Variable clustering in high dimensional linear regression models

Loïc Yengo, Julien Jacques, Christophe Biernacki

Variable clustering in high dimensional linear regression models

Loic Yengo[1,2,3], Julien Jacques[2,3] and Christophe Biernacki[2,3]

[1]CNRS UMR 8199, Laboratoire Génomique et maladies métaboliques, France.
[2]CNRS UMR 8524, Laboratoire Paul Painlevé, Université Lille 1, France.
[3]Inria Lille Nord-Europe, Équipe MODAL, France.

Abstract

For the last three decades, many scientific fields have known profound changes caused by the advent of technologies for massive data collection. What was first seen as a blessing, rapidely turned out to be termed as the curse of dimension. Reducing the dimension has therefore become a challenge in statistical learning. In high dimensional linear regression models, the quest for parsimony has long been driven by the idea that a few relevant variables may be sufficient to describe the modeled phenomenon. Recently, a new paradigm was introduced in a series of articles from which the present work derives. We propose here a model that simultaneously performs variables clustering and regression. Our approach no longer considers the regression coefficients as fixed parameters to be estimated, but as unobserved random variables following a Gaussian mixture model. The latent partition is then determined by maximum likelihood and predictions are obtained from the conditional distribution of the regression coefficients given the data. The number of latent components is chosen using a BIC criterion. Our model has very competitive predictive performances compared to standard approaches and brings significant improvements in interpretability.
Keywords: Dimension reduction, Linear regression, Variable clustering.

1 Introduction

We consider in the present article the standard linear regression model defined as

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i,\ i = 1, \ldots, n. \tag{1}$$

For some individual i, y_i is the observed response, x_{ij} is an observed value for the j-th covariate and ε_i is an error term often assumed to be normally distributed. We also often assume the ε_i's to be independent and identically distributed.

The dimension p of model (1) is tighly related to both its interpretability and ability to yield reliable prediction. Pragmatically, we can argue for the former that the more covariates we add to the model the harder becomes its interpretation. Stein [2] established besides, that the mean

prediction squared error attributable to a linear regression model increases with its dimension. Reducing the model's dimension therefore pursues the goal of minimizing prediction error while keeping the model interpretable. This problem, also referred to as the *bias-variance trade-off* in the literature, becomes more challenging when the set of covariates exceeds the sample size in classical experiments. This situation is called high dimensionality and has fueled a number of researches during the last three decades.

Variables selection is one of the most popular approaches for reducing dimensionality. Although it has a direct impact on p, stepwise algorithms for finding the best subset of predictors had a mitigated success because of their heavy computational burden. With a lower computational cost, penalized approaches rose and spread rapidly. Penalized approaches impose an identifiability constraint on the vector of regression coefficient $\beta = (\beta_1, \ldots, \beta_p)$ that depends on a tuning parameter. The least absolute shrinkage and selection operator (LASSO) [11] is probably the most emblematic of this second family of approaches. LASSO imposes an upper-bound to the L^1 norm of β. This upper-bound is a tuning parameter specified by the user.

Another relevant approach for reducing dimensionality consists in identifying patterns under which covariates can be pooled together. This idea was recently implemented in a gene expression study [15]. In that study, groups of genes were built from hierarchical clustering of gene expression levels. The authors created surrogate covariates by averaging gene expression levels within each group. Those new predictors were afterwards included in a linear regression model, replacing the primary variables. The major limitation in this approach is the independence between the prediction and clustering parts of their strategy. Consequently, effects of the surrogate covariates can be diluted if they contain primary variables with either no effect or even opposite effects on the response. To sidestep the previous limitation, Bondell and Reich [3] introduced in 2008 the octogonal shrinkage and clustering algorithm for regression (OSCAR). The OSCAR methodology belongs to the family of penalized approaches. It imposes a constraint on β that is a weighted combination of the L^1 norm and the pairwise L^∞ norm. Upper-bounding the pairwise L^∞ norm enforces the covariates to have close coefficients. When the constraint is strong enough, closeness translates into equality achieving thus a grouping property. In the aftermath of OSCAR other methodologies aiming at simultaneously performing parameter estimation and clustering were proposed. We can for instance refer to Petry [13] and She [16] works which also mixed L^1 and pairwise L^∞ penalties or Daye [9] and Shen [14] works based on alternative penalties.

