Clustering of Variables for Enhanced Interpretability of Predictive Models

Evelyne Vigneau
StatSC, Oniris, Inrae, 44 322 Nantes Cedex 03, France
E-mail: evelyne.vigneau@oniris-nantes.fr

Keywords: clustering of variables, linear model, boosting, high-dimensional data, dimensionality reduction, authentication

Received: August 18, 2020

A new strategy is proposed for building easy to interpret predictive models in the context of a high-dimensional dataset, with a large number of highly correlated explanatory variables. The strategy is based on a first step of variables clustering using the CLustering of Variables around Latent Variables (CLV) method. The exploration of the hierarchical clustering dendrogram is undertaken in order to sequentially select the explanatory variables in a group-wise fashion. For model fitting implementation, the dendrogram is used as the base-learner in an L2-boosting procedure. The proposed approach, named lmCLV, is illustrated on the basis of a toy-simulated example when the clusters and predictive equation are already known, and on a real case study dealing with the authentication of orange juices based on ^{1}H-NMR spectroscopic analysis. In both illustrative examples, this procedure was shown to have similar predictive efficiency to other methods, with additional interpretability capacity. It is available in the R package `ClustVarLV`.

Povzetek: Prispevek opisuje nov pristop za izboljšano razlago prediktivnih modelov.

1 Introduction

In the context of high-dimensional data with a large number of variables, *p*, and small number of observations, *n*, such as microarray data, metabolomic and volatolomic data (among a large variety of -omic data collected using high-throughput fingerprinting technologies), most recent statistical modelling strategies aim to achieve both efficient prediction and enhanced interpretability outcomes. For 20 years, the question of variable selection has driven a great deal of work, from discrete processes in which variables are either retained or removed, to lasso regularization processes in which several model coefficients may be shrunken towards zero [1, 2]. Regularization strategies discussions are still an active topic due to the high-dimensional of current data, either with model-based approach as in [3] or in combination with filtering step as in [4].

However, as Lasso tends to arbitrarily select one predictive variable among a group of highly correlated relevant variables, the sparsity principle of Lasso have been embedded into strategies for the selection of grouped predictors. The Elastic Net penalty [5] has been proven to be an efficient compromise between the Lasso (L1) penalty and the Ridge (L2) penalty. Fused Lasso [6] and Group Lasso [7] are designed for explanatory variables naturally ordered (as in vibrational spectroscopy) or arranged into groups (as for design factors). Sparse Group Lasso [8, 9] yields sparsity at both the group and individual feature levels, in order to select groups and predictors within a group. The Octagonal Shrinkage and Clustering Algorithm for Regression (OSCAR) [10] is another approach which combines an L_1-type penalty and a pairwise L_∞-type penalty on the model's coefficients. Thus OSCAR allows to simultaneously select variables and perform supervised clustering in the context of linear regression. The CLERE methodology [11] is also based on the clustering of the regression coefficients using a Gaussian latent mixture model. Several of the penalized least squares approaches have Bayesian analogues also developed for variable selection and grouping [12, 13].

The model we propose here stems from another family of approaches. Emphasis is put on reducing the dimensionality of the data when a large number of highly correlated explanatory variables ($p \gg n$) has been collected. We consider reduction dimension approaches that define latent components as being linear combinations of the explanatory variables and taking their correlation-structure into account. It is also desirable that these latent components are easy to interpret by making several component loadings to be exactly zeros for instance. This goal has also been addressed with approaches designed to constrain Principal Component [14, 15, 16] or PLS regression [17] loadings to be shrunk to zeros. Another strategy was to first perform a cluster analysis in order to construct interpretable components [18, 19].

Our proposal is also to first perform a clustering of the explanatory variables and then to fit a linear model between a response variable Y and p explanatory variables X_j ($j = 1, \ldots, p$) in a groupwise fashion. The algorithm adopted herein is easy to implement and is oriented towards the interpretability of the latent component introduced sequentially into the model. In the case study considered herein, each selected latent component can be assumed to be associated with a compound from its chemical spectrum in NMR (Nuclear Magnetic Resonance). With metabolomic data, it is possible to imagine that the latent components associated to subsets of metabolites may be related to specific biological pathways. Within the context of DNA microarray data, similar studies have been undertaken [20, 21] in which hierarchical cluster analysis allows to identify supergenes, obtained by averaging genes within the clusters. These supergenes are thereafter used to fit regression models, thereby attaining concise interpretation and accuracy. One of the main differences in this study, compared with previous research works [20, 21], is that representatives of the clusters of variables, or latent variables, are not necessarily an average of the observed variables, but can be components that best reflect the variability within each cluster of variables. The second difference is in the procedure adopted for the progressive construction of the regression linear model.

The clustering of the explanatory variables is based herein on the CLV (CLustering of Variables around Latent Variables) approach [22, 23], implemented in the `ClustVarLV` R package [24]. In summary, the CLV approach consists of clustering together highly correlated variables into clusters (or groups of variables), while exhibiting within each cluster a latent variable (or latent component) representative to this cluster. It turns out that each latent component is defined as a linear combination of only the variables belonging to the corresponding cluster. From this point of view, CLV components are sparse components in the space of observations, and aim to best reflect the variance-covariance structure between the explanatory variables. Using the same approach, but defining a slightly different criterion, [18] also proposed a clustering of variables method for producing interpretable principal components. [25] investigated an approach for clustering of variables, based on the identification of a mixture with bipolar Watson components defined on the hypersphere. Herein, the central feature of the clustering procedure is a bottom-up aggregation approach of the explanatory variables involving the CLV criterion.

The dendrogram obtained is then explored in order to identify and select the predictive group's latent components regarding the response variable Y. In contrast with a previous study [26], hierarchical clustering, which is the more time consuming step, is performed only once and the fitting model stage has been modified accordingly. More precisely, an L2-boosting procedure for which the base-learner model is the CLV hierarchical algorithm has been considered. It consists to, iteratively, select a group of explanatory variables. The residuals of the response variable is then regressed on the latent component associated with the selected group and the predicted response is updated with the shrunken version of this local predictor.

