CorReg

Clément Théry, Christophe Biernacki, Gaétan Loridant

ArcelorMittal Dunkerque, Université de Lille 1,équipe MΘdal Inria

March 26, 2015

Context

Proposed Models

Structure estimation

Results

Missing values

Tools

1. Steel industry databases.
2. Goal: To understand and prevent quality problems on finished product, knowing the whole process, without a priori.






Linear Regression

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

where $\varepsilon \sim \mathcal{N}(0, \sigma_Y^2 \boldsymbol{I}_n)$

MLE

β can be estimated by $\hat{\beta}$ with Ordinary Least Squares (OLS), that is the unbiased maximum likelihood estimator [Saporta, 2006, Dodge and Rousson, 2004]:

$$\hat{\beta}_{OLS} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \tag{2}$$

with variance matrix

$$\mathrm{Var}(\hat{\beta}_{OLS}) = \sigma_Y^2\,(\boldsymbol{X}'\boldsymbol{X})^{-1}. \tag{3}$$

In fact it is the Best Linear Unbiased Estimator (BLUE). The theoretical MSE is given by

$$\mathrm{MSE}(\hat{\beta}_{OLS}) = \sigma_Y^2\,\mathrm{Tr}((\boldsymbol{X}'\boldsymbol{X})^{-1}).$$

OLS

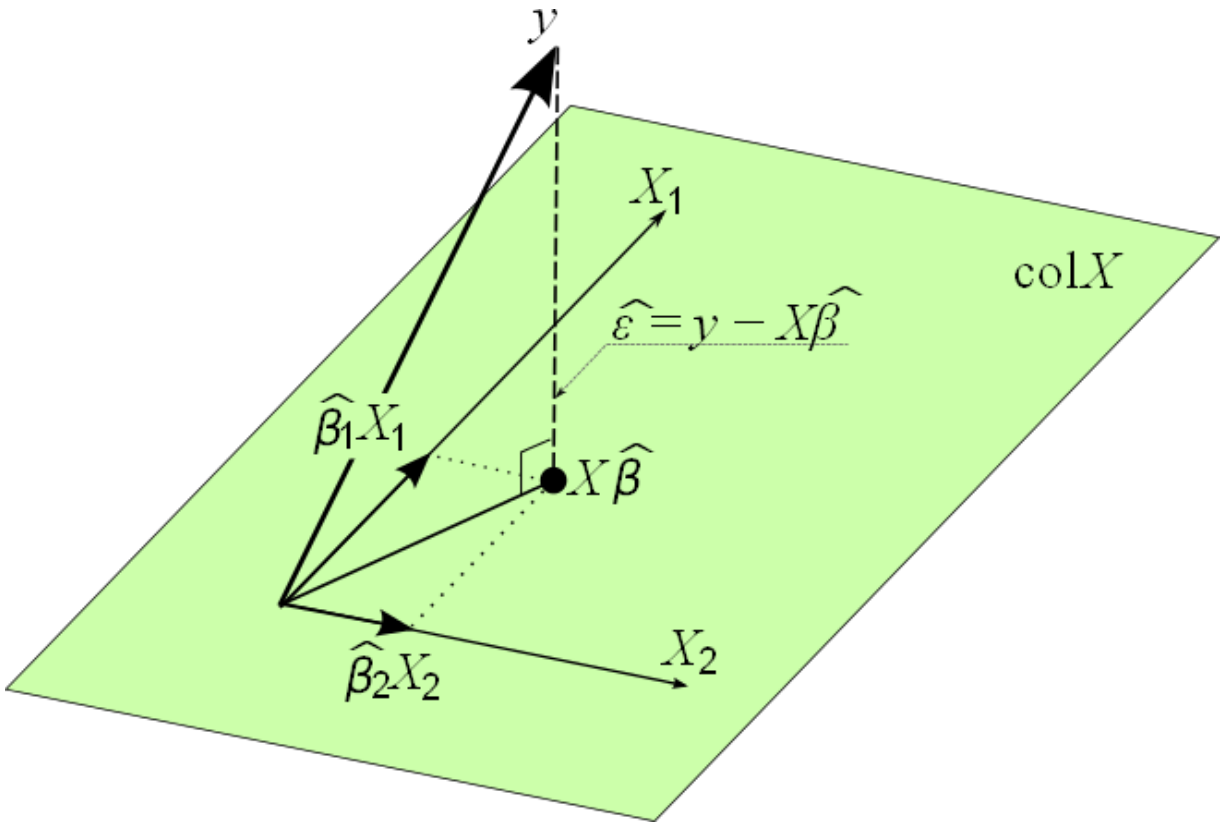


Figure: Multiple linear regression with Ordinary Least Squares seen as a projection on the $d-$dimensional hyperplane spanned by the regressors $\boldsymbol{X}$. Public domain image.

Running example

$\boldsymbol{X}^1, \boldsymbol{X}^2, \boldsymbol{X}^4, \boldsymbol{X}^5 \sim \mathcal{N}(0,1)$ and $\boldsymbol{X}^3 = \boldsymbol{X}^1 + \boldsymbol{X}^2 + \varepsilon_1$ where $\varepsilon_1 \sim \mathcal{N}(\boldsymbol{0}, \sigma_1^2 \boldsymbol{I}_n)$.
Two *scenarii* for $\boldsymbol{Y}$:
$\beta = (1,1,1,1,1)'$ and $\sigma_Y \in \{10, 20\}$.
It is clear that $\boldsymbol{X}'\boldsymbol{X}$ will become more ill-conditioned as σ_1 gets smaller. R^2 stands for the coefficient of determination which is here:

$$R^2 = 1 - \frac{\text{Var}(\varepsilon_1)}{\text{Var}(\boldsymbol{X}^3)} \tag{4}$$

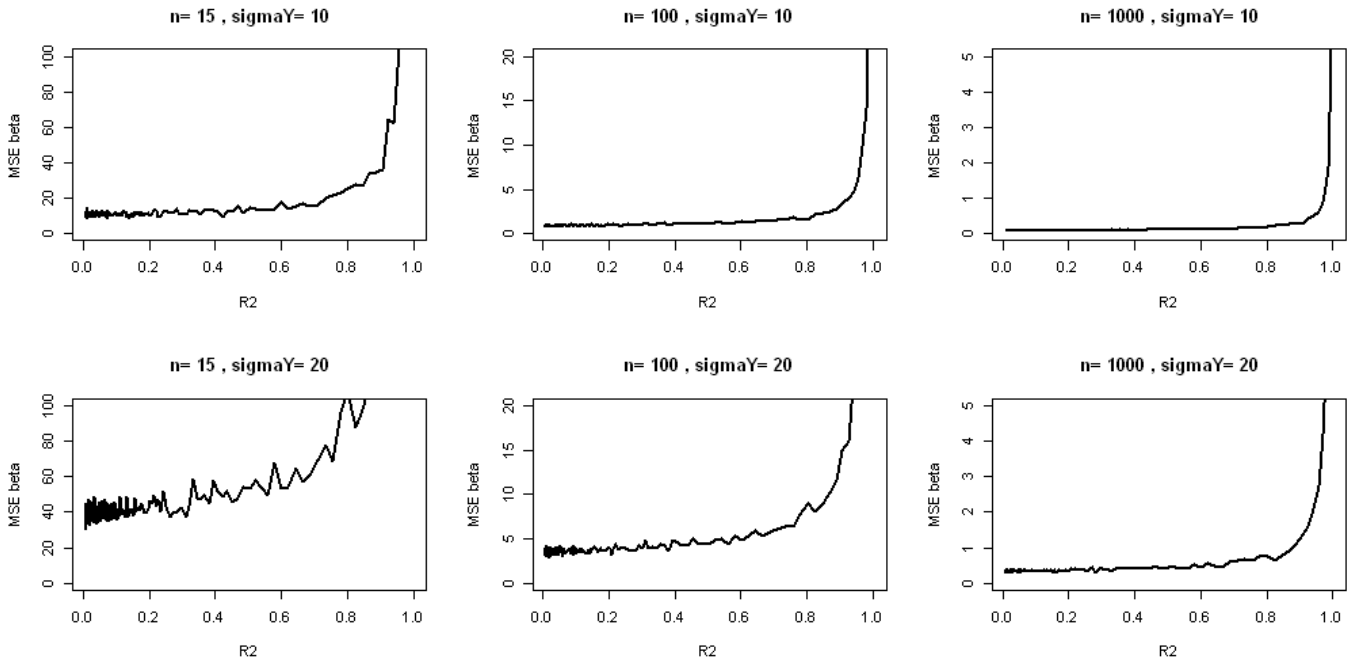


Figure: Evolution of observed Mean Squared error on $\hat{\beta}_{OLS}$ with the strength of the correlations for various sample sizes and strength of regression. $d = 5$ covariates (running example).

Ridge Regression

[Hoerl and Kennard, 1970, Marquardt and Snee, 1975] proposes a possibly biased estimator for β that can be written in terms of a parametric L_2 penalty:

$$\hat{\beta} = \text{argmin}_{\beta} \left\{ \| \boldsymbol{Y} - \boldsymbol{X}\beta \|_2^2 \right\} \text{ subject to } \| \beta \|_2^2 \leq \eta \text{ with } \eta > 0 \tag{5}$$

But this penalty is not guided by correlations. The solution of the ridge regression is given by

$$\hat{\beta} = \left(\boldsymbol{X}'\boldsymbol{X} - \lambda \boldsymbol{I}_n \right)^{-1} \boldsymbol{X}'\boldsymbol{Y} \tag{6}$$

Methods do exist to automatically choose a good value for λ [Cule and De Iorio, 2013, Er et al., 2013] and a R package called `ridge` is on CRAN [Cule, 2014].

Ridge Regression

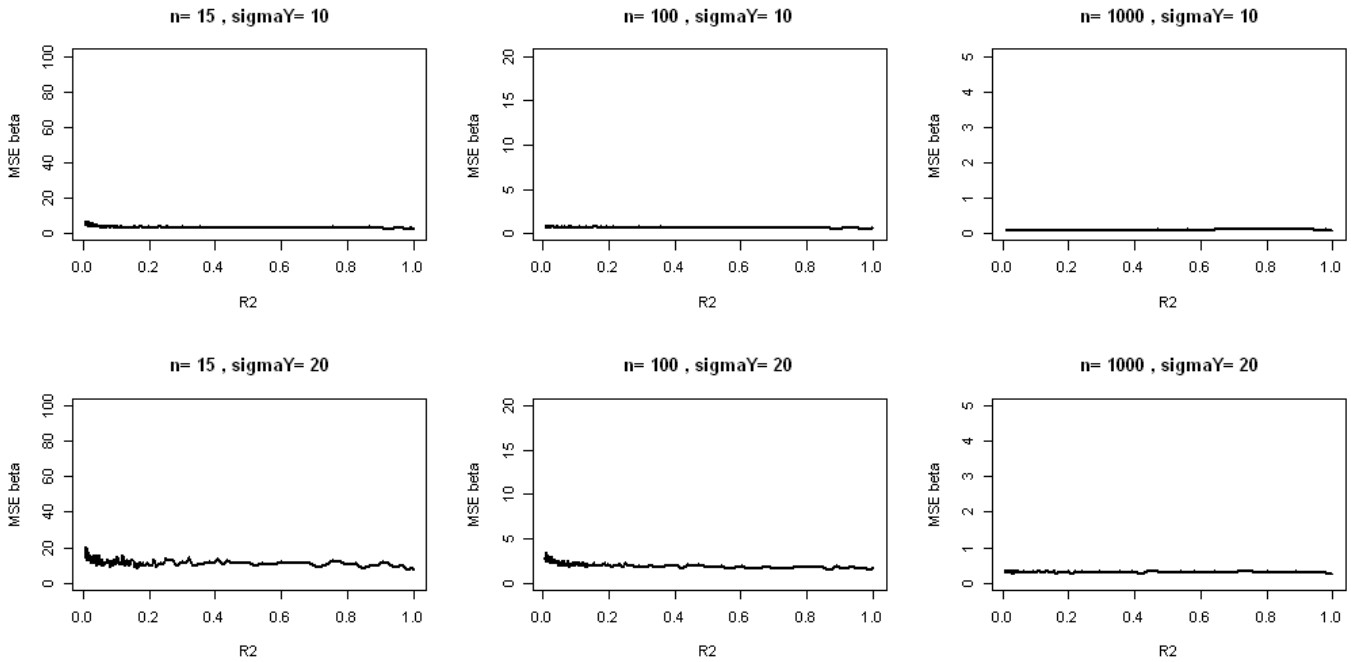


Figure: Evolution of observed Mean Squared error on $\hat{\beta}_{ridge}$ with the strength of the correlations for various sample sizes and strength of regression. $d = 5$ covariates.

