UCL
Université
catholique
de Louvain

Institut de Statistique, Biostatistique et Sciences Actuarielles

Hypothesis testing for high-dimensional LASSO models.

Membres du jury:

Prof. Pircalabelu Eugen *Promoteur*
Prof. von Sachs Rainer *Lecteur*

Mémoire présenté en vue de
l'obtention du master
en sciences statistiques
(orientation générale)
par:

Léonard Lise

Louvain-La-Neuve
Mai 2022

Acknowledgments.

I would first like to express my gratitude to my promoter, Pircalabelu Eugen, who guided me through this master thesis. He was always available with interesting advices and a careful correction that allowed me to surpass myself and to end up with a document that I am proud of.

Then, I thank my parents who have always supported me unconditionally even without understanding a word of this work. Finally, my thanks go to my boyfriend for his emotional support which was not too much.

Contents

List of Figures

List of Tables

Chapter 1

Introduction.

In the last decades the amount of data has increased exponentially. To analyse them there exist numerous techniques depending on the model. The case of the linear regression is well known when the number of observations n, is smaller than the number of variables p. However, there are many situations where the number of variables exceeds the number of observations in the sample and the situation becomes much more challenging. For a simple example, in genetic studies the variables are often counted in the tens of thousands as there are numerous genes and the number of observations is limited by the means of the surveys. In these settings identifying the main variables is a crucial task.

The linear regression model is the following : for an observation i we have a response variable Y_i and several explanatory variables $\mathbf{X}_i = (X_{i,1}, ..., X_{i,p})^\top$. In order to predict the level of variable Y_i given the explanatory variables we construct a linear model

$$\mathbf{Y} = \mathbf{X}\,\beta + \epsilon, \tag{1.0.1}$$

where $\mathbf{Y} = (Y_1, ..., Y_n)^\top$ is a vector $n \times 1$ with the responses for all observations, $\mathbf{X}$ is a matrix of size $n \times p$ that contains the explanatory variables (including an intercept if it is relevant for the model) and ϵ is a vector $n \times 1$ containing the errors of the model. Usually, $\epsilon_1, \ldots, \epsilon_n$ are assumed to be identically distributed and independent, we note them $\epsilon \sim IID(0, \sigma^2)$ where σ^2 indicates the second moment of the distribution. The vector β, of size $p \times 1$, contains the coefficients for the explanatory variables and our goal for the problem is to estimate it, as it measures the link between the explanatory variables and the response variable.

The most famous method to estimate the coefficients $\beta_j, \quad j = 1, ...p$ is the Ordinary Least Squares method (OLS). The idea is to minimize the sum of the squared errors $\sum_{i=1}^n \epsilon_i^2 = ||\,\mathbf{Y} - \mathbf{X}\,\beta\,||_2^2$ to obtain the following estimator :

$$\hat{\beta}_{OLS}(\mathbf{X}, \mathbf{Y}) = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y}\,. \tag{1.0.2}$$

The Least Squares estimator requires that the matrix $\mathbf{X}^\top\,\mathbf{X}$ be invertible, a sufficient condition is that the $\mathbf{X}$ matrix has full rank. In this work, we are interested in the high-dimensional case where the number of variables exceeds the number of observations $n \ll p$. The main

issue is that under this setting, the matrix $\mathbf{X}^\top\mathbf{X}$ is no longer invertible, the rank of this $p \times p$ matrix is at most n which is inferior to p. Thus the Least Squares estimator does not exist for the problem.

An alternative method is the LASSO introduced by Tibshirani (1996). The principle of this estimator is to select only the most relevant variables in the model and to shrink the other coefficients to zero. It works under the sparsity assumption, which means there are only few variables that are relevant in the model. Then, the estimator is constructed by imposing a penalty on the ℓ_1 norm of the estimator.

$$\hat{\beta}_{LASSO}(\mathbf{Y}, \mathbf{X}, \lambda) = \text{argmin}_{\beta \in \mathbb{R}^p} \left\{ ||\mathbf{Y} - \mathbf{X}\beta||_2^2 + \lambda ||\beta||_1 \right\}. \tag{1.0.3}$$

The ℓ_1 norm is defined as $||\beta||_1 = \sum_{j=1}^{p} |\beta_j|$. The idea is to penalize, by a factor λ, the coefficients which are too small in absolute value so to minimize the expression they are shrunk to zero. When the context is clear, we will omit the matrix $\mathbf{X}$ and the vector $\mathbf{Y}$ in the notation.

There exist many other methods to estimate the coefficient vector in the case of high-dimensional models. Examples include the Ridge estimator which imposes an ℓ_2 penalty but the advantage of the LASSO is that it selects only few variables among all the p ones, since we said that the estimator is sparse. In the high-dimensional case, reducing the dimensions and identifying the relevant variables is a crucial task.

1.1 Issues with the LASSO distribution

The LASSO estimator is a very common and convenient method, however, it has some disadvantages. First, it is not unbiased. Indeed the introduction of the constraint induces a bias on the coefficients as they will be underestimated in absolute value. In Chapter 2, we develop this point to highlight the source of the bias, however this is not the really the challenging point of this estimator.

The major inconvenience of the LASSO is the inference part, usually we measure the uncertainty of our estimation with several tools such as confidence intervals or hypothesis testing. In the case of selecting variables in a high-dimensional setting, the inference allows to identify the variables that have no significant impact on the model. The LASSO was introduced in 1996 but it took more than 15 years to obtain some results on this subject.

Why did it takes such a long wait for inference results despite the increasing popularity of the estimator ? Where does the problem come from ?

There exist many methods to obtain the distribution or at least the asymptotic distribution of an estimator. For example, the bootstrap method performs very well, why does it not apply in LASSO's case ? Actually, the issue comes from the sparsity properties, the majority of the variables are estimated to zero and if we repeat the experience a large number of times, we do not obtain a continuous density for the estimation of these null variables. There is point mass at zero and without the assumption of continuous density, the traditional methods such as subsampling do not work.

Figure 1.1.1 illustrates this phenomenon, we simulate a regression model with $p = 100$ and $n = 100$, the number of non null variables is 5. On this histogram, we see the estimation of three non active variables repeated $R = 100$. On more than 80% of the repetitions, the LASSO estimates to zero the variables and some rare times, it returns another value. Clearly the underlying density of the estimator is not continuous and comes from an even less well-known law than the Normal.

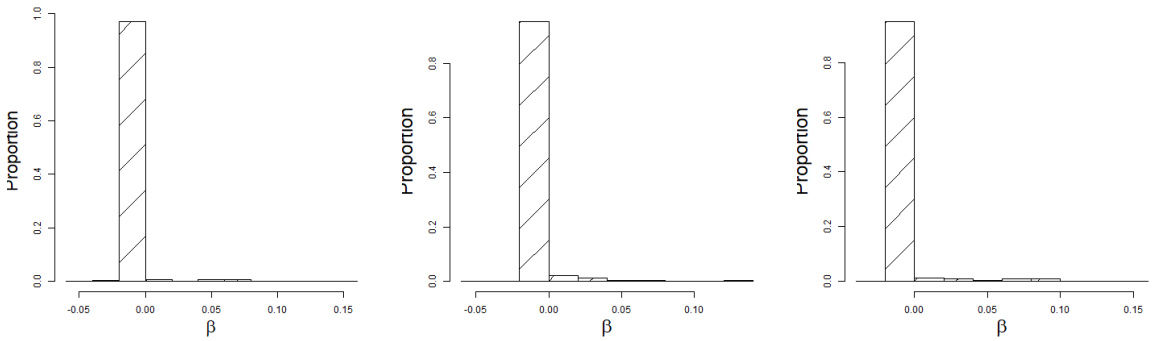

Figure 1.1.1: Histograms of the LASSO estimator for the $50th$, $51st$ and $52nd$ variables which are null. The number of variables is $p = 100$ for $n = 100$ observations, between each variables there is a correlation of $\rho = 0.1$, the number of repetitions for these histograms is $R = 100$. The density of the estimator is concentrated at zero with few outliers.

In 2014, Zhang and Zhang (2014) brings a long awaited solution to the problem. They find a new estimator derived from the LASSO which removes its bias but above all, they prove that its distribution converges to the Gaussian one. This article is quickly followed by the ones of Javanmard and Montanari (2014), van de Geer et al. (2014) which develop the idea of this 'de-biased' estimator. In this work, we investigate these methods to understand where the new estimator comes from, especially we explain how they get rid of the LASSO's issues and we compare the differences between the methods.

Part I

The de-biased LASSO estimator and properties

Chapter 2

The de-biased LASSO estimator.

The de-biased LASSO estimator (also called desparsified estimator) was first introduced by Zhang and Zhang (2014). Its purpose is the inference in high-dimensional linear models and to a lesser extent the accuracy of the estimation, in particular they focus on a unbiased estimator.

Equation 2.0.1 presents the final estimator for the problem 1.0.1. It starts from the LASSO estimator $\hat{\beta}_{LASSO}(\lambda)$ with a penalty factor λ and then the bias is removed by adding a term where a matrix M appears. The articles differ by the interpretation of this matrix but above all by the construction to obtain it.

$$\hat{\beta^d}(Y, \mathbf{X}, M, \lambda) = \hat{\beta}_{LASSO}(\lambda) + \frac{1}{n} M \mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\hat{\beta}_{LASSO}(\lambda)). \tag{2.0.1}$$

In addition to the previous goal, Zhang and Zhang (2014) look for a variable selection consistent estimator, which means their method will select the true underlying model including only the non-zero variables. On the other hand, they would likely obtain this condition even if the uniform signal strength condition were not respected. Indeed, this condition, that states that the non zero variables have a sufficiently high signal, is usually required for high-dimensional models.

In their article, the idea is to start from the LASSO estimator and to remove the bias using a relaxed projection of $X_j = (X_{1,j}, \ldots, X_{n,j})^\top$, ie. the jth columns in $\mathbf{X}$, on the orthogonal complement of the column space of $\mathbf{X}_{-j} = \{X_k | k \neq j\}$. Actually, the LASSO's bias comes from the correlation between each variables X_j and due to the high dimension of the regression, there are always some variables correlated with others in the sample. When p is greater than n the projection does not exist any more and to obtain the relaxation they used the regressions of X_j on $\mathbf{X}_{-j}$ for each variables j.

Afterwards, van de Geer et al. (2014) presents similar results using another idea to get rid of this problematic bias, they invert the Karush-Kuhn-Tucker conditions for the minimization problem of the LASSO and instead of using a relaxed projection matrix, they propose to use the inverse of $\frac{1}{n}\mathbf{X}^\top \mathbf{X}$. They introduced a pseudo inverse that is constructed by node-wise LASSO (ie. for all variables $j \in \{1, ..., p\}$ they regress the variable X_j on all the other variables $\mathbf{X}_{-j}$).

At the same time, Javanmard and Montanari (2014) explore another approach but the result is still close to the two others. To remove the bias of the LASSO, they start from the same construction but instead of looking for an estimator of the inverse of $\frac{1}{n}\mathbf{X}^\top \mathbf{X}$ or a projection, they remark that they only need a matrix that is not too far away from this inverse. They then use an optimization problem to construct such a matrix but also with the aim of controlling the variance of the final distribution.

In this chapter, we explain in detail where the estimator comes from and how the matrix M is constructed and interpreted for each method.

2.1 Bias of the LASSO estimator

First of all, we inspect the bias of the LASSO estimator because the desparsified methods arrive with this idea of correcting the bias.

Intuitively this bias comes from the penalty, the weaker coefficients are set to zero but also the strongest ones are shrunk slightly. Moreover, the bias is linked to the correlation of the variables, in a high-dimensional case when the number of observations is fixed, the correlation in the sample cannot be reduced to zero. As a result, the LASSO estimator underestimates the coefficients in terms of absolute value for high signal variables and if a covariate has a too weak signal, one can miss it. That is where the uniform signal strength condition acts.

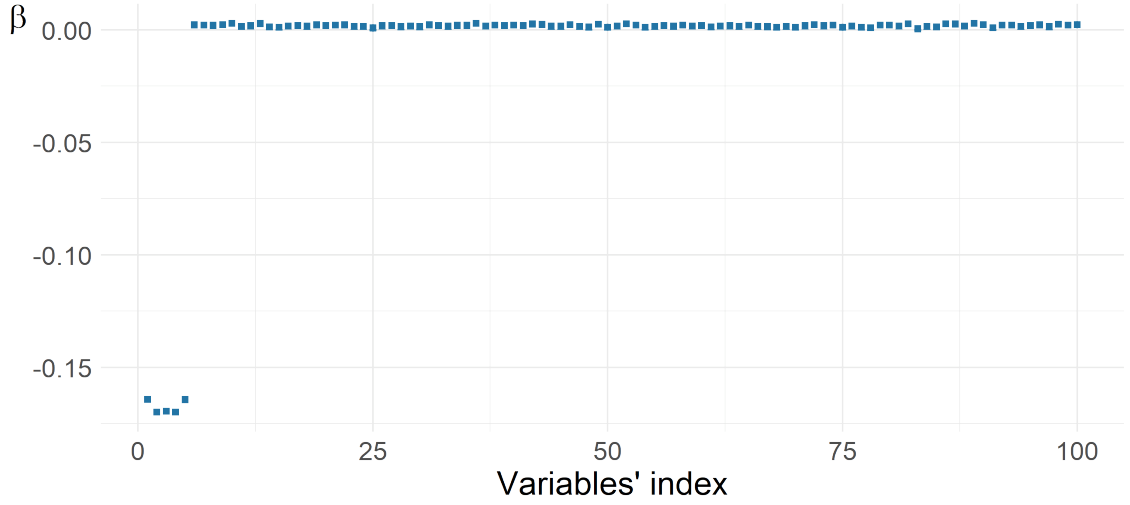

Figure 2.1.1: Bias of the LASSO when $n = 100$ and $p = 100$, with equal correlation between each variables of $\rho = 0.1$. The number of replications is $R = 100$. Only the first 5 coefficients are different from zero and we observe a negative bias for them.

This bias reduces the accuracy of the method and some effects can be totally skipped while they are important for the model. We can observe this bias empirically : Figure 2.1.1 shows the mean ones $R = 100$ repetitions of the LASSO estimator bias. The number of observations is $n = 100$, the number of variables $p = 100$ with only the first five values of β that are different from zero. The coefficients are $\beta = (2, 2, 2, 2, 2, 0, \ldots, 0)^\top$ and the variables have all the same correlation of $\rho = 0.1$. The penalty parameter λ is chosen as the one which minimizes

the cross-validation error. We clearly observe on the figure that, in expectation, the LASSO estimator does not reach the true value of the coefficients when they are different from zero. For the active variables, the bias is negative which illustrates the principle of underestimated coefficients. On average, the bias is 0.002 for the null variables and -0.168 for the non zero variables.

The bias can also be derived directly from the theory. Starting from the Ordinary Least Squares estimator, we have :

$$\begin{aligned}\hat{\beta}_{OLS} &= (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{Y} \\ &= \frac{1}{n}\Sigma^{-1}\mathbf{X}^\top \mathbf{Y} \quad \text{with} \quad \Sigma = \frac{1}{n}\mathbf{X}^\top \mathbf{X}\,.\end{aligned}$$

When $n \ll p$, the matrix Σ^{-1} cannot be obtained because the rank of the matrix $\mathbf{X}$ is majored by n while the number of columns is p. However if we find a suitable $p \times p$ matrix M to replace this quantity, we still have an estimator of the following form :

$$\hat{\beta} = \frac{1}{n}M\mathbf{X}^\top \mathbf{Y}\,.$$

Then we observe that we can rewrite this estimator as follows

$$\begin{aligned}\hat{\beta} &= \frac{1}{n}M\mathbf{X}^\top \mathbf{Y} \\ &= \frac{1}{n}M\mathbf{X}^\top(\mathbf{X}\,\beta - \epsilon) \\ &= \frac{1}{n}M\mathbf{X}^\top \mathbf{X}\,\beta - \frac{1}{n}M\mathbf{X}^\top \epsilon \\ &= \beta - \beta + M\Sigma\,\beta - \frac{1}{n}M\mathbf{X}^\top \epsilon \\ \hat{\beta} - \beta &= (M\Sigma - I_p)\beta + \frac{1}{n}M\mathbf{X}^\top\epsilon,\end{aligned}$$

where β stands for the true parameter, ϵ is the noise of the regression and I_p is the $p \times p$ identity matrix. The discussed methods are all based on this decomposition, indeed we have expressed here the error of the estimator $\hat{\beta}$ in two terms. The first is a bias, it depends on the matrix M and how close it is to Σ^{-1}. The last term is the most interesting, if the noise ϵ follows a Gaussian distribution, it will also be normally-distributed. From this expression we

can now construct the de-biased estimator :

$$\begin{aligned}
\hat{\beta}^d &= \hat{\beta} - (M\Sigma - I)\hat{\beta}_{init} \\
&= \frac{1}{n} M\mathbf{X}^\top \mathbf{Y} - (M\Sigma - I)\hat{\beta}_{init} \\
&= \frac{1}{n} M\mathbf{X}^\top \mathbf{Y} - M\Sigma\hat{\beta}_{init} + \hat{\beta}_{init} \\
&= \frac{1}{n} M\mathbf{X}^\top \mathbf{Y} - \frac{1}{n} M\mathbf{X}^\top \mathbf{X}\,\hat{\beta}_{init} + \hat{\beta}_{init} \\
&= \hat{\beta}_{init} + \frac{1}{n} M\mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\,\hat{\beta}_{init}) \\
&= \hat{\beta}_{init} + \beta - \beta + \frac{1}{n} M\mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\,\hat{\beta}_{init} - \mathbf{X}\,\beta + \mathbf{X}\,\beta) \\
&= \beta + (\hat{\beta}_{init} - \beta) + \frac{1}{n} M\mathbf{X}^\top (\epsilon - \mathbf{X}\,\hat{\beta}_{init} + \mathbf{X}\,\beta) \\
&= \beta + I_p(\hat{\beta}_{init} - \beta) + \frac{1}{n} M\mathbf{X}^\top (\epsilon - \mathbf{X}(\hat{\beta}_{init} - \beta)) \\
&= \beta + \frac{1}{n} M\mathbf{X}^\top \epsilon + (I_p - M\Sigma)(\hat{\beta}_{init} - \beta).
\end{aligned} \tag{2.1.1}$$

The last expression implies that if the estimator $\hat{\beta}_{init}$ is sufficiently close to β and if M is a good approximation to Σ^{-1}, the de-biased estimator $\hat{\beta}^d$ will be asymptotically unbiased. Moreover, the second term is asymptotically Gaussian since $\epsilon \sim \mathcal{N}(0, \sigma^2 I_n)$ which means that the de-biased estimator converges to a Normal distribution. In this work, $\hat{\beta}_{init}$ is chosen as the LASSO estimator or one of its variations.

