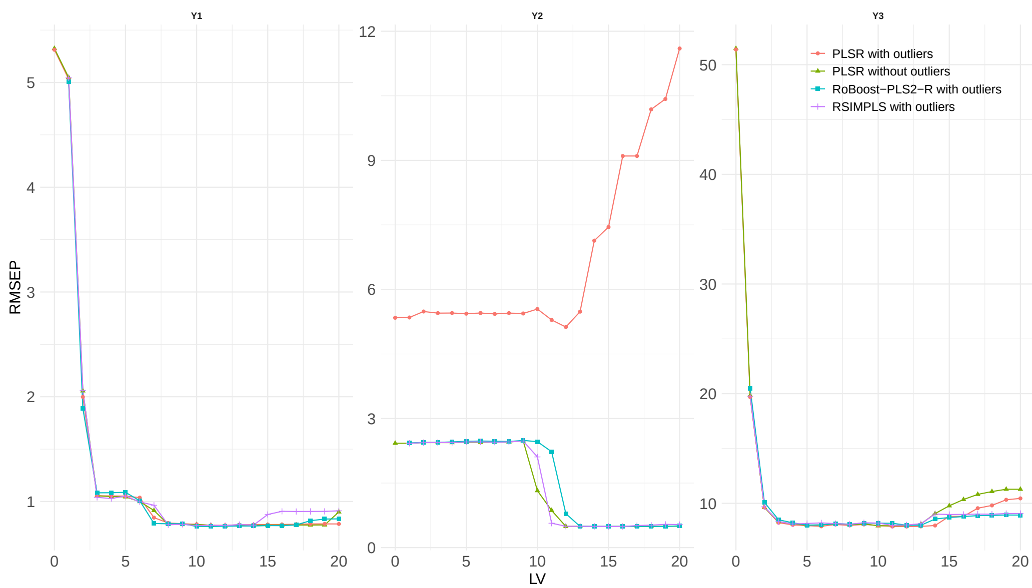


FIGURE 2 – Evolution of the RMSEP as a function of the number of latent variables for the PLS2-R with and without outliers, RSIMPLS and RoBoost-PLS2-R for the simulation 1 set

Figure 2 shows the prediction performances for each method and response variable Y on the basis of simulation 1. For the variables Y1 and Y3, the error curves obtained by PLS2-R with and without outliers, RSIMPLS and RoBoost-PLS2-R are similar. This is due to the fact that outliers are only atypical on Y2 and hence, no impact on the Y1 and Y3 predictions is expected. For the variables Y2 the error curves obtained by PLS2-R with and without outliers are different. The PLS2-R model calibrated with outliers perform poorly in inliers prediction. The prediction performance of RSIMPLS is close to the PLS2-R without outliers. This means that the RSIMPLS method can deal with these outliers and provides satisfactory results. These results show that RoBoost-PLS2-R performs as well as RSIMPLS on this dataset. Therefore, RoBoost-PLS2-R can handle the presence of outliers in the response variables regardless of the variance of the responses.