LASSO

The Least Absolute Shrinkage and Selection Operator (LASSO, [Tibshirani, 1996] and [Tibshirani et al.,]) consists in a shrinkage of the regression coefficients based on a λ parametric L_1 penalty to obtain zeros in $\hat{\beta}$ instead of the L_2 penalty of the ridge regression:

$$\hat{\beta} = \text{argmin} \left\{ \| \boldsymbol{Y} - \boldsymbol{X}\beta \|_2^2 \right\} \text{ subject to } \| \beta \|_1 \leq \lambda \text{ with } \lambda > 0.$$

LASSO

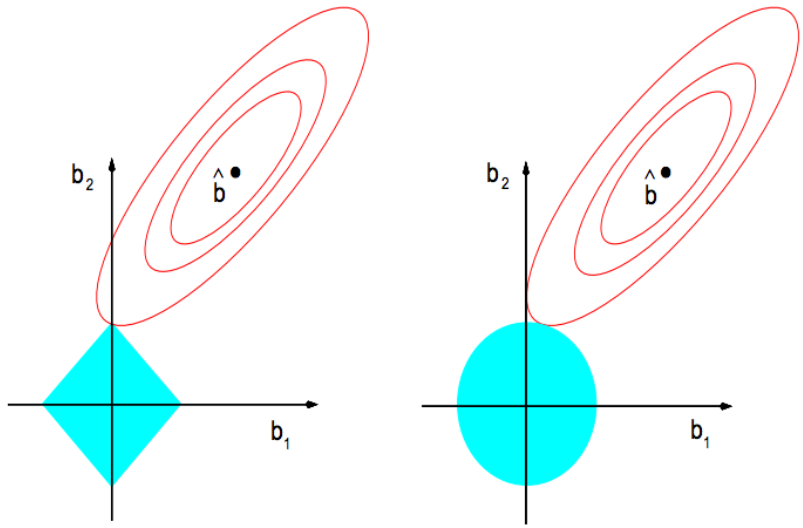


Figure: Geometric view of the Penalty for the LASSO (left) compared to ridge regression (right) as shown in the book from Hastie [Hastie et al., 2009]

Figure shows the contour of error (red) and constraint function (blue). The axis stands for the regression coefficients.

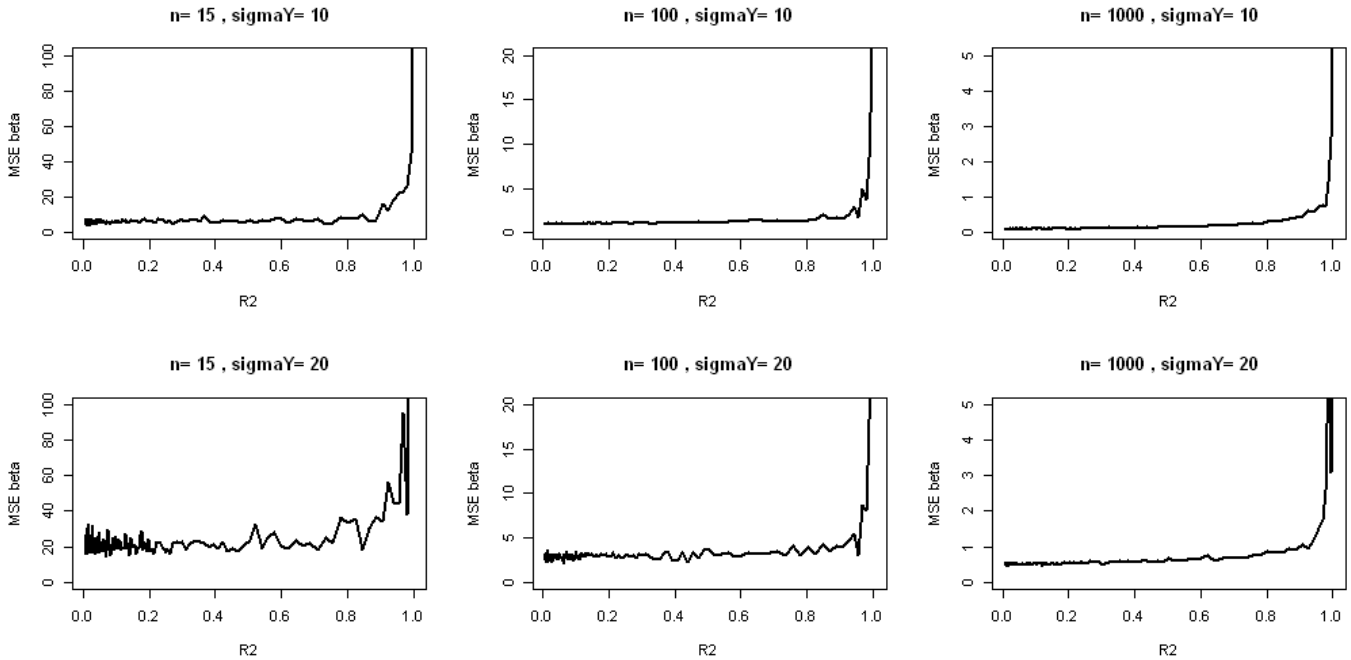


Figure: Evolution of observed Mean Squared error on $\hat{\beta}_{lar}$ with the strength of the correlations for various sample sizes and strength of regression. $d = 5$ covariates.

`lars` package on CRAN ([Hastie and Efron, 2013]).

Simultaneous Equation Modeling

$$\boldsymbol{X}_{n,d} = \boldsymbol{X}_{n,d}\beta_{d,d} + \varepsilon_{n,d}$$

No real "response variable", no real estimators for recursive SEM, no real variable selection (structure finder)

Selvarclust

$$\boldsymbol{X}^U = a + \boldsymbol{X}^R \boldsymbol{\alpha} + \boldsymbol{\varepsilon}$$

- Structure based on stepwise [Raftery and Dean, 2006, Miller, 2002]
- Made only for clustering purpose.

Hypothesis 1

There are $d_r \geq 0$ "sub-regressions", each sub-regression $j = 1, \ldots, d_r$ having the covariate $\boldsymbol{X}^{J_r^j}$ as *response* variable ($J_r^j \in \{1, \ldots, p\}$ and $J_r^j \neq J_r^{j'}$ if $j \neq j'$) and having the $d_p^j > 0$ covariates $\boldsymbol{X}^{J_p^j}$ as *predictor* variables ($J_p^j \subset \{1, \ldots, d\} \backslash J_r^j$ and $d_p^j = |J_p^j|$ the cardinal of J_p^j):

$$\boldsymbol{X}^{J_r^j} = \boldsymbol{X}^{J_p^j} \boldsymbol{\alpha}_j + \boldsymbol{\varepsilon}_j, \tag{7}$$

where $\boldsymbol{\alpha}_j \in \mathbb{R}^{d_r^j}$ ($\alpha_j^h \neq 0$ for all $j = 1, \ldots, d_r$ and $h = 1, \ldots, d_p^j$) and $\boldsymbol{\varepsilon}_j \sim \mathcal{N}_n(\boldsymbol{0}, \sigma_j^2 \boldsymbol{I})$.

Hypothesis 2

The response covariates and the predictor covariates are totally disjoint: for any sub-regression $j = 1, \ldots, d_r$, $J_p^j \subset J_f$ where $J_r = \{J_r^1, \ldots, J_r^{d_r}\}$ is set of all response covariates and $J_f = \{1, \ldots, d\} \backslash J_r$ is the set of all *non* response covariates of cardinal $d_f = d - d_r = |J_f|$. We call this hypothesis the uncrossing rule. Then:

$$\boldsymbol{Y} = \boldsymbol{X}_f \beta_f + \boldsymbol{X}_r \beta_r + \varepsilon_Y. \tag{8}$$

Hypotheses 3

We assume that all errors ε_Y and ε_j $(j = 1, \ldots, d_r)$ are *mutually independent*. It implies in particular that conditional response covariates $\{\boldsymbol{X}^{J_r^j} | \boldsymbol{X}^{J_p^j}, \boldsymbol{S}; \boldsymbol{\alpha}_j, \sigma_j^2\}$ are *mutually independent*:

$$\mathbb{P}(\boldsymbol{X}_r | \boldsymbol{X}_f, \boldsymbol{S}; \boldsymbol{\alpha}, \boldsymbol{\sigma}^2) = \prod_{j=1}^{d_r} \mathbb{P}(\boldsymbol{X}^{J_r^j} | \boldsymbol{X}^{J_p^j}, \boldsymbol{S}; \boldsymbol{\alpha}_j, \sigma_j^2). \tag{9}$$

Marginal model

We obtain for the distribution of $\{\boldsymbol{Y}|\boldsymbol{X}_f, \boldsymbol{S}; \beta, \boldsymbol{\alpha}, \sigma^2_Y, \boldsymbol{\sigma}^2\}$:

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{X}_f(\beta_f + \sum_{j=1}^{d_r} \beta_{J_r^j} \boldsymbol{\alpha}_j^*) + \sum_{j=1}^{d_r} \beta_{J_r^j} \boldsymbol{\varepsilon}_j + \boldsymbol{\varepsilon}_Y \quad (10) \\ &= \boldsymbol{X}_f \beta_f^* + \boldsymbol{\varepsilon}_Y^*, \quad (11) \end{aligned}$$

where $\boldsymbol{\alpha}_j^* \in \mathbb{R}^{d_f}$ with $(\boldsymbol{\alpha}_j^*)_{J_p^j} = \boldsymbol{\alpha}_j$ and $(\boldsymbol{\alpha}_j^*)_{J_f \setminus J_p^j} = \mathbf{0}$. We define $\boldsymbol{\alpha}^* \in \mathbb{R}^{(d_f \times d_r)}$ to use more compact notations:

$$\begin{aligned} \boldsymbol{X}_r &= \boldsymbol{X}_f \boldsymbol{\alpha}^* + \boldsymbol{\varepsilon} \\ \boldsymbol{Y} &= \boldsymbol{X}_f(\beta_f + \boldsymbol{\alpha}^* \beta_{\boldsymbol{r}}) + \boldsymbol{\varepsilon} \beta_{\boldsymbol{r}} + \boldsymbol{\varepsilon}_Y \quad (12) \end{aligned}$$

Where $\boldsymbol{\varepsilon}$ is the $n \times d_r$ matrix whose columns are the $\boldsymbol{\varepsilon}_j$, the noises of the sub-regressions.