In this decomposition, we find the expression 2.0.1 presented above for the estimator :

$$\hat{\beta^d} = \hat{\beta}_{LASSO} + \frac{1}{n} M\mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\hat{\beta}_{LASSO}).$$

We have obtained an asymptotically unbiased estimator and above all, we know its asymptotic distribution. Not only is the bias problem solved, but the challenge of finding a distribution for this estimator is overcome. The question that remains is how to optimally choose the matrix M. Indeed we need a matrix constructed from $\mathbf{X}$ which is sufficiently close to Σ^{-1} to guarantee the properties of the de-biased estimator but the matrix is not invertible.

2.2 Construction of the estimator

All the articles propose a de-biased estimator of the form 2.0.1, the differences between the methods come from the choice of the matrix M. The method of Zhang and Zhang (2014) extended by van de Geer et al. (2014) uses residuals of nodewise LASSO and Javanmard and Montanari (2014) choose another method : they select M by solving a convex optimization in order to obtain at the same time a small bias and a small variance of the estimator $\hat{\beta}^d$.

M by nodewise LASSO

In this section, we present the method to obtain the matrix M by nodewise LASSO. First, we explain the methodology of Zhang and Zhang (2014) who introduced the idea, before making

the link with the one of van de Geer et al. (2014). The two reach the same estimator but the idea behind differs a bit.

In the case where $p < n$ the j-th component of the Ordinary Least Squares estimator of β can be written as

$$\hat{\beta}_j = \frac{X_j^{\perp} Y}{(X_j^{\perp})^{\top} X_j}, \tag{2.2.1}$$

where $X_j^{\perp}$ is the projection of X_j, the j-th column of the matrix $\mathbf{X}$, on the orthogonal complement of the column space $\mathbf{X}_{-j}$. As the authors explain their method in a fixed case, we also work in this context however the idea is unchanged in a random setting. The estimator 2.2.1 can be written under the following formula by replacing Y with $\epsilon - \mathbf{X}\,\beta$ and by remarking that $X_j^{\perp} X_k = 0, \quad \forall k \neq j$ by construction :

$$\hat{\beta}_j = \beta_j + \frac{(X_j^{\perp})^{\top} \epsilon}{(X_j^{\perp})^{\top} X_j}.$$

As the expectation of ϵ is null, the estimator is unbiased. We recognize the final form of the decomposition 2.1.1 presented in the previous section but with zero in place of the third term : $\hat{\beta}_j = \beta + \frac{1}{n}(M\,\mathbf{X}^{\top})_j \epsilon + 0$, where $\frac{(X_j^{\perp})^{\top}}{(X_j^{\perp})^{\top} X_j}$ is equivalent to the j-th row of the matrix $M\,\mathbf{X}^{\top}$.

However, when $p > n$, the vector $X_j^{\perp}$ is null for every j and the estimator is not defined anymore. Zhang and Zhang (2014) has the idea to choose another vector Z_j to replace $X_j^{\perp}$. The estimator 2.2.1 becomes :

$$\hat{\beta}_j = \frac{Z_j^{\top} Y}{Z_j^{\top} X_j}.$$

In this case, we make the coefficient β_j appear again but this time, we cannot get rid of the bias term. Due to the relaxation of Z_j, we do not have the property that $Z_j^{\top} X_k = 0 \quad \forall k \neq j$ any longer. We can rewrite $\hat{\beta}_j$ as

$$\hat{\beta}_j = \beta_j + \frac{Z_j^{\top} \epsilon}{Z_j^{\top} X_j} + \sum_{k \neq j}^{p} \frac{Z_j^{\top} X_k}{Z_j^{\top} X_j} \beta_k.$$

We observe next that the bias depends on the unknown quantity β but that the Z_j and X_k are known for all k since they come from the matrix $\mathbf{X}$. The idea is to use an initial estimator, in our case the LASSO estimator, to obtain an unbiased estimator for β :

$$\hat{\beta}_j^d = \hat{\beta}_{LASSO,j} + \frac{z_j^{\top}}{z_j^{\top} X_j}(\mathbf{Y} - \mathbf{X}\,\hat{\beta}_{LASSO}).$$

The link can be made with Equation 2.0.1, as the term $\frac{Z_j^{\top}}{Z_j^{\top} X_j}$ corresponds to the j-th component of $M\,\mathbf{X}$. There remains to efficiently construct the relaxed projection $Z_j, \quad \forall j \in \{1, ..., p\}$ or equivalently, the matrix M.

Zhang and Zhang (2014) propose to use the residuals of node-wise regression to construct the vectors Z_j for $j \in \{1, ..., p\}$. They regress the column X_j on $\mathbf{X}_{-j}$ to obtain, $\forall j \in \{1, ..., p\}$:

$$\hat{\gamma}_j = \text{argmin}_\gamma \frac{||X_j - \mathbf{X}_{-j}\gamma||_2^2}{n} + \lambda_j ||\gamma||_1, \tag{2.2.2}$$

$$Z_j = X_j - \mathbf{X}_{-j}\hat{\gamma}_j. \tag{2.2.3}$$

van de Geer et al. (2014) used another construction to obtain the matrix M but the result is exactly the same for linear models. Their construction will allow the generalization for the GLMs.

The vector $\hat{\gamma}_j$ has length $p-1$ and let us denote its components by $\hat{\gamma}_j = (\hat{\gamma}_{k,1}, ..., \hat{\gamma}_{j,p})^\top$ for $k = 1, ..., p, k \neq j$. From it, they construct a $p \times p$ matrix $\hat{C}$:

$$\hat{C} = \begin{pmatrix} 1 & -\hat{\gamma}_{1,2} & ... & -\hat{\gamma}_{1,p} \\ -\hat{\gamma}_{2,1} & 1 & ... & -\hat{\gamma}_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ -\hat{\gamma}_{p,1} & -\hat{\gamma}_{p,2} & ... & 1 \end{pmatrix}. \tag{2.2.4}$$

They use again the coefficients $\hat{\gamma}_j$ to construct some residuals $\hat{\tau}_j^2 = \frac{1}{n}||X_j - \mathbf{X}_{-j}\,\hat{\gamma}||_2^2 + \lambda_j||\hat{\gamma}||_1$ and the matrix $\hat{T}^2 = \text{diag}(\hat{\tau_1}^2, ..., \hat{\tau_p}^2)$. The matrix M is eventually given by :

$$M = \hat{T}^{-2}\hat{C}. \tag{2.2.5}$$

To be convinced that the two constructions give the same estimator, we remark that the j-th row of the M matrix is $M_j = \hat{C}_j/\hat{\tau}_j^2$ and by the KKT conditions of the node-wise LASSO $\hat{\tau}_j^2 = (X_j - \mathbf{X}_{-j}\hat{\gamma}_j)^\top X_j/n$ We now have an equivalence with the notation of the first article : $(X_j - \mathbf{X}_{-j}\hat{\gamma}_j)^\top X_j/n = Z_j^\top X_j/n$.

The differences from these two constructions are not the result but the way in which the problem is approached. Zhang and Zhang (2014) has a geometric vision and a component-by-component approach while van de Geer et al. (2014) looks directly to invert the variance-covariance matrix.

M by convex minimization

Javanmard and Montanari (2014) also starts with an estimator given by 2.0.1 but they take a completely different perspective, in contrast to the previous method, they do not look for an inverse of Σ. They focus on the expected properties of the estimator and how to guarantee them.

The bias of the de-biased LASSO estimator depends on $||M\Sigma - I_p||_\infty$ thus it is sufficient to construct a matrix M that controls this quantity instead of obtaining a good estimator for the inverse of Σ. We will show in Chapter 3 that the variance of the de-biased LASSO estimator is

given by $\sigma^2(M\Sigma M^T)/n$, where σ^2 is the noise level. Thus, the matrix M constructed in order to have a minimal variance for the de-biased estimator while controlling the term $||M\Sigma - I_p||_\infty$ will give the best properties. Algorithm 1 summarizes the implemented idea.

Algorithm 1 de-biased LASSO estimator $\hat{\beta}^d$

Require: $y, \mathbf{X}, \lambda, \mu$.

1: Compute $\hat{\beta}_{LASSO}(\lambda)$ solution of 1.0.3.
2: Compute $\Sigma = (\mathbf{X}^\top \mathbf{X})/n$.
3: **for** $j = 1, ..., p$ **do**
4: Let m_j be the solution of the convex program :

$$\begin{aligned} &\text{minimize} \quad m^\top \Sigma m \\ &\text{subject to} \quad ||\Sigma m - e_j||_\infty \ll \mu, \end{aligned} \tag{2.2.6}$$

where $e_j \in \mathbb{R}^p \quad \forall j = 1, ..., p$ are the vector with one at the j-th position and zero everywhere else.

5: **end for**
6: Construct $M = (m_1, ..., m_p)^\top$ and if none of the problems are feasible, $M = I_p$.
7: Define and return the de-biased estimator :

$$\hat{\beta}^d = \hat{\beta}_{LASSO}(\lambda) + \frac{1}{n} M \mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\,\hat{\beta}_{LASSO}(\lambda)).$$

As input, the algorithm takes the design matrix $\mathbf{X}$, the response vector $\mathbf{Y}$, the penalized parameter λ for the LASSO estimator and also a scalar μ which is the cap for the matrix M. Usually, $\mu = a\sqrt{\frac{\log p}{n}}$ with $a > 0$.

The first step is the calculation of the LASSO estimator with the given penalized parameter λ and the calculation of the variance-covariance matrix Σ. Then the matrix M is constructed column by column solving independently the convex program 2.2.6. This program tries to directly minimize the variance of the final estimator keeping the bias under control by the parameter μ. Finally, the algorithm returns the de-biased estimator.

In case where Σ is the identity, M is a diagonal matrix with the same value for each rows. There the minimization becomes :

$$\begin{aligned} &\text{minimize} \quad ||m||_2^2 \\ &\text{subject to} \quad ||m - e_j||_\infty \ll \mu. \end{aligned} \tag{2.2.7}$$

As such the values on the diagonal of M will be slightly lower than 1, depending on the μ level and off diagonal, the entries are zeros. So we have something of the form $c\Sigma$ and that is the first idea presented by the authors in Javanmard and Montanari (2013) where they attacked the problem precisely in the case the variance covariance matrix is the identity.

2.3 Remarks

The de-biased estimator is also called desparsified estimator because by removing the bias, the estimator lost its sparsity. Contrary to the LASSO estimator, which is a sparse vector, for the de-biased methods, no coefficient is equal to exactly zero. This characteristic of estimators implies in particular that the bias on the null variables will be higher than for the LASSO estimator.

Figure 2.3.1 illustrates this phenomenon. We observe the bias on $R = 100$ repetitions of the LASSO estimator, the desparsified estimator from van de Geer et al. (2014), we call it *Nodewise de-biased LASSO* (ND LASSO) in reference to the construction's technique, and the bias of the desparsified estimator from Javanmard and Montanari (2014) called *Optimized de-biased LASSO* (OD LASSO) for the same reason. The design is the same as for the example in the beginning of the chapter, the number of variables is $p = 100$ for $n = 100$ observations. The correlation between each variables is 0.1 and only the first 5 entries of the vector are different from zero and they are fixed to 2, so we are in a case of high sparsity.

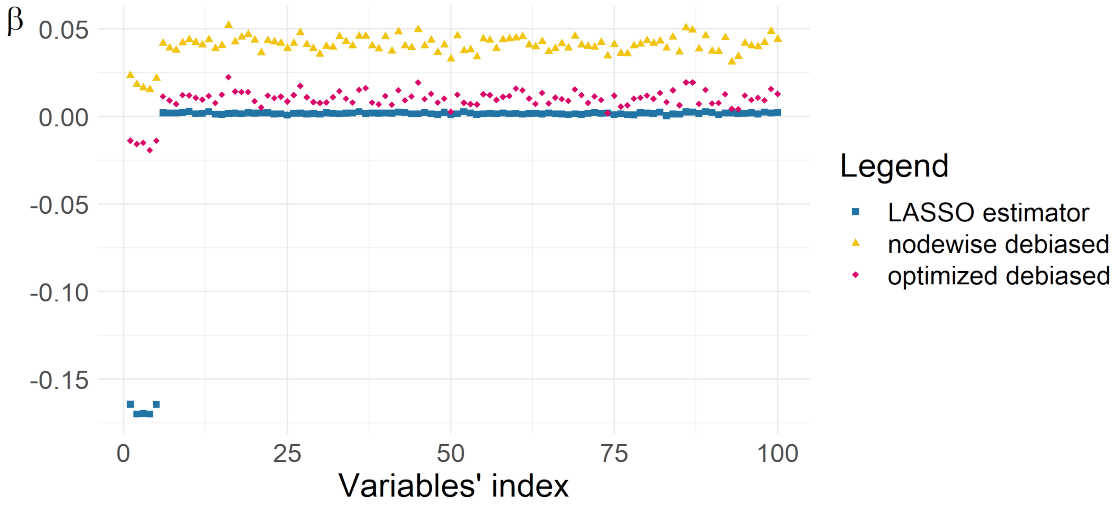

Figure 2.3.1: Bias of the LASSO and the two de-biased methods when $n = 100$ and $p = 100$, with equal correlation between each variables of $\rho = 0.1$. The number of replications is $R = 100$. Only the first 5 coefficients are different from zero.

We observe that for the LASSO estimator, there is an important bias for the non zero variables but for the other ones, the bias is almost null. Clearly, the bias for the two de-biased estimators is on another scale and there is not much difference between the non zero variables and the zero variables. On the null variable there is still a remaining bias that was not in the LASSO estimator. If we compare the two desparsified estimators, we remark that the OD estimator presents smaller bias than the ND estimator for the null variables.

This desparsification of the estimator also resolves the distribution problem, indeed sparsity in the LASSO estimator leads to a continuous density with point mass at zero but here, for the desparsified method, we do not have this behaviour anymore and then, we can conduct inference on the coefficient. Figure 2.3.2 illustrates the confidence intervals constructed from the assumed Gaussian distribution of the estimators. The validity of these intervals will be

explored in the next section.

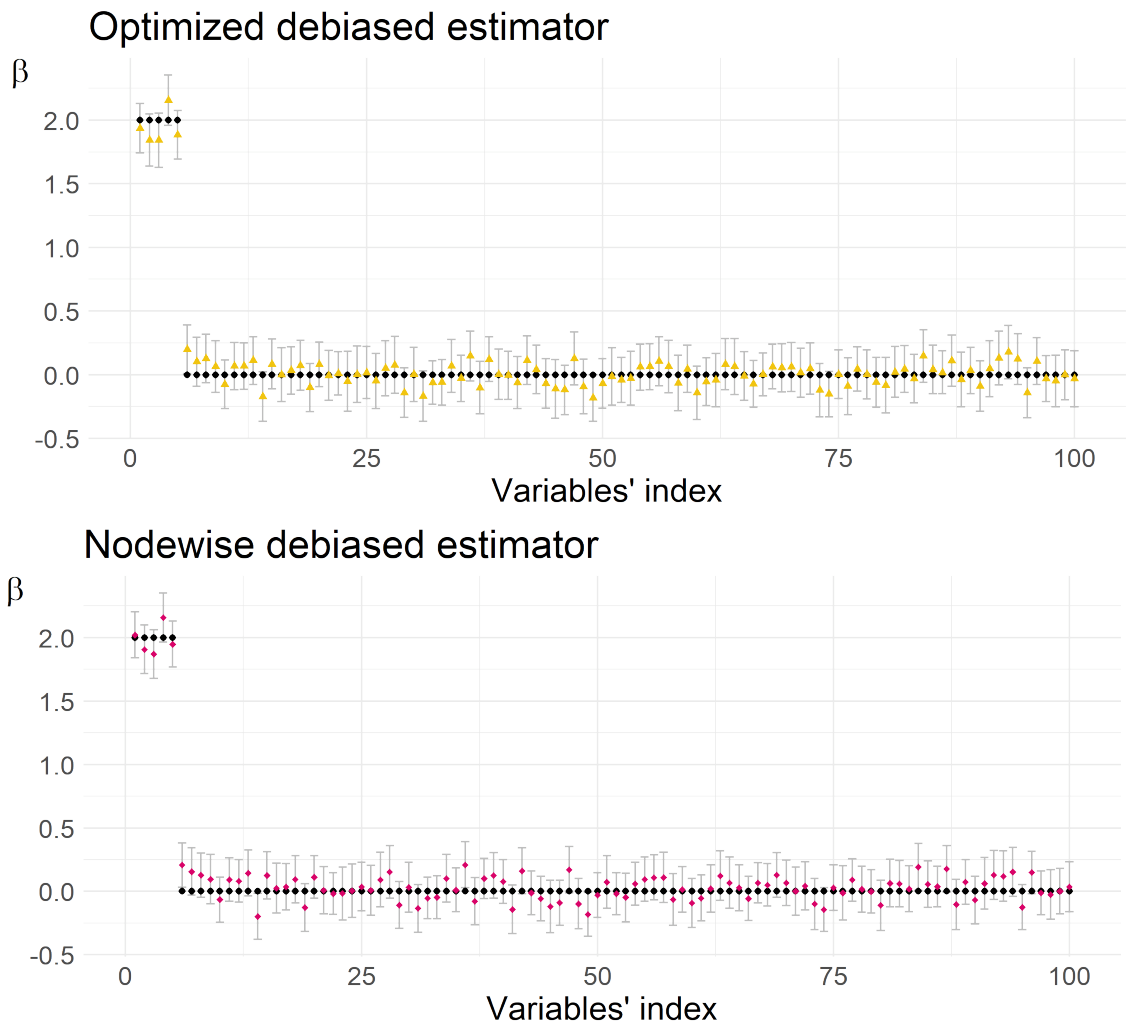

Figure 2.3.2: Confidence intervals for each variables when $n = 100$ and $p = 100$, with equal correlation between each variables of $\rho = 0.1$. Above in yellow : OD LASSO estimator and below in red : ND LASSO estimator, the true parameters are in black.