In line with the latter works, we introduce in this paper the clusterwise effect regression (CLERE), a new methodology aiming at simultaneously performing regression and clustering of covariates. CLERE considers each β_j no longer as a fixed parameter but as an unobserved random variable following a Gaussian mixture distribution. The present paper is organized as follows. In Section 2 we present our model, its parametrization as well as an identifiability condition. In Section 3, a maximum likelihood strategy is presented for estimating the model parameters as well as a criterion to select the number of latent groups. In this section is also illustrated how to make prediction using our model. Section 4 presents then some numerical experiments both on

simulated and real data. This section aims at presenting the predictive performances of our model compared to standard approaches for dimension reduction in linear regression. Another part of Section 4 presents a detailed case study on real data illustrating the improvements brought by CLERE in terms of parcimony and interpretation. Finally, the perspectives of this research are discussed in Section 5.

2 Model definition and notations

2.1 Model

As aforementioned, the number of predictors may be very large (up to 100-fold) with respect to the number of samples ($p \gg n$). It is thus impossible to uniquely estimate each coefficient β_j. We may however hypothesize the existence of g latent groups of covariates, say $(G_1, \ldots, G_g)$, within which the β_j's are sufficiently close to one another that all of them may be summarized by their average. Among possible mathematical translations of the latter assumption, we propose to consider the β_j's no longer as fixed effect parameters but as unobserved independent random variables following a Gaussian mixture distribution:

$$\beta_j \sim \sum_{k=1}^{g} \pi_k \mathcal{N}\left(b_k, \gamma^2\right). \tag{2}$$

In other words, we assume for each β_j the existence of a Bernoulli distributed random variable, z_{jk} which equals 1, with probability π_k, when β_j is drawn from the k-th component of the mixture. Our model can then be written:

$$\begin{cases} y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i \\ \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \\ \beta_j | \mathbf{z}_j \sim \mathcal{N}\left(\sum_{k=1}^{g} b_k z_{jk}, \gamma^2\right) \\ \mathbf{z}_j = (z_{j1}, \ldots, z_{jg}) \sim \mathcal{M}\left(\pi_1, \ldots, \pi_g\right). \end{cases} \tag{3}$$

We add the following identifiability condition to model (3):

$$\forall k = 1, \ldots, g \ \sum_{j=1}^{p} z_{jk} \geq 1. \tag{4}$$

This condition basically says that none of the groups should be empty.

2.2 Notations

We introduce here some matricial and vectorial notations:
$\mathbf{y} = (y_1, \ldots, y_n)'$, $\beta = (\beta_1, \ldots, \beta_p)'$, $\mathbf{X} = (x_{ij})$, $\mathbf{Z} = (z_{jk})$, $\mathbf{b} = (b_1 \ldots b_g)'$ and $\pi = (\pi_1, \ldots, \pi_g)'$.
Let $\log p(\mathbf{y}|\mathbf{X}; \theta)$ denote the log-likelihood of the model (3) assessed for the parameter $\theta =$

$(\beta_0, \mathbf{b}, \pi, \sigma^2, \gamma^2)$.

3 Estimation, prediction and model selection

3.1 Maximum Likelihood Estimation

The estimation strategy studied in this paper is the maximum likelihood estimation (MLE). The log-likelihood $\log p(\mathbf{y}|\mathbf{X};\theta)$ is defined as

$$\log p(\mathbf{y}|\mathbf{X};\theta) = \log \left[\sum_{\mathbf{Z}} \int p(\mathbf{y}, \beta, \mathbf{Z}|\mathbf{X};\theta) \mathrm{d}\beta \right]. \tag{5}$$

It involves integration over unobserved data $(\beta, \mathbf{Z})$ which renders impossible a direct maximization to estimate θ.

The Expectation-Maximization (EM) algorithm [10] has been introduced to perform MLE in the presence of unobserved data. The EM algorithm is an iterative method, which starts with initial estimates of the parameters and updates these estimates at each iteration until convergence is achieved. We propose in the following subsections its implementation in the special case of model (3).

3.1.1 Initialization

The algorithm is initialized using primary estimates ${\beta_j}^{(0)}$ of each β_j. The latter can be either obtained from univariate regression coefficients or from penalized approaches like the LASSO or the ridge regression.

Model (2) is then fitted using $\beta^{(0)} = \left(\beta_1^{(0)}, \ldots, \beta_p^{(0)}\right)$ as observed data, to produce starting values for $\mathbf{b}$, π and γ^2. An initial partition $\mathbf{Z}^{(0)}$ is also naturally derived from the previous fit. β_0 and σ^2 are then initialized using $\beta^{(0)}$ as following:

$$\beta_0^{(0)} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j^{(0)} x_{ij} \right) \text{ and } {\sigma^2}^{(0)} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \beta_0^{(0)} - \sum_{j=1}^{p} \beta_j^{(0)} x_{ij} \right)^2.$$