The proposed methodology combining variables clustering and iterative linear model fitting, designated as *lmCLV*, is described in Section 2. Section 3 includes a simple simulated dataset in order to illustrate the behavior of the procedure in a known context, as well as one real case study dealing with the authentication of orange juice based on ^{1}H-NMR spectroscopy.

2 Methodology

2.1 Notation

We consider the high-dimensional linear model :

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon, \qquad (1)$$

where $\mathbf{y} = [y_i]$ is a $n \times 1$-dimensional random response vector; $\mathbf{X} = [\mathbf{x}_1|\ldots|\mathbf{x}_p] = [x_{ij}]$, a $n \times p$-dimensional quantitative explanatory variables matrix (with $i = 1,\ldots,n$ and $j = 1,\ldots,p$); β, a $p \times 1$ regression coefficients vector and ε the $n\times 1$-dimensional random vector of the residuals, with zero mean and constant variance.

We consider contexts where $p > n$ or $p \gg n$ and where the explanatory variables may be arranged into groups of highly correlated variables.

Both the explanatory variables matrix and the response vector are assumed to be column-centered. In addition, the user may choose to standardize, or not, the variables to a unit variance.

2.2 *lmCLV*'s outlines

lmCLV combines two main methods:

1. The CLV method which is performed first.The similarities between the variables, herein evaluated on the basis of their covariance or correlation coefficients, are depicted by a tree diagram which is used as the learner for the model fitting method. The clustering of variables, using CLV method, is detailed in Sect.2.3.

2. The L2-boosting procedure which provides an efficient iterative model fitting method. The outline of the L2-boosting algorithm is depicted in Sect.2.4, and the way the CLV base-learner is used in the course of this procedure is explained in Sect.2.5.

2.3 Clustering of the variables

The clustering of the explanatory variables using the CLV method for directional groups [22, 24] aims to both identify clusters of variables and define a latent component associated with each cluster.

For a given number of clusters, K, the aim of the CLV method is to seek a partition $\mathcal{G}_K = \{G_1,\ldots G_K\}$ of the variables into K disjoint clusters and a matrix $\mathbf{C}_K = [\mathbf{c}_1|\ldots|\mathbf{c}_K]$ of K latent variables, each being associated with one cluster, so as to maximize the internal coherence within the clusters. When the agreement between the variables is assessed using their covariance or correlation coefficients, regardless of the sign of these coefficients, the clusters we are looking for are named directional groups. In this case, the aim is to define the partition, $\mathcal{G}_K$, and group's latent variables matrix, $\mathbf{C}_K$, so that to get the optimal value T^* of the CLV criterion, so that:

$$T^* := \max_{(\mathcal{G}_{\mathcal{K}}, \mathbf{C_K})} \sum_{k=1}^{K} \sum_{j=1}^{p} \delta_{kj}\ cov(\mathbf{x}_j, \mathbf{c}_k)^2, \qquad (2)$$

under the constraints $||\mathbf{c}_k|| = 1$.

In eq.(2), $\delta_{kj} = 1$ if the j^{th} variable belongs to the group G_k, and $\delta_{kj} = 0$ otherwise. In other terms, (δ_{kj}) is the generic term of a binary matrix $\mathbf{\Delta}$ of the group's membership of the p variables. $\mathbf{\Delta}$ has only one nonzero element per row.

It is easy to show [22] that when the partition $\mathcal{G}_{\mathcal{K}}$ is fixed, the latent variable in a cluster $G_k\,(k = 1, ..., K)$ is defined as the first standardized principal component of a data matrix formed by the variables belonging to this cluster. The latent variable $\mathbf{c}_k$ associated with the cluster $G_k\,(k = 1, ..., K)$ can therefore be expressed as a linear combination of the variables of this group:

$$\mathbf{c}_k = \sum_{j/\delta_{kj}=1} v_j \mathbf{x}_j. \qquad (3)$$

For various numbers of clusters, from $K = p$ clusters, in which each variable is considered to form a cluster by itself, to $K = 1$, where there is a single cluster containing all the variables, an ascendant hierarchical clustering algorithm is also proposed with respect to the CLV criterion and aggregating rules detailed in [22]. The strategy proposed herein consists of firstly constructing the whole hierarchy of the p explanatory variables and to explore repeatedly the dendrogram obtained.

2.4 L2-boosting procedure for model fitting

The second feature of the strategy is a boosting procedure, which can also be represented as a functional gradient descent (FGD) algorithm [27, 28], performed on the residuals of an iteratively updated shrunken linear model for the prediction of the response variable $\mathbf{y}$. The outline of the algorithm can be depicted as follows:

1. Set $m = 0$, and initialize $\hat{\mathbf{y}}^{(0)}$, for instance by choosing $\hat{\mathbf{y}}^{(0)} = \mathbf{1_n}\bar{y}$, with $\mathbf{1_n}$ a vector of ones of length n, and $\bar{y} = 1/n \sum_{i=1}^{n} y_i$,

2. Increase m by 1, and compute the residuals $e_i^{(m)} = y_i - \hat{y_i}^{(m-1)}$ for $i = 1, \ldots, n$;

3. Apply the base-learner procedure to the actual residuals. The aim at this step is to identify the "best" CLV latent component denoted $\mathbf{c}^{*(m)}$. This base-learner procedure will be described in Sect.2.5;

4. Update the predictive function, i.e. $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \nu\,\alpha^{(m)}\,\mathbf{c}^{*(m)}$ where $0 < \nu \leq 1$ is a shrinkage parameter and $\alpha^{(m)}$ the Ordinary Least Squares (OLS) coefficient estimate of the linear regression of $\mathbf{c}^{*(m)}$ on $\mathbf{e}^{(m)} = [e_1^{(m)}, \ldots, e_n^{(m)}]$.

