Regression for correlated variables : application in steel industry

Clément THERY

January 27, 2014

- resume

- keywords : linear regression, correlations, sem, selection, graphs, pretreatment, plugin

1 Introduction

When one wants to explain a phenomenon based on some covariates, the first statistical method tried frequently is the linear regression. It provides a predictive model with a good interpretability and is simple to learn for non-statistician. Therefore, linear regression is used in nearly all the fields where statistics are made, from industry (ballistic models to calibrate the process) to sociology (predicting some numerical properties of a population). Linear regression is a very classic situation that faces an also classical problem : the variance of the estimators. This variance increases based on two aspects :

- The dimension p (number of covariates) of the model : the more covariates you have the greater variance you get.
- The correlations within the covariates : strongly correlated covariates give bad-conditioning and increase variance of the estimators .

With the rise of informatic, datasets contains more and more covariates and thus more and more useless covariates. So dimension reduction becomes a necessity. Moreover, when you use more covariates, you increase the chance to have correlated ones. For example, this work takes place in an industrial context with a big set of covariates (many parameters of the whole process without any a priori) highly correlated (physical laws, process rules, etc). In such a context, variance of the estimators can lead to arbitrary results or even no results at all. Prediction and interpretation are both strongly needed, with a preference for interpretation in industrial context (better to improve the process when possible than to only predict defects).

When estimating the parameters of the regression we have to compute the inverse of a matrix[8] which will be ill-conditioned or even singular if some covariates depend linearly from each other. For a model defined by

$$Y = X\beta + \varepsilon \tag{1}$$

where X is the $n \times p$ matrix of the explicative variables, Y the response vector and $\varepsilon \sim \mathcal{N}(0, \sigma_Y^2)$ we have the following Ordinary Least Squares (OLS) estimators :

$$\hat{\beta} = \left(X'X\right)^{-1} X'Y \tag{2}$$

Because it is the minimum-variance unbiased estimator, penalized methods try to reduce the variance introducing some bias to improve the bias-variance trade-off and get better prediction. Ridge regression[7] proposes a biased estimator :

$$\hat{\beta} = (X'X + kI)^{-1}X'Y \text{ with } k \geq 0 \tag{3}$$

But Ridge regression is not efficient to select covariates (it's an assumed choice) because coefficients tends to 0 but don't reach 0. So it gives difficult interpretations and is not adapted for our industrial context. We need to reduce the dimension of the model. Our goal is not just to predict but also to understand the response variable.

Real datasets implies many irrelevant variables (datasets based on the whole process without any a priori) so we have to use variable selection methods.

We note classical norms: $\parallel \beta \parallel_2^2 = \sum_{i=1}^p (\beta_i)^2$ and $\parallel \beta \parallel_1 = \sum_{i=1}^p |\beta_i|$

The Least Absolute Shrinkage and Selection Operator (LASSO)[9] consists in a shrinkage of the regression coefficients based on a λ parametric L_1 penalty.

$$\hat{\beta} = \text{argmin}\left\{\parallel Y - X\beta \parallel_2^2\right\} \text{ subject to } \parallel \beta \parallel_1 \leq \lambda \tag{4}$$

The Least Angle Regression[3] (LAR) Algorithm offers a very efficient way to obtain the whole LASSO path and is very attractive. It requires only the same order of magnitude of computational effort as OLS applied to the full set of covariates. And it really selects covariates with coeffients set exactly to 0. But LASSO also faces consistency problems[13] when confronted with correlated covariates. This point will be developped further (see 4.5) with numerical results. Another limitation of the LASSO is that it preserves at most n predictors (troublesome when in high dimension).

Elastic net[14] is a method developped to be a compromise between Ridge regression and the LASSO. Elastic net can be written:

$$\hat{\beta} = (1 + \lambda_2)\,\text{argmin}\left\{\parallel Y - X\beta \parallel_2^2\right\}, \text{ subject to } (1-\alpha) \parallel \beta \parallel_1 + \alpha \parallel \beta \parallel_2^2 \leq t \text{ for some } t \tag{5}$$

where $\alpha = \frac{\lambda_2}{(\lambda_1 + \lambda_2)}$ It seems to give good predictions. But it is based on the grouping effect and if the dataset contains two identical variables they will obtain the same coefficient whereas LASSO will choose between one of them and will then obtain same predictions with a more parsimonious model.

The CLusterwise Effect REgression[11] (CLERE) tries to reduce the dimension by considering the β_j no longer as fixed effect parameters but as unobserved independant random variables whith β following a Gaussian Mixture distribution.

$$\begin{aligned} \beta_j|\mathbf{z}_j &\sim \mathcal{N}(\sum_{k=1}^{g} b_k z_{jk}, \gamma^2) \text{ with} && (6)\\ \mathbf{z}_j &= (z_{j1}, \ldots, z_{jg}) \sim \mathcal{M}(\pi_1, \ldots, \pi_g) \text{ and} && (7)\\ \forall k = 1, \ldots, g \qquad & \sum_{j=1}^{p} z_{jk} \geq 1 && (8) \end{aligned}$$

The idea is to hope that this mixture will have few enough components to have a number of parameters to estimate significantly lower than p. In such a case, it improves interpretability and ability to yeld reliable prediction with a smaller variance on $\hat{\beta}$. But we need to suppose having many covariates with the same level of effect on the response variable and seems to be less efficient in prediction than elastic net.

When some try to reduce the dimension and then just hope to have small correlations in the remaining dimensions, we propose to focus on the correlations, giving a model with orthogonal covariates. In fact we search the greatest set of orthogonal covariates to keep the maximum but with an orthogonality constraint. This can be viewed as a pretreatment on the dataset allowing to use then dimension reduction tools without suffering from correlations. We only consider strong correlations (i.e. : problematic ones) thus we keep most of the information contained in the dataset. We will in a second time be able to use the remaining part of the information (sequential approach).