3.1.2 (Stochastic) Expectation step

During iteration (d) of the algorithm, the log-likelihood of the full data $\log p(\mathbf{y}, \beta, \mathbf{Z}|\mathbf{X}; \theta^{(d)})$ has the following expression

$$\begin{aligned}\log p(\mathbf{y}, \beta, \mathbf{Z}|\mathbf{X}; \theta^{(d)}) &= \log p(\mathbf{y}|\beta, \mathbf{X}; \beta_0^{(d)}, \sigma^{2(d)}) + \log p(\beta, \mathbf{Z}|\mathbf{X}; \mathbf{b}^{(d)}, \pi^{(d)}, \gamma^{2(d)}) \\ &= -\frac{n}{2}\log\left(2\pi\sigma^{2(d)}\right) - \frac{1}{2\sigma^{2(d)}}\sum_{i=1}^{n}\left(y_i - \beta_0^{(d)} - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 \\ &\quad -\frac{p}{2}\log\left(2\pi\gamma^{2(d)}\right) + \sum_{j=1}^{p}\sum_{k=1}^{g} z_{jk}\left(\log\pi_k^{(d)} - \frac{\left(\beta_j - b_k^{(d)}\right)^2}{2\gamma^{2(d)}}\right).\end{aligned}$$

In classical EM algorithm, the E-step requires, at each iteration, the calculation of the expectation of the log-likelihood of the full data $\log p\left(\mathbf{y}, \beta, \mathbf{Z}|\mathbf{X}; \theta^{(d)}\right)$, with respect to the conditional distribution of unobserved data given observed data. This quantity generally denoted as $Q\left(\theta|\theta^{(d)}\right)$, does not have a closed form in model (3). We therefore approximate $Q\left(\theta|\theta^{(d)}\right)$ using Monte Carlo simulations. This stochastic version of the EM algorithm was introduced in [6] under the name of Monte Carlo EM (MCEM) algorithm. A Gibbs sampling scheme is proposed to generate draws from the probability distribution $p\left(\beta, \mathbf{Z}|\mathbf{y}, \mathbf{X}; \theta^{(d)}\right)$. In model (3), Gibbs sampling requires the definition of the conditional distributions $p\left(\beta|\mathbf{Z}, \mathbf{y}, \mathbf{X}; \theta^{(d)}\right)$ and $p\left(\mathbf{Z}|\beta, \mathbf{y}, \mathbf{X}; \theta^{(d)}\right)$. The latter distributions are given below in Equations (6) and (7):

$$\begin{cases} \beta|\mathbf{Z}, \mathbf{y}; \theta^{(d)} \sim \mathcal{N}\left(\mu^{(d)}, \Sigma^{(d)}\right) \\ \mu^{(d)} = \left[\mathbf{X}'\mathbf{X} + \frac{\sigma^{2(d)}}{\gamma^{2(d)}}\mathrm{I}_p\right]^{-1}\mathbf{X}'\mathbf{y} + \frac{\sigma^{2(d)}}{\gamma^{2(d)}}\left[\mathbf{X}'\mathbf{X} + \frac{\sigma^{2(d)}}{\gamma^{2(d)}}\mathrm{I}_p\right]^{-1}\mathbf{Z}\mathbf{b}^{(d)} \\ \Sigma^{(d)} = \sigma^{2(d)}\left[\mathbf{X}'\mathbf{X} + \frac{\sigma^{2(d)}}{\gamma^{2(d)}}\mathrm{I}_p\right]^{-1} \end{cases} \tag{6}$$

and

$$p\left(z_{jk} = 1|\beta; \theta^{(d)}\right) \propto \pi_k^{(d)} \exp\left(-\frac{\left(\beta_j - b_k^{(d)}\right)^2}{2\gamma^{2(d)}}\right). \tag{7}$$

Now suppose we have sampled $\left[\left(\beta^{(1,d)}, \mathbf{Z}^{(1,d)}\right), \ldots, \left(\beta^{(M_d,d)}, \mathbf{Z}^{(M_d,d)}\right)\right]$ from $p\left(\beta, \mathbf{Z}|\mathbf{y}, \mathbf{X}; \theta^{(d)}\right)$ and verifying the condition (4). The approximated E-step can then be written as follows:

$$Q\left(\theta|\theta^{(d)}\right) = \mathbb{E}\left[\log p(\mathbf{y}, \beta, \mathbf{Z}|\mathbf{X}; \theta^{(d)})|\mathbf{y}, \mathbf{X}; \theta^{(d)}\right] \approx \frac{1}{M_d}\sum_{m=1}^{M_d}\log p(\mathbf{y}, \beta^{(m,d)}, \mathbf{Z}^{(m,d)}|\mathbf{X}; \theta^{(d)}). \tag{8}$$

The computational time and the convergence of the algorithm is governed by the choice of M_d. In [6], the authors suggested using small values for M_d (around 20) when starting the algorithm and increases this value along with number of iterations. In this paper however M_d was set to a

constant large value.