5. Return to step 2., until $m = M$ (M being a large predetermined integer).

This procedure depends on two parameters: the stopping iteration parameter, M, and the shrinkage parameter, ν, which can be determined via cross-validation or other information criterion as previously suggested in [28].

In practice, because our base-learner procedure returns one-dimensional components associated with sequential group-wise selections of the explanatory variables, we have often observed (as it will be shown in Sect.3) that a relatively high value of ν (greater than 0.5) generally performs better. Moreover, as the predictive ability of the model appeared to be relatively stable, large values of M (say 50 or more) are not necessarily useful.

2.5 Base-learner procedure

The third step of the algorithm presented in Sect.2.4 is the core of the proposed strategy. At each iteration, m, of the algorithm, the aim is to select a cluster of explanatory variables and their associated representative, i.e. the group's latent component $\mathbf{c}^{*(m)}$. This choice is guided by CLV hierarchical clustering of the p explanatory predictors of $\mathbf{X} = [\mathbf{x}_1| \ldots |\mathbf{x}_p]$ (Sect.2.3).

- For each size of partition (from p clusters to one cluster), we first aim to identify the CLV latent component which has the largest correlation (in absolute value) with the actual residuals $\mathbf{e}^{(m)}$. For $q = 1, \ldots p$, that is for each partition $\mathcal{G}_{p-q+1}$, we define :

$$\mathbf{c}_q^* := \max_{k \in \{1, \ldots, (p-q+1)\}} |cor(\mathbf{c}_k, \mathbf{e}^{(m)})|. \qquad (4)$$

- The next step consists of choosing a specific level q between 1 and p, that is a latent component $\mathbf{c}_q^*$ and its associated group of predictors G_q^*, in such a way that G_q^* is as large as possible while accommodating an undimentionality criterion. The unidimensionality condition is assessed herein using the modified Kaiser-Guttman (KG) rule [29].

  If we denote m this specific level:

$$m := arg \max_{q \in \{1, \ldots, p\}} (|G_q^*| \,/\, \lambda_1 > \mathcal{L} \text{ and } \lambda_2 \leq \mathcal{L}), \qquad (5)$$

  where $|G_q^*|$ denotes the cardinal of G_q^*, λ_1 and λ_2 are respectively the first and the second largest eigenvalues of the correlation matrix of the explanatory variables belonging to G_q^* and the threshold $\mathcal{L}$ is defined according to [29] by:

$$\mathcal{L} = 1 + 2\sqrt{\frac{|G_q^*| - 1}{p - 1}}.$$

The latent component $\mathbf{c}^{*(m)}$ and the groups of explanatory variables in G_m^* are returned to the main algorithm (Sect.2.4) which continues at step **4**.

Finally, it can be noticed that at each step the base learner returns a latent component which is itself a linear combination of a subset of the explanatory variables (eq.3). It is therefore easy to reformulate the prediction function in terms of a linear combinations of the p predictors, and to obtain an estimate of the coefficients' vector β (eq.1).

3 Applications

3.1 A toy-simulated example

The *lmCLV* procedure is illustrated in this section using a simple example based on a simulated model as in [26]. We considered herein $p = 70$ explanatory variables supposed to be measured on $n = 100$ observations. Moreover, these variables were assumed to be structured into five groups ($\mathcal{G}_1$ to $\mathcal{G}_5$) of various sizes: $\mathcal{G}_1$ was the largest group consisting of 35 variables, $\mathcal{G}_2$ was the smallest group with 5 variables, whereas $\mathcal{G}_3, \mathcal{G}_4$ and $\mathcal{G}_5$ were 10 variables each.

Each group of variables was generated around a prototypical variable. The five prototypes were centered and standardized random variables with a known structure of covariance. In practice, n realizations of a vector $(\mathcal{Z}_1, \ldots, \mathcal{Z}_5)^t$ were generated from a centered multivariate normal distribution with a covariance matrix $\mathbf{\Sigma}$:

$$\mathbf{\Sigma} = \begin{pmatrix} 1 & 0.5 & 0.5 & 0.5 & 0.1 \\ 0.5 & 1 & 0.5 & 0.1 & 0.1 \\ 0.5 & 0.5 & 1 & 0.1 & 0.1 \\ 0.5 & 0.1 & 0.1 & 1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 1 \end{pmatrix}. \tag{6}$$

Let us denote $\mathbf{Z}$ the n x 5 generated prototypical matrix. Then, the variables of each group were randomly simulated according to a multivariate normal distribution $\mathcal{N}(\mathbf{0_n}, \mathbf{I_n})$, where $\mathbf{0_n}$ represents the n-dimensional null vector and $\mathbf{I_n}$ the $n \times n$ identity matrix, as follows:

$$\mathbf{x}_j = \omega_j \, \mathbf{Z}\mathbf{\Lambda}^t + \varepsilon_j \text{ for } j \in \{1, \ldots, p\}, \tag{7}$$

where $\mathbf{\Lambda}$ is a p x 5 binary matrix defining the allocation of explanatory variables into the 5 groups. The columnwise marginal sum of $\mathbf{\Lambda}$ was $[35, 5, 10, 10, 10]$. In eq.(7), $\omega_j \in \{+1, -1\}$ was used to randomly create positive or negative correlations between each simulated variable $\mathbf{x}_j$ in a group and the associated prototypical variable, in order to generate directional groups of variables.

Finally the response variable $\mathbf{y}$ ($\in \mathbf{R}^n$) was generated with the following model:

$$\mathbf{y} = \mathbf{Z}\mathbf{b} + \varepsilon \text{ with } \mathbf{b} = [1, 5, 3, 0, 0]^t, \tag{8}$$

where ε resulting from a multivariate normal distribution $\mathcal{N}(\mathbf{0_n}, \mathbf{I_n})$.

The response $\mathbf{y}$ was supposed to be mainly related to the variables of the smallest group, $\mathcal{G}_2$, moderately to the variables in $\mathcal{G}_3$ and the smallest for the most numerous variables in $\mathcal{G}_1$. According to the parameters of the simulation, the expected correlation coefficients between $\mathbf{y}$ and five prototypical variables are 0.69, 0.96, 0.82, 0.18 and 0.12, respectively. This toy-simulated example was designed to represent a simplified, but realistic, case study.