First, we observe again the non sparsity of the estimators, no variable is exactly equal to zero. Inspecting the confidence interval, all the true values for non zero coefficients fall in it. For the non active variables, a large majority of the true parameters are in the intervals. By a visual analysis, we cannot observe a real difference between the methods, the confidence interval does not seem wider for one method than for the other. We inspect in detail the performance of the methods on several numerical simulations later in this work.

These methods seem quite useful however they have some disadvantages. One important issue is the complexity of the problems, the node-wise regression requires to compute p different LASSO regressions before arranging the vectors to obtain the matrix M. When p is large, and the purpose of the method is to work with a lot of variables, obtaining the matrix M takes lot of time, even with fast algorithms for the LASSO regressions. For the second method, the

issue is the same. The optimization problem needs to be solved for each of the p variables and its complexity is equivalent to a LASSO regression, especially when the matrix Σ is far from the identity.

Chapter 3

Theoretical Results.

In the three articles the authors prove that, under some conditions, the desparsified estimator $\hat{\beta}^d$ is asymptotically Gaussian with mean β and this important result allows the long awaited construction of confidence intervals for hypothesis testing.

The main conditions behind the results is the high sparsity of the model, that means the number of non zero entries in β, $s = |S| = |\{j : \beta_j \neq 0, j = 1, ..., p\}|$, is not too large relatively to n. The other condition is about the matrix M, it requires that it is sufficiently close to its target, the inverse of population covariance. In the three articles, they found that an appropriate level for the sparsity is of order $\frac{\sqrt{n}}{\log p}$, this is a quite strong condition since the classical LASSO only requires order $\frac{n}{\log p}$ to work.

In the next sections, we inspect the results of the desparsified LASSO estimator and compare the method of van de Geer et al. (2014) and the one of Javanmard and Montanari (2014). Then, we discuss the case where the noise level has to be estimated and the case where this noise does not follow a Gaussian distribution as assumed up to now.

3.1 Main results

In the previous chapter, we showed in 2.1.1 that the bias of the de-sparsified estimator can be decomposed into two terms. The first one is asymptotically normal if the noise is Gaussian and the second can be seen as a bias. The next results present bounds on this error term if the matrix M is carefully chosen.

First, we need to introduce some definitions and notations.

Definition 3.1.1. Given a matrix $\mathbf{X}$ and a support set S with cardinality $s = |S|$, the corresponding compatibility constant is defined as

$$\phi^2(\Sigma, S) = \min_{b \in \mathbb{R}^p} \left\{ s \frac{b^\top \Sigma b}{||b_S||_1^2} : ||b_{S^c}||_1 \leq 3||b_S||_1 \right\},$$

where b_A is the vector made up from the entries b_j with $j \in A$ and S^c is the complementary set of S. We say that Σ satisfies the compatibility condition for S with constant ϕ_0 if $\phi(\Sigma, S) \geq \phi_0$.

We say that it holds for the design matrix $\mathbf{X}$, if it holds for $\Sigma = \mathbf{X}^\top \mathbf{X}/n$.

Definition 3.1.2. The generalized coherence parameter of $\mathbf{X}$ and M, is

$$\mu_*(\mathbf{X}, M) = ||M\Sigma - I_p||_\infty,$$

where for a $p \times p$ matrix A, $||A||_\infty = \max_{1 \le l \le p} \sum_{j=1}^{p} |a_{l,j}|$ is the maximum of the absolute sum of a row of the matrix.

The compatibility condition ensures that the initial estimator $\hat{\beta}_{LASSO}$ converges to β in ℓ_1 norm while the second quantity was introduced by Javanmard and Montanari (2014). In their algorithm they construct M such that the generalized coherence parameter is smaller than a certain fixed level μ.

An important result to remember is the Theorem 6.1 from Bühlmann and van de Geer (2011). As it presents the conditions under which the LASSO estimator converges to the true parameter β and it will be largely used to show the good properties of the desparsified estimator.

Theorem 3.1.3 (Bühlmann and van de Geer (2011), Theorem 6.1)**.** *Suppose the compatibility condition holds for* $\mathbf{X}$ *and* S *with constant* ϕ_0*. Let* $t > 0$ *be arbitrary. Then for* $\lambda \ge 4\sigma\sqrt{(t^2 + \log p)/n}$*, with probability at least* $1 - 2e^{-t^2}$*,*

$$|| \mathbf{X}(\hat{\beta}_{LASSO} - \beta)||_2^2/n + \lambda||\hat{\beta}_{LASSO} - \beta\,||_1 \le 4\lambda^2 \frac{s}{\phi_0^2}$$

The proof of this result is not addressed in this work but it is detailed in the book of the authors.

3.1.1 Fixed Design

We first treat the case where the matrix $\mathbf{X}$ is non-random and fixed. The condition (A1) on the design matrix $\mathbf{X}$ is sufficient to obtain the desired results.

(A1) The compatibility condition holds for $\mathbf{X}$ with compatibility constant ϕ_0. Moreover, assume $\max_j(\Sigma)_{j,j} \le K^2$ for a constant $0 < K < \infty$.

The condition is not surprising, the compatibility condition is necessary for the convergence of the LASSO estimator and the second part of the condition ensures just that the matrix Σ has bounded entries, it is quite intuitive, if the entries of the matrix explode, approaching its inverse will be a very complicated task.

The following theorems bound in probability the bias term of the desparsified LASSO estimator. The first theorem comes from van de Geer et al. (2014) and the bound depends on the penalization parameters λ_j used in the ND LASSO for the construction of the matrix M. The second theorem, from Javanmard and Montanari (2014), presents a bound that depends on the generalized coherence parameter they have introduced. By their construction of M, this bound guarantees the convergence of the estimator.

Theorem 3.1.4 (van de Geer et al. (2014), Theorem 2.1)**.** *Consider the linear model in 1.0.1 with Gaussian error* $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ *and assume condition (A1). Let* $t > 0$ *be arbitrary.*

When using the LASSO in 1.0.3 with $\lambda \geq 2K\sigma\sqrt{2(t^2+\log p)/n}$ and the LASSO for node-wise regression in 2.2.5 we have

$$\begin{aligned}\sqrt{n}(\hat{\beta}^d - \beta) &= W + \Delta,\\ W = \frac{M\,\mathbf{X}^\top \epsilon}{\sqrt{n}} &\sim \mathcal{N}(0, \sigma^2 M\Sigma M^\top),\\ \mathbb{P}\left[||\Delta||_\infty \geq 8\sqrt{n}\frac{\lambda s}{\phi_0}\max_j \frac{\lambda_j}{\hat{\tau}_j^2}\right] &\leq 2e^{-t^2}.\end{aligned}$$

Proof. The decomposition of $\sqrt{n}(\hat{\beta}^d - \beta)$ was already presented in Chapter 2, the error term Δ is equal to $\sqrt{n}(M\Sigma - I)(\hat{\beta}_{LASSO} - \beta)$. It is clear that $W = \frac{M\,\mathbf{X}^\top \epsilon}{\sqrt{n}}$ follows a Gaussian distribution since the only random term is ϵ which is Gaussian by assumption. There remains to bound the error term. Under condition (A1), the LASSO estimator is close to its target in ℓ_1 norm. One can show, based on Theorem 3.1.3, that, with probability at least $1 - 2e^{-t^2}$,

$$\begin{aligned}||\hat{\beta}_{LASSO} - \beta\,||_1 &\leq 8\lambda\frac{s}{\phi_0^2},\\ ||\,\mathbf{X}(\hat{\beta}_{LASSO} - \beta)||_2^2 &\leq 8n\lambda^2\frac{s}{\phi_0^2}.\end{aligned}$$

On the other hand, by the KKT conditions applied to node-wise LASSO, for all $j \in \{1, ..., p\}$ we have

$$\begin{aligned}-\mathbf{X}_{-j}^\top(X_j - \mathbf{X}_{-j}\hat{\gamma}_j)/n + \lambda_j\hat{\nu} &= 0,\\ ||\hat{\nu}||_\infty &\leq 1,\\ \hat{\nu}_k = \text{sign}(\hat{\gamma}_{j,k}) \quad &\text{if } \hat{\gamma}_j \neq 0.\end{aligned}$$

By recalling that the definition of $\hat{\tau}_j^2$ is $||X_j - \mathbf{X}_{-j}\hat{\gamma}_j||_2^2/n + \lambda_j||\hat{\gamma}_j||_1$, we obtain :

$$\hat{\tau}_j^2 = (X_j - \mathbf{X}_{-j}\hat{\gamma}_j)^\top X_j/n.$$

Let us denote the jth row of M by M_j, it is a $1 \times p$ vector. By construction of the matrix, we have $M_j = \hat{C}_j/\hat{\tau}_j^2$ where $\hat{C}_j$ is a $1 \times p$ vector denoting the jth row of $\hat{C}$ defined in 2.2.4 and $\hat{\tau}_j^2$ is a scalar. Multiplying both sides of the previous equation by $\hat{\tau}_j^{-2}$ gives

$$1 = X_j^\top\,\mathbf{X}\,M_j^\top/n.$$

And the KKT conditions also give

$$\begin{aligned}\lambda_j\hat{\nu} &= \mathbf{X}_{-j}^\top(X_j - \mathbf{X}_{-j}\hat{\gamma}_j)/n\\ \lambda_j\hat{\nu}\hat{\tau}^{-2} &= \mathbf{X}_{-j}^\top(X_j - \mathbf{X}_{-j}\hat{\gamma}_j)\hat{\tau}^{-2}/n\\ \lambda_j\hat{\nu}\hat{\tau}^{-2} &= \mathbf{X}_{-j}^\top\,\mathbf{X}\,M_j^\top/n\\ ||\lambda_j\hat{\nu}\hat{\tau}^{-2}||_\infty &= ||\mathbf{X}_{-j}^\top\,\mathbf{X}\,M_j^\top||_\infty/n\end{aligned}$$

$$1 \times \frac{\lambda_j}{\hat{\tau}_j^2} \geq \frac{\lambda_j}{\hat{\tau}^2}||\hat{\nu}||_\infty = ||\mathbf{X}_{-j}^\top \mathbf{X}\, M_j^\top||_\infty/n$$
$$||\mathbf{X}_{-j}^\top \mathbf{X}\, M_j^\top||_\infty/n \leq \frac{\lambda_j}{\hat{\tau}_j^2},$$

where the last inequality is derived from the fact that $||\hat{\nu}||_\infty \leq 1$ by the KKT conditions. Finally with e_j the vector with 1 on the jth position and 0 otherwise :

$$||\Sigma M_j^\top - e_j||_\infty = ||\mathbf{X}_{-j}^\top \mathbf{X}\, M_j^\top||_\infty/n \leq \frac{\lambda_j}{\hat{\tau}_j^2}.$$

This implies a bound on the following term

$$||M\Sigma - I||_\infty \leq \max_{j=1,\dots,p} ||\Sigma M_j^\top - e_j||_\infty \leq \max_{j=1,\dots,p} \frac{\lambda_j}{\hat{\tau}_j^2}.$$

By combining the terms we obtain :

$$\begin{aligned} ||\Delta||_\infty &= \sqrt{n}||(M\Sigma - I)(\hat{\beta}_{LASSO} - \beta)||_\infty \\ &\leq \sqrt{n}||M\Sigma - I||_\infty||\hat{\beta}_{LASSO} - \beta\,||_1 \\ &\leq \sqrt{n}\left(\max_j \frac{\lambda_j}{\hat{\tau}_j^2}\right) 8\lambda\frac{s}{\phi_0} \end{aligned}$$

with probability at least $1 - 2e^{-t^2}$ and that concludes the proof.

□

Theorem 3.1.5 (Javanmard and Montanari (2014), Theorem 6)**.** *Let $\mathbf{X} \in \mathbb{R}^{n\times p}$ be any deterministic matrix and $\hat{\beta}^d$ the de-sparsified estimator in Equation 2.0.1. Then we have :*

$$\sqrt{n}(\hat{\beta}^d - \beta) = W + \Delta,$$
$$W = \frac{M\,\mathbf{X}^\top \epsilon}{\sqrt{n}} \sim \mathcal{N}(0, \sigma^2 M\Sigma M^\top).$$

Further assume that $\mathbf{X}$ satisfies condition (A1) and has generalized coherence parameter $\mu_* = \mu_*(\mathbf{X}, M)$. Then for $\lambda = \sigma\sqrt{(c^2 \log p)/n}$, with some $c > 0$, we have

$$\mathbb{P}\left[||\Delta||_\infty \geq \frac{4c\mu_*\sigma s}{\phi_0^2}\sqrt{\log p}\right] \leq 2p^{-c_0},$$
$$\textit{with} \quad c_0 = \frac{c^2}{32K} - 1.$$

Proof. The idea is the same as for the previous theorem, we need to show that Δ is not too large entry-wise. The result from Bühlmann and van de Geer (2011) is also used to bound $||\hat{\beta}_{LASSO} - \beta\,||_1$.

First of all, we have from above that,

$$||\Delta||_\infty \leq \sqrt{n}||M\Sigma - I||_\infty||\hat{\beta}_{LASSO} - \beta\,||_1 = \sqrt{n}\mu_*||\hat{\beta}_{LASSO} - \beta\,||_1.$$

Then, by Bühlmann and van de Geer (2011), Theorem 6.1, for all $\lambda \geq 2K\sigma\sqrt{2(t^2 + \log p)/n}$,

$$\mathbb{P}\left[||\hat{\beta}_{LASSO} - \beta\,||_1 \geq 4\lambda\frac{s}{\phi_0^2}\right] \leq 2e^{t^2}.$$

That implies the bound on Δ :

$$\mathbb{P}\left[||\Delta||_\infty \geq \sqrt{n}\mu_* 4\lambda\frac{s}{\phi_0^2}\right] \leq 2e^{t^2}.$$

This works for any arbitrary $t > 0$ and in the theorem, t is chosen such that $2e^{t^2} = p^{c_0}$. □

These two theorems bound the ℓ_∞ norm of the error term Δ in probability if the sparsity index is $s = o(\sqrt{n}/\log p)$. We remark that they have exactly the same assumptions, namely the compatibility condition and a bound on the diagonal entries of the Σ matrix. Each theorem assumes also that the noise is Gaussian and identically distributed, this assumption will be discussed later and a results for the non-Gaussian case will be proposed. The bound for Δ depends also on the regularization parameter λ, directly for the ND LASSO estimator and via the constant c for the OD LASSO, and on the inverse of the compatibility constant ϕ_0^2.

3.1.2 Random Design

In a random design, the matrix $\mathbf{X}$ is not fixed anymore, the challenge here is to provide conditions to guarantee the validity of the desparsified LASSO estimator for inference for a large family of design matrices.

Definition 3.1.6. The sub-Gaussian norm of a random variable X, denoted by $||X||_{\psi_2}$ is defined as

$$||X||_{\psi_2} = \sup_{q \leq 1} \frac{1}{\sqrt{q}}(\mathbb{E}[|X|^q])^{\frac{1}{q}}.$$

For a random vector $X \in \mathbb{R}^n$, its sub-Gaussian norm is defined as

$$||X||_{\psi_2} = \sup_{x \in S^{n-1}} ||\langle X, x\rangle||_{\psi_2}$$

where S^{n-1} denotes the unit sphere in $\mathbb{R}^n$.

The next conditions will be used to prove the distribution of the desparsified LASSO.

(A2) The rows of $\mathbf{X}$ are iid realizations from a Gaussian distribution, Σ has strictly positive eigenvalues and Λ_1, the smallest eigenvalue of Σ, is such that $\frac{1}{\Lambda_1} = \mathcal{O}(1)$. Moreover, $\max_j \Sigma_{j,j} = \mathcal{O}(1)$.

(A3) Σ is such that its eigenvalues are strictly positive and finite, ie $\exists \quad C_{\min}, C_{\max} \in]0, \infty[\quad st. \quad C_{\min} < \Lambda_1 < \ldots < \Lambda_p < C_{\max}$ and $\max_j(\Sigma)_{j,j} \leq 1$. Moreover, $\mathbf{X}\,\Sigma^{-1/2}$ has sub-Gaussian rows with zero mean and sub-Gaussian norm $\kappa \in]0, \infty[$.