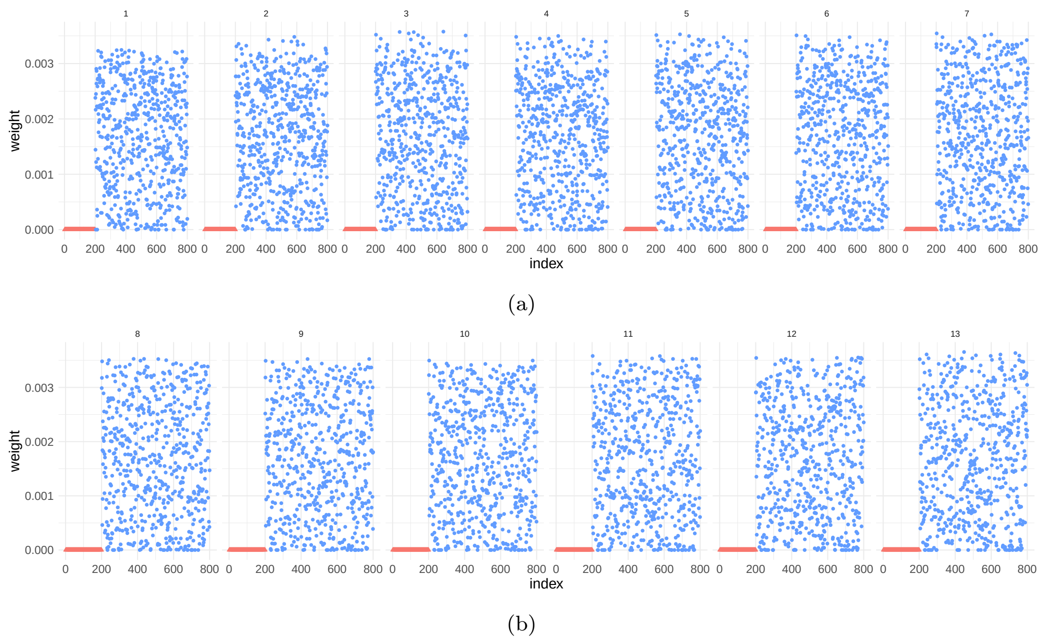


FIGURE 3 – Weights assigned to samples by the RoBoost-PLS2-R method for the simulation set 1 according to the number of LV from 1 to 13. Outliers and inliers are in red and blue, respectively.

Figure 3 shows the weights assigned to the samples of simulation 1 by the RoBoost-PLS2-R method as a function of the number of LV with the best performing hyperparameters. It can be noted that outliers have a very low weight while some inliers have a weight close to zero. This may be due to three reasons. Firstly, the hyperparameters of bisquare function must be strict enough to assign a weight close to 0 to the outliers for each LV. Taking into account that some inliers could be very similar to some outliers, assignation of low weights to these inliers could be expected. Secondly, the weights associated to Y-residuals are a combination of weights defined for each Y variables. The hyperparameter $beta$(see Section 3) is assumed to be constant for each variable in Y. This means that the higher the number of variables, the more dispersed the weights assigned to the inliers could be. To achieve a more homogeneous weighting on the outliers, the multivariate aspect of Y should be taken into account. For example, a potential solution can be to calculate the robust Mahalanobis distance at the centre of the data on the residuals of Y for each Latent Variable. Thirdly, some outliers are not detrimental to the model but are also irrelevant and can therefore have a low weight without impacting on the prediction performance of the model. In conclusion, RoBoost-PLS2-R has assigned a low weight to a large number of samples without impacting on the prediction performance of the model. However, it is potentially possible to improve this approach by modifying the weighting criteria associated with the Y residuals.