The choice of the shrinkage parameter ν is discussed on the basis of the Root Mean Squared Error criterion evaluated using a five-fold cross-validation procedure, denoted $RMSE_{CV}$. This criterion was assessed at each iteration m from 0 (null model) up to $M = 20$. Figure 1 shows that as the parameter ν becomes smaller, the decrease in prediction errors becomes slower. In this example, a value of $\nu = 0.7$ or 0.8 led to a low level of errors ($RMSEP_{CV} = 2.29$) after only three iterations of the algorithm. In the following, ν was fixed to 0.7.

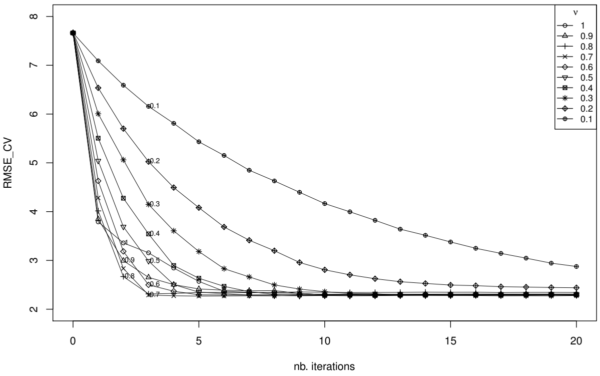

Figure 1: Evolution of the Root Mean Squared Error in Cross-validation with *lmCLV*, according to the number of iterations, and for various values of the shrinkage parameter, ν (from 1 to 0.1).

At the first iteration, and for each fold of the cross-validation procedure, the variable numbers 36 to 40 ($\mathcal{G}_2$) were in the cluster associated with the selected CLV latent component (for one of the five folds the variable 33 was also added, and for another fold the variable 46 was added). At the second iteration, the $\mathcal{G}_3$ variables (41 to 50) were always retained (except for one fold, for which the variable 48 was lacking). Finally, at the third iteration, as expected, all the variables associated with $\mathcal{G}_1$, number 1 to 35, were in the selected cluster of variables.

For comparison of *lmCLV* with two usual regression methods based on latent components, namely Principal Components Regression (PCR) and Partial Least Squares Regression (PLSR), the same dataset was considered, using the same five consecutive segments for Cross-Validation. The evolution of the $RMSE_{CV}$ according to the number of latent components included are shown in Figure 2(b). The left hand side panel (Figure 2(a)) shows the evolution of the errors in prediction, more precisely the $RMSE$, evaluated on the calibration set formed by the whole dataset. As can be observed in Figure 2(a), PLSR and, to a weaker extend, PCR are prone to over-fitting, with the $RMSE$ value reaching 0.94 when $p = 70$ components are included (i.e. the OLS solution). On the contrary the $RMSE$ with *lmCLV* remained relatively stable with increasing numbers of iterations. As a matter of fact, the higher the number of iterations, the more often the same groups of variables, and thus the same latent components, are likely to be included in the model. However, the loadings of these repeated latent components are becoming increasingly smaller, which has no impact on the overall quality of the model. With three latent components, extracted during three iterations of the algorithm, *lmCLV* made it possible to obtain a value of $RMSE = 2.23$ and of $RMSE_{CV} = 2.29$. Regarding the $RMSE_{CV}$ criterion, in this example, *lmCLV* performed a little better than PLSR and PCR. As with *lmCLV*, the PLSR solution with three components could be retained. However, except for the first PLS component which was quite well correlated ($r = -0.94$) with the third latent component retained with *lmCLV* (also well correlated with the third prototypical variable), the other components were not pairwise well correlated. In fact, the PLS components are two by two uncorrelated, which is not the case for components from CLV.

The component-wise gradient boosting, or L2-boosting approach [28, 30] has also been considered for comparison purposes using the `glmboost` function included in the R package `mboost`. We used the same cross-validation structure as previously and tested step length values $\nu = 0.01$ and 0.1 to 1 by step of 0.1, with a maximal number of iteration of 1000. The best combination of the shrinkage parameter ν and the number of iterations, was chosen according to the $RMSE_{CV}$ criterion. For this example, the best combination was $\nu = 0.3$ and a stopping iteration number $m_{stop} = 30$, leading to a $RMSE_{CV}$ value of 2.45, which is similar to the optimal values observed with previous methods. With both of these parameters, 23 variables were selected: seven from $\mathcal{G}_1$, the five variables from $\mathcal{G}_2$, six variables from $\mathcal{G}_3$, and two variables from each of the last two group.

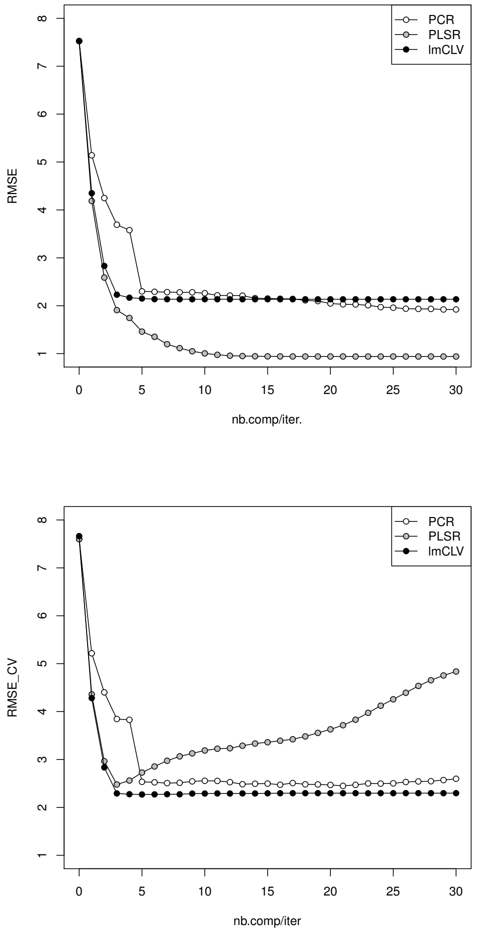

Figure 2: Evolution of the Root Mean Squared Errors.