3.1.3 Maximization step

The M-step consists in maximizing $Q\left(\theta|\theta^{(d)}\right)$ with respect to θ. We get the following update equations:

$$\pi_k^{(d+1)} = \frac{1}{M_d p} \sum_{m=1}^{M_d} \sum_{j=1}^{p} z_{jk}^{(m,d)}, \tag{9}$$

$$b_k^{(d+1)} = \frac{1}{M_d p \pi_k^{(d+1)}} \sum_{m=1}^{M_d} \sum_{j=1}^{p} z_{jk}^{(m,d)} \beta_j^{(m,d)}, \tag{10}$$

$$\gamma^{2(d+1)} = \frac{1}{M_d p} \sum_{m=1}^{M_d} \sum_{j=1}^{p} \sum_{k=1}^{g} z_{jk}^{(m,d)} \left(\beta_j^{(m,d)} - b_k^{(d+1)}\right)^2, \tag{11}$$

$$\beta_0^{(d+1)} = \frac{1}{n} \sum_{i=1}^{n} \left[y_i - \sum_{j=1}^{p} \left(\frac{1}{M_d} \sum_{m=1}^{M_d} \beta_j^{(m,d)} \right) x_{ij} \right], \tag{12}$$

$$\sigma^{2(d+1)} = \frac{1}{n M_d} \sum_{m=1}^{M_d} \sum_{i=1}^{n} \left(y_i - \beta_0^{(d+1)} - \sum_{j=1}^{p} \beta_j^{(m,d)} x_{ij} \right)^2. \tag{13}$$

3.1.4 Stopping rule

Choosing a stopping rule for MCEM is far to be obvious. When first introduced in [6], only visual criteria were proposed. Caffo and co-workers [12] proposed to stop the algorithm when $Q\left(\theta^{(d+1)}|\theta^{(d)}\right) - Q\left(\theta^{(d)}|\theta^{(d)}\right)$ is stochastically small. When M_d is large enough, the central limit theorem, gives us the following asymptotic equation:

$$\mathbb{P}\left(\sqrt{M_d} \frac{Q\left(\theta^{(d+1)}|\theta^{(d)}\right) - Q\left(\theta^{(d)}|\theta^{(d)}\right)}{\sigma\left[Q\left(\theta^{(d)}|\theta^{(d)}\right)\right]} \geq t \right) = 1 - \phi\left(t\right), \tag{14}$$

where ϕ is the cumulative probability function of a scaled and center normal distribution and $\sigma\left[Q\left(\theta^{(d)}|\theta^{(d)}\right)\right]$ is defined as

$$\sigma\left[Q\left(\theta^{(d)}|\theta^{(d)}\right)\right] = \frac{1}{M_d} \sum_{m=1}^{M_d} \left[\log p(\mathbf{y}, \beta^{(m,d)}, \mathbf{Z}^{(m,d)}|\mathbf{X}; \theta^{(d)}) - Q\left(\theta^{(d)}|\theta^{(d)}\right)\right]^2. \tag{15}$$

The algorithm can then be stopped when the probability defined in Equation (14) belows a

user-specified threshold. In the present article we terminated the algorithm after a predefined large number of EM iterations.

3.2 Prediction

If $\mathbf{X}^v$ denotes a new design matrix for which we want to predict the response $\mathbf{y}^v$, then we can define the predicted response $\widehat{\mathbf{y}}$ as

$$\widehat{\mathbf{y}} = \mathbf{X}^v \mathbb{E}\left[\beta | \mathbf{y}, \mathbf{X}; \widehat{\theta}\right] \tag{16}$$

where $\widehat{\theta}$ is the maximum likelihood estimate of θ.

3.3 Model selection

Model (3) depends on a tuning parameter g, which is the assumed number of groups of covariates. In few situations, this number can be chosen *a priori*, however in a more general setting a strategy should be proposed to make such a choice. We propose the Bayesian information criterion [5] (BIC) as a means to select g. This criterion was prefered to other criteria based on estimates of the out-of-sample prediction error like cross-validation (CV) because of its low computational cost. Moreover, the number of parameters of model (3) which is $2(g+1)$ is often small compared to the sample size. Therefore we may expect such asymptotic criterion to yield good results in this context. In model (3) the BIC when g is fixed has the following expression:

$$BIC = -2 \log p(\mathbf{y} | \mathbf{X}; \widehat{\theta}) + 2(g+1) \log(n). \tag{17}$$

As the calculation of the likelihood is still untractable, we can derive from Equation (5), an approximation of the BIC criterion using Monte Carlo simulations similar to the previous stochastic E-step.