The conditions (A2) and (A3) are similar in the sense that they both require that all the eigenvalues of the variance covariance matrix are strictly positive. However, condition (A3) imposes a finite sub-Gaussian norm on $\mathbf{X}\,\Sigma^{-1/2}$ while in the other condition the design is Gaussian and $\frac{1}{\Lambda_1}$ is controlled. As a sub-Gaussian norm is less restrictive than a Gaussian design, the (A2) condition is a bit stronger than the (A3) one.

Theorem 3.1.7 (van de Geer et al. (2014), Theorem 2.2)**.** *Consider the linear model in 1.0.1 with Gaussian error $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$, where $\sigma^2 = \mathcal{O}(1)$. Assume (A2) and the sparsity conditions $s = o(\frac{\sqrt{n}}{\log p})$ and $\max_j s_j = o(\frac{n}{\log p})$, where s_j is the number of the non zero entries of the jth row of the inverse of the empirical covariance matrix Σ^{-1}. Consider a suitable choice of the regularization parameter $\lambda \propto \sqrt{(\log p)/n}$ for the LASSO in 1.0.3 and $\lambda_j \propto \sqrt{(\log p)/n}$ uniformly in j for the nodewise regression 2.2.5. Then*

$$\begin{aligned}
&\sqrt{n}(\hat{\beta}^d - \beta) = W + \Delta,\\
&W = \frac{M\,\mathbf{X}^\top \epsilon}{\sqrt{n}} \text{ and } W|\,\mathbf{X} \sim \mathcal{N}_n(0, \sigma^2 M\Sigma M^\top),\\
&||\Delta||_\infty = o_{\mathbb{P}}(1).
\end{aligned}$$

Furthermore, $||M\Sigma M^\top - \Sigma^{-1}||_\infty = o_{\mathbb{P}}(1)$.

This result ensures that the error term of the de-biased LASSO is bounded in probability but also that the matrix M is close to its target Σ^{-1}. The proof follows the same idea as the one in the fixed design, however the work is slightly more complex. We will use the two following intermediary results to reach the conclusion.

Lemma 3.1.8. *Assume (A2) and $s = o(\frac{n}{\log p})$. Then there is a constant $L = \mathcal{O}(1)$ depending on Λ_1 only such that with probability tending to one, the compatibility condition holds with compatibility constant $\phi_0^2 \geq L^2$.*

This first lemma ensures that the condition (A2) is sufficient to get the compatibility condition. With this result, we can apply some arguments of the first theorem's proof. For the demonstration of this result, we refere to Raskutti et al. (2010), as it is a direct application of their Theorem 1.

Lemma 3.1.9. *Assume (A2) and row sparsity in Σ^{-1}: $\max_j s_j = o(\frac{n}{\log p})$. Then for $\lambda_j \propto \sqrt{(\log p)/n}$, uniformly in j,*

$$\max_j \frac{1}{\hat{\tau}_j^2} = \mathcal{O}_{\mathbb{P}}(1).$$

This second lemma bounds the inverse of the estimator for the node-wise LASSO variance which is a quantity used to construct the matrix M and that appears in the bound for Δ presented in the fixed case. We observe that it requires some sparsity in the rows of Σ, a condition that was not present for the result on the OD LASSO estimator.

Proof. By the first lemma, with probability tending to one, the compatibility assumption holds for the node-wise LASSO regression. The idea is to show that the terms in $\hat{\tau}_j^2$ are different from zero and that they converge for each j.

Recall that :

$$\tau_j^2 = \mathbb{E}[(X_{1,j} - \sum_{k \neq j} \gamma_{k,j} X_{1,k})^2],$$
$$\text{and} \quad \hat{\tau}_j^2 = ||X_j - \mathbf{X}_{-j}\hat{\gamma}_j||_2^2/n + \lambda_j ||\hat{\gamma}_j||_1.$$

As for each j, $||\mathbf{X}_{-j}(\hat{\gamma}_j - \gamma_j)||_2^2/n = \mathcal{O}_{\mathbb{P}}(s_j(\log p)/n) = \mathcal{O}_{\mathbb{P}}(s_j\lambda_j^2)$, the first equality comes from Bühlmann and van de Geer (2011) and the second uses that $\lambda \propto \sqrt{(\log p)/n}$ as assumed in the statement.

Then because ${\hat{\tau}_j}^2 = \tau_j^2 + \mathcal{O}(n^{-1/2})$,

$$\begin{aligned} ||X_j - \mathbf{X}_{-j}\hat{\gamma}_j||_2^2/n &= ||X_j - \mathbf{X}_{-j}\gamma_j||_2^2/n + ||\mathbf{X}_{-j}(\hat{\gamma}_j - \gamma_j)||_2^2/n + 2(X_j - \mathbf{X}_{-j}\gamma_j)^\top \mathbf{X}_{-j}(\hat{\gamma}_j - \gamma_j) \\ &= \tau_j^2 + \mathcal{O}_{\mathbb{P}}(n^{-1/2}) + \mathcal{O}_{\mathbb{P}}(\lambda_j^2 s_j) + \mathcal{O}_{\mathbb{P}}(\lambda_j\sqrt{s_j}) \\ &= \tau_j^2 + o_{\mathbb{P}}(1). \end{aligned}$$

Moreover the error variances τ_j^2 do not diverge. Indeed, $\tau_j^2 \leq \mathbb{E}[X_{1,j}^2] = \Sigma_{j,j} = \mathcal{O}(1)$. On the other hand, $\tau_j^2 = 1/(\Sigma^{-1})_{j,j} \geq \Lambda_1^2 > 0$. This allows us to conclude that the first part of $\hat{\tau}_j^2$ behaves well.

The second term in $\hat{\tau}_j^2$ is also bounded. By assumption, none of λ_j or $||\hat{\gamma}_j||_1$ are zero. Then

$$\begin{aligned} &||\gamma_j||_1 \leq \sqrt{s_j}||\gamma_j||_2 \leq \sqrt{s_j\Sigma_{j,j}}/\Lambda_1, \\ &\lambda_j||\hat{\gamma}_j||_1 \leq \lambda_j||\gamma_j||_1 + \lambda_j||\hat{\gamma}_j - \gamma_j||_1 = \lambda_j\mathcal{O}(\sqrt{s_j}) + \lambda_j\mathcal{O}(\lambda_j s_j) = o_{\mathbb{P}}(1). \end{aligned}$$

Hence this concludes the proof. □

We are now tackling the theorem that interests us.

Proof of Theorem 3.1.7. First, let us bound the error term Δ. The first lemma says that the compatibility conditions holds for the setting. We can thus apply the argument of the first theorem's proof to obtain :

$$\begin{aligned} \Delta &= \sqrt{n}(M\Sigma - I_p)(\hat{\beta}_{LASSO} - \beta), \\ ||\Delta||_\infty &= \sqrt{n}||(M\Sigma - I_p)(\hat{\beta}_{\text{LASSO}} - \beta)||_\infty, \\ &\leq \sqrt{n}||M\Sigma - I_p||_\infty||\hat{\beta}_{LASSO} - \beta\,||_2, \\ &\leq \sqrt{n}||\hat{\beta}_{LASSO} - \beta\,||_2 \max_j \frac{\lambda_j}{\hat{\tau}_j^2}. \end{aligned}$$

By the second Lemma, $\max_j 1/\hat{\tau}_j^2 = \mathcal{O}_{\mathbb{P}}(1)$. Moreover, the parameter λ_j is proportional to $\sqrt{(\log p)/n}$ and the term $||\hat{\beta}_{\text{LASSO}} - \beta\,||_2$ is also bounded :

$$\begin{aligned} ||\,\mathbf{X}(\hat{\beta}_{LASSO} - \beta)||_2^2/n &= (\hat{\beta}_{LASSO} - \beta)^\top\Sigma(\hat{\beta}_{LASSO} - \beta) \\ &\leq (\hat{\beta}_{LASSO} - \beta)^\top\Sigma(\hat{\beta}_{LASSO} - \beta)C + \sqrt{\frac{\log p}{n}}||\hat{\beta}_{LASSO} - \beta\,||_1 C, \end{aligned}$$

where the inequality is true with probability tending to one for some well chosen constant C. This result comes from (Raskutti et al., 2010, Theorem 1). Hence,

$$|| \mathbf{X}(\hat{\beta}_{LASSO} - \beta)||_2^2/n = \mathcal{O}_{\mathbb{P}}(\frac{s \log p}{n}).$$

So if we choose $\lambda \propto \sqrt{(\log p)/n}$ for $\hat{\beta}_{LASSO}$, we have the following bound :

$$||\hat{\beta}_{LASSO} - \beta \,||_2 = \mathcal{O}_{\mathbb{P}} \left(\sqrt{\frac{s \log p}{n}} \right).$$

This bound is sufficient to control the error term for the desparsified estimator :

$$\begin{aligned} ||\Delta||_\infty &\leq \sqrt{n}||\hat{\beta}_{LASSO} - \beta \,||_2 \max_j \frac{\lambda_j}{\hat{\tau}_j^2}, \\ &= \sqrt{n}\mathcal{O}_{\mathbb{P}} \left(\sqrt{\frac{s \log p}{n}} \right) \mathcal{O}_{\mathbb{P}}(1) \mathcal{O} \left(\sqrt{\frac{\log p}{n}} \right), \\ &= \mathcal{O}_{\mathbb{P}} \left(\frac{s \log p}{\sqrt{n}} \right), \\ &= o_{\mathbb{P}}(1). \end{aligned}$$

The last equality comes from the sparsity assumption on s, we have assumed $s = o(\frac{\sqrt{n}}{\log p})$.

There now remains to prove the second statement. Using similar argumentation to that in the proof of Lemma 3.1.9, we have that uniformly in j,

$$||M_j||_1 = \mathcal{O}_{\mathbb{P}}(\sqrt{s_j}).$$

Furthermore, $M\Sigma M^\top = (M\Sigma - I_p)M^\top + M^\top$. Then, using again Lemma 3.1.9 :

$$||(M\Sigma - I_p)M^\top||_\infty \leq \max_j \lambda_j ||M_j||_1/\hat{\tau}_j^2 = o_{\mathbb{P}}(1).$$

Finally,

$$\begin{aligned} ||M - \Sigma^{-1}||_\infty &\leq \max_j ||M_j - \Sigma_j^{-1}||_2 \\ &\leq \max_j \lambda_j \sqrt{s_j} = o_{\mathbb{P}}(1). \end{aligned}$$

We can now finish the proof, as we have

$$\begin{aligned} ||M\Sigma M^\top - \Sigma^{-1}||_\infty &\leq ||(M\Sigma - I_p)M^\top||_\infty + ||M - \Sigma^{-1}||_\infty, \\ &\leq o_{\mathbb{P}}(1). \end{aligned}$$

□

Theorem 3.1.10 (Javanmard and Montanari (2014), Theorem 8)**.** *Consider the linear model in 1.0.1 with Gaussian error $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ and $\hat{\beta}^d$ in Algorithm 1 with $\mu = a\sqrt{(\log p)/n}$, for $a > 0$. Under assumption (A3), for $n \geq \max(\nu s \log(p/s), 1600\kappa^4 \log p, a^2/(4e^2\kappa^4) \log p)$ and for $\lambda = \sigma\sqrt{(c^2 \log p)/n}$, with κ the sub-Gaussian norm of $\mathbf{X}\Sigma^{-1/2}$, ν that depends on $C_{\max}/C_{\min}$ and on κ and some $c > 0$, we have*

$$\begin{aligned}
&\sqrt{n}(\hat{\beta}^d - \beta) = W + \Delta, \\
&W = \frac{M\,\mathbf{X}^\top \epsilon}{\sqrt{n}} \text{ and } W|\,\mathbf{X} \sim \mathcal{N}(0, \sigma^2 M\Sigma M^\top), \\
&\mathbb{P}\left[||\Delta||_\infty \geq \frac{16ac\sigma}{C_{\min}} \frac{s \log p}{n}\right] \leq 4e^{-1/(4c\kappa^4)n} + 4p^{\min(-c^2/48-1,(a^2 C_{\min})/(24e^2\kappa^4 C_{\max})-2)}.
\end{aligned}$$

Finally, this tail bound holds for any choice of M that is only function of the design matrix $\mathbf{X}$, and satisfies the feasibility of the program 2.2.6, ie $||M\Sigma - I||_\infty \leq \mu$.

To obtain this result, the authors base themselves on a previous theorem which states that under the assumption (A3) and with sufficiently large n, the compatibility constant holds with high probability. The result also stipulates that the optimization problem 2.2.6 is feasible for $\mu = a\sqrt{(\log p)/n}$ with high probability.

Theorem 3.1.11 (Javanmard and Montanari (2014), Theorem 7)**.** *Assume condition (A3). Then*

a) If n is sufficiently large, the compatibility condition holds with high probability, ie.

$$\begin{aligned}
&\mathcal{E}(\phi_0) = \{\mathbf{X} \in \mathbb{R}^{n\times p} : \min_{S,|S|\leq s} \phi(\Sigma, S) \geq \phi_0, \max_j \Sigma_{j,j} \leq K\}, \\
&\mathbb{P}[\mathbf{X} \in \mathcal{E}(\phi_0)] \geq 1 - 4e^{-c_1 n}.
\end{aligned}$$

b) For $a > 0$, let $\mathcal{G}(a)$ be the event that the problem 2.2.6 is feasible for $\mu = a\sqrt{(\log p)/n}$, ie.

$$\mathcal{G}(a) = \left\{\mathbf{X} \in \mathbb{R}^{n\times p} : \min_M \mu(\mathbf{X}, M) < a\sqrt{\frac{\log p}{n}}\right\}.$$

Then for n sufficiently large and for c' constant depending on κ and on $C_{\min}, C_{\max}$,

$$\mathbb{P}(\mathbf{X} \in \mathcal{G}(a)) > 1 - 2p^{-c'}.$$

In this theorem we have introduced $\mathcal{E}(\phi_0)$, the set of matrices $\mathbf{X}$ that satisfy the compatibility condition. If our design matrix belongs to this set, and if the sparsity is enough large, the convergence of the initial LASSO estimator is guarantee by 3.1.3. The goal, in a random design, is to show that the probability that $\mathbf{X} \in \mathcal{E}(\phi_0)$ is high. This is the problem that point a tackles. In the point b, the set $\mathcal{G}(a)$ contains all matrices $\mathbf{X}$ for which the minimization program 2.2.6 is solvable. If the matrix $\mathbf{X}$ does not belong to it, Algorithm 1 will fail. What said the theorem is that for sufficiently large n, with high probability, the matrix $\mathbf{X}$ is such that the LASSO converges and the Algorithm 1 works.

Using this theorem, the proof for consistence of the desparsified estimator and its convergence is direct.

Proof of Theorem 3.1.10. Let us denote the quantity in bound of Δ in the theorem by Δ_0 :

$$\Delta_0 = \frac{16ac\sigma}{C_{\min}} \frac{s \log p}{n}$$

We have then, with $\phi_0 = \sqrt{C_{\min}}/2$:

$$\begin{aligned}
\mathbb{P}(||\Delta||_\infty \geq \Delta_0) &= \mathbb{P}\left(\{||\Delta||_\infty \geq \Delta_0\} \cap \mathcal{E}(\phi_0) \cap \mathcal{G}(a)\right) + \mathbb{P}(\mathcal{E}^c(\phi_0) \cap \{||\Delta||_\infty \geq \Delta_0\}) + \mathbb{P}(\mathcal{G}^c(a) \cap \{||\Delta||_\infty \geq \Delta_0 \\
&\leq \mathbb{P}\left(\{||\Delta||_\infty \geq \Delta_0\} \cap \mathcal{E}(\phi_0) \cap \mathcal{G}(a)\right) + \mathbb{P}(\mathcal{E}^c(\phi_0)) + \mathbb{P}(\mathcal{G}^c(a)), \\
&\leq \mathbb{P}\left(\{||\Delta||_\infty \geq \Delta_0\} \cap \mathcal{E}(\phi_0) \cap \mathcal{G}(a)\right) + 4e^{-c_1 n} + 2p^{-c_2}
\end{aligned}$$

The notation $\mathcal{A}^c$ is for the complement event of $\mathcal{A}$ and second inequality comes from Theorem 3.1.11.

We conclude the proof by applying the theorem for fixed design since we have conditioned on $\mathbf{X}$ and since the matrix satisfies the compatibility condition as well as solve the optimization program. This implies that

$$\begin{aligned}
\mathbb{P}\left(\{||\Delta||_\infty \geq \Delta_0\} \cap \mathcal{E}(\phi_0) \cap \mathcal{G}(a)\right) &\leq \sup_{\mathbf{X} \in \mathcal{E}(\phi_0) \cap \mathcal{G}(a)} \mathbb{P}(||\Delta||_\infty \geq \Delta_0 | \, \mathbf{X}), \\
&\leq 2p^{c_3}.
\end{aligned}$$

□

The proof of Theorem 3.1.11 is given in Javanmard and Montanari (2014) as well as a more detailed statement, here we only make a sketch to explain the reasoning and identify where assumption (A3) was used.

Sketch of the proof for Theorem 3.1.11. For the point a, the condition can be decomposed into two events :

$$\begin{aligned}
B_{1,n} &:= \{\mathbf{X} \in \mathbb{R}^{n \times p} : \min_{S, |S| \leq s} \phi(\Sigma, S) \geq \phi_0\}, \\
B_{2,n} &:= \{\mathbf{X} \in \mathbb{R}^{n \times p} : \max_j \Sigma_{j,j} \leq K\}.
\end{aligned}$$

If we can bound these two events, the first part of the Theorem will be proven.