3.2 Orange juice authentication case study

In this case study, a ^{1}H NMR spectroscopic profiling approach was investigated to discriminate between authentic and adulterated juices [31]. In this study, we considered the adulteration of orange juice (*Citrus sinensis*) with clementine juice (*Citrus reticulata*). Supplementation of substitution with cheaper or easier to find similar fruit is one of the type of fraud conducted within the fruit juice industry. For the experiment, twenty pure orange juices and ten pure clementine juices were selected from the Eurofins database. They are deemed to be representative of the variability of the fruit juices available on the market. From these juices, 120 blends were prepared by mixing one of the twenty orange juices and one of the ten clementine juices in known proportions. The proportion of the clementine juice in a mix were 10%, 20%, . . ., or 60%. The experimental design is described in more detail in [31]. The database was completed with the twenty authentic orange juices, for a total of 140 juice samples, and these were analyzed on an ^{1}H Nuclear Magnetic Resonance (NMR) spectroscopy platform.

The NMR variables are associated with chemical shifts, given in ppm. In the following, two spectral regions were simultaneously considered: The region from 6 to 9 ppm and the region from 0.5 to 2.3 ppm. The first spectral region mostly includes chemical shifts associated with aromatic components and the second one due to amino-acid-specific spectral shifts (among others). Between these regions lies the typical ^{1}H NMR spectra for orange juice sugars [32], which cannot discriminate between *Citrus sinensis* and *Citrus reticulata* juice. After a preprocessing binning step, 300 chemical shift variables were collected between 6 to 9 ppm, and 180 variables between 0.5 to 2.3 ppm.

The data matrix is therefore composed of n=140 observations and p=480 variables. The log-transformed data are available in the R package `ClustVarLV` (dataset `authen_NMR`). The mean of the log-transformed signals for both spectral regions are shown in Figure 3.

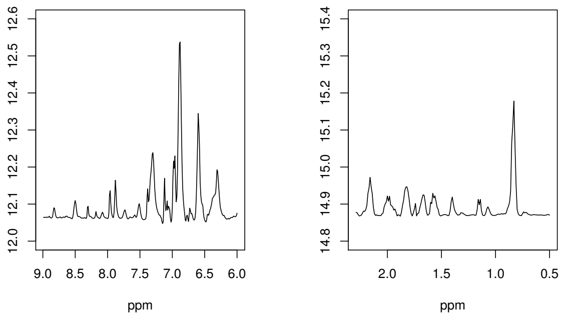

Figure 3: Mean spectrum of the log-transformed NMR signals according to the spectral region. Two subplots are shown due to the differences with order of magnitude in both regions.

In the following, a pareto-scaling was applied to each variable. This scaling, which consists of dividing each variable by the square root of its standard deviation, was shown in this case study [31] to be preferable to the usual pre-scaling by the standard deviation. Moreover, the splitting of the observations into ten segments was defined for Cross-Validation purposes. A proportional stratified allocation rule has been adopted in such a way that each segment contained two observations of each of the seven experimental levels, ranging from 0 to 60% of co-fruit added to pure orange juices.

Figure 4 shows the evolution of the errors in prediction, evaluated by means of the $RMSE_{CV}$ criterion, when the shrinkage parameter ν was set to 0.2, 0.5 or 0.8. On this basis, we choose to consider a value of the shrinkage parameter $\nu = 0.5$ for more detailed analysis of the model.

Besides the predictive purpose, one of the key features of *lmCLV* is to identify groups of explanatory variables and their associated latent variables that are, step by step, selected and involved in the model. The introduction sequence of the CLV latent variables provides an interesting insight. However, it should be noted that the smaller the shrinkage parameter, the more often the same group of explanatory variables, by the means of its latent variable, will appear. Therefore, some of the *lmCLV* algorithm's outputs are provided for each exhibited group, in order of its first occurrence, rather than by iteration. Thus, the OLS coefficients, $\alpha^{(m)}$, associated with a selected latent component $\mathbf{c}^{*(m)}$ (see step 4 in Sect.2.4) are aggregated on all iterations, m, to which this specific latent component is selected, up to the predefined maximum number of iterations, M. The same applies to the β coefficients of each of the (pre-processed) explanatory variables belonging to a selected group. Finally, a group importance criterion has been introduced. This importance is assessed as the sum, over all its occurrences, of the decrease in residual variance allowed by introducing the shrunken OLS estimate of the associated latent component.

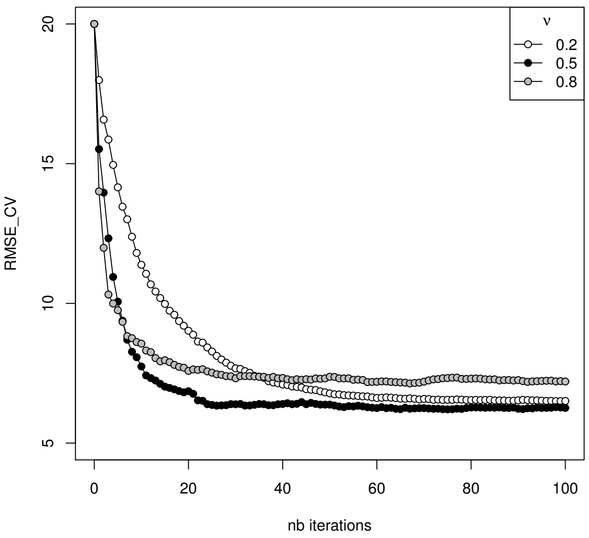

Figure 4: Evolution of the Root Mean Squared Error in Cross-validation with *lmCLV*, according to the number of iterations, and the values of the shrinkage parameter, ν, in the orange juice authentication case study.