Our work is based on the assumption that if we know that correlations are a problem and if we precisely know the correlations, we could use this knowledge to avoid the problem. The idea is to suppose explicitly a linear structure between the covariates. It gives a recursive Structural Equation Model (SEM)[2]. That can be viewed as a system of linear regression.

$$Y = X\beta_Y + \varepsilon_Y \tag{9}$$

$$X = XB + \varepsilon_X \tag{10}$$

Recursive sem don't really have a specific estimator because general system estimators (Seemingly Unrelated Regression (SUR)[12] and Two-Stage Least Squares (2SLS)) are equivalent to independent Ordinary Least Squares when applied to recursive SEM [10]. Our work can be viewed as a new way of estimating recursive SEM based on their own structure. In this work, we decide to distinguish the response variable from the other variables that are on the left of a regression. Thus we don't have a system of regressions but one regression on our response variable and a system of subregressions (without the response variable). The structure is supposed to be the source of the correlations and allows us to define a reduced set of independent covariates. Thus we reduce dimension and correlations in the same time. The structure justifies the eviction of the redundant covariates without significant information loss. It can be seen as a pretreatment on the dataset based on the hypothesis of a strong structure between the covariates (i.e. : small ε_X).

We can use any variable selection method on the reduced dataset with improved efficiency (reduced variance) due to dimension reduction and correlation suppression. So we obtain two kinds of zeros in our first model : coerced zeros due to correlations (redundant information) and estimated ones with classical variable selection methods applied on remaining variables. This two kinds of zero won't be interpreted in the same way and thus consistency issues don't mean interpretation issues any more. So we dodge the drawbacks of both grouping effect and variable selection.

But to work, we need an explicit structure between the covariates. SEM are often used in social sciences and economy where a structure is supposed "by hand" but here we want to be able to find a structure without any a priori (possibility to include some known structure remains). Graphical LASSO [4] offers a method to obtain a structure based on the precision matrix (inverse of the variance-covariance matrix). It consists in a selection in the precision matrix, setting some covariances to zero. But the resulting matrix is symmetric and we need an oriented graph for our SEM. So we developped an MCMC algorithm to find it (R package CorReg on Rforge). However, Graphical LASSO can be used in the initialization step of our MCMC. This structure is based on gaussian mixture models to fit better real datasets and to allow identifiability of the structure in terms of complexity (number of parameters).

This paper will first present the reduced model, its properties and the algorithm used to find the structure. Then we talk about further usage of the method, both estimating residuals of the reduced model and managing missing values in the dataset. We will finish with some numerical results on simulated and real industrial datasets before concluding and giving some perspectives.

2 Model to decorrelate the covariates

We have a p correlated covariates X to explain a response variabe Y. Let Z be the adjacency matrix that defines which covariates is depending on which others. That is $Z_{i,j} = \mathbf{1}_{(X^j depends\ onX^i)}$. We consider dependencies in a generative point of view ("depends on" means "is generated according to") so Z is not symmetric and has no cycles.

We can describe the structure Z by $S = (p_2, I_2, p_1, I_1)$ defined by :

$$p_2 = \sum_{j=1}^{p} \mathbf{1}_{(\exists i, Z_{i,j} \neq 0)} \text{ the number of sub-regressions} \tag{11}$$

$$I_2 = (I_2^1, \dots, I_2^{p_2}) \text{ vector of the indices of the dependent covariates} \tag{12}$$

$$I_1 = (I_1^1, \dots, I_1^{p_2}) \text{ with} \tag{13}$$

$$I_1^j = \{i | Z_{i,j} = 1\} \text{ indices of the covariates explaining } X^j \tag{14}$$

$$p_1 = (p_1^1, \dots, p_1^{p_2}) \text{ where } p_1^j = \sharp I_1^j \tag{15}$$

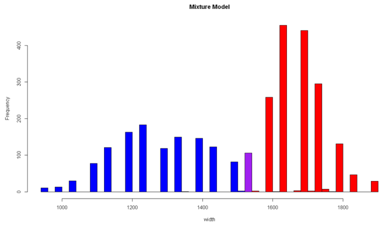


Figure 1: Industrial variables seem to follow gaussian mixture models

We suppose $I_1 \cap I_2 = \emptyset$, *i.e.* dependent variables don't explain other variables in X.

We note $I_2^c = \{1, \dots, p\} \setminus I_2$ Then our generative model can be written :

$$\begin{aligned}
Y_{|X,S} = Y_{|X} &= XA + \varepsilon_Y = X^{I_2^c} A_{I_2^c} + X^{I_2} A_{I_2} + \varepsilon_Y \text{ with } \varepsilon_Y \sim \mathcal{N}(0, \sigma_Y^2) && (16)\\
\forall j \in I_2 : \ X^j_{|X^{I_1^j}, S} &= X^{I_1^j} B^j_{I_1^j} + \varepsilon_j \text{ with } \varepsilon_j \sim \mathcal{N}(0, \sigma_j^2) && (17)\\
\forall j \notin I_2 : \ X^j &= f(\theta_j) \text{ free law} && (18)
\end{aligned}$$

Where $B^j_{I_1^j}$ is the p_1^j-sized vector of the coefficients of the subregression.