Plug-in model

$$\varepsilon_Y^* = \varepsilon\beta_r + \varepsilon_Y. \tag{13}$$

Then the Best Linear Unbiased Estimator (BLUE) for β_r is given (MLE estimator) by:

$$\hat{\beta}_r = (\varepsilon'\varepsilon)^{-1}\varepsilon'\varepsilon_Y^*. \tag{14}$$

And we have the following estimators:

$$\begin{aligned} \hat{\varepsilon} &= \boldsymbol{X}_r - \boldsymbol{X}_f\hat{\alpha}^* \text{ and} \\ \hat{\varepsilon}_Y^* &= \boldsymbol{Y} - \boldsymbol{X}_f\hat{\beta}_f^* \end{aligned}$$

that we can use by plug-in.

Plug-in model

$$\hat{\beta}_r^{\varepsilon} = (\hat{\varepsilon}'\hat{\varepsilon})^{-1}\hat{\varepsilon}'\hat{\varepsilon}_Y^*$$

that depends on all covariates in $\boldsymbol{X}$ and relies on the estimated coefficients of sub-regressions $\hat{\alpha}^*$ and on the estimate $\hat{\beta}_f^*$ of the coefficients in the marginal model. Then we can estimate $\boldsymbol{Y}$ by:

$$\hat{\boldsymbol{Y}}_{plug-in} = \boldsymbol{X}_f\hat{\beta}_f^* + \hat{\varepsilon}\hat{\beta}_r^{\varepsilon}. \quad (15)$$

We can improve estimation of β_f (in terms of bias) by doing an additional identification step. We know that $\beta_f^* = \beta_f + \alpha^*\beta_r$ so we naturally define the following estimator:

$$\hat{\beta}_f^{\varepsilon} = \hat{\beta}_f^* - \hat{\alpha}^*\hat{\beta}_r^{\varepsilon}.$$

Marginal model's properties

$$\mathbb{E}(\hat{\beta}_f^*) = \beta_f + \sum_{j=1}^{d_r} \beta_{J_r^j} \boldsymbol{\alpha}_j^* \text{ and } \mathbb{E}(\hat{\beta}_r^*) = \mathbf{0}$$

$$\text{Var}(\hat{\beta}_f^*) = (\sigma_Y^2 + \sum_{j=1}^{d_r} \sigma_j^2 \beta_{J_r^j}^2)(\boldsymbol{X}_f' \boldsymbol{X}_f)^{-1} \quad \text{and} \quad \text{Var}(\hat{\beta}_r^*) = \mathbf{0}. \quad (16)$$

Plug-in model's properties

Asymptotically unbiased (consistent estimators + continuous mapping theorem)

$$\begin{aligned} \mathrm{Var}(\hat{\beta}_r^{\varepsilon}) &= \sigma_Y^2 \left[\hat{\varepsilon}'\hat{\varepsilon}[\hat{\varepsilon}'(\boldsymbol{I}_n - \boldsymbol{H}_f)\hat{\varepsilon}]^{-1}\hat{\varepsilon}'\hat{\varepsilon}\right]^{-1} \\ &= \sigma_Y^2 \left[(\boldsymbol{X}_r^{\varepsilon})'\boldsymbol{X}_r^{\varepsilon}\right]^{-1} \end{aligned}$$

$$\mathrm{Var}(\hat{\beta}_f^{\varepsilon}) = \sigma_Y^2 \left[(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1} + \hat{\alpha}^* \left[(\boldsymbol{X}_r^{\varepsilon})'\boldsymbol{X}_r^{\varepsilon}\right]^{-1} (\hat{\alpha}^*)'\right].$$

$$\mathrm{Var}_{ols}(\hat{\beta}_f) =$$
$$\sigma_Y^2 \left[(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1} + (\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1}\boldsymbol{X}_f'\boldsymbol{X}_r[\boldsymbol{X}_r'\boldsymbol{X}_r - \boldsymbol{X}_r'\boldsymbol{X}_f(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1}\boldsymbol{X}_f'\boldsymbol{X}_r]^{-1}\boldsymbol{X}_r'\boldsymbol{X}_f(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1} \right],$$

$$= \sigma_Y^2 \left[(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1} + \underbrace{(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1}\boldsymbol{X}_f'\boldsymbol{X}_r}_{\hat{\boldsymbol{\alpha}}^*_{ols}}[\boldsymbol{X}_r'(\boldsymbol{I}_n - \boldsymbol{H}_f)\boldsymbol{X}_r]^{-1} \underbrace{\boldsymbol{X}_r'\boldsymbol{X}_f(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1}}_{(\hat{\boldsymbol{\alpha}}^*_{ols})'} \right]$$

$$\begin{aligned} \mathrm{Var}_{ols}(\hat{\beta}_r) &= \sigma_Y^2 \left[\boldsymbol{X}_r'\boldsymbol{X}_r - \boldsymbol{X}_r'\boldsymbol{X}_f(\boldsymbol{X}_f'\boldsymbol{X}_f)^{-1}\boldsymbol{X}_f'\boldsymbol{X}_r \right]^{-1} \\ &= \sigma_Y^2 \left[\boldsymbol{X}_r'(\boldsymbol{I}_n - \boldsymbol{H}_f)\boldsymbol{X}_r \right]^{-1}. \end{aligned} \tag{17}$$

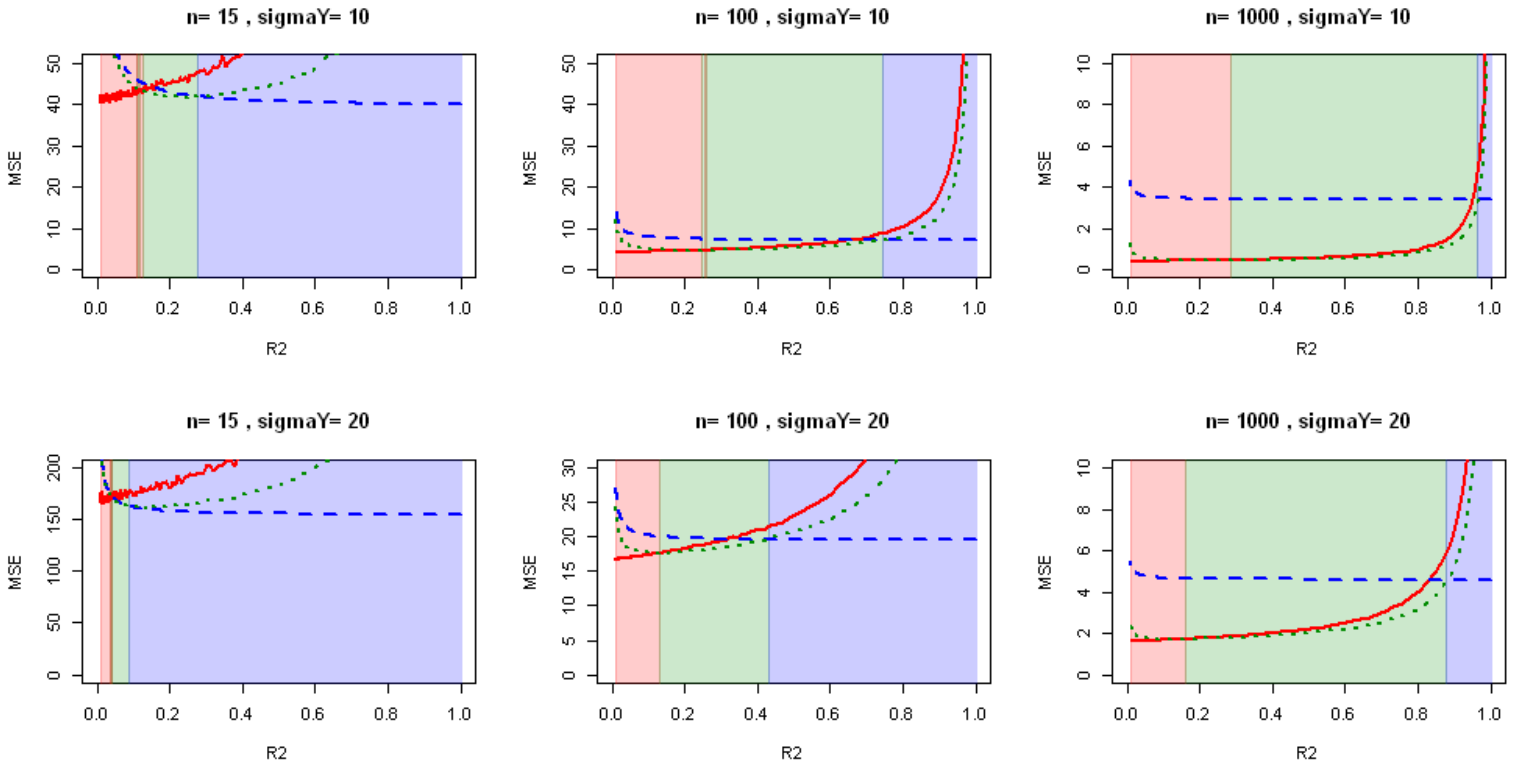


Figure: MSE on $\hat{\beta}$ of OLS (plain red) and `CorReg` marginal (blue dashed) and `CorReg` plug-in (green dotted) estimators for varying R^2 of the sub-regression, n and σ_Y. Results obtained on the running example with $d = 5$ covariates.

Lasso Consistency

Consistency issues of the LASSO are well known and Zhao [Zhao and Yu, 2006] gives a very simple example to illustrate it. We have taken the same example to show how our method is better to find the true relevant covariates. Here $d = 3$ and $n = 1\,000$.
We define $\boldsymbol{X}^1, \boldsymbol{X}^2, \varepsilon_Y, \varepsilon_1 \quad i.i.d. \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_n)$ and then

$$\begin{aligned} \boldsymbol{X}^3 &= \frac{2}{3}\boldsymbol{X}^1 + \frac{2}{3}\boldsymbol{X}^2 + \frac{1}{3}\varepsilon_1 \text{ and} \\ \boldsymbol{Y} &= 2\boldsymbol{X}^1 + 3\boldsymbol{X}^2 + \varepsilon_Y. \end{aligned}$$

Lasso Consistency

True $\boldsymbol{S}$ was found 991 times on 1 000 tries.We look at the consistency that is the real stake:

	Classical LASSO	CorReg marginal + LASSO	CorReg full plug-in + LASSO
True $\boldsymbol{S}$	0	**1000**	**835**
$\hat{\boldsymbol{S}}$	0	**991**	**829**

Table: Number of consistent models found on 1 000 tries.

$$G = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

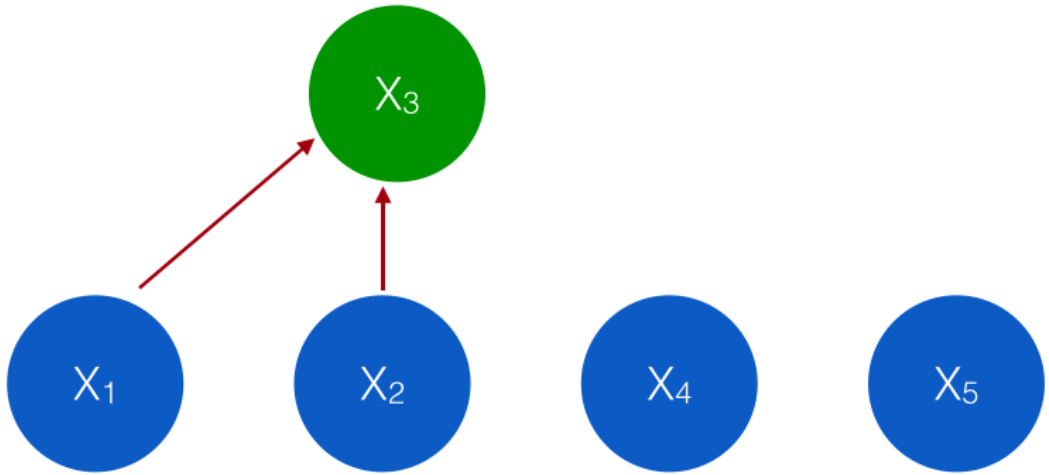


Figure: The bipartite graph associated to the running example. $\boldsymbol{X}_r$ is green and $\boldsymbol{X}_f$ is blue.