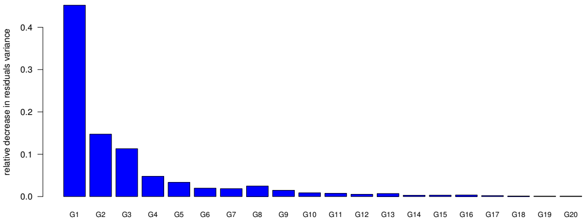

Figure 5: Relative Group Importance for the orange juice authentication case study. The Group Importance values are expressed relatively to the total variance of the response variable $\mathbf{y}$.

In our case study, using the whole data set, with *lmCLV* parameters $\nu = 0.5$ and $M = 25$, the group importance estimates are depicted in Figure 5. This reveals the presence of a group of spectral variables (G1) of high importance, as well as two other important groups (G2 and G3). Each group is numbered according to its first occurrence, which corresponds rather well with its importance in the model.

The first group involved nine spectral variables at 7.52, 7.51, 7.50, 6.77, 2.10, 2.08, 2.07, 2.06 and 2.04 ppm. The spectral range between 7.50 and 7.52 ppm, combined with the signal at 6.77 ppm and around 2 ppm, is particularly interesting and could be associated with 4 amino-3-methylbenzoic acid (see for instance the Spectral Database for Organic Compounds [33]. This information is quite stable: the spectral variables at 7.52, 7.51, 6.77 and 2.08 ppm had been selected in the first group at each of the 10 iterations of the CV procedure. The same variables have also be noted in [31], but their presumed association with the same compound could not be so clearly highlighted. The second group of spectral variables consisted of ten variables (7.15, 7.10-7.09, 1.10-1.05 and 0.96 ppm). The CV procedure showed that the range between 1.09 and 1.05 ppm and the peak at 7.10-7.09 ppm was simultaneously retrieved 7 times out of 10 as the second or third group. Lastly, the third group consisted of nine spectral variables, and specifically a range between 1.65 and 1.60 ppm. The relationship between these first three group components and the proportion of clementine juice in a mix is shown in Figure 6.

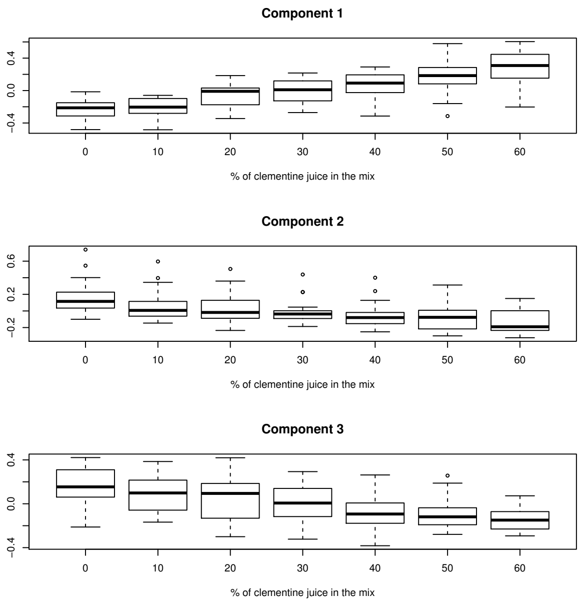

Figure 6: relationship between the first three group components and the proportion of clementine juice added to a mix.

In Table 3.2, the predictive ability of *lmCLV* with $\nu = 0.5$ is compared with those of various alternatives among the most popular statistical learning approaches [1, 27]: (i) Sparse PLS Regression [17] which is very widely used in chemometrics; (ii) Elastic-Net [5], a penalized regression method allowing sparsity and grouping effect in variable selection; (iii) L2-boosting [28, 30], a simple boosting approach for squared error loss function which is based on a regression model, unlike the base learner involved in *lmCLV*; and (iv) Random Forest [34], one of the most popular and intuitive machine learning method for high dimensional predictive problem. The $RMSE_{CV}$ were evaluated using the same 10-fold Cross-Validation division as above. For Sparse PLS Regression (sPLSR) the R package `spls` was used, with parameter $\eta \in \{0, 0.1, \ldots, 0.9\}$ and K, the number of components, varying from 1 to 20. Elastic-Net (E-Net) was adjusted using the R package `glmnet` and by considering the Gaussian family model with mixing parameter α fixed to 0.5. The parameter λ was chosen as the mean of optimal values determined for each CV fold. For the L2-boosting approach (L2-boost), in the R package `mboost`, values of parameter ν from 0 (0.01 precisely) to 1, by 0.1, with a maximal number of 1000 iterations, was explored. RandomForest was considered because this machine learning approach often proves to be effective in high-dimensional classification or regression problems and provides an interesting variables importance ranking. The R package `randomForest` was used with parameter $ntree = 5000$, and default values for the two other parameters, $mtry$ (i.e. 160) and $nodesize$ (i.e. 5).

method[a]	parameters setting	$RMSE_{CV}$	# var.selected
lmCLV	$\nu = 0.5, M = 25$	6.366	145
sPLSR	$\eta = 0.3, K = 5$	6.427	150
E-Net	$\alpha = 0.5, \lambda = 0.08$	6.335	90
L2-boost	$\nu = 0.2, M = 209$	6.742	75
RdF	$(5000, 160, 5)$[b]	9.992	146[c]

[a] All the methods was applied using available R packages. Their paramaters were determined on the basis of Cross-Validation with the same samples division.

[b] $ntree = 5000, mtry = 160, nodesize = 5.$

[c] Variables with a standardized variable importance value greater than 3 were selected.

Table 1: Root Mean Squared Error in cross-validated prediction of the percentage of clementine juice added to the orange juices, according to various methods.

As shown in Table 3.2, *lmCLV* has an expected prediction performance similar to Sparse PLS Regression, Elastic-Net and the L2-boosting approach. Quite surprisingly, in this case study, Random Forest did not perform very well. In fact, one can observe that by Random Forest the prediction of the percentage of co-fruit added was overestimated or underestimated at the extremes of the experimental scale, i.e. for 0-10% or 50-60%.