5.2. Simulation 2

5.2.1. Data visualisation

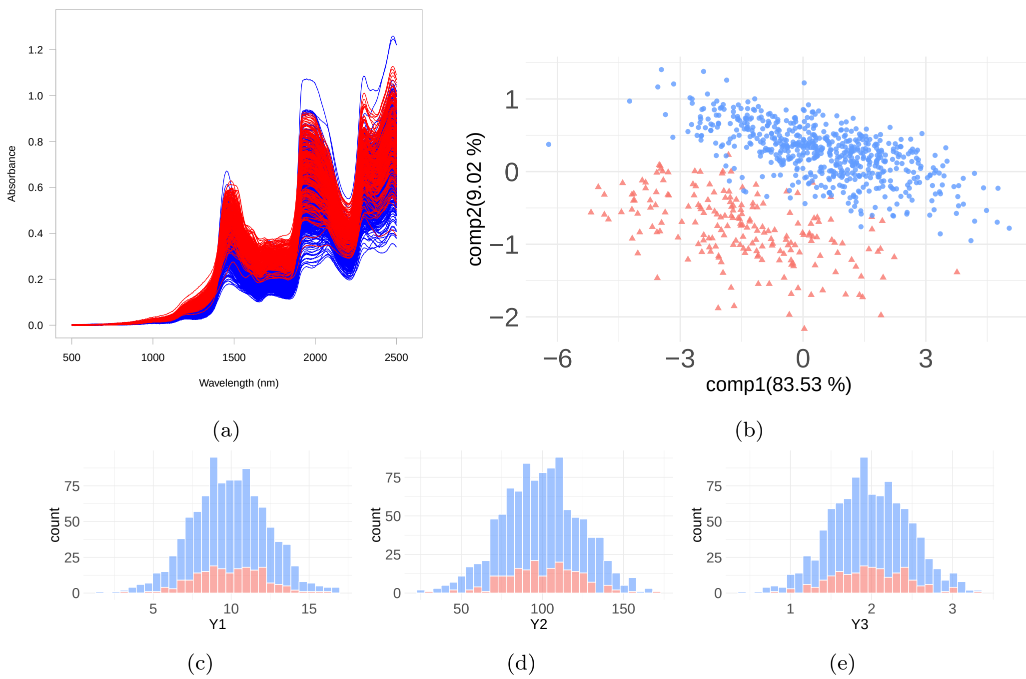


FIGURE 4 – Graphical representation of simulation 2 : (a) spectral data (b) PCA score plot of the two first components (c) value distribution of Y1 response variable (d) value distribution of Y2 response variable (e) value distribution of Y3 response variable. Outliers are shown in red and inliers in blue.

Figure 4 shows the graphical representation of simulation 2. From spectra plot of the sample (Figure 4 a), it can be seen that outliers are not identifiable. Indeed, in this simulation, outliers are different only for spectral signatures and hence, they contribute slightly to the construction of the spectra. Figures 4b represents the score plot on the two first principal components. Two centroids can be seen but there is no clear separation between outliers and inliers. This is due to the outliers having their major compounds in common (see Table 2). From the value distributions plot of the responses (see : Figures 4c,d,e), it can be seen that outliers and inliers present similar distribution in all Y response variables. Outliers are different only on the basis of the spectral signatures that compose them.

5.2.2. *Method evaluation*

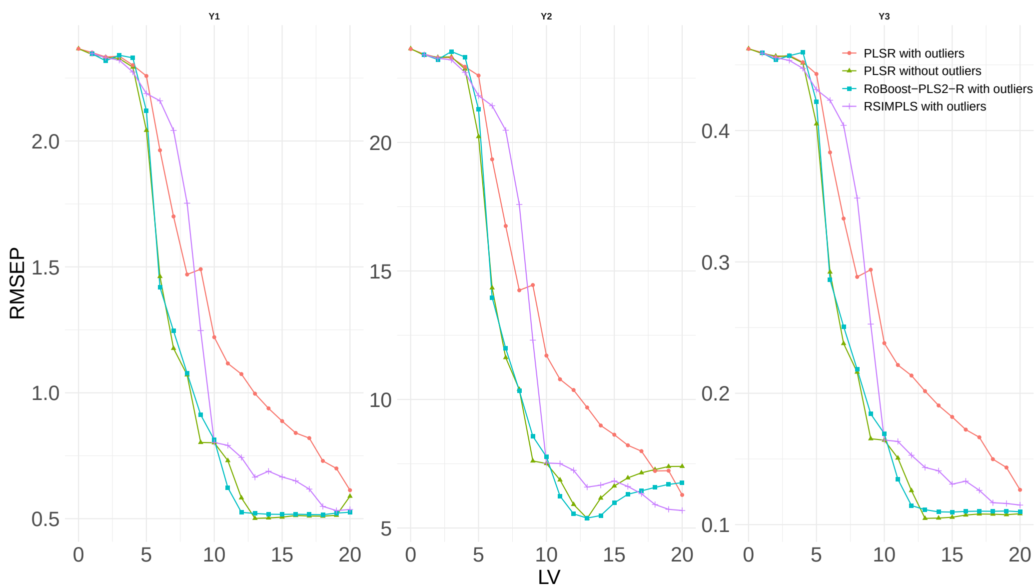
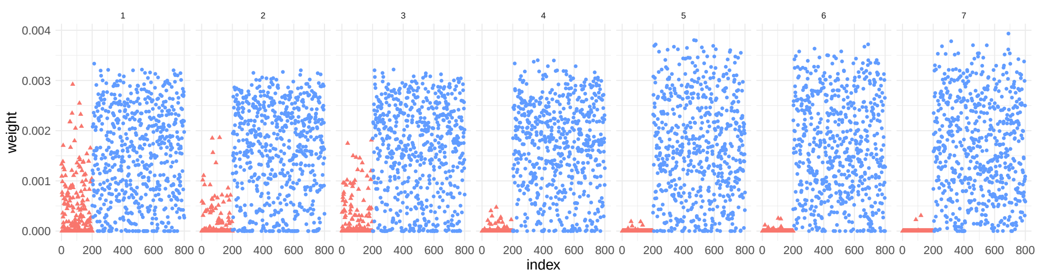