4 Numerical experiments

In this section we compare in terms of prediction error, our approach CLERE with standard dimension reduction approaches. The methods selected for comparison are the variables selection using LARS algorithm [1], the ridge regression [4], the elastic net [8] and the LASSO [11]. All these methods are implemented in freely available R packages *lars* and *glmnet* (for ridge, LASSO and elastic net). Those packages were used with default options. When running CLERE, the maximum number of EM iterations was set 1000 and the number M_d of Monte Carlo samples was set to 500.

4.1 Simulated data

4.1.1 Description

The simulated data are presented under three scenarios. For each scenario, 100 training data sets were simulated from the standard linear regression model (1). All training data sets consist of $n = 50$ simulated individuals with $p = 100$ variables. In each scenario a validation set consisting of 500 individuals was used to calculate the scaled mean squared prediction error.

If $\left(\mathbf{y}^t, \mathbf{X}^t\right)$ and $(\mathbf{y}^v, \mathbf{X}^v)$ are respectively the training and validation data sets, then the scaled mean squared prediction error MSE is calculated as:

$$\text{MSE} = \frac{\|\mathbf{y}^v - \widehat{\mathbf{y}}\left(\mathbf{X}^v, \mathbf{y}^t, \mathbf{X}^t\right)\|_2}{\|\mathbf{y}^v\|_2} \tag{18}$$

where $\widehat{\mathbf{y}}\left(\mathbf{X}^v, \mathbf{y}^t, \mathbf{X}^t\right)$ is the predicted response and $\|.\|_2$ stands for the L^2 norm. For CLERE, predictions are obtained using Equation (16). Each of the methods selected for comparison provide a fitted value $\widehat{\beta}$ for β. A predicted response under the design $\mathbf{X}^v$ is then calculated as $\mathbf{X}^v\widehat{\beta}$. In all simulations, design matrices $\mathbf{X}^t$ and $\mathbf{X}^v$ were simulated as independently normally distributed:

$$\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{19}$$

where $\mathbf{R} = \left(r_{jj'}\right)$ is a $p \times p$ defined by $r_{jj'} = 0.5^{|j-j'|}$. In all scenarios, parameters β_0 and σ^2 equal respectively 0 and 100.
The three scenarios are presented below.

1. In scenario 1, the vector β of regression coefficients is given by:

$$\beta = (\underbrace{0, \ldots, 0}_{36}, \underbrace{1, \ldots, 1}_{28}, \underbrace{3, \ldots, 3}_{20}, \underbrace{7, \ldots, 7}_{12}, \underbrace{15, \ldots, 15}_{4})'.$$

2. In scenario 2, the vector β of regression coefficients is given by:

$$\beta = (\underbrace{0, \ldots, 0}_{36}, \underbrace{4, \ldots, 4}_{28}, \underbrace{24, \ldots, 24}_{20}, \underbrace{124, \ldots, 124}_{12}, \underbrace{624, \ldots, 624}_{4})'.$$

3. In scenario 3, the regression coefficients are chosen uniformally between -10 and +10. This can be mathematically written with the following equation:

$$\forall j,\ \beta_j = -10 + (j-1) \times \frac{20}{99}.$$

Scenarios 1 and 2 were chosen to favor variables selection approaches like the LASSO. In those

scenarios indeed 36 out of 100 covariates do not influence the response. Moreover the number of effective variables decreases with their effect size. Scenario 3 was proposed to illustrate the relative predictive performances of CLERE under the assumption that almost all covariates contributes to the response.

4.1.2 Results

Table 1 summarizes the MSE calculated under each scenario. We also considered a measure of model complexity being either the number of non-zero parameters or simply the number of parameters for CLERE. Using the latter measure, the present simulation study illustrates that CLERE selects the simplest model in all the considered scenarios.

Scenario		100× averaged MSE (Std. Err)	Averaged number of parameters
1	Lars	52.3 (1.59)	49
	LASSO	16.3 (0.39)	41
	Ridge	59.4 (0.45)	100
	Elastic Net	14.3 (0.34)	49
	CLERE (g=5)	21.3 (0.65)	12
	CLERE	18.8 (0.73)	17
2	Lars	9.86 (0.93)	49
	LASSO	1.14 (0.04)	33
	Ridge	65.7 (0.37)	100
	Elastic Net	1.06 (0.04)	33
	CLERE (g=5)	0.42 (0.14)	12
	CLERE	0.32 (0.12)	15
3	Lars	71.0 (1.72)	49
	LASSO	35.6 (0.82)	45
	Ridge	53.4 (0.59)	100
	Elastic Net	23.8 (0.69)	65
	CLERE	26.8 (0.69)	20

Table 1: Averaged MSE for simulated data under the three scenarios. The average number of non-zero parameters estimated for each method was also reported. Where not specified, the number of groups g is chosen using BIC criterion.