Hypothesis 4: Full generative model

All covariates $\boldsymbol{X}^j$ with $j \in J_f$ are mutually independent and arise from the following Gaussian mixture of K_j components

$$\forall 1 \leq i \leq n, \mathbb{P}(x_{i,j}|\boldsymbol{S}; \boldsymbol{\pi}_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \sum_{h=1}^{K_j} \pi_{j,h} \Phi(x_{i,j}; \mu_{j,h}, \Sigma_{j,h}),$$

where $\boldsymbol{\pi}_j = (\pi_{j,1}, \ldots, \pi_{j,K_j})$ is the vector of mixing proportions with $\forall 1 \leq h \leq k_j, \pi_{j,h} > 0$ and $\sum_{h=1}^{K_j} \pi_{j,h} = 1$, $\boldsymbol{\mu}_j = (\mu_{j,1}, \ldots, \mu_{j,K_j})$ is the vector of centres and $\boldsymbol{\Sigma}_j = (\Sigma_{j,1}, \ldots, \Sigma_{j,K_j})$ is the vector of variances and Φ is the Gaussian density function.
We stack together all these mixture parameters in $\boldsymbol{\theta} = (\boldsymbol{\pi}_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j; j \in J_f)$. We now have a full generative model on $\boldsymbol{X}$.

Penalized BIC

we propose to introduce some information in $\mathbb{P}(\boldsymbol{S})$ promoting simple models through the following *hierarchical* uniform distribution denoted by $\mathbb{P}_H(\boldsymbol{S})$:

$$\begin{aligned}
\mathbb{P}_H(\boldsymbol{S}) &= \mathbb{P}_H(\boldsymbol{J}_r, \boldsymbol{J}_p) \\
&= \mathbb{P}_H(\boldsymbol{J}_r, \boldsymbol{J}_p, d_r, \boldsymbol{d}_p) \\
&= \mathbb{P}_U(\boldsymbol{J}_p | \boldsymbol{d}_p, \boldsymbol{J}_r, d_r) \times \mathbb{P}_U(\boldsymbol{d}_p | \boldsymbol{J}_r, d_r) \times \mathbb{P}_U(\boldsymbol{J}_r | d_r) \times \mathbb{P}_U(d_r) \\
&= \left[\prod_{j=1}^{d_r} \binom{d - d_r}{d_p^j} \right]^{-1} \times [d - d_r]^{-d_r} \times \left[\binom{d}{d_r} \right]^{-1} \times [d+1]^{-1}
\end{aligned}$$

At each step the Markov chain moves with probability:

$$\forall \boldsymbol{S} \in \mathcal{S}_d, \forall \boldsymbol{S}^+ \in \mathcal{V}(\boldsymbol{S}) : \mathbb{P}(\boldsymbol{S}^+ | \mathcal{V}(\boldsymbol{S})) = \frac{\exp\left(-\mathrm{BIC}_*(\boldsymbol{S}^+)\right)}{\sum_{\tilde{\boldsymbol{S}} \in \mathcal{V}(\boldsymbol{S})} \exp\left(-\mathrm{BIC}_*(\tilde{\boldsymbol{S}})\right)}$$

where $\mathcal{V}(\boldsymbol{S})$ is a neighbourhood and $\mathcal{S}_d$ is the set of feasible structures within d variables.

For each step (q), starting from $\boldsymbol{S} \in \mathcal{S}_d$ we define a neighbourhood:

$$\mathcal{V}(\boldsymbol{S}) \quad = \quad \{\boldsymbol{S}\} \cup \{\boldsymbol{S}^{(i,j)} \in \mathcal{S}_d | (i,j) \in \mathcal{A}_{(q)}\}$$

where $\mathcal{A}_{(q)}$ is a set of couples $(i,j) \in \{1,\dots,d\}^2$ with $i \neq j$ drawn at the step (q) according to a strategy defined below and corresponding to the directed edge of the graph to modify (add or remove). And we have for $\tilde{\boldsymbol{S}} = \boldsymbol{S}^{(i,j)}$:

$$\begin{aligned} \forall (k,l) \neq (i,j), \quad \tilde{\boldsymbol{G}}_{k,l} &= \boldsymbol{G}_{k,l} \\ \tilde{\boldsymbol{G}}_{i,j} &= 1 - \boldsymbol{G}_{i,j} \end{aligned}$$

where $\tilde{\boldsymbol{G}}$ is the adjacency matrix associated to $\tilde{\boldsymbol{S}}$. Any strategy can be chosen for $\mathcal{A}_{(q)}$, from uniform distribution to specific heuristics.

New definition of $\tilde{\boldsymbol{G}}$: Modification of the selected directed edge (i, j) on the graph:

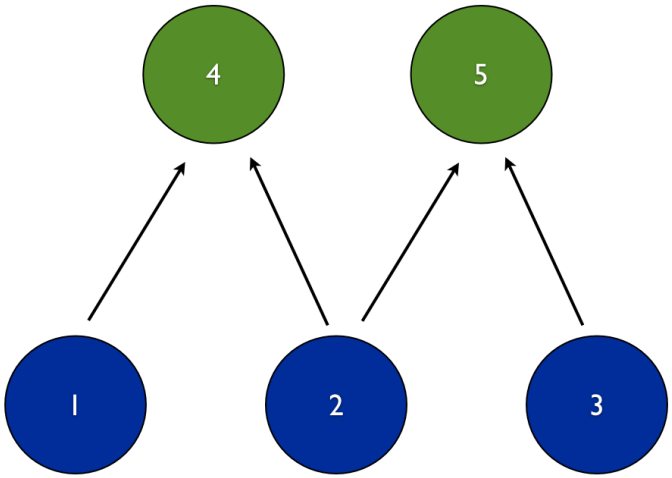
$$\begin{aligned} & \tilde{\boldsymbol{G}}_{i,j} = 1 - \boldsymbol{G}_{i,j} \text{ as usual and} \\ \forall k \neq i, l \neq j, \quad & \tilde{\boldsymbol{G}}_{k,l} = \boldsymbol{G}_{k,l} \end{aligned}$$

Column-wise relaxation : newly predictive covariate cannot be regressed anymore:

$$\forall k \in \{1, \ldots, d\} \setminus \{i\}, \tilde{\boldsymbol{G}}_{k,j} = \boldsymbol{G}_{i,j} \boldsymbol{G}_{k,j} \tag{18}$$

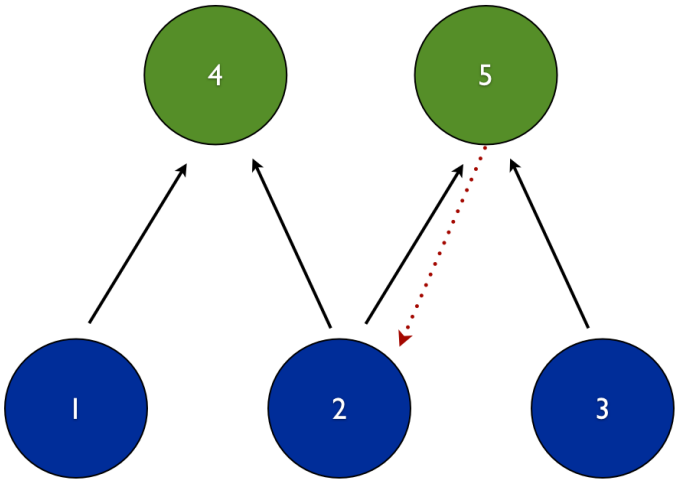
Row-wise relaxation: newly regressed covariate cannot be predictive anymore:

$$\forall l \in \{1, \ldots, d\} \setminus \{j\}, \tilde{\boldsymbol{G}}_{i,l} = \boldsymbol{G}_{i,j} \boldsymbol{G}_{i,l} \text{ (row-wise relaxation)}$$



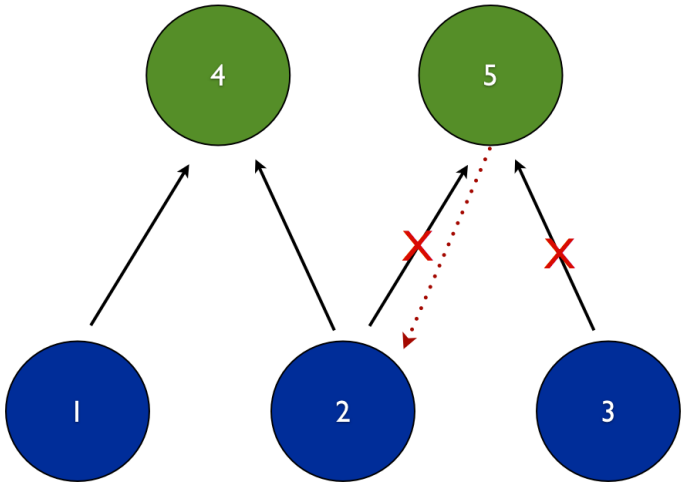
$$\boldsymbol{G} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure: We start from a structure $\boldsymbol{S} = ((4,5),(\{1,2\},\{2,3\}))$ and its associated matrix G



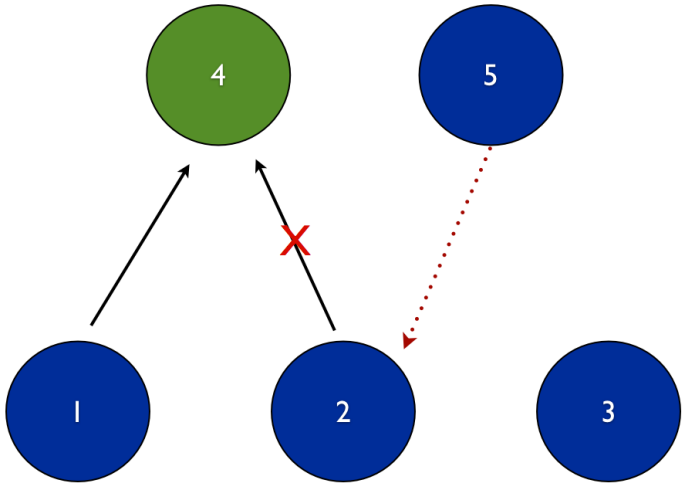
$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 \end{pmatrix}$$

Figure: We want to define the candidate $\tilde{\boldsymbol{S}} = \boldsymbol{S}^{(5,2)}$ and its associated matrix but the structure obtained would not be feasible (breaking the uncrossing rule).