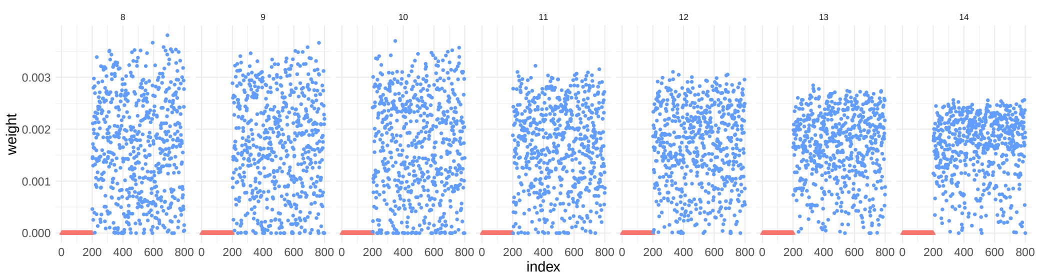
FIGURE 5 – Evolution of the RMSEP as a function of the number of latent variables for the PLS2-R with and without outliers, RSIMPLS and RoBoost-PLS2-R for the simulation 2 set

The figure 5 represents the prediction performances of the applied methods on validation set for each response variable on the basis of the simulation. As expected, the outliers impact negatively the predictive capacity of the PLS2-R for all responses. For the RSIMPLS method, all performance curves are between those of the PLS2-R method with and without outliers. However, with a large number of latent variables, the prediction performances of RSIMPLS approach the best performance of PLS2-R without outliers. This may be due to the fact that RSIMPLS does not directly take into account the residuals of X but also that the estimation of the leverage effect is not directly taken into account. Indeed, in RSIMPLS it is the cross-covariance matrices $\mathbf{C_{xy}}$ and the empirical covariance matrix $\mathbf{C_x}$ that are robustly estimated.

For the RoBoost-PLS2-R method, it can also be seen that for the three responses, performance curves are close to those of PLS2-R without outliers. However the optimal number of components is higher for RoBoost-PLS2-R than the PLS2-R without outliers. To conclude, these results highlight the fact that RoBoost-PLS2-R can reach the best performance of PLS2-R without outliers. Thus, RoBoost-PLS2-R can handle these X-outliers for the prediction of multiple responses.



(a)



(b)

FIGURE 6 – Weights assigned to samples in simulation set 2 according to the chosen number of latent variables from 1 to 14. Outliers and inliers are in red and blue, respectively

Figure 6 shows the weight assigned to samples by RoBoost-PLS2-R according to the number of LV. It can be observed that the weights of outliers decrease progressively when the number of LV increases. This gradual decrease is partly explained by the fact that both outliers and inliers were simulated using common majority spectral signatures. Indeed, only some minor spectral signatures differentiate the inliers from the outliers (see Section 4). After 8 latent variables, all outliers have a weight equal to 0, whereas almost all inliers present a high weight. Nevertheless, it is possible to note that the majority of the inliers have a strong weight and therefore a large number of them are used to calculate the model.