We note that (16) and (17) also give :

$$\begin{aligned}
Y &= X^{I_2^c}(A_{I_2^c} + \sum_{j \in I_2} B^j_{I_1} A_j) + \sum_{j \in I_2} \varepsilon_j A_j + \varepsilon_Y && (19)\\
&= X^{I_2^c} \tilde{A}_{I_2^c} + \tilde{\varepsilon} = X\tilde{A} + \tilde{\varepsilon} && (20)\\
\text{where } \tilde{A}_{I_2} &= 0 && (21)\\
\tilde{A}_{I_2^c} &= A_{I_2^c} + \sum_{j \in I_2} B^j_{I_1} A_j && (22)\\
\tilde{\varepsilon} &= \sum_{j \in I_2} \varepsilon_j A_j + \varepsilon_Y && (23)
\end{aligned}$$

2.1 Structure identifiability

$$\mathbf{P}(X) = \mathbf{P}(X^{I_2^c}, X^{I_2}) = \mathbf{P}(X^{I_2} | X^{I_2^c}) \mathbf{P}(X^{I_2^c}) \tag{24}$$

When estimating the quality of the model we look at the likelihood (to compute the Bayesian Information Criterion (BIC)):

$$\mathcal{L}_B(X) = \prod_{\substack{1 \leq i \leq n \\ j \in I_2}} \mathbf{P}(X_{i,j} | X^{I_1^j}) \prod_{\substack{1 \leq i \leq n \\ j \in I_1^j}} \mathbf{P}(X_{i,j}) \tag{25}$$

We suppose $\forall j \in I_2^c : X^j$ follows gaussian <u>mixture</u> models to better fit industrial variables (Figure 1) estimated separately (we use the Rmixmod package for R). In the following, $\otimes$ and $\oplus$ denote respectively the kronecker product and sum.

$$\begin{aligned}
\forall j \in I_2 : \ X^j_{|X^{I_1^j}} &\sim \mathcal{N}(X^{I_1^j} B^j_{I_1^j}, \sigma_j^2) && (26)\\
\forall j \in I_2^c : \ X^j &\sim \mathcal{GM}(\pi_j; \mu_j; \sigma_j^2) \text{ with } \pi_j, \mu_j, \sigma_j^2 \text{ vectors of size } k_j && (27)\\
\text{And we obtain, } \forall j \in I_2 : \ X^j &\sim \mathcal{GM}(\bigotimes_{\substack{i \in I_1^j \\ Z_{i,j}=1}} \pi_i \ ; \bigoplus_{\substack{i \in I_1^j \\ Z_{i,j}=1}} B_{i,j}\mu_i \ ; \sigma_j^2 + \bigoplus_{\substack{i \in I_1^j \\ Z_{i,j}=1}} B_{i,j}^2 \sigma_i^2) && (28)
\end{aligned}$$

So when we compare (27) and (28) we see that the number of classes in I_2 variables differs when subregressions are of length > 1 (almost 2 predictors) with multiple-class predictors (so the kronecker product is effective). We call this difference in component number "identifiability" in the sense that we try to find the model with the fewest component in X^{I_1} (we use the BIC so when comparing two models with the same likelihood, the one with lesser parameters will win). So in this case we have a "better" model (the simplest one) in the group of equivalent models (permuting variables in each subregression).

Remark : the uncrossing constraint ($I_2 \cap I_1 = \emptyset$) significatively reduces the number of feasible models and thus increases identifiability (it also reduces the number of models equivalent in interpretation).

And if we try to permute one subregression we obtain for $k \in I_2$ and $j \in I_1^k$:

$$X^k - \varepsilon_X^k - \sum_{\substack{i \in I_1^k \\ i \neq j}} X^i B_{i,k} \sim \mathcal{GM}\left(\tilde{\pi}_j \, ; \tilde{\mu}_j \, ; \tilde{\sigma}_j^2\right) \text{ where} \tag{29}$$

$$\tilde{\pi}_j = (\bigotimes_{i \in I_1} \pi_i) \otimes \bigotimes_{\substack{i \in I_1 \\ i \neq j}} \pi_i \tag{30}$$

$$\tilde{\mu}_j = \frac{(\bigoplus_{i \in I_1} B_{i,k}\mu_i) \oplus \bigoplus_{\substack{i \in I_1 \\ i \neq j}} (-B_{i,k}\mu_i)}{B_{j,k}} \tag{31}$$

$$\tilde{\sigma}_j^2 = \frac{2d_k^2 + (\bigoplus_{i \in I_1} B_{i,k}^2 \sigma_i^2) \oplus \bigoplus_{\substack{i \in I_1 \\ i \neq j}} B_{i,k}^2 \sigma_i^2}{B_{j,k}^2} \tag{32}$$

Thus if we call identifiability the ability to find the model with the smallest variance on the subregression residuals, we have identifiability on the structure if $X^k - \varepsilon_X^k - \sum_{\substack{i \in I_1 \\ i \neq j}} X^i B_{i,k} \nsim X^j$. That is the case if $d_k^2 > 0$ (non-exact subregression) because the variance of the noise of the subregression is doubled when the subregression is permuted. So even with gaussian variables, non-exact subregression are a sufficient condition for identifiability (in the meaning of smallest-variance subregressions). But you need to look the marginal laws because the joint one relies on the structure and does not use marginal laws of the left-sided covariates.

If there are exact subregressions, classical methods will fail (singular matrix) and the structure won't be identifiable. But it only means that several structure will have the same likelihood and they will have the same interpretation. So it's not a problem. Moreover, when an exact subregression is found, we can delete one of the implied variables without any loss of information and the structure will define a list of variable from which to delete. CorReg (Our R package) prints a warning to point out exact subregressions when found.

2.2 Estimator and properties

Classical methods like Ordinary Least Squares (OLS) estimate $Y|X$ and obtain (Maximum Likelihood Estimation):

$$\hat{A} = (X'X)^{-1}X'Y \text{ (ill-conditoned matrix to inverse)} \tag{33}$$

With following properties :

$$E[\hat{A}|X] = A \tag{34}$$

$$Var[\hat{A}|X] = \sigma_Y^2 (X'X)^{-1} \tag{35}$$

And when correlations are strong, the matrix to invert is ill-conditioned and the variance explodes.