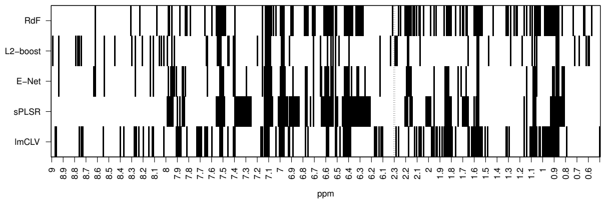

Figure 7: Location of the spectral variables involved in the models according to the method considered, in the orange juice authentication case study.

The number of variables involved in the model fitted on the whole dataset, using the predetermined model's parameters, is indicated in the last column of Table 3.2. As shown in Figure 7, while Sparse PLS Regression has the tendency to identify fewer but larger spectral ranges, the L2-boosting approach retained a relatively small number of variables associated with narrow ranges. For *lmCLV*, 20 groups and their latent variable were identified. Since several small spectral ranges were merged within the same group, the total number of spectral variables involved was rather high. However, the order of extraction and the grouping effect, which are a specificity of *lmCLV*, cannot be revealed in Figure 7. Globally, 19 spectral variables were retained with the five methods considered herein. We systematically found the variables at 7.51-7.52 ppm, but not the other variables that belonged to the first group extracted with *lmCLV*. The variables between 7.09 to 7.12 ppm as well as the variable at 1.07 ppm are also present, as in the second group extracted with *lmCLV*. We finally noted that the areas at 6.96-6.98 ppm, 1.56-1.57 ppm, 0.86-0.88 ppm were selected with the five methods.

4 Conclusion

In this study, we introduced a strategy for linear model fitting based on the hypothesis that high-dimensional datasets often include highly correlated variables having similar effect on the response variable. This is specifically the case for modern scanning instruments such as those used in spectroscopy (infrared, near-infrared, Raman, NMR,...) which are able to collect a large number of sequential spectral variables, several of them being representative of one feature/signal. Omic data, and specifically metabolomic data, contain a large quantity of measured elements that are components of the same metabolic pathways and that constitute the biological information that is sought. The basis of the approach is to then cluster the explanatory variables first, as other authors have already proposed [19, 18]. In *lmCLV*, the clustering stage is based on the CLV method and consists of identifying unidimensional latent components which represent clusters of variables. These latent components play the role of the predictors in the second stage which results in introducing the explanatory variables in a group-wise fashion.

In constrast with [19] for the CRL approach (Cluster Representative Lasso), and [20] or [21] in which each cluster representative is defined as the mean variable of a group of variables, the CLV latent components are defined at the same time as the clusters, and are derived from the eigen decomposition of the within cluster covariance matrix. In addition, for the construction of the regression model itself, we have adopted an L2-boosting approach, rather than a Lasso approach. This makes it possible to build a simple and efficient algorithm in which the CLV dendrogram constitutes the base-learner.

Compared to our previous study [26], the algorithm proposed here requires much less computer resources. In the previous study, the clustering stage was performed on the data matrix combining the residuals of the response variable and the explanatory variables, and was repeated for each iteration. However, the clustering of several hundred variables requires the largest part of the computation time. Using the new version of *lmCLV*, for the case study on orange juice authentication (Sect. 3.2), which involved 480 explanatory variables, the whole procedure ($\nu = 0.5$, $M = 100$) took 1 min 48 sec on one 3.4 GHz processor, including 1 min 25 sec for the clustering stage alone. For the implementation of this algorithm, a function `lmCLV` will be available in version 2.0.2 of the R package `ClustVarLV`.

In both examples presented in this study, *lmCLV* was shown to have similar predictive efficiency to other methods, and importantly, provided additional interpretability capacity. The use of latent components that are easy to interpret, because each of them is associated with a group of collinear variables, is consistent with the development of modern simplifying approaches for modelization. However, compared to Lasso-based methodologies, dimensionality reduction based on variables clustering makes it easier to identify directions of interest for prediction and makes it possible to highlight functional links between explanatory variables where they exist.

Acknowledgements

The author is grateful to Freddy Thomas (Authenticity unit, Eurofins Analytics France, Nantes, France) for the opportunity to present the case study on orange juice authentication.

References

[1] T. Hastie, R. Tibshirani and J. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, (Second ed.). Springer Series in Statistics. New-York: Springer

[2] R. Tibshirani (1996). Regression Shrinkage and Selection Via the Lasso, , *J. Roy. Stat. Soc. B*, **58**, 267–288.
`https://doi.org/10.1111/j.2517-6161.1996.tb02080.x`,

[3] G. Celeux, C. Maugis-Rabusseau and M.Sedki (2019). Variable selection in model-based clustering and discriminant analysis with a regularization approach. *Data Anal. Classif.*, **13**, 259–278.
`https://doi.org/10.1007/s11634-018-0322-5`

[4] Z.Y. Algamal and M.H. Lee (2019). A two-stage sparse logistic regression for optimal gene selection in high-dimensional microarray data classification, *Adv. Data Anal. Classif.*, **13**, 753—771.
`https://doi.org/10.1007/s11634-018-0334-1`

[5] H. Zou and T. Hastie (2005). Regularization and variable selection via the elastic net. *J. Roy. Stat. Soc.*

B, **67**, 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x

[6] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu and K. Knight (2005). Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. B*, **67**, 91–108. https://doi.org/10.1111/j.1467-9868.2005.00490.x

[7] M. Yuan and Y. Lin (2006). Model selection and estimation in regression with grouped variables. *J. Roy. Stat. Soc. Ser. B*, **68**, 49–67. https://doi.org/10.1111/j.1467-9868.2005.00532.x

[8] J. Friedman, T. Hastie, and R. Tibshirani (2010). *A note on the group lasso and a sparse group lasso*. Technical report, Statistics Department, Stanford University.