5.3. Real data set

5.3.1. Data visualisation

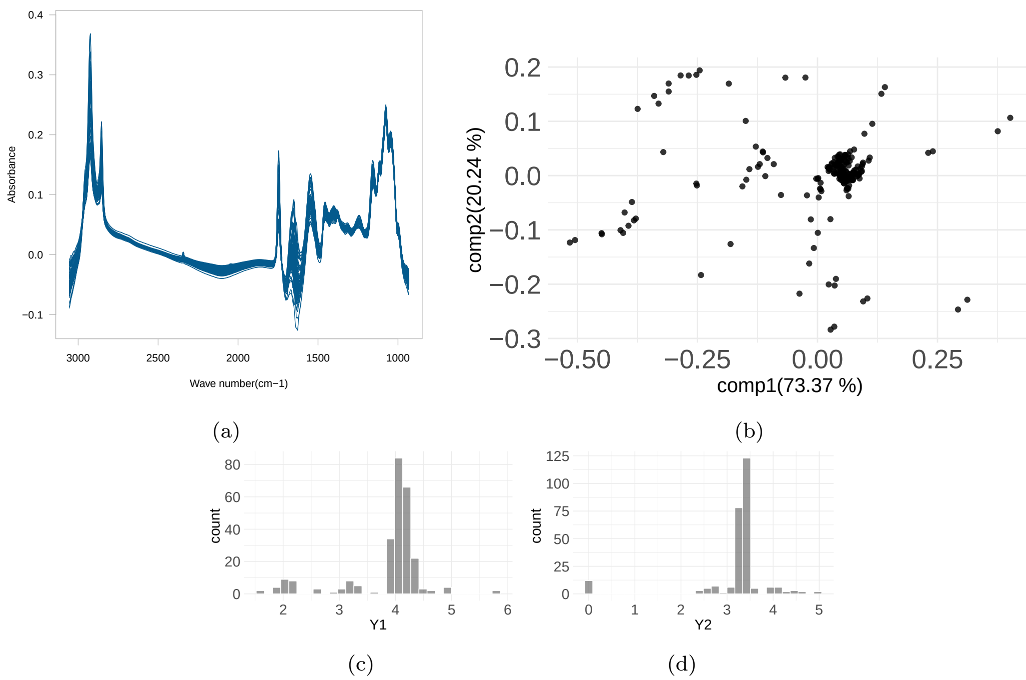


FIGURE 7 – Graphical representation of real data set : (a) spectral data (b) PCA score plot of the two first components (c) value distribution of Y1 (c) value distribution of Y2

Figure 7 shows the graphical representation of real data set. From the spectra plot (Figure. 7a), it can be seen that there is no visible atypical spectrum. This means that is not possible to identify or detect outliers in this data set based on spectra visualisation. Figure 7b shows the PCA score plot of the two first components. It can be observed that some samples scores are really different from those of other samples. It is possible that some atypical samples are outliers but some sample can be also relevant to calculate a model. From the value distributions plot of the responses (see Figures 7c,d), it can be seen that some samples show extreme response values in Y1 and

Y2. In conclusion, this real data set potentially contains samples that are detrimental to the model.

5.3.2. Method evaluation

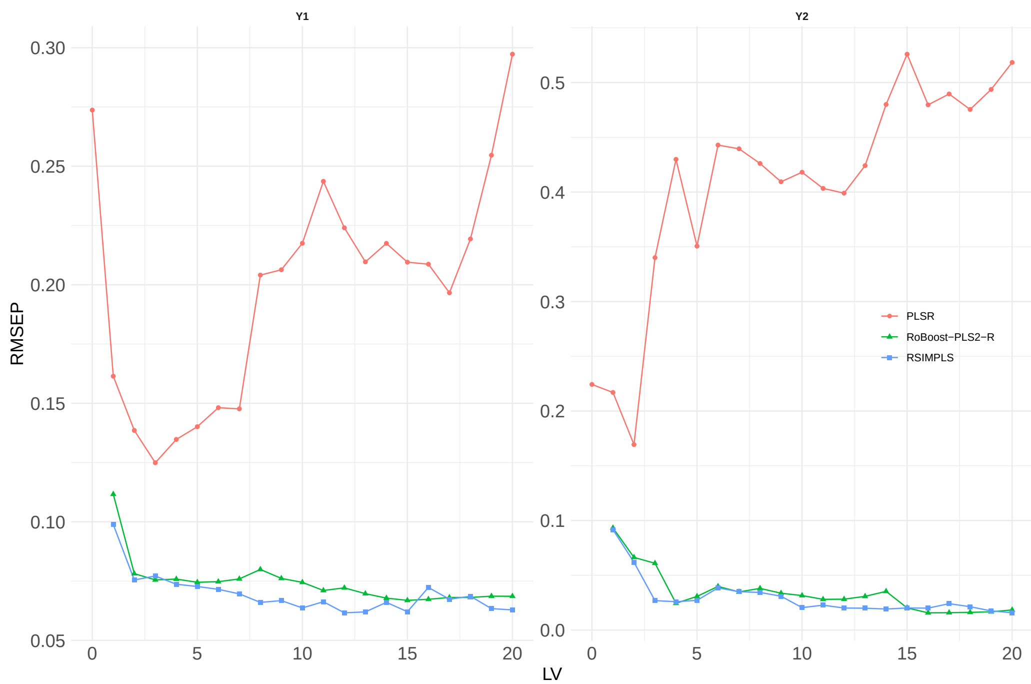
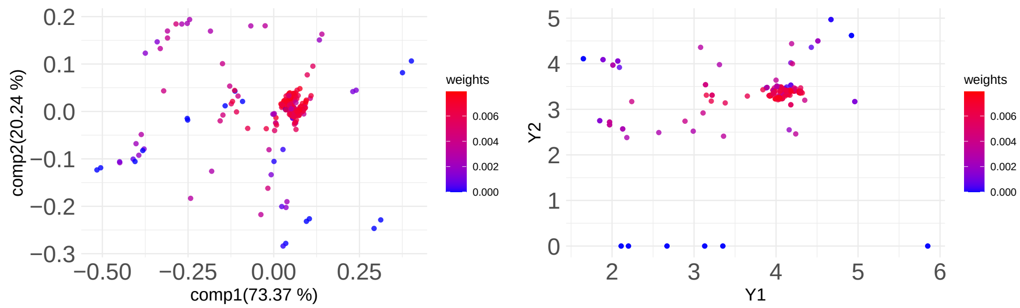