Our idea is to reduce the variance so we explain Y only with X^{I_1} knowing (17) and (20)

$$Y = X^{I_2^c} \tilde{A}_{I_2^c} + \tilde{\varepsilon} \tag{36}$$

So the new estimator simply is :

$$\hat{\tilde{A}}_{I_2^c} = (X'_{I_2^c} X^{I_2^c})^{-1} X'_{I_2^c} Y \tag{37}$$

$$\hat{\tilde{A}}_{I_2} = 0 \tag{38}$$

and we get the following properties :

$$\begin{aligned} E[\hat{\hat{A}}|X] &= \tilde{A} && (39)\\ Var[\hat{\hat{A}}_{I_2^c}|X] &= (\sigma_Y^2 + \sum_{j\in I_2} \sigma_j^2 A_j^2)(X'_{I_2^c}X^{I_2^c})^{-1} && (40)\\ Var[\hat{\hat{A}}_{I_2}|X] &= 0 && (41) \end{aligned}$$

We see that the variance is reduced (no correlations and smaller matrix give better conditioning) for small values of σ_j *i.e.* strong correlations.

Both classical and our new estimators of Y are unbiased [8] (true model).

There is no theoretical guarantee that our model is better. It's just a compromise between numerical issues caused by correlations for estimation and selection versus increased variability due to structural hypothesis. Therefore we made some simulations to compare both methods (see the end of this paper). This new model is reduced even without variable selection and is just a linear regression so every method for variable selection in linear regression can be used. Hence we hope to obtain a parsimonious model.

The explicit structure between the covariates helps to understand the model and the complex link between the covariate and the response variable so we call this model explicative.

When we use a variable selection method on it we obtain two kinds of 0 :

1. Because of the structure we coerce $\hat{\hat{A}}^{I_2} = 0$. This kind of zero means redundant information but the covariate can be correlated with the response variable. So we don't have the grouping effect (so we are more parsimonious) and we don't suffer from false interpretation (LASSO would).

2. Variable selection methods can lead to get some exact zeros in $\hat{\hat{A}}^{I_1}$. This kind of zero means that implied covariate has no significant effect on the response variable. And because variables in X^{I_1} are orthogonal, we know that it is not misleading interpretation due to correlations.

2.3 Why grouping effect is misleading

In industrial context, when a model explain why things go wrong, one will try to fix the problem. If $X_1 = X_2 + e$ and we have the grouping effect, we will obtain a model like $Y = aX_1 + aX_2$. Then when one will try to modify Y he will modify one of the covariates and both will change so he won't get expected results. Nothing constrains us to give only one equation. It is clearly better to give the user another equation (or system for more complex models) describing the correlations. So you get the following model : $Y = aX_1 + aX_2$ AND $X_1 = X_2 + e$. So you have more information and are able to decide better actions. With such a model, grouping effect is no more useful because when saying $Y = 2aX_2$ AND $X_1 = X_2 + e$ you clearly show that X_2 is correlated with both Y and X_1. So it is possible to combine the advantages of grouping effect and selection just giving several equations. Each equation here is very simple so you don't really increase complexity of the model. Uncrossed model (nilpotent Z) guarantee to keep a simple structure easily interpretable.

3 Estimating subregressions

All our work is based on a linear structure between the covariates. Let's define $\mathcal{S}$ and $\mathcal{Z}$ the ensemble of feasible structures and the ensemble of corresponding adjacency matrices.

$Z \in \mathcal{Z} \Leftrightarrow$:

- Z is binary
- $ZZ = 0$ (Z is not crossed). Equivalent to $I_1 \cap I_2 = \emptyset$.

So $\mathcal{Z}$ is just the set of the binary square nilpotent matrices of size p. Z is an adjacency matrix and we know [1] that Z^p shows the number of paths of length p (linking $p+1$ vertices). So we suppose that Z is nilpotent, meaning it does not contain any non-trivial path. This strong hypothesis also strongly reduces the size of $\mathcal{Z}$.

Now we have made hypothesis on the distribution, we can use them to compare the structures with the Bayesian Information criterion (BIC) [6]. But BIC tends to give too complex structures because we test a great range of models. Thus we choose to penalise the complexity a bit more with specific a priori laws(uniform laws for the number of subregression and the complexity of each subregression instead of uniform law on S) :

$$\begin{aligned}
P(S) &= P(I_1|p_1, I_2, p_2)P(p_1|I_2, p_2)P(I_2|p_2)P(p_2) && (42)\\
P(I_1|p_1, I_2, p_2) &= \prod_{j=1}^{p_2} P(I_1^j|p_1^j, I_2, p_2) && (43)\\
P(I_1^j|p_1^j, I_2, p_2) &= \begin{pmatrix} p-p_2 \\ p_1^j \end{pmatrix}^{-1} = \frac{p_1^j!(p-p_2-p_1^j)!}{(p-p_2)!} && (44)\\
P(p_1|I_2, p_2) &= \prod_{j=1}^{p_2} P(p_1^j|I_2, p_2) && (45)\\
P(p_1^j|I_2, p_2) &= \frac{1}{p-p_2} && (46)\\
P(I_2|p_2) &= \begin{pmatrix} p \\ p_2 \end{pmatrix}^{-1} = \frac{p_2!(p-p_2)!}{p!} && (47)\\
P(p_2) &= \frac{1}{p_2} && (48)\\
P(S) &= \left(\prod_{j=1}^{p_2} \begin{pmatrix} p-p_2 \\ p_1^j \end{pmatrix}^{-1}\right) \left(\frac{1}{p-p_2}\right)^{p_2} \frac{p_2!(p-p_2)!}{p!}\frac{1}{p_2} && (49)\\
\ln P(S) &= -\sum_{j=1}^{p_2} \ln \begin{pmatrix} p-p_2 \\ p_1^j \end{pmatrix} - p_2\ln(p-p_2) - \ln \begin{pmatrix} p \\ p_2 \end{pmatrix} - \ln(p_2) && (50)
\end{aligned}$$

Then we have

$$\begin{aligned}
P(S|X) &\propto P(X|S)P(S) && (51)\\
\ln(P(S|X)) &= \ln(P(X|S)) + \ln(P(S)) + cste && (52)\\
BIC^* &= BIC + \ln(P(S)) && (53)
\end{aligned}$$

It increases penalty on complexity for $p_2 < \frac{p}{2}$ thus in the following we will use $BIC*$ under this hypothesis (that becomes a constraint in the MCMC).

$$BIC(X|S) = \sum_{j=1}^{p} BIC(X^j|S) \qquad (54)$$

Where

$$BIC(X^j|S) = -2\mathcal{L}_{|S}(X^j, \theta_j) + K_j \log(n) \qquad (55)$$

Where K_j is the number of parameters to estimate. We will now use the following notation : $BIC(S) = BIC(X|S)$

If we have some hypothesis on the distribution of some variables (exponentially distributed for example) we can compute corresponding BIC separately and then improve the efficiency of the algorithm (it will find a structure only if it is really relevant).