To bound $B_{1,n}$, the idea is to use the concept of restricted eigenvalue that was introduced by Bickel et al. (2009) and then the result which states that if the population variance covariance satisfies the restricted eigenvalue condition, the sample covariance will also satisfy it (Rudelson and Zhou, 2013).

To control the second event $B_{2,n}$, the authors use the hypothesis that $\mathbf{X}\,\Sigma^{-1/2}$ has sub-Gaussian rows to apply a Bernstein-type inequality (Vershynin, 2012).

For the second part of the theorem, the Bernstein-type inequality is also used. □

The two theorems give, as in the fixed-design case, a bound on $||\Delta||_\infty$. The result of Javanmard and Montanari (2014) explicitly presents the link with the eigenvalues of Σ. They also require that n is higher than several parameters depending directly on the models. The conditions of 3.1.7 are slightly stronger, indeed they assume $\mathbf{X}$ has iid. Gaussian rows while (A3) only asks the rows to be independent with some finite sub-Gaussian norm. They also assume sparsity in the matrix Σ^{-1} and this is not something that one can easy verify. If $\sigma, C_{\min}$ are of order one, Theorem 3.1.10 gives $||\Delta||_\infty = O(\frac{s \log p}{\sqrt{n}})$ which is the same as in Theorem 3.1.7 due to the sparsity assumption.

3.1.3 Discussion on the assumptions

For the fixed design, the methods work under the same assumption on the setting, they first need the convergence of the LASSO estimator and then, that their constructed matrix M is close to Σ^{-1}. In a random design setting, they set conditions to guarantee that each matrix will verify the requirement for fixed design. We have already highlighted that assumptions (A2) and (A3) present some differences. (A2) is about the control on the first eigenvalue inverse of Σ and to assume that $\mathbf{X}$ is Gaussian, for (A3), the condition imposes a finite sub-Gaussian norm to $\mathbf{X}\,\Sigma^{-1/2}$ in order to guarantee the convergence of Algorithm 1. This last assumption is so weaker than condition (A2).

For the sparsity assumption, this is the same for both methods, we need $s = o(\sqrt{n}/\log p)$ but moreover, the ND LASSO also requires that the row of the inverse variance covariance matrix is sparse.

More recently, Javanmard and Montanari (2018) show that the condition for the sparsity is too strong and can be a relaxed. It is sufficient that the sparsity level s is of order $o(\frac{n}{(\log p)^2})$ to obtain the convergence of the desparsified estimator and its distribution. However, this result holds only in the hypothetical case where the inverse covariance matrix Σ^{-1} is known and used to construct the estimator. They go further in their reflection and prove that for the case where the matrix M is constructed by node-wise LASSO, assuming in addition that $\min(s, s_{\Sigma^{-1}}) = o(\frac{\sqrt{n}}{\log p})$, where $s_{\Sigma^{-1}}$ is the number of non-null entries in Σ^{-1}, guarantees the convergence to the Normal distribution. It means that the model requires sparsity in the coefficients but also in the inverse of the variance-covariance matrix. However, this is not an easy task, even if the variance covariance matrix is sparse, depending on its structure, its inverse can be dense, as the sparsity properties of a matrix are not conserved by the inverse operation.

Another assumption in the three articles for the convergence to a Gaussian distribution is that the parameter λ is correctly chosen. In these articles they use $\lambda = C\sigma\sqrt{\frac{\log p}{n}}$ for some C usually greater than 2 in their proofs. This choice is justified by the good properties of this parameter but it is not necessary optimal. In Miolane and Montanari (2021), the authors prove that the convergence of the parameter is uniform over the penalty parameter. The proof is under a Gaussian design for the matrix $\mathbf{X}$ but it indicates that the results can be extended for other sizes of the penalty parameter.

The conditions for the convergence of the de-biased LASSO estimator were extended to cover the case where the model is misspecified (Bühlmann and van de Geer, 2015). More precisely, they consider the case where the true underline model is non linear and that we estimate it as if it were, that means we estimate the projection of the true model on a linear model. Adding

a few assumptions on the design, they prove that the desparsified estimator converges to the one in the projection and then the selected variables in the model estimated by the de-biased LASSO are also non-zero in the unknown true model. These results prove some robustness of the method.

3.2 Inference

The previous theorems allow us to obtain explicit formulas for confidence intervals and to perform hypothesis testing. We present them in this section before discussing the practiced case where the noise level is unknown.

Confidence intervals: Let us assume $\alpha \in [0,1]$, then an asymptotical confidence interval at level $1-\alpha$ for the parameter β_j is given by

$$\begin{aligned} IC_j(\alpha) &= [\hat{\beta}_j^d - \delta(\alpha,n); \hat{\beta}_j^d + \delta(\alpha,n)], \\ \delta(\alpha,n) &= \Phi^{-1}(1-\frac{\alpha}{2})\frac{\sigma}{\sqrt{n}}(M\Sigma M^\top)_{j,j}^{1/2}, \end{aligned} \tag{3.2.1}$$

where the function Φ denotes the cumulative distribution function of the standardized Normal. This confidence interval indicates that if we repeated the experiences a large number of times, on $1-\alpha$ percentage of cases, the true parameter will be contained in the interval.

Indeed, using the previous theorems

$$\begin{aligned} \lim_{n\to\infty} \mathbb{P}(\beta_j \in IC_j(\alpha)) = &\lim_{n\to\infty} \mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}} \leq \Phi^{-1}(1-\frac{\alpha}{2})\right) \\ &- \lim_{n\to\infty} \mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}} \leq -\Phi^{-1}(1-\frac{\alpha}{2})\right) \\ =&1-\alpha. \end{aligned}$$

Hypothesis testing: For a coefficient β_j, we are interested in testing the null hypothesis that the parameter is equal to zero against the hypothesis that it is different:

$$\begin{aligned} H_0 :&\quad \beta_j = 0, \\ H_a :&\quad \beta_j \neq 0. \end{aligned}$$

This test is really interesting because it tells us if there is a linear link between the variable X_j and the response variable $\mathbf{Y}$. Again, the theory developed in the previous section gives us a formula for the p-value, where

$$P_j = 2\left(1 - \Phi\left(\frac{\sqrt{n}|\hat{\beta}_j^d|}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}}\right)\right). \tag{3.2.2}$$

For $\alpha \in [0,1]$, the associated decision rule is

$$\hat{T}_j = \begin{cases} 1 & \text{if } P_j \leq \alpha, \\ 0 & \text{otherwise.} \end{cases}$$

A decision $\hat{T}_j = 1$ corresponds to rejecting the null hypothesis H_0 and $\hat{T}_j = 0$ is the case where the null hypothesis is not rejected, hence we cannot affirm that the coefficient is significantly different from zero.

Unknown variance

In practice the variance σ^2 of the noise is unknown and the inference for the parameter depends directly on this quantity. To replace σ^2 with a consistent estimator while keeping the properties of the confidence interval, we need some work.

Javanmard and Montanari (2014) have proposed sufficient conditions for an estimator of the noise level to guarantee keeping the distribution of the de-biased estimator. They require that the estimator $\hat{\sigma}(\mathbf{X})$ is consistent in the sense that it converges in probability for all the possible vectors β.

Lemma 3.2.1. *Assume condition (A3) and $s = o(\frac{\sqrt{n}}{\log p})$. Let $\hat{\beta}^d$ be the solution of Algorithm 1 with $\mu = a\sqrt{\log p/n}$ and $\lambda = \sigma\sqrt{c^2 \log p/n}$ where a and c are large enough constants. Then if $\hat{\sigma}(\mathbf{X})$ is a consistent estimator for σ in the sense that $\forall \epsilon > 0$,*

$$\lim_{n\to\infty} \sup_{\beta\in\mathbb{R}^p, ||\,\beta\,||_0 \leq s} \mathbb{P}\left(\left|\frac{\hat{\sigma}}{\sigma} - 1\right| \geq \epsilon\right) = 0,$$

where $||\,\beta\,||_0$ denotes the number of non zero entries of the vector. We have

$$\lim_{n\to\infty} \sup_{\beta\in\mathbb{R}^p, ||\,\beta\,||_0 \leq s} \left| \mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\hat{\sigma}(M\Sigma M^\top)_{j,j}^{1/2}} \leq x\right) - \Phi(x)\right| = 0.$$

Proof. We will prove that

$$\limsup_{n\to\infty} \left(\sup_{\beta\in\mathbb{R}^p, ||\,\beta\,||_0 \leq s} \mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\hat{\sigma}(M\Sigma M^\top)_{j,j}^{1/2}} \leq x\right)\right) \leq \Phi(x).$$

An analogous argument shows that for the $\liminf_{n\to\infty}$ and given these two bounds, the lemma is proven.

From Theorem 3.1.10, we have that

$$\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}} = \frac{e_j^\top M\, \mathbf{X}^\top \epsilon}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}} + \frac{\Delta_j}{\sigma(M\Sigma M^\top)_{j,j}^{1/2}},$$

$$= V^\top \epsilon + \frac{\Delta_j}{\sigma(M\Sigma M^\top)^{1/2}_{j,j}},$$

where $V = \mathbf{X}\, M^\top e_j / (\sigma(M\Sigma M^\top)^{1/2}_{j,j})$. The variables V and ϵ are independent due to the independence of $\mathbf{X}$ and ϵ and $||V||_2 = 1$. Moreover, by assumption $\epsilon \sim \mathcal{N}(0,1)$, hence

$$\mathbb{P}(V^\top \epsilon \leq x) = \mathbb{E}[\mathbb{P}(V^\top \epsilon \leq x | V)] = \mathbb{E}[\Phi(x)|V] = \Phi(x).$$

In other words, $V^\top \epsilon \sim \mathcal{N}(0,1)$.

Let $u > 0$. Then

$$\begin{aligned}
\mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}^d_j - \beta_j)}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \leq x\right) &= \mathbb{P}\left(\frac{\sigma}{\hat{\sigma}} V^\top \epsilon + \frac{\Delta_j}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \leq x\right), \\
&\leq \mathbb{P}\left(\frac{\sigma}{\hat{\sigma}} V^\top \epsilon \leq x + u\right) + \mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \geq u\right), \\
&\leq \mathbb{P}\left(V^\top \epsilon \leq \frac{\hat{\sigma}}{\sigma}(x+u) \middle| \left|\frac{\hat{\sigma}}{\sigma}\right| \leq u+1\right) \mathbb{P}\left(\left|\frac{\hat{\sigma}}{\sigma}\right| \leq u+1\right) + \mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \geq u\right), \\
&\leq \Phi(x + ux + u^2 + u)\left(1 - \mathbb{P}\left(\left|\frac{\hat{\sigma}}{\sigma} - 1\right| \geq u\right)\right) + \mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \geq u\right).
\end{aligned}$$

By taking the limit on each side, we have

$$\begin{aligned}
\limsup_{n\to\infty} \sup_{\beta \in \mathbb{R}^p, ||\,\beta\,||_0 \leq s} \mathbb{P}\left(\frac{\sqrt{n}(\hat{\beta}^d_j - \beta_j)}{\sigma(M\Sigma M^\top)^{1/2}_{j,j}} \leq x\right) \leq \\
\Phi(x + ux + u^2 + u) + \lim_{n\to\infty} \sup_{\beta \in \mathbb{R}^p, ||\,\beta\,||_0 \leq s} \mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \geq u\right).
\end{aligned}$$

As u is completely arbitrary, there only remains to prove that this second term is equal to zero. For this, we use the following technical lemma. We do not show its proof here but it can be found in Javanmard and Montanari (2014).

Lemma 3.2.2. *Under the same assumptions as in Lemma 3.2.1,*

$$\mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)^{1/2}_{j,j}} \geq u\right) \leq \mathbb{P}\left(\frac{5}{\sigma}|\Delta_j| \geq u\right) + \mathbb{P}\left(\frac{\hat{\sigma}}{\sigma} \leq \frac{1}{2}\right) + \mathbb{P}(\Sigma_{j,j} \geq \sqrt{2}).$$

Since $\hat{\sigma}$ is consistent, $\mathbb{P}(\frac{\hat{\sigma}}{\sigma} \leq \frac{1}{2}) \to 0$ and the last term always goes to zero from the proof of Theorem 3.1.10.

Hence using the consistency of $\hat{\sigma}^2$,

$$\begin{aligned}\lim_{n\to\infty} \sup_{\beta\in\mathbb{R}^p,|| \beta ||_0\leq s} \mathbb{P}\left(\frac{|\Delta_j|}{\hat{\sigma}(M\Sigma M^\top)_{j,j}^{1/2}} \geq u\right) &\leq \lim_{n\to\infty} \sup_{\beta\in\mathbb{R}^p,|| \beta ||_0\leq s} \mathbb{P}\left(||\Delta||_\infty \geq \frac{u\sigma}{5}\right) \\ &\leq \lim_{n\to\infty}(4e^{c_1 n} + 4p^{-\min(\tilde{c},c_2)}) = 0,\end{aligned}$$

with $c_1, c_2, \tilde{c}$ as in 3.1.10.

□

This lemma ensures that using a good estimator for σ^2 leads to correct confidence intervals and p-values for the OD estimator of Javanmard and Montanari (2014). In case of unknown variance, the formula for the confidence intervals as well as the procedure for hypothesis testing are built in the same ways as above, replacing σ by $\hat{\sigma}$. The other article van de Geer et al. (2014) has also suggested to take a studentized version of the distribution to take into account of the estimation of the noise level.

In high dimension regression, there exist many ways to estimate this parameter. One of the most common is the scaled LASSO (Sun and Zhang, 2012). The idea is to minimize the loss function with two parameters, the coefficient β and the noise level σ^2.

$$(\hat{\beta}, \hat{\sigma}^2) = \arg\min_{\beta\in\mathbb{R}^p,\sigma^2>0} \frac{1}{\sigma 2n}|| \mathbf{Y} - \mathbf{X}\beta ||_2^2 + \frac{\sigma^2}{2} + \lambda|| \beta ||_1 \tag{3.2.3}$$

This estimator for σ^2 satisfies the consistency criterion for the construction of the inference's tools. The proof is developed in Appendix C of Javanmard and Montanari (2014).

3.3 Non Gaussian Noise

The inference for the desparsified estimator is given when the noise in regression is identically Gaussian distributed. In this section, we will present results where this assumption for the noise is relaxed. The question is to find whether the distribution for the estimator can still be derived in such a setting.

As a reminder, the Gaussian distribution of the desparsified estimator comes from the fact that $W = M\mathbf{X}^\top \epsilon/\sqrt{n}| \mathbf{X} \sim \mathcal{N}(0, \sigma^2 M\Sigma M^\top)$ since $\epsilon_i \sim \mathcal{N}(0, \sigma^2) \quad \forall i \in \{1, ..., n\}$.

When the ϵ does not follow a Gaussian distribution anymore, the normality of the estimator is no longer guaranteed. To resolve this issue, Javanmard and Montanari (2014) proposes to use the Lindeberg condition in order to apply the Central Limit Theorem and obtain the Gaussian distribution for the $\hat{\beta}^d$.