FIGURE 8 – Evolution of the RMSEP as a function of the number of latent variables for the PLS2-R, RSIMPLS and RoBoost-PLS2-R for the real data set

Figure 8 represents the prediction performances of the methods on validation set for each reference Y. As there are not all known outliers in the calibration set, it was not possible to define a PLS2-R with and without outliers. Therefore, only the PLS2-R has been calculated on the data with potential outliers. In the figure 8 it can be seen that for both responses the PLSR performance curve is higher than those of the two robust methods.This means that RSIMPLS and RoBoost-PLS2-R method have higher prediction performances than the PLS2-R method applied on this data set. Therefore, some samples are detrimental in the calibration set to the calculation of a PLS2-R model that predicts the samples in the validation set. The two methods RoBoost-PLS2-R and RSIMPLS have close results in terms of RMSEP for a number of latent variables close to 15. This means that both methods were able to deal with potential outliers samples and therefore enable more accurate predictions.



(a) (b)

FIGURE 9 – Graphical Representation of the mean weights (for 15 LV) assigned by RoBoost-PLS2-R through PCA score plot of the first two components(a) and Y2 as a function of Y1(b). A colour gradient from blue to red represents the weights assigned to the samples (smallest to largest).

Figure 9 shows the weights assigned to the samples by RoBoost-PLS2-R through PCA score plot of the first two components and the Y2 as a function of Y1 plot. It can be seen in figure 9a that not all samples far from the centre were considered as potential outliers (*i.e.* with low weights). Some extreme samples seem to be relevant for the model and were therefore given high weights. The figure 9b shows that some samples have extreme Y-values (0). These samples have a 0 average weight in RoBoost-PLS2-R. This is due to missing value. In this data set, missing data has a value of 0 assigned. It can be concluded through these observations that the RoBoost-PLS2-R method can eliminate outliers on Y but also on X while limiting the assignment of low weights to extreme samples.

6. Conclusion

In this paper, RoBoost-PLS2-R method is proposed to predict multi-response. This method was evaluated and compared to reference methods on two simulated data sets and one real data set containing different outlier scenarios. For all data sets, prediction performances of RoBoost-PLS2-R are close to those of PLS2-R models calibrated without outliers and to RSIMPLS method. Simulations have shown that RoBoost-PLS2-R extension was very effective when outliers are defined by their spectral properties. In the case of real data, results obtained for both robust methods are better than the PLS2-R method. To conclude, RoBoost-PLS2-R seems to be a reliable and robust regression tool for predicting multi-response variables when data potentially contain outliers. However, some method developments are possible. First of all, the estimation of the criterion evaluated on the Y-residuals can be estimated in another way to take into account the multivariate aspect of Y. In addition, the optimisation of the hyperparameters allowing the weighting of the individuals is complex, it would be relevant to look at automatic parameterisation approaches. Moreover, it could be interesting to use the formalism of the RoBoost-PLS2-R method for cases of categorical variables and thus propose a robust discriminant method. Finally, new RoBoost-PLS2-R algorithm now enables the estimation of regression coefficients contrary to the previous algorithm proposed for RoBoost-PLS1-R. It would be interesting to study these regression coefficients to assess the method's behaviour outside the prediction capacities. In future work, it would be relevant to use the RoBoost formalism for concrete applications involving multi-response variables.