Although scenarios 1 and 2 are directly derived from model (3), they differ in terms of cluster separation. In accordance with standard clustering approaches, predicting performances of CLERE increases with cluster separation. This is illustrated by the dramatic improvement of CLERE from scenario 1 to scenario 2. In scenario 2 indeed the regression coefficients were seperated enough. CLERE therefore outperformed all other methods. Moreover, even if the cluster separation was small in scenario 1, the BIC criterion suprisingly led in average to choose a larger

number of groups. However, CLERE had better performances when the number of groups was tuned for each simulated data set. In Scenario 3, the regression coefficients were not separated at all. However, CLERE managed to yield competitive performances.

4.2 Real data

4.2.1 Description

The real data used in this section was published in [7]. This data set consists of $n = 60$ mice for which the expression of 83 gene transcripts from liver tissues was measured and $p = 145$ microsatellite markers were genotyped. One challenging issue of Genetics consists in connecting gene expression levels with variations in the genomic sequence. Microsatellite markers are such variations. The latter markers are discrete quantitative variables taking values in $\{1, 2, 3\}$, while gene expression levels are real quantitative variables. Instead of considering each transcript as a response, we performed a principal component analysis over the gene expression data to come up with a reduced number of outcomes. The first nine principal components (PC) accounted for more than 97% of the total inertia. We then proposed a linear regression model for each of those selected PCs using the microsatellites markers as covariates. The selected PCs are subsequently denoted $PC1, \ldots, PC9$. Since no proper validation data sets were available, all methods were compared in terms of out-of-sample prediction error estimated via 5-fold cross-validation (CV).

4.2.2 Overall results

Table 2 summarizes the MSE for each selected PC and each method. Similarly to numerical experiments on simulated data, variables selection using Lars algorithm yielded, for each PC, very large prediction error. All other methods had however comparable prediction error. For 5 PCs, CLERE showed the best performances in terms of prediction error. The LASSO tended to select very simple models even compared to CLERE. For $PC8$ for instance, the LASSO yielded a prediction error slightly larger than CLERE with only 2 non-zero parameters. Nevertheless, CLERE selected simpler models in average. When averaging the MSE over all PCs, CLERE showed very competitive performances since it was the second best method right after ridge regression.

4.2.3 Focus on $PC1$

We have illustrated above that CLERE is a very competitive method for prediction. In this subsection we now present how CLERE can be used for interpretation purpose. A focus is therefore laid on $PC1$ as a single response variable. The data were no longer partitioned as previously did for cross-validation.

Using the whole data set, 3 groups were chosen using the BIC criterion (see Figure 1).

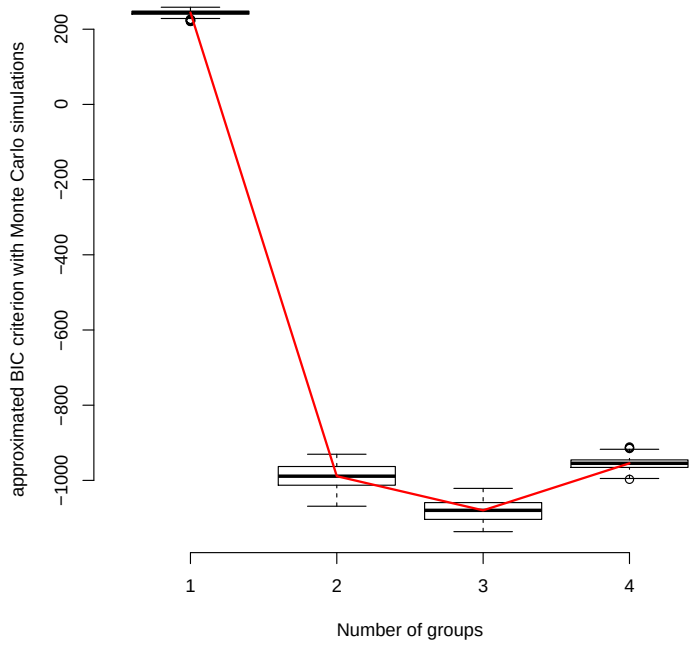

Figure 1: Selection procedure for the number of groups. Here $g = 3$ was selected as it minimizes the BIC. The BIC is approximated using Monte Carlo simulations.