For each component $j \in \{1, ..., p\}$ since $||\Delta||_\infty = o(1)$, we have

$$\frac{\sqrt{n}(\hat{\beta}_j^d - \beta_j)}{\sigma((M\Sigma M^\top)_{j,j})^{1/2}} = \frac{1}{\sqrt{n}} \frac{m_j^\top \mathbf{X}^\top \epsilon}{\sigma\sqrt{(m_j^\top \Sigma m_j)_{j,j}}} + o(1),$$

$$= \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \frac{m_j^\top \mathbf{X}_i^\top \epsilon_i}{\sigma\sqrt{(m_j^\top \Sigma m_j)_{j,j}}} + o(1).$$

The terms $U_i = \frac{m_j^\top \mathbf{X}_i^\top \epsilon_i}{\sigma\sqrt{(m_j^\top \Sigma m_j)_{j,j}}}$ are independent conditionally on $\mathbf{X}$, with a mean of zero and such that $\sum_{i=1}^{n} \mathbb{E}[U_i^2 | \mathbf{X}] = 1$. Indeed, under the hypothesis that ϵ comes from an unknown distribution with a mean of zero and with homoscedastic variance σ^2, we have

$$\mathbb{E}[U_i | \mathbf{X}] = \frac{m_j^\top \mathbf{X}_i^\top}{\sigma((m_j^\top \Sigma m_j)_{j,j})^{1/2}} \mathbb{E}[\epsilon_i | \mathbf{X}] = 0,$$
$$\mathbb{E}[U_i^2 | \mathbf{X}] = \frac{(m_j^\top \mathbf{X}_i^\top)^2}{\sigma^2 (m_j^\top \Sigma m_j)_{j,j}} \mathbb{E}[\epsilon_i^2 | \mathbf{X}] = \frac{1}{n} \frac{(m_j^\top \Sigma m_j)_{j,j}}{\sigma^2 (m_j^\top \Sigma m_j)_{j,j}} \sigma^2 = \frac{1}{n}.$$

Thus the CLT gives a Gaussian distribution to the estimator if the Lindeberg condition is satisfied, i.e. if for every $t > 0$ we have that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[U_i^2 \mathbb{1}_{\{|U_i| > t\sqrt{n}\}} | \mathbf{X}] = 0.$$

Then,

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n} U_i | \mathbf{X} \sim \mathcal{N}(0, 1).$$

In order to respect this condition for $\hat{\beta}^d$, Algorithm 1 is adapted. More precisely a condition is added to the minimization problem which becomes :

$$\begin{aligned} \text{minimize} \quad & m^\top \Sigma m \\ \text{subject to} \quad & ||\Sigma m - e_j||_\infty \leq \mu, \\ & ||\mathbf{X} m||_\infty \leq n^\alpha \quad \text{for an arbitrary fixed} \quad \alpha \in]1/4, 1/2[. \end{aligned} \tag{3.3.1}$$

Theorem 3.3.1. *Assume that for all $i = 1, \ldots, n$ ϵ_i are independent with $\mathbb{E}[\epsilon_i] = 0, \mathbb{E}[\epsilon^2] = \sigma$ and $\mathbb{E}[|\epsilon|^{2+a}] \leq C\sigma^{2+a}$ for some $a, C > 0$. Let M be the matrix obtained by 3.3.1. Under the assumptions of the theorem 3.1.10, with sparsity level $s = o(\sqrt{n}/\log p)$, an asymptotic confidence interval for the level $1 - \alpha$ is constructed as in 3.2.1 and p-values for the null hypothesis $H_{0,j} : \beta_j = 0$ are formed as before in 3.2.2.*

Proof. Continuing the above reasoning, we have to show that for a fixed β_j, all variables U_i verify the Lindeberg condition for $i = 1, ..., n$. To recall :

$$U_i = \frac{m_j^\top \mathbf{X}_i^\top \epsilon_i}{\sigma\sqrt{(m_j^\top \Sigma m_j)_{j,j}}}$$

Let c_n denote $(m_j \Sigma m_j)^{1/2}$. By Javanmard and Montanari (2014), Lemma 12, $\liminf_{n\to\infty} c_n \geq c_\infty > 0$, almost surely. If the optimizations problems in 3.3.1 are all feasible, then

$$|U_i| \leq \frac{||\mathbf{X} m_j||_\infty ||\epsilon||_\infty}{c_n \sigma} \leq \frac{n^\alpha ||\epsilon||_\infty}{c_n \sigma}.$$

Hence,

$$\begin{aligned}
\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[U_i^2 \mathbb{1}_{\{|U_i|>t\sqrt{n}\}} | \, \mathbf{X}\right] &\leq \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[U_i^2 \mathbb{1}_{\{||\epsilon||_\infty/\sigma > t c_n n^{1/2-\alpha}\}} | \, \mathbf{X}], \\
&= \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \frac{m_j^\top X_i X_i^\top m_j}{m_j^\top \Sigma m_j} \mathbb{E}\left[\left(\frac{\epsilon_i}{\sigma}\right)^2 \mathbb{1}_{\{||\epsilon||_\infty/\sigma > t c_n n^{1/2-\alpha}\}} | \, \mathbf{X}\right], \\
&\leq \lim_{n\to\infty} \mathbb{E}\left[\left(\frac{\epsilon_1}{\sigma}\right)^2 \mathbb{1}_{\{|\frac{\epsilon_1}{\sigma}|> t c_\infty n^{1/2-\alpha}\}}\right], \\
&\leq c(t) \lim_{n\to\infty} n^{-a(1/2-\alpha)} \, \mathbb{E}\left[\left|\frac{\epsilon_1}{\sigma}\right|^{2+a}\right] = 0,
\end{aligned}$$

where $c(t)$ is a function depending on t and c_∞. Then, using this condition, the Lindeberg Central Limit Theorem ensures that $1/\sqrt{n}\sum_i U_i | \, \mathbf{X}$ converges weakly to a Gaussian distribution. For all $i \in \{1, \ldots, n\}$,

$$\lim_{n\to\infty} \mathbb{P}(\frac{1}{\sqrt{n}}\sum_{i=1}^{n} U_i \leq x | \, \mathbf{X}) = \Phi(x).$$

□

This idea of using the Lindeberg condition to obtain the CLT, can also be applied to the estimator of van de Geer et al. (2014). However, as the matrix M is constructed to be an estimator of Σ^{-1}, nothing guarantees that the condition holds. For some design matrix $\mathbf{X}$, the desparsified estimator will not have a Gaussian distribution. In spite of that, they show in their Theorem 2.4 that the matrix M constructed by node-wise LASSO is still close to Σ^{-1} if the error term is not Gaussian. The term Δ is also bounded in absolute value for each component or, under some other conditions, in ℓ_∞ norm.

Figure 3.3.1 illustrates by an example the performance of the estimator when the noise is non Gaussian. In this toy example, the noise follows an exponential distribution with parameter $m = 0.667$. The number of observations is 100 for 500 coefficients to estimate but only the 100 first are shown, the correlation between each variable is $\rho = 0.1$.

In this case, we observe that the ND LASSO method misses the first non active coefficients but for the rest, the coverage seems quite correct. For the OD LASSO method, this is different, for several coefficients the confidence interval does not contain the true parameter. We see that here the method fails to estimate correctly the uncertainty of the coefficients even if the coefficients seem correct.

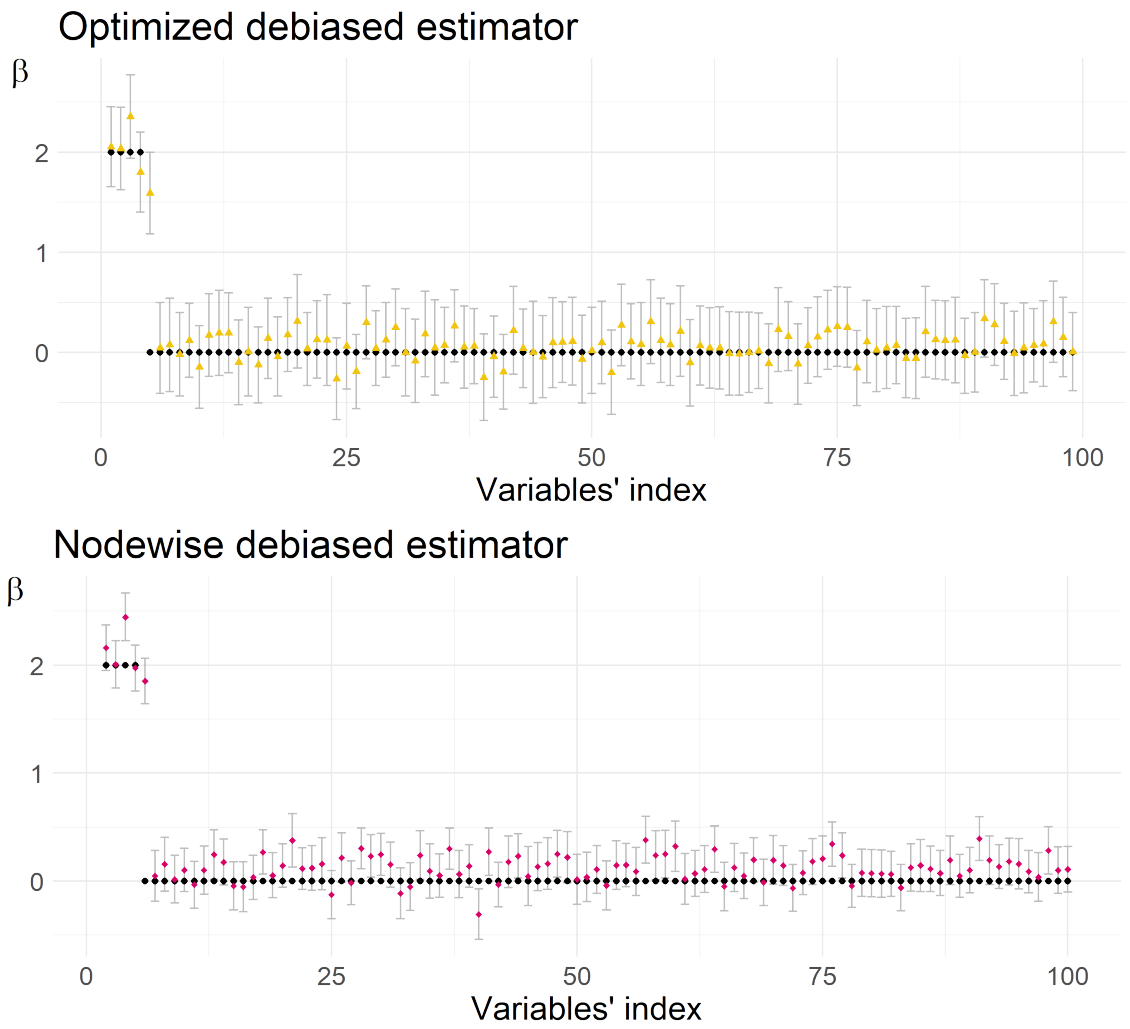

Figure 3.3.1: Confidence intervals for each variables when the noise follows an Exponential distribution with parameter $m = 0.667$. The design is $n = 100$ and $p = 500$ but for more readability only the first 100 coefficients are shown. The correlation between each variable is the same with $\rho = 0.1$. Above in yellow : OD LASSO estimator and below in red : ND LASSO estimator, the true parameters are in black.

Chapter 4

Numerical results : Simulations.

In this chapter, we compare the performances of two de-biased methods in numerical simulations. The estimators are also compared with some alternatives in high dimension for the estimation as well as for the inference.

As a reminder, the first desparsified method is introduced by Zhang and Zhang (2014) and developed by van de Geer et al. (2014) and we call it the Node-wise De-biased LASSO (ND LASSO) estimator. To perform the estimation, we use their `R` package `hdi` and more precisely the function `lasso.proj()`.

The second method is the one from Javanmard and Montanari (2014) which constructs the M matrix by a minimization problem, named Optimized De-biased LASSO (OD LASSO) estimator. We also use the source code provided by the authors on Montanari's webpage.

The competitors for the comparison are the following : the classical LASSO method, the Ridge estimator 4.1.1, the elastic net 4.1.2 and the SCAD method 4.1.3 for the accuracy and for the inference part, the Exact Post Selection Inference (EPSI) method that also obtains confidence intervals in high-dimensions. These methods are presented in the next section.

4.1 Classical alternatives for high-dimensional regression

In this section, we introduce some competitors to the desparsified LASSO and their main properties. These competitors can all work in high-dimensional settings but they do not all allow for the construction of confidence intervals and hypothesis testing. Among the methods presented here, only the EPSI method provides tools for inference in high-dimensions.

Ridge estimator

This estimator, published 30 years before the LASSO, is introduced by Hoerl and Kennard (1970). Its purpose is to prevent the bias induced by the multicollinearity of the variables. The issue of multicollinearity occurs when some explanatory variables are highly correlated, usually with a correlation higher than 0.7. It leads to a less accurate estimation. The idea here is to add a constraint on the ℓ_2 norm of coefficients in the loss function, leading to a non sparse estimator that is always derivable, even when the rank of the matrix $\mathbf{X}$ is not full.

$$\beta_{\text{ridge}} = \arg\min_{\beta} || \mathbf{Y} - \mathbf{X}\,\beta||_2^2 + \lambda||\beta||_2^2. \tag{4.1.1}$$

This estimator also requires the choice of a penalization parameter λ. Its bias is non null but in low dimensions, it has lower variance than the classic Ordinary Least Squared estimator as well as lower mean squared error. We can obtain the theoretical distribution of the estimator but there is no population estimator for its variance in high-dimensions, thus inference can not be conducted.

$$\beta_{\text{ridge}} \sim \mathcal{N}((\mathbf{X}^\top\mathbf{X}+\lambda I)^{-1}\mathbf{X}^\top\mathbf{X}\,\beta, \sigma^2(\mathbf{X}^\top\mathbf{X}+\lambda I)^{-1}\mathbf{X}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda I)^{-1}).$$

Elastic Net

When some non zero variables are strongly correlated, the LASSO chooses one of them and the others are shrunk to zero. The Elastic Net estimator was introduced to obtain a sparse estimator that keeps the structure of correlated variables (Zou and Hastie, 2005). The idea is similar to the one of the Ridge estimator : they add, in addition to the constraint on the ℓ_1 norm of the coefficients, a constraint on the ℓ_2 norm.

$$\beta_{\text{elastic net}} = \text{argmin}_\beta \left(|| \mathbf{Y} - \mathbf{X}\,\beta||_2^2 + \lambda \left(\alpha||\beta||_1 + \frac{(1-\alpha)}{2}||\beta||_2^2 \right) \right), \tag{4.1.2}$$

where $\alpha \in [0,1]$. The λ parameter is a penalization parameter that can be chosen by cross validation but the α parameter is chosen arbitrarily in function of the structure of the data, if it is suspected that the data presents high correlation, the value of the parameter will be close to zero, else we keep it close to 1. The case $\alpha = 0$ corresponds to the Ridge estimator and $\alpha = 1$ comes back to the LASSO estimator. Unlike the Ridge estimator, this one is sparse and has no known distribution when $\alpha > 0$.

This method as well as the LASSO and the ridge regression are implemented using the `R` package `glmnet` (Friedman et al., 2010).

SCAD

The SCAD estimator's purpose is to correct the bias of the LASSO estimator (Fan and Li, 2001), however, unlike the de-biased methods investigated here, the correction is made in the construction of the estimator and not a posteriori. As the bias of the LASSO comes from the penalization on the ℓ_1 norm of the coefficients, the penalization depends on the size of the coefficients. The principle of this method is that on high signal variables, the penalization is greatly reduced. The estimator is sparse and under sparsity assumption, it gives coefficients with a very small bias.

$$\beta_{\text{SCAD}} = \text{argmin}_\beta \left(|| \mathbf{Y} - \mathbf{X}\,\beta||_2^2 + \sum_{j=1}^{p} P_\lambda(|\beta_j|) \right), \tag{4.1.3}$$

where the function P_λ is defined by :

$$P_\lambda(|\beta_j|) = \begin{cases} \lambda|\beta_j|, & \text{if} \quad |\beta_j| \leq \lambda. \\ \frac{2\lambda a|\beta_j| - \beta_j^2 - \lambda^2}{2(a-1)}, & \text{if} \quad \lambda < |\beta_j| \leq a\lambda. \\ \frac{\lambda^2(a+1)}{2}, & \text{otherwise.} \end{cases}$$

Usually, the parameter a is set to 3.7. On the penalized function P_λ we observe that when β_j is greater than $a\lambda$, for a certain a, the penalization does not depend on the size of the coefficients anymore. On the other hand, when β_j is estimated as smaller than λ, the penalization is strong and it incentivizes the estimate to be zero. It is very clear if we look at the derivative of the function P_λ as a function of the coefficient β_j :

$$P'_\lambda(|\beta_j|) = \begin{cases} \lambda, & \text{if} \quad |\beta_j| \leq \lambda. \\ \frac{\lambda a - \beta_j}{a-1}, & \text{if} \quad \lambda < |\beta_j| \leq a\lambda. \\ 0, & \text{otherwise.} \end{cases}$$

In the case $|\beta_j| \leq \lambda$, the penalization of the coefficients is the same as the one for classical LASSO but if $|\beta_j|$ is larger than $a\lambda$, there is no more penalization on the size of the coefficient anymore.

There exist other methods with this idea to adapt the penalty to the size of the coefficients. For example, the adaptive LASSO (Zou, 2006) adjusts the penalization with weights that correspond to the inverse of the absolute value of the coefficient estimated by an initial estimator.

The `R` package `ncvreg` will be used to carry out the SCAD estimation.

Exact post selection inference

This last method does not provide an estimator for the coefficients β_j with $j \in \{1, ..., p\}$ but instead it forms confidence intervals using a generic estimator as we detail below.

The general idea of selection inference comes from the following methodology.

With one method, the coefficients of the model are selected. For example, the coefficients are selected if they are estimated different from zero by the LASSO. Other selection methods are available, such as the stepwise regression which adds or drops a variable of the model as a function of a criterion. For the second step, since the model composed by the selected variable is low dimensional, a classical method with a known distribution is used to estimate once more the coefficients, generally the Ordinary Least Squared.

However, this method does not provide correct confidence intervals. Indeed, the choice of the model by the first regression is random and some error can occur, the p-value of the estimated variables will tend to be really small since we disregard that these variables have already been selected so they tend to have a strong correlation with the response variable and to be estimated to be more significant.

To summarize, the idea behind the post selection inference theory is that the choice of the selected model is random and the error in this choice needs to be taken into account to obtain valid result. There are several ways to carry out this inference and in this work we have chosen to use the Exact Post Selection Inference (EPSI) adapted to the LASSO from Lee et al. (2016).

Let us first introduce some notation. We denote by $\mathcal{M}$ a regression model and the true model, under the hypothesis that it exists, is $\mathcal{M}_0$. The selected model will be denoted by $\widehat{\mathcal{M}}$, in this work we obtain it from a LASSO regression. The coefficient for a variable j in the model $\mathcal{M}$ is denoted by $\beta_j^{\mathcal{M}}$. One important thing to have in mind is that the coefficients depend on the model because they are estimated given the effect of other variables. Then for two different models $\mathcal{M}_1$ and $\mathcal{M}_2$ that both contain the variable j, in general $\beta_j^{\mathcal{M}_1} \neq \beta_j^{\mathcal{M}_2}$.