It would also be interesting to modify the strategy for visualising the weights of individuals in the calibration. Indeed, here the weights are displayed for each latent variable, so it could be interesting to find a strategy to obtain a weight for each individual allowing to summarise all the weights of each latent variable.

Références

[1] S. Wold, M. Sjostrom, L. Eriksson, PLS regression : a basic tool of chemometrics, Chemometrics and Intelligent Laboratory Systems 58 (2) (2001) 109–130. doi:10.1016/S0169-7439(01)00155-1.

[2] M. Griep, I. Wakeling, P. Vankeerberghen, D. Massart, Comparison of semirobust and robust partial least squares procedures, Chemometrics and Intelligent Laboratory Systems 29 (1) (1995) 37–50. doi:10.1016/0169-7439(95)80078-N.

[3] I. Stanimirova, S. Serneels, P. J. Van Espen, B. Walczak, How to construct a multiple regression model for data with missing elements and outlying objects, Analytica Chimica Acta 581 (2) (2007) 324–332. doi:10.1016/j.aca.2006.08.014.

[4] R. J. Pell, Multiple outlier detection for multivariate calibration using robust statistical techniques, Chemometrics and Intelligent Laboratory Systems 52 (1) (2000) 87–104. doi:10.1016/S0169-7439(00)00082-4.

[5] J. A. Gil, R. Romera, On robust partial least squares (PLS) methods, Journal of Chemometrics 12 (6) (1998) 365–378. doi:10.1002/(SICI)1099-128X(199811/12)12:6<365::AID-CEM519>3.0.CO;2-G.

[6] J. González, D. Peña, R. Romera, A robust partial least squares regression method with applications, Journal of Chemometrics 23 (2) (2009) 78–90. doi:10.1002/cem.1195.

[7] I. N. Wakelinc, H. J. H. Macfie, A robust PLS procedure, Journal of Chemometrics 6 (4) (1992) 189–198. doi:10.1002/cem.1180060404.

[8] J. Peng, S. Peng, Y. Hu, Partial least squares and random sample consensus in outlier detection, Analytica Chimica Acta 719 (2012) 24–29. doi:10.1016/j.aca.2011.12.058.

[9] P. Filzmoser, R. Maronna, M. Werner, Outlier identification in high dimensions, Computational Statistics & Data Analysis 52 (3) (2008) 1694–1711. doi:10.1016/j.csda.2007.05.018.

[10] M. Hubert, K. V. Branden, Robust methods for partial least squares regression, Journal of Chemometrics 17 (10) (2003) 537–549. doi:10.1002/cem.822.

[11] U. Kruger, Y. Zhou, X. Wang, D. Rooney, J. Thompson, Robust partial least squares regression : Part II, new algorithm and benchmark studies, Journal of Chemometrics 22 (1) (2008) 14–22, _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1002/cem.1095. doi:10.1002/cem.1095.

[12] I. Hoffmann, S. Serneels, P. Filzmoser, C. Croux, Sparse partial robust M regression, Chemometrics and Intelligent Laboratory Systems 149 (2015) 50–59. doi:10.1016/j.chemolab.2015.09.019.

[13] P. Filzmoser, S. Serneels, R. Maronna, C. Croux, Robust multivariate methods in Chemometrics, arXiv :2006.01617 [stat] (2020) 393–430ArXiv : 2006.01617. doi:10.1016/B978-0-12-409547-2.14642-6.

[14] M. Hubert, K. V. Branden, Robust methods for partial least squares regression, Journal of Chemometrics 17 (10) (2003) 537–549. doi:10.1002/cem.822.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cem.822

[15] M. Metz, F. Abdelghafour, J.-M. Roger, M. Lesnoff, A novel robust PLS regression method inspired from boosting principles: RoBoost-PLSR, Analytica Chimica Acta (2021) 338823doi:10.1016/j.aca.2021.338823.
URL https://linkinghub.elsevier.com/retrieve/pii/S0003267021006498

[16] M. Metz, A. Biancolillo, M. Lesnoff, J.-M. Roger, A note on spectral data simulation, Chemometrics and Intelligent Laboratory Systems 200 (2020) 103979. doi:10.1016/j.chemolab.2020.103979.