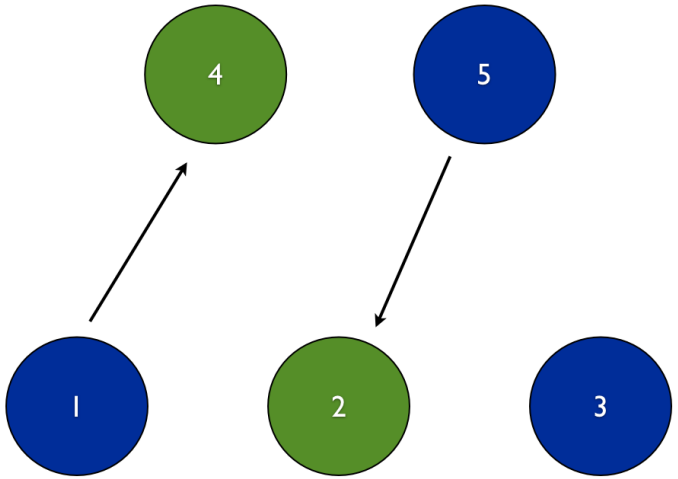
$$\begin{pmatrix} 0 & 0 & 0 & 1 & \mathbf{0} \\ 0 & 0 & 0 & 1 & \mathbf{0} \\ 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & \mathbf{1} & 0 & 0 & \mathbf{0} \end{pmatrix}$$

Figure: Column-wise relaxation: newly predictive covariate cannot be regressed anymore.



$$\tilde{G} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Figure: Row-wise relaxation: newly predictive covariate cannot be regressed anymore.



$$\tilde{\boldsymbol{G}} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Figure: We get a feasible candidate $\tilde{\boldsymbol{S}} = ((2,4),(\{5\},\{1\}))$ that does differ from $\boldsymbol{S}$ in many points.

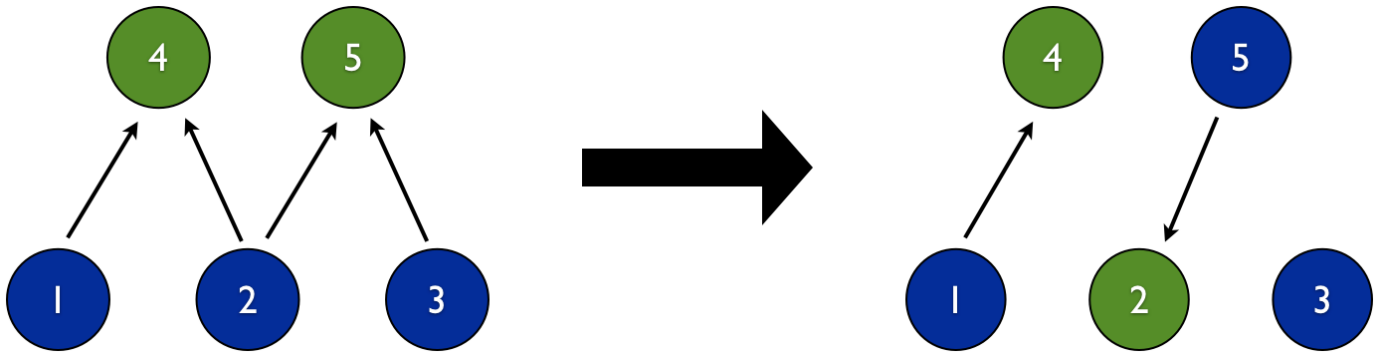


Figure: All this modifications are made in only one step in the MCMC, meaning an increased scope for the neighbourhoods.

Pruning

Every sub-graph of a bipartite graph is bipartite thus every sub-graph can be reached. We propose an heuristic change in the strategy with:

$$\mathcal{A}_{(q)} = \{(i, J_r^j), i \in J_f, j \in \{1, \ldots, d_r\} : i \in J_p^j\}.$$

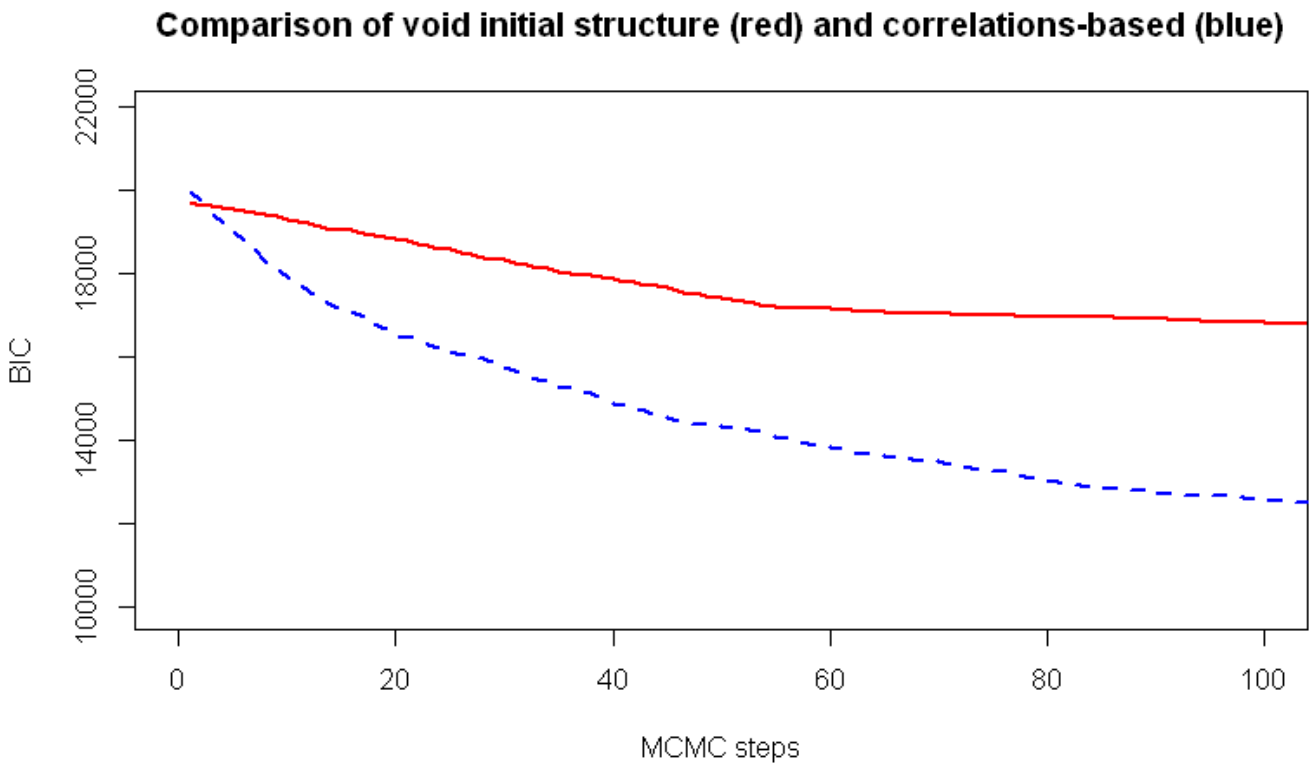


Figure: Evolution of the BIC (criterion to minimize in the MCMC) for each method.

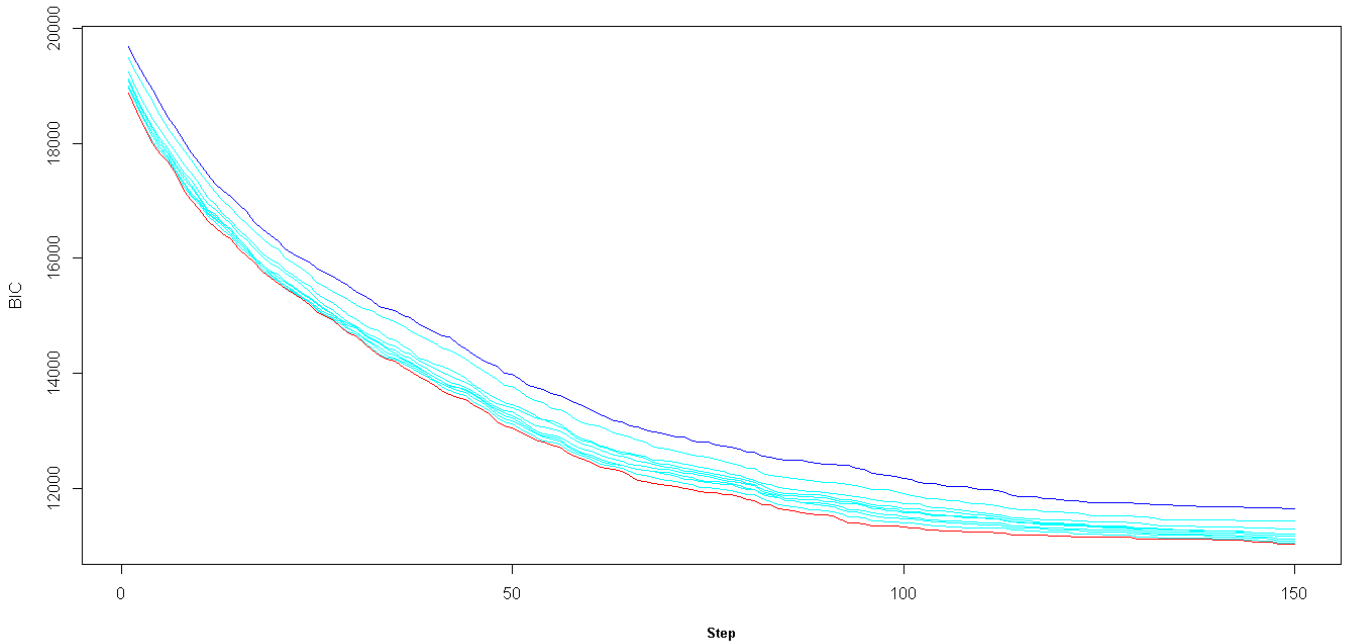


Figure: Comparison of distinct number of correlation-based initializations for the MCMC. Dark blue=1, red=10.

In the following, the chain was launched with twenty initializations each time, based on the correlation matrix.