3.1 The Markov chain

3.1.1 Formalism with Z

For each step, starting from $Z \in \mathcal{Z}$ we define a neighbourhood $\mathcal{V}_{Z,j}$ with $j \sim \mathcal{U}(\{1,\dots,p\})$ like this :

$$\begin{aligned}\mathcal{V}_{Z,j} = \{\tilde{Z} \in \mathcal{Z} \quad | \quad & \exists! i, \tilde{Z}_{i,j} = 1 - Z_{i,j}, \text{ and } \forall (k,l) \neq (i,j) : \\ & \tilde{Z}_{j,l} = (1-\tilde{Z}_{i,j})Z_{j,l} \text{ (row-wise relaxation),} \\ & \tilde{Z}_{k,i} = (1-\tilde{Z}_{i,j})Z_{k,i} \text{ (column-wise relaxation)} \\ & \tilde{Z}_{k,l} = Z_{k,l} \\ & \} \cup \{Z\}\end{aligned}$$

We have $|\mathcal{V}_{Z,j}| = p$ but some other constraints can be added on the definition of $\mathcal{Z}$ and will consequently modify the size of the neighbourhood (for example a maximum complexity for the subregressions or the whole structure, a maximum number of sub-regressions, etc). CorReg allows to modify this neighbourhood to better fit users constraints.

3.1.2 Formalism with S

First we remember that S is completely described with I_1 : So we will only describe the variations in I_1 at each step and other parts of S will follow according to the previous definition. for each step, starting from $S \in \mathcal{S}$ we define a neighbourhood $\mathcal{V}_{S,j}$ with $j \sim \mathcal{U}(\{1,\dots,p\})$ like this :

$$\mathcal{V}_{S,j} \quad = \quad \{S^{(i,j)} | 1 \leq i \leq p\} \cup \{S\} \tag{56}$$

With $S^{(i,j)}$ defined by the following algorithm :

- if $i \notin I_i^j$ (add):
  - $I_1^j = I_1^j \cup \{i\}$
  - $I_1^i = \emptyset$ (explicative variables can't depend on others : column-wise relaxation)
  - $I_1 = I_1 \setminus \{j\}$ (dependent variables can't explain others : row-wise relaxation)
- if $i \in I_1^j$ (remove): $I_1^j = I_1^j \setminus \{i\}$

And then :

$$\begin{aligned} I_2 &= \{j | \sharp I_1^j > 0\} && (57) \\ p_2 &= \sharp I_2 && (58) \\ \forall 1 \leq j \leq p : p_1^j &= \sharp I_1^j && (59) \end{aligned}$$

We have $|\mathcal{V}_{S,j}| = p$ but some other constraints can be added on the definition of $\mathcal{S}$ and will consequently modify the size of the neighbourhood (for example a maximum complexity for the subregressions or the whole structure, a maximum number of sub-regressions, etc). CorReg allows to modify this neighbourhood to better fit users constraints.

3.1.3 properties

The algorithm follows a time-homogeneous markov chain whose transition matrix $\mathcal{P}$ has $|\mathcal{S}|$ rows and columns (combinatory so we'll just compute the probabilities when we need them). And $\mathcal{S}$ is a finite state space.

We want

$$\mathcal{P}(S,\tilde{S}) = \mathbf{1}_{[\exists j, \tilde{S} \in \mathcal{V}_{S,j}]} P(\tilde{S}|X) \tag{60}$$

Because the walk follows a regular and thus ergodic markov chain with a finite state space, it has exactly one stationary distribution [5] : π and every rows of $\lim_{k\to\infty} \mathcal{P}^k = W$ equals π.

With $\forall S \in \mathcal{S}$:

$$\begin{aligned} 0 \leq \quad \pi(S) & \leq 1 \\ \sum_{S\in\mathcal{S}} \pi(S) &= 1 \\ \pi(S) &= \sum_{\tilde{S}\in\mathcal{S}} \pi(\tilde{S})\mathcal{P}(\tilde{S},S) \qquad (61) \\ & \qquad (62) \end{aligned}$$

We make a first approximation (53) :

$$P(S|X) \approx exp(BIC^*(S)) \qquad (63)$$

We define [6], :

$$q(\tilde{S},\mathcal{V}_{S,j}) = \mathbf{1}_{\{\tilde{S}\in\mathcal{V}_{S,j}\}} \frac{exp(\frac{-1}{2}\Delta BIC(\tilde{S},\mathcal{V}_{S,j}))}{\sum_{S_l\in\mathcal{V}_{S,j}} exp(\frac{-1}{2}\Delta BIC(S_l,\mathcal{V}_{S,j}))} \qquad (64)$$

Where $\Delta BIC(S,\mathcal{V}_{S,j}) = BIC(S) - \min\{BIC(\tilde{S})|\tilde{S} \in \mathcal{V}_{S,j}\}$ is the gap between a structure and the worst structure in the neighbourhood in terms of BIC.

And then we can note $\forall(S,\tilde{S}) \in \mathcal{S}^2$:

$$\mathcal{P}(S,\tilde{S}) = \frac{1}{p}\sum_{j=1}^{p} q(\tilde{S},\mathcal{V}_{S,j})$$

The output will be the best structure seen in terms of BIC. If we have some knowledge about some sub-regressions (physical models for example) we can add them in the found structure. So the model is really expert-friendly.