There are two approaches to the problem, the first one is the *conditional coverage*. We are interested in constructing a confidence interval for $\beta_j^{\mathcal{M}}$ only if the model $\mathcal{M}$ is selected, i.e. $\widehat{\mathcal{M}} = \mathcal{M}$. The idea is to condition on this event. Thus, we look for a confidence interval $C_j^{\mathcal{M}}$ such that :

$$\mathbb{P}\left[\beta_j^{\mathcal{M}} \in C_j^{\mathcal{M}}; \widehat{\mathcal{M}} = \mathcal{M}\right] \geq 1 - \alpha.$$

The second approach is the *simultaneous coverage* : For a selected model $\widehat{\mathcal{M}}$, we want to control the probability of making errors instead of focusing on each variable. Berk et al. (2013) introduces the Familywise error rate (FWER) and proposes to choose confidence intervals such that this quantity is under a certain level α.

$$\text{FWER} = \mathbb{P}\left[\beta_j^{\widehat{\mathcal{M}}} \notin \ C_j^{\widehat{\mathcal{M}}}; \quad \forall j \in \widehat{\mathcal{M}}\right].$$

As controlling this probability is a challenging task if there are many variables in the model $\widehat{\mathcal{M}}$, Benjamini and Yekutieli (2005) have suggested focusing on the expected proportion of error :

$$\text{FCR} = \mathbb{E}\left[\frac{|\{j \in \widehat{\mathcal{M}}; \beta_j^{\widehat{\mathcal{M}}} \notin \ C_j^{\widehat{\mathcal{M}}}\}|}{|\widehat{\mathcal{M}}|}; |\widehat{\mathcal{M}}| > 0\right].$$

In this approach, instead of focusing on one coefficient β_j, we have a more global view of the model.

We immediately see that the control of the conditional coverage for all selected variables in a model $\widehat{\mathcal{M}}$ implies the control of FWER or FCR in this model. Indeed, if the probability to belong to the confidence interval, given the model is controlled to be higher than $1 - \alpha$, then the probability of the complementary event is lower than α.

The exact post selection inference address the conditional coverage control. The distribution of the following quantity is studied :

$$\beta_j^{\mathcal{M}} | \{\widehat{\mathcal{M}} = \mathcal{M}\}.$$

Using the KKT conditions for the LASSO minimization problem, we are able to write the event $\{\widehat{\mathcal{M}} = \mathcal{M}, \hat{s}^{\mathcal{M}} = s^{\mathcal{M}}\}$ under the form of a polyhedron $\{y \in \mathbb{R}^n : A(\mathcal{M}, s^{\mathcal{M}})y \leq b(\mathcal{M}, s^{\mathcal{M}})\}$ where A is a matrix and b a vector depending both on $\mathbf{X}^{\mathcal{M}}$, the matrix of the selected variable, and on $s^{\mathcal{M}}$, the sign of the coefficients. In their Theorem 5.2, Lee et al. (2016) obtain the conditional distribution of $\beta_j^{\mathcal{M}}$:

$$\beta_j^{\mathcal{M}} | \{\widehat{\mathcal{M}} = \mathcal{M}, \hat{s}^{\mathcal{M}} = s^{\mathcal{M}}\} \sim \mathcal{TN}\left(\beta_j, \sigma^2 ||\mathbf{X}^{\mathcal{M},\top} e_j||_2^2, V_s^-, V_s^+\right).$$

The $\mathcal{TN}()$ distribution is the Truncated Normal distribution between V_s^- and V_s^+ and these two quantities depend on the matrix $\mathbf{X}^{\mathcal{M}}$ and on $s^{\widehat{M}}$. The result can be extended to an union of polyhedra in order to obtain the distribution of $\beta_j^{\mathcal{M}} | \{\widehat{\mathcal{M}} = \mathcal{M}\}$. Finally, the explicit formula for the confidence interval at level $1 - \alpha$ is given by :

$$C_j^{\mathcal{M}}(\alpha) = \left\{\beta_j^{\mathcal{M}}; \frac{\alpha}{2} \leq F^V(\beta_j^{\mathcal{M}}) \leq 1 - \frac{\alpha}{2}\right\}.$$

The function F is the cdf of the truncated normal distribution of the intervals set $V = \bigcup_{s \in \{-1,1\}^{|\mathcal{M}|}} [V_s^-, V_s^+]$. This recent technique has a different interpretation from the other ones. The confidence interval for the coefficient β_j is conditional on the event that this coefficient has been selected and taking this random selection into account can provide really wide intervals in some cases.

Practically speaking, the estimation of the confidence intervals' lower and upper bounds is constructed by the `R` package `SelectiveInference`.

4.2 The settings

In order to compare the estimators, we use different performance metrics on several designs. In each design, we vary the parameters of the models.

$$Y = \mathbf{X}\,\beta + \epsilon \quad \text{where } \epsilon_i \sim \mathcal{N}(0, \sigma^2), \forall i = 1, ..., n.$$

The coefficient β is a $p \times 1$ vector with only the s first values different from zero and equal to 2, $\beta = (2, \ldots, 2, 0, \ldots, 0)^\top$. The number of variables p and the number of observations n are equal to 100 or 1000 and we define two cases, one with high sparsity when $s = 5$ and another one with weak sparsity, when $s = 50$. The signal-to-noise ratio also defined two cases, the first with $\sigma^2 = 0.8$ is high signal-to-noise ratio and the second with $\sigma^2 = 1.5$ is a low signal-to-noise ratio. We expected that the performances of the estimators will be negatively impacted by a low sparsity or a low signal-to-noise ratio.

The sets of values we used are :

$$p \in \{100, 1000\}, \qquad n \in \{100, 1000\},$$
$$s \in \{5, 50\}, \qquad \sigma^2 \in \{1, 1.5\},$$

$\rho \in \{0.1, 0.8\}$.

The matrix of observations is generated as $\mathbf{X} \sim \mathcal{N}(0, \Sigma_0)$ and the structure of the variance covariance matrix varies. In the first case we consider an equicorrelation structure, all the values of the matrix are the same and for the second case, we look at a Toeplitz correlation structure, the correlation decreases when the variables are more distant in the model.

$$\Sigma_{equi} = \begin{pmatrix} 1 & \rho & \dots & \rho \\ \rho & 1 & \dots & \rho \\ \vdots & & \ddots & \vdots \\ \rho & \rho & \dots & 1 \end{pmatrix}, \qquad \Sigma_{Toeplitz} = \begin{pmatrix} 1 & \rho & \rho^2 & \dots & \rho^p \\ \rho & 1 & \rho & \dots & \rho^{p-1} \\ \rho^2 & \rho & 1 & \dots & \rho^{p-2} \\ \vdots & \vdots & & \ddots & \vdots \\ \rho^p & \rho^{p-1} & \rho^{p-2} & \dots & 1 \end{pmatrix}.$$

The parameter ρ itself defined two case, the first uses a low correlation between the variables, $\rho = 0.1$, and the second high correlated case $\rho = 0.8$.

For each fixed design, we repeated the data generation and the estimation $R = 100$. In all, a total of 56 designs have been created.

4.3 Penalization parameter

The methods presented depend highly on the penalization parameter λ. It determines the number of non zero variables in the LASSO regression. A too small parameter will incorporate too many variables in the model and with a high parameter, we can miss important coefficients. To allow more comparison between the methods, the λ parameter is the same through the different methods based on the LASSO. Both de-biased methods start with the same initial vector, the post selection inference also constructs confidence interval for this coefficient vector.

There is no consensus on the method to select this parameter. However in Miolane and Montanari (2021), it was shown that the de-biased LASSO estimator converges uniformly for all regularization parameter, their simulation study also highlighted that the cross-validation strategy performs well in reducing the risk. This is a common selection method and we describe here how it works : The dataset is divided into K different folds, then for each fold k, we estimate the model using all observations except the ones in the kth fold and we estimate the prediction error for these excluded observations.

$$\text{MSPE}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} (Y_i - \hat{Y}_{i,-k})^2.$$

The number of observations in the fold is n_k and $\hat{Y}_{i,-k}$ denotes the predicted value for individual i when the model is estimated without the kth fold. We define the K-fol cross-validation error and we look for the parameter λ that minimizes this value, on a grid of 100 values,

$$\text{K-cv} = \frac{1}{K} \sum_{k=1}^{K} \text{MSPE}_k,$$

$$\hat{\lambda}_{cv} = \arg \min_{\lambda_1,\ldots,\lambda_l} \text{K-cv}_\lambda.$$

The idea behind the cross-validation is to avoid over-fitting the data and to find a parameter such that the prediction power of the model is better. In our case, we have taken a more empirical approach, the parameter is chosen by following the $1se$ rule which looks for a model with a good prediction power but also with a high sparsity.

This rule suggests taking the greatest $\hat{\lambda}'$ such that

$$\text{K-cv}_{\hat{\lambda}'} \leq \text{K-cv}_{\hat{\lambda}} + \text{SE}_{\hat{\lambda}},$$
$$\text{SE}_{\hat{\lambda}} = \sqrt{\frac{1}{K}\frac{\sum_{k=1}^{K}(\text{K-cv}_{k,\hat{\lambda}} - \text{K-cv}_\lambda)^2}{K-1}}.$$

We choose the most parsimonious model, meaning the model with the greatest sparsity, such that the cross validation error of this model is at most SE away from the minimum cross-validation error.

4.4 Accuracy

For measuring the accuracy of the estimators in each design, we use the bias and the Mean Squared Error. For each $j \in \{1, \ldots, p\}$:

$$\widehat{\text{Bias}}(\beta_j) = \frac{1}{R}\sum_{r=1}^{R}(\hat{\beta}_j - \beta_{j,0}),$$
$$\widehat{\text{MSE}}(\beta_j) = \frac{1}{R}\sum_{r=1}^{R}(\hat{\beta}_j - \beta_{j,0})^2,$$

where $\beta_{j,0}$ is the true coefficient for the variable X_j and R is the number of replication. The bias measures the deviation of the estimation with the true parameter and the MSE takes into account the variability of the estimator.

For more clarity, we compare the average bias on the active and non active variables and the average MSE as well, as we expect significant differences between them. These quantities are computed as follows :

$$\widehat{\text{Bias}}_A = \frac{1}{s}\sum_{j=1}^{s}\widehat{\text{Bias}}(\beta_j), \qquad \widehat{\text{Bias}}_{NA} = \frac{1}{p-s}\sum_{j=s+1}^{p}\widehat{\text{Bias}}(\beta_j),$$
$$\widehat{\text{MSE}}_A = \frac{1}{s}\sum_{j=1}^{s}\widehat{\text{MSE}}(\beta_j), \qquad \widehat{\text{MSE}}_{NA} = \frac{1}{p-s}\sum_{j=s+1}^{p}\widehat{\text{MSE}}(\beta_j).$$

Tables 4.4.1 and 4.4.2 shows these quantities for all the design with equicorrelated structure for Σ. The results for the designs with Toeplitz correlation are located in Appendix, Tables 7.5.5 and 7.5.6. For some designs values in the table are 'NA', it means that the method does not provide enough results for the comparison, we discuss of these events after.

						OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
			design	n	sd	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	-0.01	0.01	0.03	0.04	-0.17	0.00	-1.80	0.04	-0.20	0.00	0.00	0.00
			2	1000	0.8	-0.00	0.00	0.01	0.01	-0.07	0.00	-0.14	0.01	-0.08	0.00	-0.00	0.00
			3	100	1.5	NA	NA	0.03	0.12	-0.24	0.01	-1.89	0.08	-0.38	0.02	0.00	0.00
			4	1000	1.5	-0.00	0.00	0.00	0.01	-0.08	0.00	-0.61	0.03	-0.16	0.01	0.00	0.00
		$\rho = 0.8$	5	100	0.8	-0.08	0.09	0.01	0.19	-0.18	0.08	-1.26	0.51	-0.19	0.09	-0.58	0.23
			6	1000	0.8	0.00	-0.00	0.05	0.05	-0.04	0.00	-0.09	0.08	-0.05	0.00	0.00	-0.00
			7	100	1.5	0.05	0.68	-0.14	0.31	-0.38	0.30	-1.08	0.86	-0.43	0.35	-0.58	0.35
			8	1000	1.5	0.00	-0.00	0.11	0.12	-0.07	0.01	-0.23	0.23	-0.17	0.11	-0.00	0.00
	$s = 50$	$\rho = 0.1$	9	100	0.8	-0.03	0.02	0.04	0.08	-0.31	0.00	-1.80	0.04	-0.38	0.01	0.01	-0.00
			10	1000	0.8	-0.00	0.00	0.01	0.02	-0.13	0.00	-0.19	0.01	-0.14	0.00	-0.00	0.00
			11	100	1.5	-0.33	0.03	-0.10	0.01	-0.47	0.02	-1.88	0.08	-0.67	0.03	-0.03	0.00
			12	1000	1.5	-0.00	0.00	0.00	0.01	-0.14	0.00	-0.66	0.03	-0.27	0.01	0.00	-0.00
		$\rho = 0.8$	13	100	0.8	-0.09	0.10	-0.01	0.20	-0.20	0.10	-1.26	0.51	-0.20	0.11	-0.55	0.23
			14	1000	0.8	0.00	-0.00	0.02	0.03	-0.03	0.01	-0.07	0.06	-0.03	0.01	0.00	-0.00
			15	100	1.5	0.09	0.73	-0.13	0.32	-0.39	0.31	-1.08	0.86	-0.43	0.35	-0.58	0.35
			16	1000	1.5	0.00	-0.00	0.10	0.10	-0.07	0.02	-0.50	0.49	-0.16	0.11	0.00	0.00
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	-0.07	0.03	-0.01	0.05	-0.26	0.00	-1.97	0.01	-0.36	0.00	-0.00	0.00
			34	1000	0.8	-0.00	0.01	0.01	0.01	-0.08	0.00	-1.78	0.01	-0.10	0.00	-0.00	0.00
			35	100	1.5	-0.37	-0.04	-0.16	-0.01	-0.44	0.00	-1.98	0.01	-0.65	0.00	-0.04	0.00
		$\rho = 0.8$	37	100	0.8	-1.05	0.35	-0.85	0.49	-1.59	0.05	-1.85	0.08	-1.60	0.05	-1.69	0.05
			38	1000	0.8	0.03	0.03	0.05	0.05	-0.06	0.00	-1.62	0.08	-0.07	0.00	-0.04	0.00
			39	100	1.5	NA	NA	-1.28	0.35	-1.75	0.08	-1.73	0.09	-1.67	0.08	-1.83	0.08
	$s = 50$	$\rho = 0.1$	41	100	0.8	-0.13	0.06	-0.03	0.09	-0.48	0.00	-1.97	0.01	-0.63	0.00	-0.00	0.00
			43	100	1.5	-0.65	0.05	-0.31	0.02	-0.83	0.00	-1.98	0.01	-1.09	0.01	-0.26	0.00
			44	1000	1.5	NA	NA	-0.10	0.01	-0.30	0.00	-1.97	0.01	-0.45	0.00	-0.02	0.00
		$\rho = 0.8$	45	100	0.8	-1.06	0.35	-0.86	0.48	-1.59	0.05	-1.85	0.08	-1.60	0.05	-1.70	0.05
			47	100	1.5	NA	NA	-0.41	0.63	-1.32	0.06	-1.73	0.09	-1.47	0.07	-1.44	0.06
			48	1000	1.5	NA	NA	0.04	0.08	-0.17	0.01	-1.76	0.09	-0.43	0.02	-0.23	0.01

Table 4.4.1: Average bias on active and non active variables on $R = 100$ repetitions for the desparsified LASSO methods and all the competitors, the correlation structure of the design matrix is the equicorrelated one.

						OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
			design	n	sd	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	3.76	0.20	3.91	0.20	3.21	0.20	0.20	0.19	3.10	0.20	3.82	0.20
			2	1000	0.8	3.80	0.20	3.82	0.20	3.54	0.20	3.29	0.20	3.52	0.20	3.80	0.20
			3	100	1.5	NA	NA	3.90	0.22	2.97	0.20	0.19	0.19	2.52	0.20	4.83	0.20
			4	1000	1.5	3.79	0.20	3.81	0.20	3.52	0.20	1.86	0.20	3.22	0.20	3.80	0.20
		$\rho = 0.8$	5	100	0.8	2.00	1.86	2.21	1.72	1.83	1.86	1.11	1.28	1.81	1.85	5.45	2.30
			6	1000	0.8	2.00	2.00	2.10	1.90	1.91	2.00	1.82	1.85	1.91	1.99	2.00	2.00
			7	100	1.5	6.01	4.20	2.88	1.97	2.61	1.81	1.01	1.02	1.77	1.61	3.77	1.99
			8	1000	1.5	2.00	2.00	2.23	1.77	1.88	1.98	1.79	1.76	1.69	1.82	2.01	2.00
	$s = 50$	$\rho = 0.1$	9	100	0.8	3.75	0.22	4.00	0.21	2.75	0.20	0.20	0.20	2.56	0.20	3.86	0.20
			10	1000	0.8	3.80	0.20	3.85	0.20	3.32	0.20	3.11	0.20	3.28	0.20	3.80	0.20
			11	100	1.5	2.81	0.22	3.59	0.29	2.35	0.20	0.19	0.19	1.80	0.20	5.11	0.20
			12	1000	1.5	3.79	0.21	3.82	0.21	3.28	0.20	1.73	0.20	2.85	0.20	3.81	0.20
		$\rho = 0.8$	13	100	0.8	2.04	1.86	2.23	1.72	1.87	1.83	1.11	1.28	1.84	1.83	5.49	2.38
			14	1000	0.8	2.00	2.00	2.05	1.95	1.94	1.98	1.87	1.89	1.94	1.98	2.00	2.00
			15	100	1.5	6.14	4.21	3.02	2.06	2.65	1.84	1.01	1.02	1.82	1.63	4.61	2.09
			16	1000	1.5	2.01	2.01	2.21	1.80	1.89	1.97	1.63	1.68	1.70	1.81	2.02	2.00
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	3.71	0.03	3.91	0.03	3.02	0.02	0.04	0.02	2.69	0.02	3.96	0.02
			34	1000	0.8	3.96	0.02	3.99	0.02	3.64	0.02	0.08	0.02	3.58	0.02	3.96	0.02
			35	100	1.5	2.69	0.03	3.39	0.04	2.46	0.02	0.04	0.02	1.87	0.02	4.89	0.02
		$\rho = 0.8$	37	100	0.8	1.74	0.47	2.18	0.60	0.91	0.26	0.19	0.19	0.78	0.25	1.51	0.33
			38	1000	0.8	3.93	0.20	3.98	0.19	3.58	0.20	0.27	0.19	3.53	0.20	3.65	0.20
			39	100	1.5	NA	NA	1.02	0.53	0.52	0.36	0.22	0.20	0.47	0.25	0.40	0.37
	$s = 50$	$\rho = 0.1$	41	100	0.8	-0.13	0.06	-0.03	0.09	-0.48	0.00	-1.97	0.01	-0.63	0.00	-0.00	0.00
			43	100	1.5	1.99	0.04	3.02	0.09	1.51	0.02	0.04	0.02	0.96	0.02	4.65	0.02
			44	1000	1.5	NA	NA	3.58	0.03	2.88	0.02	0.04	0.02	2.37	0.02	4.67	0.02
		$\rho = 0.8$	45	100	0.8	1.74	0.47	2.17	0.59	0.92	0.27	0.19	0.19	0.79	0.25	1.44	0.33
			47	100	1.5	NA	NA	4.08	0.74	2.06	0.31	0.22	0.20	0.86	0.25	1.87	0.32
			48	1000	1.5	NA	NA	3.96	0.20	3.19	0.20	0.21	0.20	2.38	0.20	3.01	0.20

Table 4.4.2: Average Mean Squared Error on active and non active variables computed on $R = 100$ repetitions, the correlation structure is the equicorrelated one.