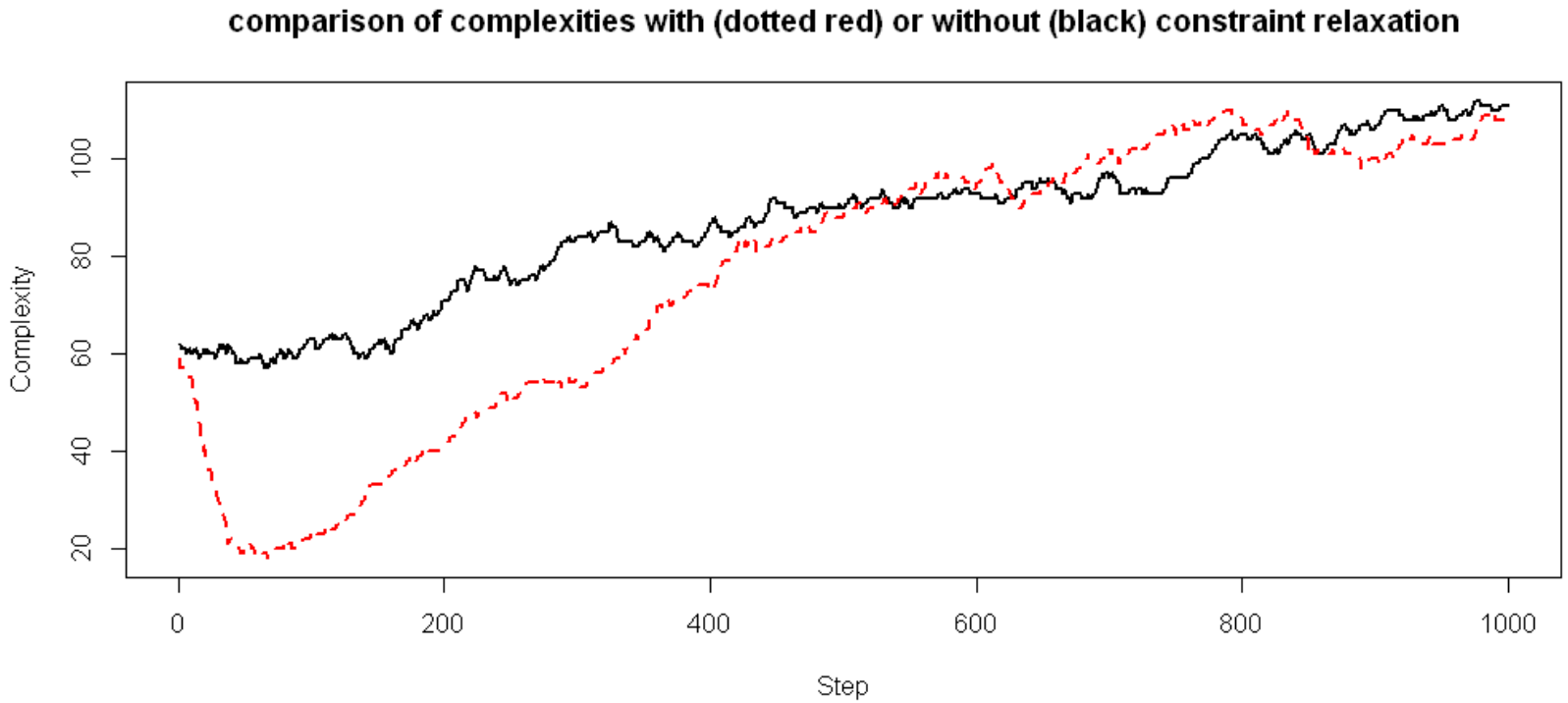


Figure: Comparison of complexity evolution with or without constraint relaxation.

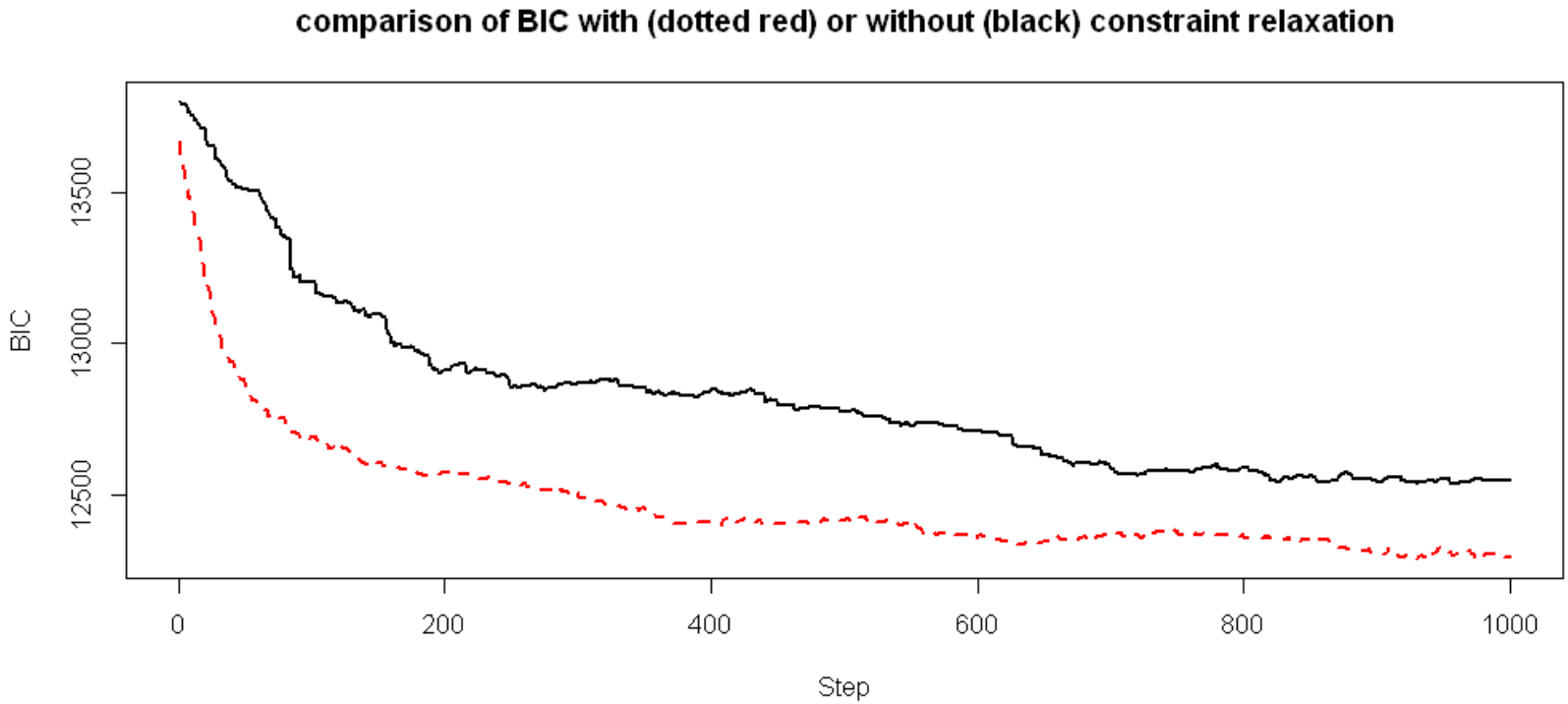


Figure: Comparison of BIC evolution with or without constraint relaxation.

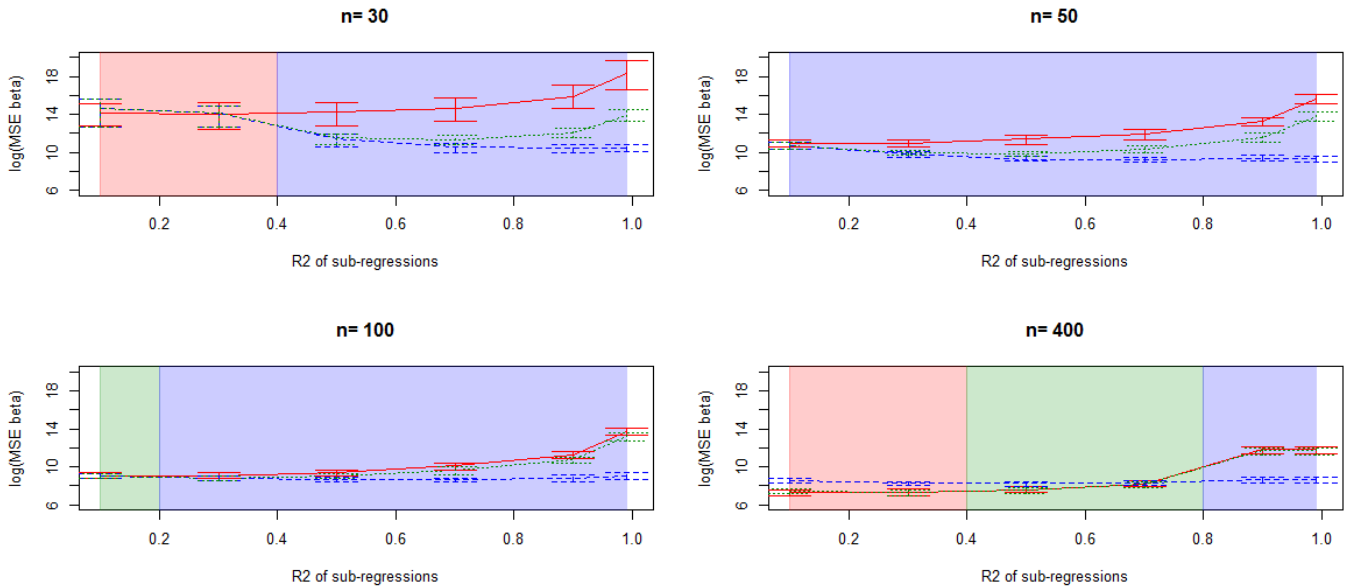


Figure: Comparison of the MSE on $\hat{\beta}$, plain red=classical (complete) model, dashed blue=marginal model, dotted green=plug-in model. OLS estimators

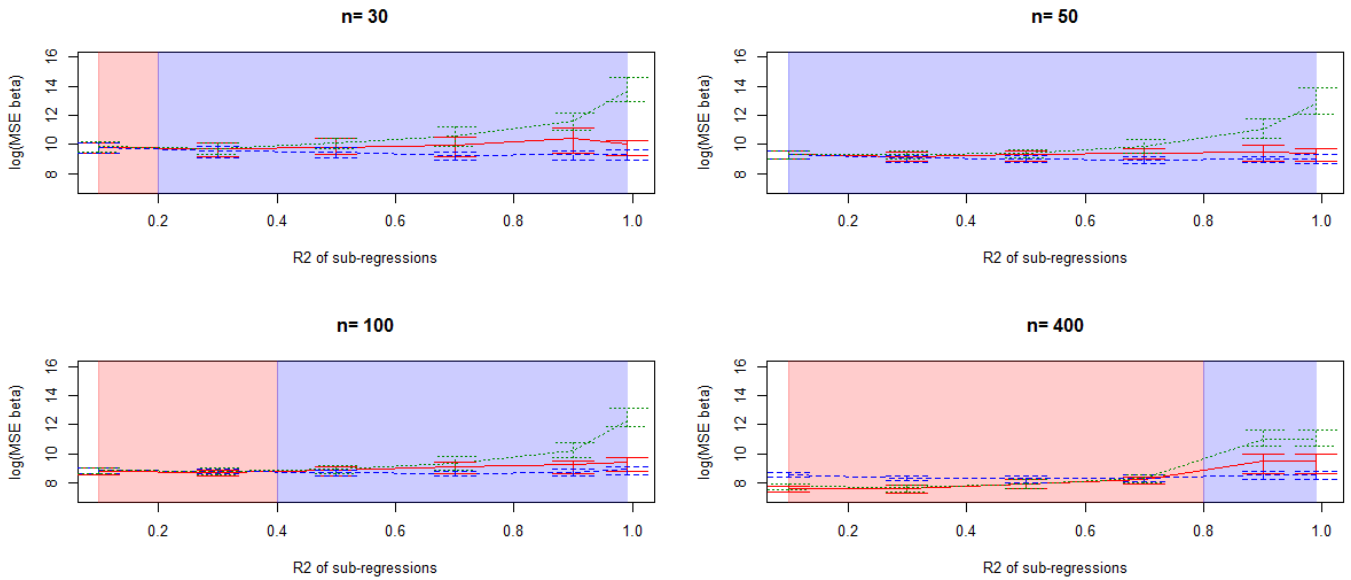
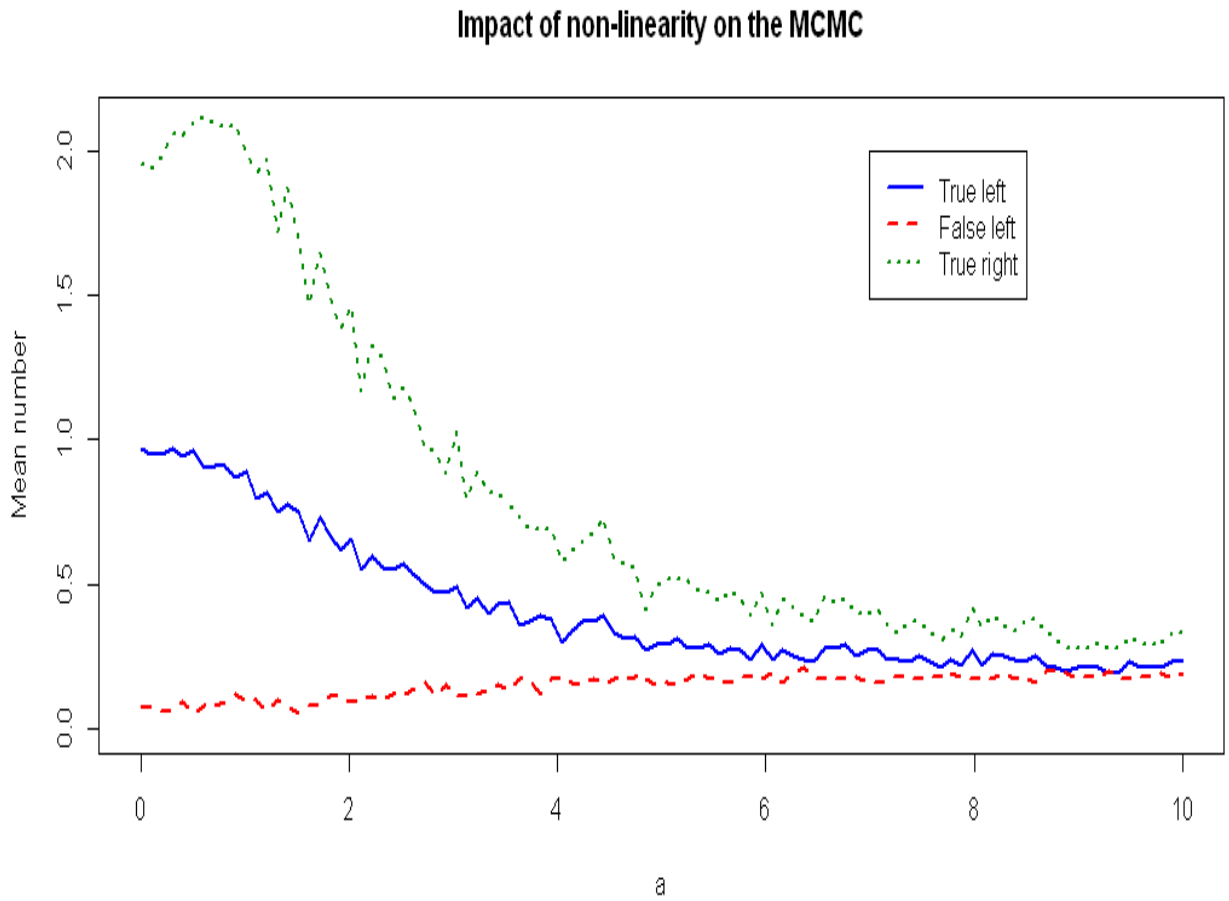