The estimated parameters are given in Table 3. Two groups with moderated positive effects and one group with strong negative effect were identified. In Section 3.2, we presented how to make predictions with CLERE using the vector $\mathbb{E}\left[\beta|\mathbf{y}, \mathbf{X}; \widehat{\theta}\right]$. The latter vector of expectations can be interpreted as a vector of regression coefficients. Consequently, the small estimated value for parameter γ^2 ($\widehat{\gamma^2} = 3.0 \times 10^{-6}$) leads those expectations to be strongly concentrated around the $\widehat{b}_k$'s. CLERE yielded thus a very parsimonious regression model.

The second group, associated with $\widehat{b}_2 = -0.931$, was of interest since it gathers the 11 variables showing the strongest impact on the response. In Table 4, we compared for those variables the regression coefficients obtained with Lars, the LASSO, the ridge regression and the elastic net. The five methods yielded sign and size consistent regression coefficients for almost all the markers highlighted in Table 4. One exception was however noticed for D13Mit16. On the other hand CLERE showed that some variables dropped by the other methods may still be of interest. Overall this analysis emphasized the ability of CLERE to consistently identify influencial covariates using a very parsimonious model. Moreover, this analysis identified the clusters of markers that may be relevantly investigated for a biological characterization.

5 Discussion

We proposed in this paper a new method for simultaneous variables clustering and regression. This work comes in the aftermath of a series of recently published approaches aiming at reducing the dimension in linear regression models by collapsing the covariates into groups. Contrarily to those previous works, our approach is not based on penalized least squares problem. However we assumed the existence of a latent structure within the variables that depends only on their unobserved regression coefficients. In such framework, no distributional assumption regarding the covariates is necessary for achieving the clustering. The latent structure is modeled using a Gaussian mixture model whose parameters are estimated via an EM algorithm. A stochastic version, namely the MCEM, of the latter algorithm was proposed since the E-step was untractable. Even if MCEM has become a standard in many applications, it is noteworthy that its computational cost is not neglectable. Indeed, running the estimation with 3 groups on the data set presented in Section 4.2.3 took 30 seconds for CLERE but less than 1 second for the other approaches. Although CLERE seemed to be relatively slow, the estimation time remained however manageable. Improvements in speeding up the estimation through parallel computating consist in a natural perspective for this work, especially since we are aiming at tackling ultra-high dimensional regression problems in forthcoming researches. We proposed in this paper the BIC criterion for choosing the number of latent groups. This criterion was prefered over different existing criteria such as the out-of-sample prediction error because of its small computational cost. Other information-based criteria will be explored in further works.
Our approach showed good predictive performances both on simulated and real data compared to the LASSO, the ridge regression, the elastic net and Lars. These good performances were accompanied by a lower complexity in terms of number of fitted parameters. CLERE also brought improvements in terms of interpretability since each fit provides a clustering of the covariates. Variables selection may be considered as a special case of clustering, unuseful covariates being clustered together. As a consequence, if a constraint is imposed on the parameter space, then CLERE can also be used as a variable selection tool. Such constraint may for instance lead to assume one group k to have its mean b_k and its associated variance equal to zero. This is a new model which however may be easily derived from the approach presented here. Many applications deal with response variable that may not be continous. Another promising extension of our model is therefore towards generalized linear models. This extension may be achieved straightforwardly.

References

[1] Efron B., Hastie T., Johnstone I., and Tibshirani R. Least angle regression. *Annals of Statistics*, 32:407–499, 2004.

[2] Stein C. Estimation of the mean of a multivariate normal distribution. *Ann. Statist.*, 9:1135–1151, 1981.

[3] Bondell H. D. and Reich B. J. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, 64:115–123, 2008.

[4] Hoerl A. E. and Kennard W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

[5] Schwarz G. Estimating the dimension of a model. *Ann. Statist.*, 6:461–464, 1978.

[6] Wei C. G. and Tanner M. A. A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 85:699–704, 1990.

[7] Chun H. and Keles S. Expression quantitative trait loci mapping with multivariate sparse partial least squares regression. *Genetics*, 182:79–90, 2009.

[8] Zou H. and Hastie T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.

[9] Daye Z. J. and Jeng X. J. Shrinkage and model selection with correlated variables via weighted fusion. *Computational Statistics & Data Analysis*, 53(4):1284–1298, February 2009.

[10] Dempster A. P., Laird M. N., and Rubin D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 39:1–22, 1977.

[11] Tibshirani R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[12] Caffo B. S., Jank W., and Jones G. L. Ascent-based Monte Carlo Expectation-Maximization. *Journal of the Royal Statistical Society Series B*, 67(2):235–251, 2005.

[13] Petry S. and Tutz G. Shrinkage and variable selection by polytopes. *Technical report No. 053, Department of Statistics, University of Munich*, 2009.