3.1.4 initialisation

The initial structure can be based on a first warming algorithm taking the correlations into account. Ones are randomly placed into Z, weighted by the absolute value of the correlations. Then this structure can be reduced by the hadamard product with the binary matrix obtained by Graphical Lasso[4].

3.1.5 reduced neighbourhood

If the algorithm did not have time to converge, it can be finished with a few step fr which the neighboorhood would only contain smaller candidates. It is equivalent to ask for each element in the structure if the criterion would be better without it. Thus it can be seen as a final cleaning step. But in fact, it's just contui

3.1.6 multiple try

One would rather test multiple short chains than lose time in initialisation or long chains

4 Numerical results on simulated datasets

Here are some results on simulated datasets.

4.1 Finding the structure

4.1.1 How to evaluate found structure ?

The first criterion is BIC^* which is minimised in the MCMC. We can compare this criterion for both the true Structure and the found one. But if the structures differ, the BIC^* will also differ and comparison won't show how far the found structure is from the true one. So we need more precise criterions to compare the true structure S and the found one $\tilde{S}$. Global indicators :

- True left : the number of found dependent variables that really are dependent $TL = \sharp(I_2 \cap \tilde{I}_2)$
- Wrong left : the number of found dependent variables that are not dependent $WL = \sharp(\tilde{I}_2) - TL$
- Missing left : the number of really dependent variables not found $ML = \sharp I_2 - TL$
- Δp_2 : the gap between the number of sub-regression in both model : $\Delta p_2 = \sharp I_2 - \sharp(\tilde{I}_2)$. The sign defines if $\tilde{S}$ is too complex or too simple
- $\Delta compl$: the difference in complexity between both model : $\Delta compl = \sum_{j \in p_2} p_1^j - \sum_{j \in \tilde{p}_2} \tilde{p}_1^j$

And we have aussi some criterion to compare between the structures :

- Mean true found R^2 (or $sigma^2$) : the mean of R^2 (in the true model) for sub-regressions found by the algorithm and existing in the true model (comparing only the left-side variable). $MTF_{R^2} = \frac{1}{TL} \sum_{j \in I_2 \cap \tilde{I}_2} R^2(j)$.
- Mean Missing R^2 (or $sigma^2$) : the mean of R^2 for sub-regressions only in the true structure (comparing only the left-side variable).$MM_{R^2} = \frac{1}{ML} \sum_{j \in I_2 \setminus (I_2 \cap \tilde{I}_2)} R^2(j)$.
- Mean found R^2 (or $sigma^2$) : the mean of R^2 (in the found model) for sub-regressions found by the algorithm and existing in the true model (comparing only the left-side variable). $MF_{R^2} = \frac{1}{TL} \sum_{j \in I_2 \cap \tilde{I}_2} \tilde{R}^2(j)$.
- Mean Wrong R^2 (or $sigma^2$) : the mean of R^2 for sub-regressions in only in the found structure (comparing only the left-side variable).$MW_{R^2} = \frac{1}{WL} \sum_{j \in \tilde{I}_2 \setminus (I_2 \cap \tilde{I}_2)} \tilde{R}^2(j)$.
- Mean R^2 : the mean of R^2 of all sub-regression for a given structure. We can then compare this value for both structures
- Mean σ^2 : the mean of σ^2 of all sub-regression for a given structure. We can then compare this value for both structures
- Complexity : global complexity of a model : $compl = \sum_{j \in p_2} p_1^j$. We can then compare both global complexities.

4.1.2 Results

Tableau de la forme :

n	time	trueBIC	BICempty	BIC_opt	True1	False1	missing1	Δp2	True_left	False_left
40	??	??	??	??	??	??	??	??	??	??
60	??	??	??	??	??	??	??	??	??	??
100	??	??	??	??	??	??	??	??	??	??

Table 1: p variables. Markov chain was XX seconds long for $n = 100$(mean observed).

Ordre des critères de comparaison : MSE sur X, Vraigauche, fauxgauche, bics (les 3), vrais1, faux1, missing1,deltap2

L'idée serait de n'avoir qu'une seule configuration (vu qu'ici on ne dépend pas de Y) et garder la même pour tous les tableaux suivants (pour pouvoir s'appuyer sur celui-ci dans l'interprétation). Tous les tableaux seraient générés en même temps. Pour chaque base générée, on génèrerait plusieurs Y de plusieurs manières pour avoir tous les cas sur les mêmes données. La parallélisation des expériences se ferait alors sur le nombre de réplications. Les résultats seraient toujours basés sur Zchapeau (et donc Bchapeau).

On devrait constater que quand l'algo a le temps de converger, on trouve pour n petit des BICs meilleurs que le vrai modèle (d'où un bruit sur la structure). quand n augmente, ce surapprentissage devrait disparaître et on devrait donc converger vers le vrai Z (et le vrai BIC).

4.2 Y depends only on some covariates in X^{I_1}

4.2.1 without selection

n	OLS	(sd)	explicative	(sd)	predictive	(sd)
40	NA	NA	??	??	??	??
60	??	??	??	??	??	??
100	??	??	??	??	??	??

Table 2: Y only depends on X^{I_1}. $p = 50$ and $\mathrm{Var}(Y) \simeq 3.10^8$

On doit constater qu'on est meilleurs que OLS, que l'explicatif gagne (vrai modèle possible) mais que le prédictif reste bon. On doit aussi voir que quand n grandit OLS commence à redevenir correct.

4.2.2 with selection

Avec le même Y que pour le cas sans sélection, (et les mêmes données) on teste simplement d'autres modèles :

- package lm
- modèle complet avec lasso (et LAR)
- modèle complet avec elastic net
- modèle complet ridge
- modèle explicatif elastic net
- modèle prédictif elastic net
- modèle explicatif LASSO
- modèle prédictif LASSO

Il y aurait un tableau par valeur de n pour pouvoir donner en plus des MSE des valeurs de sparsité et de validité du modèle (comparaison des positions des 0). Je n'ai pas pour l'instant de quoi utiliser CLERE mais l'article CLERE montre qu'elastic net est meilleur en prédiction donc pour les MSE ce n'est pas trop un problème.