Figure: Comparison of the MSE on $\hat{\beta}$, plain red=classical (complete) model, dashed blue=marginal model, dotted green=plug-in model. LASSO estimators

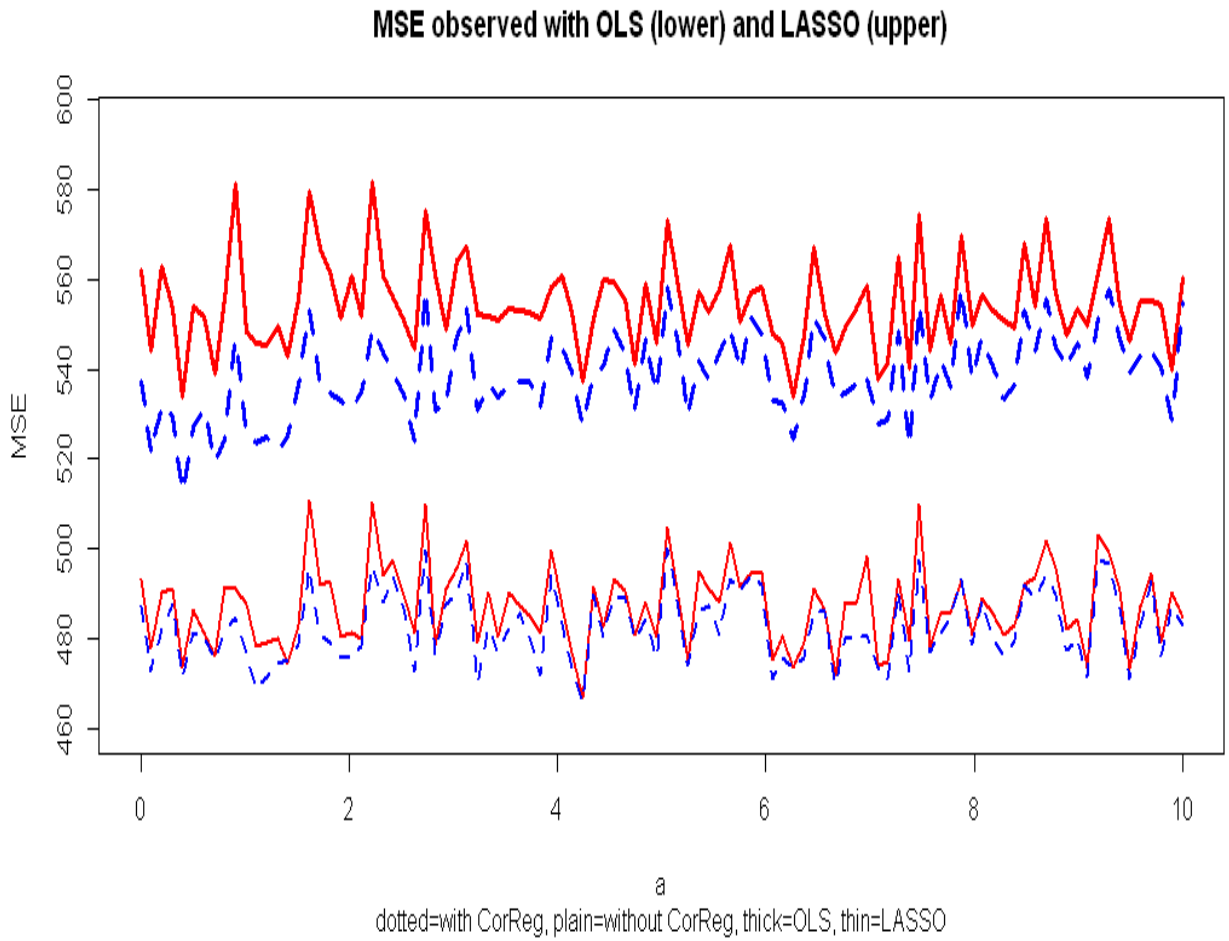
We define $\boldsymbol{X}^7 = a(\boldsymbol{X}^1)^2 + \boldsymbol{X}^2 + \boldsymbol{X}^3 + \varepsilon_1$. The matrix $\boldsymbol{X}$ is then scaled before doing

$$\boldsymbol{Y} = \sum_{i=1}^{7} \boldsymbol{X}^i + \varepsilon_Y.$$

We let a vary between 0 and 10 to increase progressively the non-linear part of the sub-regression. Once again, simulations has been made 100 times and the MSE were computed with 1 000 individuals validation samples.



(a) Evolution of the quality of $\hat{S}$ when the paramater a increases



(a) MSE on the main regression for OLS(thick) and LASSO (thin) used both with (plain) or without CorReg (dotted).

We have :

- a quality parameter (confidential) as response variable,
- $d = 205$ variables from the whole process to explain it.

We get a training set of $n = 3\,000$ products described by these 205 variables from the industrial process and also a validation sample of 847 products.
The objective here is not only to predict non-quality but to understand and then to avoid it.

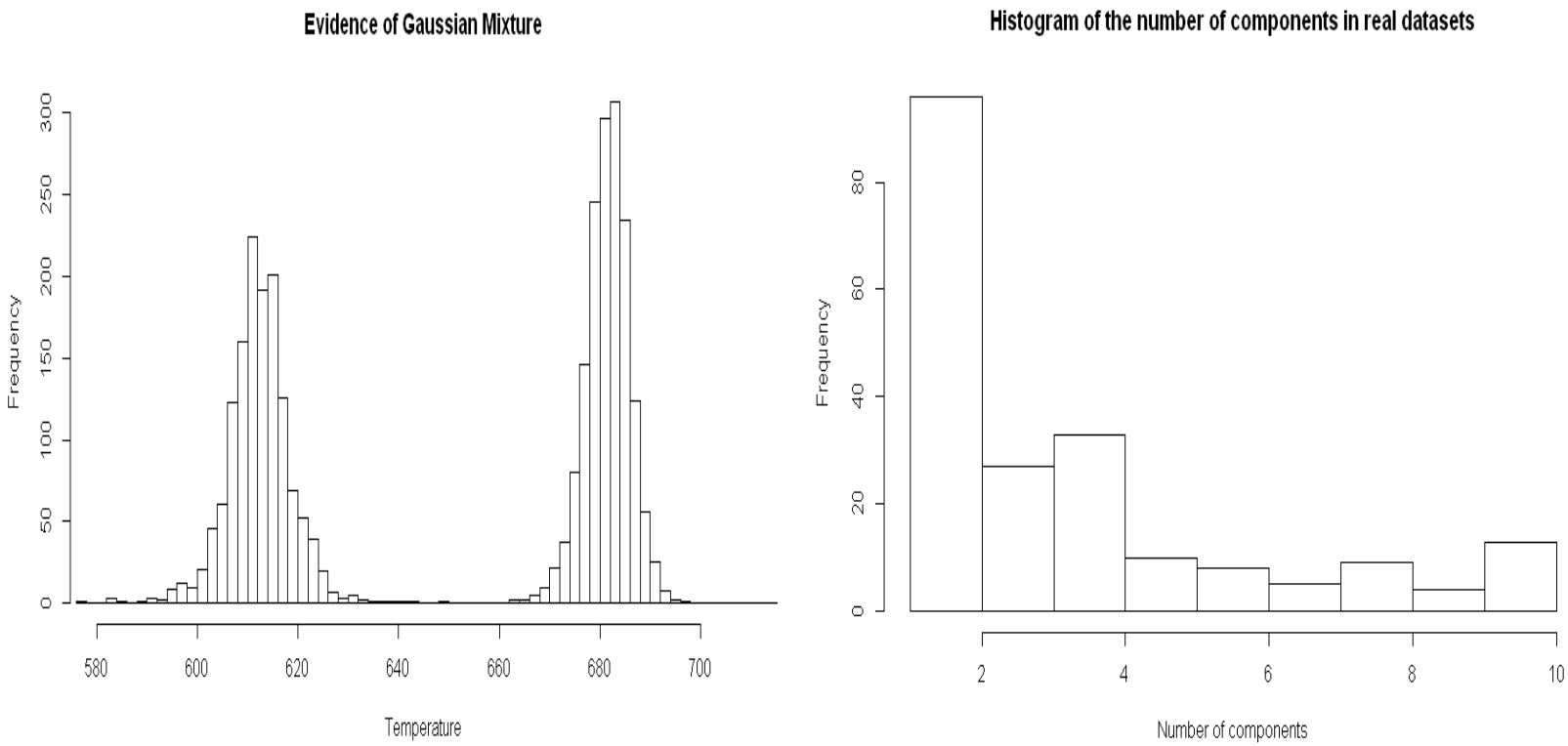


Figure: Quality case study: (left) Example of a non-Gaussian real variable easily modeled by a Gaussian mixture, (right) distribution of the number of components found for each covariate.