[14] Shen X. and Huang H. Grouping pursuit in regression. *Journal of American Statistical Association*, 105:727–739, 2010.

[15] Park M. Y., Hastie T., and Tibshirani R. Averaged gene expressions for regression. *Biostatistics*, pages 212–227, 2007.

[16] She Y. and Stanford University. *Sparse Regression with Exact Clustering*. Stanford University, 2008.

PC		averaged 5-fold CV statistic (Std. Err)	Averaged number of parameters
1	Lars	112 (30.9)	47
	LASSO	1.13 (0.05)	3
	Ridge	1.11 (0.04)	145
	Elastic Net	1.31 (0.09)	15
	CLERE	1.07 (0.09)	5
2	Lars	18.8 (5.84)	47
	LASSO	1.32 (0.22)	15
	Ridge	0.92 (0.04)	145
	Elastic Net	1.17 (0.20)	33
	CLERE	1.06 (0.12)	10
3	Lars	19.43 (5.61)	47
	LASSO	0.99 (0.09)	12
	Ridge	0.98 (0.06)	145
	Elastic Net	1.08 (0.10)	26
	CLERE	1.15 (0.18)	9
4	Lars	48.8 (9.20)	47
	LASSO	1.10 (0.02)	3
	Ridge	1.05 (0.01)	145
	Elastic Net	1.13 (0.04)	8
	CLERE	1.03 (0.01)	4
5	Lars	29.4 (11.6)	47
	LASSO	1.22 (0.08)	5
	Ridge	1.13 (0.02)	145
	Elastic Net	1.31 (0.08)	9
	CLERE	1.09 (0.02)	5
6	Lars	28.7 (9.80)	47
	LASSO	1.17 (0.08)	13
	Ridge	0.94 (0.04)	145
	Elastic Net	1.10 (0.08)	30
	CLERE	1.27 (0.15)	7
7	Lars	28.5 (11.3)	47
	LASSO	1.23 (0.08)	17
	Ridge	0.94 (0.04)	145
	Elastic Net	1.21 (0.09)	37
	CLERE	1.28 (0.1)	10
8	Lars	26.4 (4.00)	47
	LASSO	1.06 (0.02)	2
	Ridge	1.08 (0.03)	145
	Elastic Net	1.11 (0.02)	8
	CLERE	0.99 (0.05)	5
9	Lars	33.4 (12.7)	47
	LASSO	1.12 (0.06)	6
	Ridge	1.07 (0.04)	145
	Elastic Net	1.14 (0.07)	14
	CLERE	1.01 (0.03)	5

Table 2: Out-of-sample prediction error estimated using 5-fold CV for each method and each PC for mice data from [7]. The averaged number of fitted parameters, as a measure of model complexity, is also reported.

$\widehat{\beta_0}$	$\widehat{b_1}$	$\widehat{b_2}$	$\widehat{b_3}$	$\widehat{\pi}_1$	$\widehat{\pi}_2$	$\widehat{\pi}_3$	$\widehat{\gamma^2}$	$\widehat{\sigma^2}$
2.32×10^{-2}	7.87×10^{-2}	-9.32×10^{-1}	7.63×10^{-2}	0.870	0.076	0.054	3.0×10^{-6}	7.35

Table 3: Maximum likelihood estimate obtained for CLERE when fitting mice data using $PC1$ as response variable.

Markers	Chromosome	Lars	LASSO	Ridge	Elastic net	CLERE
D1Mit87	1	.	.	-0.0265	.	-0.9318
D3Mit19	3	-0.2347	-0.8962	-0.1940	-0.5670	-0.9316
D4Mit149	4	.	.	-0.0855	.	-0.9316
D4Mit237	4	-2.7478	-0.8661	-0.1714	-0.4767	-0.9318
D7Mit56	7	-0.2011	-0.0484	-0.1026	-0.1516	-0.9318
D7Mit76	7	.	-0.0116	-0.1026	-0.1514	-0.9317
D8Mit42	8	0.0119	.	-0.0430	.	-0.9319
D9Mit15	9	-3.1530	-1.6102	-0.2826	-1.0474	-0.9318
D13Mit16	13	1.2867	.	0.0530	0.0823	-0.9318
D15Mit174	15	-1.7012	-0.9335	-0.1149	-0.4312	-0.9319
D19Mit34	19	.	.	-0.0449	-0.0303	-0.9317

Table 4: Microsatellite markers having the strongest impact on $PC1$. Regression coefficients for those variables are reported for all compared methods. For CLERE regression coefficients are obtained using $\mathbb{E}\left[\beta|\mathbf{y}, \mathbf{X}; \widehat{\theta}\right]$. "." means 0.