4.3 Y depends only on some covariates in X^{I_2}

même chose qu'avant mais on part avec un handicap. les notions d'explicatif et prédictif finaux devraient alors prendre tout leur sens.

4.4 global case

Y dépend un peu de tout le monde... me semble trop compliqué car beaucoup trop de cas possibles. la conclusion étant de toute manières qu'on sera quelque part entre les deux cas précédents. Je mettrais bien des exemples simples et poussés (3 variables explicatives comme dans l'article sur la consistance du lasso) pour que les gens puissent facilement refaire le test chez eux, même sans notre package (hypothèse du vrai Z). la simplicité de l'exemple permettrait aussi de voir ce qui se passe si Zchapeau est une version permutée du vraiZ. On testerait là aussi tous les modèles concurrents abordés plus haut.

Attention : on a une variabilité due à la validation croisée. Sur les mêmes données, quand on lance plusieurs fois la sélection on ne trouve pas toujours exactement les mêmes 0 (tout de même relativement stable, peut s'arranger en choisissant un meilleur K pour la validation croisée).

4.5 Consistency Issues

Consistency issues of the LASSO are well known and Zhao [13] gives a very simple example to illustrate it. We have taken the same example to show how our method is more consistent. Here $p = 3$ and $n = 1000$.We define $X_1, X_2, \varepsilon_Y, \varepsilon_X i.i.d. \sim \mathcal{N}(0,1)$ and then $X_3 = \frac{2}{3}X_1 + \frac{2}{3}X_2 + \frac{1}{3}\varepsilon_X$ and $Y = 2X_1 + 3X_2 + \varepsilon_Y$. We compare consistencies of complete,explicative and predictive model with LASSO (and LAR) for selection. It happens that the algorithm don't find the true strucure but a permuted one so we look at the results obtained with the true Z (but $\hat{B}$ is used) and with the structure found by the Markov chain after a few seconds.

True Z is found 340 times on 1000 tries.

	Classical LASSO	Explicative	Predictive
True Z	1.006479	**1.005468**	**1.006093**
$\hat{Z}$	**1.006479**	1.884175	1.006517

Table 3: MSE observer on a validation sample (1000 individuals)

We observe as we hoped that explicative model is better when using true Z (coercing real zeros) and that explicative with $\hat{Z}$ is penalized (coercing wrong coefficients to be zeros). But the main point is that the predictive model stay better than the classical one whith the true Z and corrects enough the explicative model to follow the classical LASSO closely when using $\hat{Z}$. And when we look at the consistency :

	Classical LASSO	Explicative	Predictive
True Z	0	1000	830
$\hat{Z}$	0	340	**621**

Table 4: number of consistent model found (Y depending on X_1, X_2 and only them) on 1000 tries

299 times on 1000 tries, the predictive model using $\hat{Z}$ is better than classical LASSO in terms of MSE and consistent (classical LASSO is never consistent).

We also made the same experiment but with X_1, X_2 (and consequently X_3) following gaussian mixtures (to improve identifiability) randomly generated by our R package. True Z is now found 714 times on 1000 tries . So it confirms that non-gaussian models are easier to identify.

	Classical LASSO	Explicative	Predictive
True Z	1.571029	**1.569559**	**1.570801**
$\hat{Z}$	1.005402	1.465768	**1.005066**

Table 5: MSE observed on a validation sample (1000 individuals)

And when we look at the consistency :

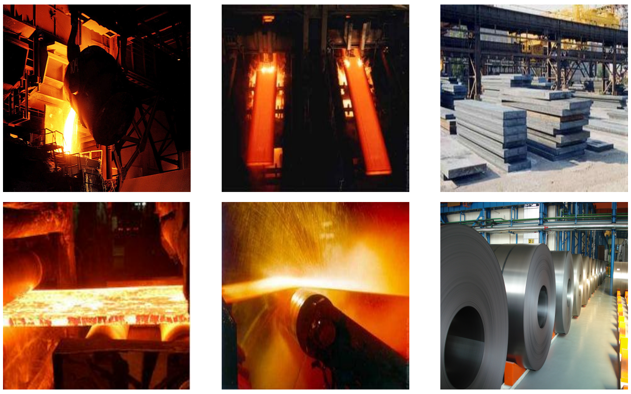
	Classical LASSO	Explicative	Predictive
True Z	0	1000	789
$\hat{Z}$	0	714	**608**

Table 6: number of consistent model found (Y depending on X_1, X_2 and only them) on 1000 tries

299 times on 1000 tries, the predictive model using $\hat{Z}$ is better than classical LASSO in terms of MSE and consistent (classical LASSO is never consistent).

5 Numerical results on real datasets

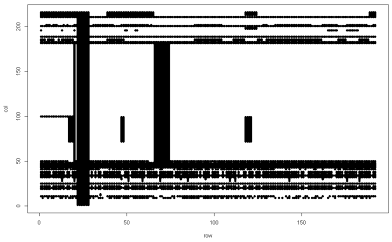
This work takes place in steel industry context, with quality oriented objective : to understand and prevent quality problems on finished product, knowing the whole process. In particular, we focus on regression problems.



Industrial context often means specific issues :

- Highly correlated parameters (parameters depends on the targeted product, physical models,...).
- Sometimes more variables than individuals.
- Missing values

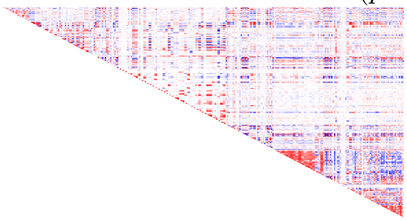
Figure 2: Missing values on a real industrial dataset



5.1 The dataset

- variables from the whole process
- The stakes : hundreds euros per ton (for information: Dunkerque site produces up to 7.5 millions tons a year)

Figure 3: Correlation matrix of the dataset ($p = 293$, $n = 3000$)



Some observed correlations with physical meaning :

- Width and Weight : 0.905

- Temperature before and after a tool : 0.983
- Roughness of both faces : 0.919
- Mean & Max of a curve : 0.911

The method was tested on 205 variables without missing values.