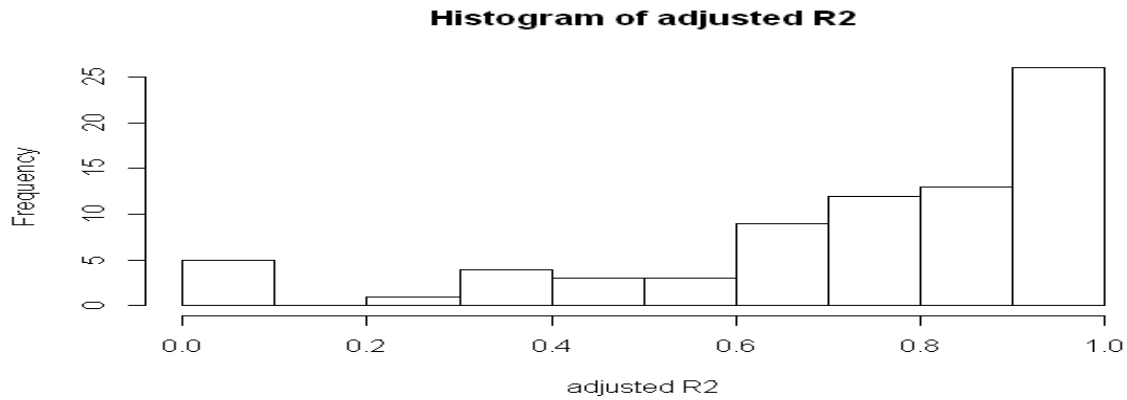


Figure: Quality case study: histogram of the adjusted R^2_{adj} for the $d_r = 76$ sub-regressions.

The width and the weight of a steel slab gives $|\rho| = 0.905$, the temperature before and after some tool gives $|\rho| = 0.983$, the roughness of both faces of the product gives $|\rho| = 0.919$ and a particular mean and a particular max gives $|\rho| = 0.911$.

Method	Indicator	CorReg's reduced model)	complete model
OLS	MSE	13.30	14.03
	complexity	130	206
LASSO	MSE	12.77	12.96
	complexity	24	21
elasticnet	MSE	**12.15**	13.52
	complexity	40	78
ridge	MSE	12.69	13.09
	complexity	130	206

Table: Quality case study: Results obtained on a validation sample ($n = 847$ individuals). In bold, the best MSE value.

Hope

Full generative model + explicit dependencies = missing values management

The number of component k can be huge (combinations of all the components of the covariates in $\boldsymbol{X}_f$). In fact we have

$$K = \prod_{j \in J_f} K_j$$

where K_j is the number of components of the Gaussian mixture followed by $\boldsymbol{X}_j$ as defined in Hypothesis 4. It is clear that K can really explode even if some components may be identical (but it happens with zero probability). For instance, if $\boldsymbol{X}$ contains only 10 independent Gaussian mixtures and with only 2 components each, then $\boldsymbol{X}$ will have up to $K = 2^{10} = 1\ 024$ components. And if these mixtures have 3 components each, then it rises up to $K = 59\ 049$ components.

Problem

- Too much components
- The likelihood may not be linear

EM is not possible in practice.

Stochastic EM

It is possible to simplify the problem by using Stochastic EM.
But the number of components stays a problem.

Solution

Gibbs Sampling for stochastic imputation in the Stochastic EM

We define

$$c_j = \frac{1}{n} \sum_{i=1}^{n} c_{i,j},$$

where $c_{i,j}$ is the number of parameters to estimate in $\mathbb{P}(x_{i,j}|\boldsymbol{X}_i \setminus \boldsymbol{X}_i^j)$. If all predictors $\boldsymbol{X}_i^{J_p^j}$ are observed then we have to estimate d_p^j coefficients and an intercept and the variance of the residual so $c_{i,j} = d_p^j + 2$. But for each missing predictor, we have to estimate the parameters of its distribution (Gaussian mixture with proportion, mean and variance to estimate for each component) so we have

$$c_{i,j} = d_p^j + 2 + \sum_{\substack{l \in J_p^j \\ M_{i,l}=1}} (3K_l - 1)$$

(the sum of the proportions is 1 so one estimation is useless for each mixture).

Results

Very slow and too much variability

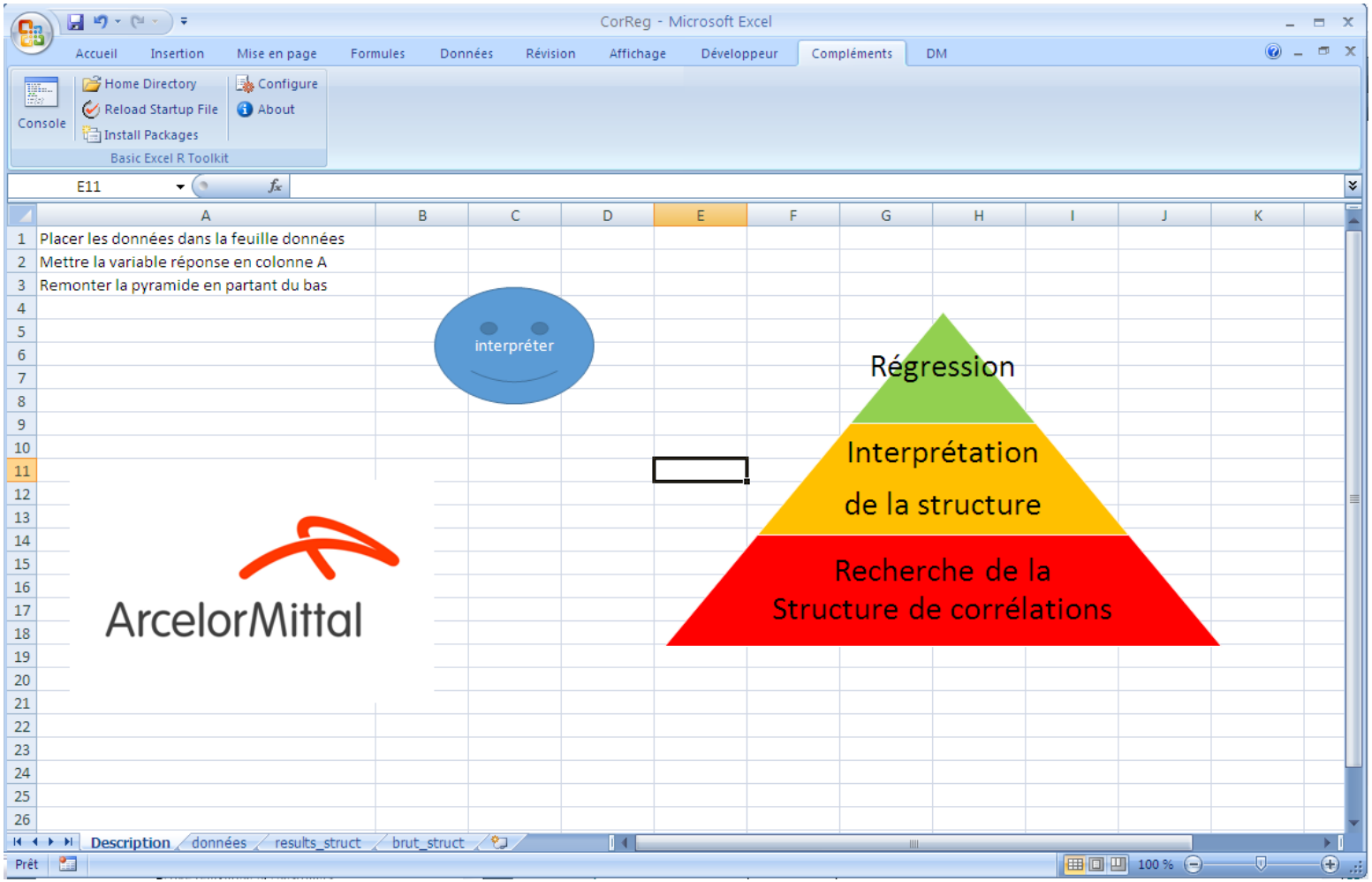


Figure: Screenshot of the Graphical User Interface of `CorReg` in Excel.

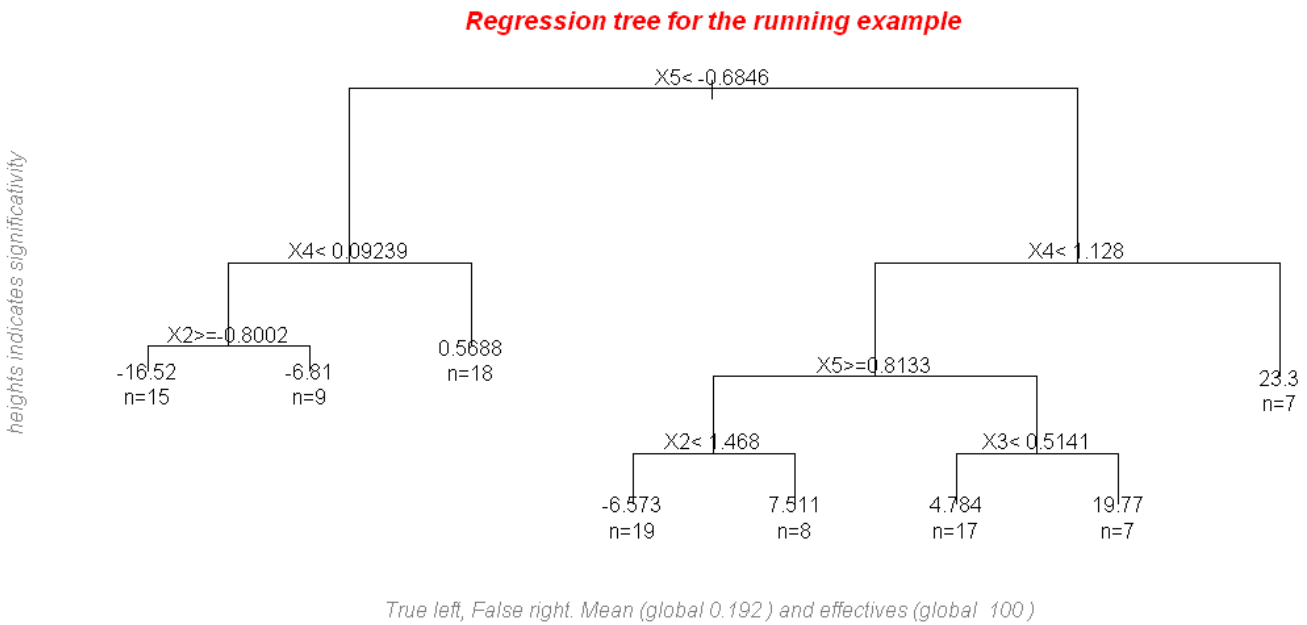


Figure: Regression tree obtain with the package `CorReg` (graphical layer on top of the `rpart` package) on the running example.

arbres de décision, showdata, Conan, *etc.*


Cule, E. (2014).
ridge: Ridge Regression with automatic selection of the penalty parameter.
R package version 2.1-3.


Cule, E. and De Iorio, M. (2013).
Ridge regression in prediction problems: automatic choice of the ridge parameter.
Genetic epidemiology, 37(7):704–714.


Dodge, Y. and Rousson, V. (2004).
Analyse de régression appliquée: manuel et exercices corrigés (coll. eco sup,).
Recherche, 67:02.


Er, M. J., Shao, Z., and Wang, N. (2013).
A systematic method to guide the choice of ridge parameter in ridge extreme learning machine

Merci pour ce moment...