[9] B. Zeng, X. M. Wen and L.Zhu (2017). A linkfree sparse group variable selection method for single-index model. *J. Appl. Stat.*, **44**, 2388–2400. https://doi.org/10.1080/02664763.2016.1254731

[10] Bondell, H. D. and B. J. Reich (2008). Simultaneous regression shrinkage, variable selection and clustering of predictors with OSCAR. *Biometrics*, **64**, 115-123. https://doi.org/10.1111/j.1541-0420.2007.00843.x

[11] L. Yengo, J. Jacques, C. Biernack, and M. Canoui (2016). Variable clustering in highdimensional linear regression: the R package CLERE. *R journal*, **8**, 92–106. https://doi.org/10.32614/RJ-2016-006

[12] S. M. Curtis and S. K. Ghosh (2011). A bayesian approach to multicollinearity and the simultaneous selection and clustering of predictors in linear regression. *J. Stat. Theory Pract.*, **5**, 715–735. https://doi.org/10.1080/15598608.2011.10483741

[13] S. Chakraborty and A. C. Lozano (2019). A graph laplacian prior for bayesian variable selection and grouping. *Comput. Stat. Data An.*, **136**, 72–91. https://doi.org/10.1016/j.csda.2019.01.003

[14] I. T. Jolliffe, N. T. Trendafilov and M. Uddin (2003). A modified Principal Component technique based on the lasso. *J. Comput. Graph. Stat.*, **12**, 531–547. https://doi.org/10.1198/1061860032148

[15] H. A. Chipman and H. Gu (2005). Interpretable dimension reduction. *J. Appl. Stat.*, **32**, 969–987. https://doi.org/10.1080/02664760500168648

[16] T. F. Cox and D. S. Arnold (2018). Simple components. *J. Appl. Stat.*, **45**, 83–99. https://doi.org/10.1080/02664763.2016.1268104

[17] H. Chun and S. Keles (2010). Sparse partial least squares for simultaneous dimension reduction and variable selection. *J. Roy. Stat. Soc. B*, **72**, 3–25. https://doi.org/10.1111/j.1467-9868.2009.00723.x

[18] D. G. Enki, N. T. Trendafilov and I. T. Jolliffe (2013). A clustering approach to interpretable principal components. *J. Appl. Stat.*, **40**, 583–599. https://doi.org/10.1080/02664763.2012.749846

[19] P. Bühlmann, P. Rütimann, S. van de Geer and C.-H. Zhan (2013). Correlated variables in regression: clustering and sparse estimation. *J. Stat. Plan. Infer.*, **143**, 1835–1858. https://doi.org/10.1016/j.jspi.2013.05.019

[20] T. Hastie, R. Tibshirani, D. Botstein and P. Brown (2001). Supervised harvesting of expression trees. *Genom. Biol.*, **2**, 1–12. https://doi.org/10.1186/gb-2001-2-1-research0003

[21] M. Y. Park, T. Hastie and R. Tibshirani (2007). Averaged gene expressions for regression. *Biostat.*, **8**, 212–227. https://doi.org/10.1093/biostatistics/kxl002

[22] E. Vigneau and E. Qannari (2003). Clustering of variables around latent components. *Comm. Stat.-Simul Comput.*, **32**, 1131–1150. https://doi.org/10.1081/SAC-120023882

[23] E. Vigneau and M. Chen (2016). Dimensionality reduction by clustering of variables while setting aside atypical variables. Electron. *J. Appl. Stat. An.*, **9**, 134–153. https://doi.org/10.1285/i20705948v9n1p134

[24] E. Vigneau, M. Chen and E. M. Qannari (2015). ClustVarLV: An R package for the clustering of variables around latent variables. *R journal*, **7**, 134–148. https://doi.org/10.32614/RJ-2015-026

[25] A. Figueiredo and P. Gomes (2015). Clustering of variables based on Watson distribution on hypersphere: A comparison of algorithms. *Comm. Stat. - Simul Comput.*, **44**, 2622–2635. https://doi.org/10.1080/03610918.2014.901353

[26] M. Chen and E. Vigneau (2016). Supervised clustering of variables. *Adv. Data Anal. Classif.*, **10**, 85–101. https://doi.org/10.1007/s11634-014-0191-5

[27] B. Efron. and T. Hastie (2016). *Computer age statistical inference: algorithms, evidence and data science*. New York: Cambridge University Press. https://doi.org/10.1017/cbo9781316576533

[28] P. Bühlmann and T. Hothorn (2007). Boosting algorithms: Regularization, prediction and model fitting. *Stat. Sc.*, **22**, 477–505. http://dx.doi.org/10.1214/07-STS242

[29] D. Karlis, G. Saporta and A. Spinakis (2003). A simple rule for the selection of principal components.*Comm. Stat. - Theor. M.*, **32**, 643–666. https://doi.org/10.1081/STA-120018556

[30] B. Hofner, A. Mayr, N. Robinzonov and M. Schmid (2014). Model-based boosting in R: A hands-on tutorial using the R package mboost. *Comput. Stat.*, **29**, 3–35. https://doi.org/10.1007/s00180-012-0382-5

[31] E. Vigneau and F. Thomas (2012). Model calibration and feature selection for orange juice authentication by 1H NMR spectroscopy. *Chemometr. Intell. Lab. Sys.*, **117**, 22–30. https://doi.org/10.1016/j.chemolab.2011.05.006

[32] P. Rinke, S. Moitrier, E. Humpfer, S. Keller, M. Moertter, M. Godejohann, G. Hoffmann, H. Schaefer and M. Spraul (2007). An 1H-NMR technique for high troughput screening in quality and authenticity control of fruit juice and fruit juice raw materials- SGF-profiling. *Fruit Process*, **1**, 10–18.

[33] SDBSWeb (2020). Spectral Database for organic Compounds. https://sdbs.db.aist.go.jp (National Institute of Advanced Industrial Science and Technology)

[34] L. Breiman (2001). Random forests. *Mach. Learn.*,**45**, 5–32. https://doi.org/10.1023/A:1010933404324