5.2 Results

The algorithm gives a structure with 82 subregressions with a mean of 5 regressors. Some found subregressions with physical meaning :

- Mean.weight = f (Min.weight , Max.weight , Sigma.weight) and other same-shaped subregressions.
- Width = f (Mean.flow , Mean.speed.CC)

  True Physical model (not linear) :

  - Width = flow / (speed * thickness) (thickness is constant)

Some of the other subression represent physical models used to regulate the process and that were forgotten by the metallurgist we worked with. Found model has selected relevant variables (verified with metallurgist).

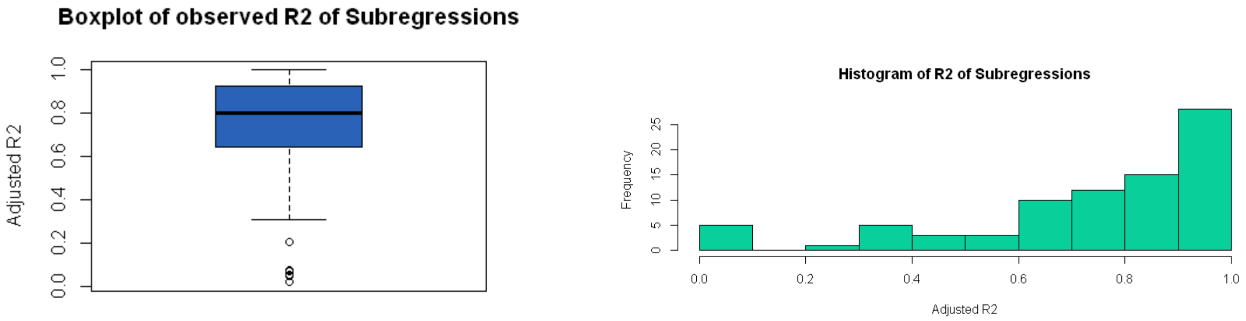


Figure 4: Adjusted R2 of found subregressions (industrial dataset)

We used Elastic Net[14] on this dataset for selection (get better results than LASSO). Here are the observed MSE on a $n = 847$ validation sample. Predictive model (sequential elastic net base on estimated structure and using all the variables) is $5,82\%$ better (Figure 5.2) than elastic net computed on the whole dataset.

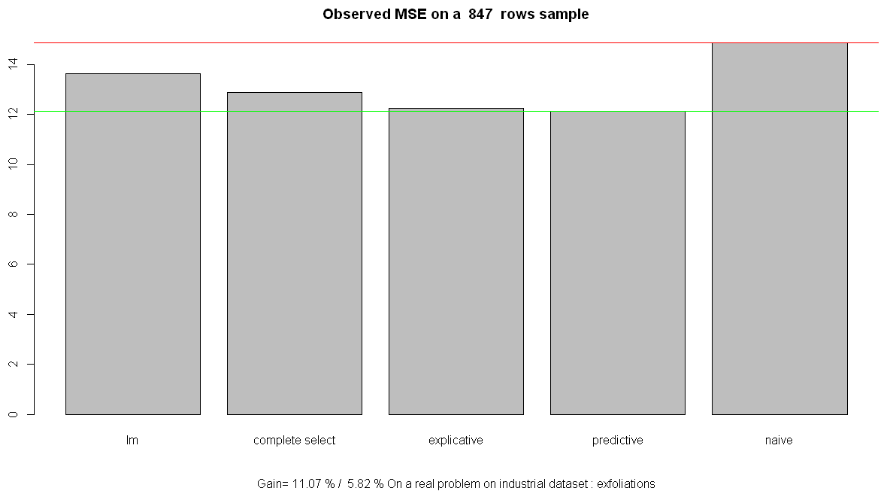


Figure 5: MSE comparison on industrial dataset

6 Conclusion

We have seen that correlations can lead to serious estimation and variable selection problems in linear regression and that in such a context, it can be useful to explicitly model the structure between the covariates and to use this structure (even sequentially) to avoid correlations issues. We also show that real industrial context faces this kind of situations so our model can help to interprete and predict physical phenomenon efficiently and to help to manage missing values. But for now we still need a full dataset to learn the structure between the covariates and the method only works with numerical values. Further work is needed to face these two challenges.

References

[1] Norman Biggs. *Algebraic graph theory.* Cambridge University Press, 1993.

[2] R. Davidson and J.G. MacKinnon. Estimation and inference in econometrics. *Oxford University Press Catalogue*, 1993.

[3] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[4] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[5] Charles Miller Grinstead and James Laurie Snell. *Introduction to probability.* American Mathematical Soc., 1997.

[6] É. Lebarbier and T. Mary-Huard. Une introduction au critère bic: fondements théoriques et interprétation. *Journal de la SFdS*, 147(1):39–57, 2006.

[7] D.W. Marquardt and R.D. Snee. Ridge regression in practice. *American Statistician*, pages 3–20, 1975.

[8] G. Saporta. *Probabilités, analyse des données et statistique.* Editions Technip, 2006.

[9] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[10] N.H. Timm. *Applied multivariate analysis.* Springer Verlag, 2002.

[11] Loic Yengo, Julien Jacques, Christophe Biernacki, et al. Variable clustering in high dimensional linear regression models. 2012.

[12] A. Zellner. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *Journal of the American statistical Association*, pages 348–368, 1962.

[13] Peng Zhao and Bin Yu. On model selection consistency of lasso. *J. Mach. Learn. Res.*, 7:2541–2563, December 2006.

[14] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.