For the bias, we observe first that, as announced, the LASSO estimator is biased for the active variables and that the bias decreases when the sample size increases. The de-biased methods are able to reduce significantly this bias with a small cost on the bias for the non active variables. The design 45 with low sparsity, high-dimension and high correlation is the most difficult for the accuracy. The method with the smallest bias, all designs combined, is the SCAD. It is an expected result since that is the main purpose of the method. On the other hand, the Ridge estimator presents the highest bias but it decreases when the correlation increases, which highlights that the estimator stands the multicollinearity issues. For the design with Toeplitz correlation structure, the bias is much smaller than with equicorrelated correlation but the same remarks apply.

The MSE also takes the variability of the estimator into account. For the non active variables, the measure for the LASSO method and for the two desparsified method are really similar in the cases with $p = 100$ despite the fact that the estimators were no more sparse. On active variables, the MSE of the de-biased method are higher than the one for the LASSO but it remains in the same order of magnitude. Comparing the de-biased methods with each other, we observe that the OD LASSO has slightly smaller MSE, it is expected since the matrix M is chosen such that the estimator has the best performances. The method with the smallest MSE for all the designs is the Ridge. In the cases with Toeplitz correlation, we observe the same behaviour as for the equicorrelated correlation structure except for the SCAD method which has sometimes high MSE on the active variable.

From these results, we conclude that the de-biased methods correct, as expected, the LASSO bias for the active variables and this correction does not have too much cost in terms of MSE. Between the two de-biased methods, despite the fact that the OD LASSO has slightly smaller MSE, none of them really stands out for its accuracy performances.

4.5 Prediction

In this section, we compare the different methods in terms of their prediction power. This is not the purpose of the desparsified methods but it can be interesting to observe whether their estimated coefficients can predict well the response variable outside of the training sample. Therefore, we look at the test error which is the following quantity :

$$\text{Test error} = \frac{1}{m}\sum_{i=1}^{m}(Y_i - \hat{Y}_i)^2,$$

where the observations $Y_i, i = 1, ..., m$ belong to a test set excluded from the construction of the model. The size of the test set m is 15% of the original dataset. This quantity is an unbiased estimator for the Expected Prediction Error of the model. The methods with the smallest expected prediction error will be the most accurate in the predictions.

Table 4.5.1 shows the prediction error for the case where the correlation structure is equicorrelated, the case with Toeplitz correlation is displayed in Appendix, Table 7.5.4. First, for all the designs, the prediction errors for the de-biased LASSO methods are higher than for the other methods, the method with minimal test error is the LASSO. When the sample size is small, $n = 100$, the SCAD method does not perform well, however, what is most obvious

are the poor performances of the desparsified method in high-dimensional setting. When $p = 500$ and $n = 100$, the prediction error explodes for these methods, up to 20 times more than for the other methods. Even if the prediction is not the goal of estimators, such bad results are unexpected. This behaviour is found to a lesser extent in the Toeplitz correlation cases, for design 61 where the SCAD and the LASSO estimator have a test error around 200, the OD LASSO presents 1200 for the test error and the prediction error for the ND LASSO increases up to 2000. In general, the test error is lower for the OD LASSO than for the ND LASSO.

			design	n	sd	**OD LASSO**	**ND LASSO**	**LASSO**	**RIDGE**	**EN**	**SCAD**
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	1	2	1	20	1	1
			2	1000	0.8	1	1	1	1	1	1
			3	100	1.5	NA	70	0	4	0	1
			4	1000	1.5	1	1	1	1	1	1
		$\rho = 0.8$	5	100	0.8	10	25	13	139	12	250
			6	1000	0.8	1	4	1	2	1	1
			7	100	1.5	18077	1414	28	80	18	108
			8	1000	1.5	1	126	7	6	7	1
	$s = 50$	$\rho = 0.1$	9	100	0.8	3	6	2	21	2	2
			10	1000	0.8	2	3	2	3	2	2
			11	100	1.5	109	12	2	4	2	3
			12	1000	1.5	2	4	2	3	2	2
		$\rho = 0.8$	13	100	0.8	15	27	18	148	17	244
			14	1000	0.8	2	3	2	3	2	2
			15	100	1.5	17067	1336	31	84	20	116
			16	1000	1.5	2	110	8	16	8	2
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	98	213	1	23	1	0
			34	1000	0.8	6	14	1	15	1	1
			35	100	1.5	6715	622	1	5	1	1
		$\rho = 0.8$	37	100	0.8	8353	17711	227	152	210	389
			38	1000	0.8	135	348	1	131	2	2
			39	100	1.5	NA	37516	137	60	80	196
	$s = 50$	$\rho = 0.1$	41	100	0.8	322	693	3	23	4	2
			43	100	1.5	9299	909	2	5	2	3
			44	1000	1.5	NA	103	2	6	2	2
		$\rho = 0.8$	45	100	0.8	8246	17424	232	157	217	390
			47	100	1.5	NA	167421	72	57	42	137
			48	1000	1.5	NA	4063	9	59	10	23

Table 4.5.1: Prediction error computed with a test set of size $15\%n$ for the desparsified estimators and the competitors, the number of repetitions is $R = 100$ and the correlation structure of the design matrix is the equicorrelated one.

These results reminds us that the prediction is not the purpose of the de-biased methods and with such poor performances, they should not be used in this case. Between the two, the OD LASSO is the least worst.

4.6 Distribution

The major motivation for the desparsified LASSO methods is their distributions. In this section we verify the promised results by Chapter 3. We measure and compare the coverage and the length of the confidence intervals, defined as

$$\widehat{\text{Cov}}(\beta)_j = \frac{1}{R}\sum_{r=1}^{R} \mathbb{1}_{\{\beta_{j,0}\in C_j(\alpha)\}},$$
$$\widehat{\text{Length}}(\beta)_j = \frac{1}{R}\sum_{r=1}^{R} \text{length}(C_j(\alpha)),$$

where $\mathbb{1}_A$ is the indicator function that returns 1 if the event A is satisfied and 0 otherwise. The length of the interval function $C_j(\alpha)$ is the difference between the upper and the lower bounds, $\text{length}([a,b]) := b-a$.

The coverage measures the number of times the confidence interval recovers the true parameter. As the parameter α is equal to 0.05, we expect a coverage of 95% for the methods. The length of the interval is an indicator for the accuracy of the inference, a small length is desired. As before, to have a clearer comparison, we look at the average of these measures on the active and non active variables.

$$\widehat{\text{Cov}}_A = \frac{1}{s}\sum_{j=1}^{s} \widehat{\text{Cov}}(\beta_j), \qquad \widehat{\text{Cov}}_{NA} = \frac{1}{p-s}\sum_{j=s+1}^{p} \widehat{\text{Cov}}(\beta_j),$$
$$\widehat{\text{Length}}_A = \frac{1}{s}\sum_{j=1}^{s} \widehat{\text{Length}}(\beta_j), \qquad \widehat{\text{Length}}_{NA} = \frac{1}{p-s}\sum_{j=s+1}^{p} \widehat{\text{Length}}(\beta_j).$$

In Table 4.6.1 the measures for the coverage in the cases with equicorrelated structure are presented and Table 7.5.7 presents the length of the corresponding confidence intervals. The measures in cases of Toeplitz correlation can be found in Appendix, Tables 7.5.8 and 7.5.9. First of all, let remind that the EPSI method has a different interpretation that the two others, as we are in the case of a conditional confidence interval, it means the confidence interval is given only for the selected variables by the LASSO methods. Thus when a variable is estimated to zero by the LASSO, the EPSI does not construct inference for it and it is not taken into account for the coverage.

In this context, it is normal that the results for the coverage and the length of the confidence intervals are different than the ones for the other methods. Indeed an active variable not selected by the LASSO will have associated a confidence interval and the coverage for the active variable will probably be higher. Also, the mean is not taken on the same number of variables for each repetition because it depends on the selected set.

Looking at Table 4.6.1, we observe that the coverage for the EPSI method is really high, somewhat too large for the 95% target. For the de-biased methods, the coverage for the non active variables is close to the target in the majority of cases but for the active variables, the

estimators perform less well. When the correlation is low, $\rho = 0.1$, the coverage for the active variables is around 75%. For example, design 43 has poor coverage of 65% for the active variables. In general, the coverage of OD LASSO is better than the one of ND LASSO.

In the cases of Toeplitz correlation, the coverage is even lower when $\rho = 0.1$, it is somewhat expected since the correlation in weaker for this structure. However, it is surprising that the methods perform so poorly in these cases. As the results for the accuracy were not really impacted by a low correlation, it seems that the issue comes from the construction of the estimators' variance.

Table 7.5.7 in Appendix shows that the length of the confidence intervals are much larger for the OD LASSO when the correlation is high. For example, in design 5, the length for the OD LASSO is ten times the one for the ND LASSO but the coverage of the OD LASSO is also much closer to its target. For the ND LASSO estimator, the lengths are higher when the signal-to-noise ratio is low. In general, EPSI presents confidence intervals with larger length, it is expected since the method proposes conditional confidence intervals that take into account the choice of the model.

In conclusion, the desparsified methods present good coverages for testing the non active variable. For the active variables, the coverage for the desparsified methods is not really satisfying when the correlation is low but when the correlation is high, the length of the confidence intervals for ND is correctly adjusted and the estimator performs well.

4.7 Convergence issues and numerical stability

A certain number of times, the estimation leads to an error in the simulation, it has happened for the OD LASSO and for the EPSI methods. These errors, coupled with the long computing time for the desparsified estimator, have forced us to lower our ambitions and to present the results only on 100 repetitions instead of the 500 planned. In the tables, when there are not enough results for a method, we have replaced its value by 'NA' and the methods without any errors are computed on the first 100 obtained repetitions. In this section, we explore these errors and try to explain them.

For the EPSI method, the error takes the form of incoherent values for the confidence interval bounds, for example the lower bound will be higher than the upper one and with some very high values such as 34 and the upper one is -42 while the true coefficient is equal to 2. For some other variables in the same repetitions, the bounds are simply set to be infinite.

The authors of the `R` package explain that their optimization problem can fail, indeed they have to find a minimizer for a polyhedra union and the problem is not always guaranteed to be solvable. Moreover, they indicate that their implemented code is really sensitive to the design. To increase the number of successes for our simulations, we have raised the number of the number repetitions up to 5000 and even there, for several designs, we were not able to obtain the 100 required values for the comparison.

Table 4.7.1 indicates the percentage of success for the OD LASSO and the EPSI computed on the first $R = 100$ repetitions. The most problematic designs for the EPSI are the design 7, design 15, design 38, design 48 and design 55, the methods does not provide one estimator after all the repetitions. Except for the 55, for each of these design, the structure of the

variance-covariance matrix is equicorrelate. The only difference between designs 7 and 15 is the sparsity, both of them have 100 variables for 100 observations, a high correlation and a low signal-to-noise. The success of the methods seems highly influences by the correlation in the design.

In contrast, the design where no error has occurred is the 24 with a low-dimensional setting and a Toeplitz structure for the correlation, however for this model the sparsity is low and $\rho = 0.8$. This leads us to conclude that the EPSI method is highly sensitive to the correlation structure of the covariates, the level of the noise and the number of observations, however, the sparsity does not affect the numerical stability of the method.

			design	n	sd	**OD LASSO**	**EPSI**
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	1.00	0.46
			2	1000	0.8	1.00	0.90
			3	100	1.5	0.42	0.10
			4	1000	1.5	1.00	0.37
		$\rho = 0.8$	5	100	0.8	1.00	0.09
			6	1000	0.8	1.00	0.53
			7	100	1.5	0.70	0.00
			8	1000	1.5	1.00	0.06
	$s = 50$	$\rho = 0.1$	9	100	0.8	1.00	0.45
			10	1000	0.8	1.00	0.91
			11	100	1.5	0.40	0.11
			12	1000	1.5	1.00	0.33
		$\rho = 0.8$	13	100	0.8	1.00	0.12
			14	1000	0.8	1.00	0.45
			15	100	1.5	0.70	0.00
			16	1000	1.5	1.00	0.05
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	1.00	0.10
			34	1000	0.8	1.00	0.69
			35	100	1.5	0.06	0.43
		$\rho = 0.8$	37	100	0.8	1	0.67
			38	1000	0.8	1.00	0.00
			39	100	1.5	0.22	0.00
	$s = 50$	$\rho = 0.1$	41	100	0.8	1.00	0.10
			43	100	1.5	0.39	0.01
			44	1000	1.5	0.20	0.05
		$\rho = 0.8$	45	100	0.8	1.00	0.00
			47	100	1.5	0.19	0.00
			48	1000	1.5	0.02	0.00

Table 4.7.1: Success rate of the EPSI and OD LASSO methods computed on $R = 100$ repetitions.

For the OD LASSO, the errors happen more when the design matrix has a large number of variables. When the minimization problem 2.2.6 cannot be solved for an index $j \in \{1, \ldots, p\}$,

the algorithm 1 replaces m_j by e_j the jth unit vector in $\mathbb{R}^p$ and after that, the matrix M is constructed with all of these vectors. If the optimization problem cannot be solved for all the variables, the matrix M becomes the identity matrix. However in this case, the conditions to obtain a good estimator are not respected, we do not have any guarantee on the behaviour of $||\Sigma M - I||_2$. It leads that the de-biased estimator will no more converge to the true parameter. Moreover, a problem occurs when estimating the variance and this is what returns us an error in the author's code : the estimated variance is infinite and all variables are included in the model.

In Table 4.7.1, we observe that the errors happen in the $p = 1000$ cases especially when the matrix Σ has a high correlation, when the matrix has an equicorrelated structure. A low signal-to-noise ratio also increases the chance of failure. That indicates that if the variables are highly correlated with small signal, the method can often fails, especially if number of variables exceeds the number of observations.

In Theorem 3.1.11, the authors exposed the conditions under the ones the problem is solvable. More precisely, they bound the probability that the problem is not solvable by $2p^c$ where c is a constant depending on κ, the bound for the sub-Gaussian norm of the matrix $\mathbf{X}\,\Sigma^{-1/2}$, and on Λ_1 and Λ_p the minimal and maximal eigenvalues of Σ. They precise also that their theorem applies if the number of observations is large enough compared to p and s.

We observe well that the errors happen in case of the correlation between the variables is high which impacts the parameters κ as well as Λ_1 and Λ_p. And we find issues if n is too small compared to p, which suggests that there are not enough observations to guarantee the second part of Theorem 3.1.11.

4.8 Non Gaussian noise

Before finishing the chapter, we explore a last variation of the setting where we continue the discussion initiated in the end of previous Chapter. The normality of the noise is relaxed and we want to know if we keep the normal distribution of the estimators. In Chapter 3, we have proved that the OD LASSO estimator will keep its distribution but not necessarily the ND LASSO one.

We perform the simulations on a reduced setting. The noise is generated with an exponential distribution and the rate m takes 1 or 0.667 as value, the variance of the exponential is $1/m^2$ so $m = 1$ corresponds to a high-signal to-noise ratio, and $m = 0.667$ is low signal-to-noise ratio with the same variance as for the low signal variance for Gaussian noise of level $\sigma = 1.5$. In the majority of cases, we are under a high-dimensional setting. We define a case with high sparsity $s = 5$ and another with low sparsity $s = 40$, and a case with high correlated variables $\rho = 0.8$ instead of $\rho = 0.1$, the structure of the covariance matrix is always the equicorrelated one. The data generation process and the estimation model is repeated $R = 100$ under 16 settings.

$$p \in \{100, 500\}, \qquad n \in \{50, 100\},$$
$$s \in \{5, 40\}, \qquad \rho \in \{0.1, 0.8\},$$
$$m \in \{1, 0.667\}.$$

						OD LASSO		**ND LASSO**		**EPSI**	
			design	n	sd	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	0.76	0.97	0.72	0.91	0.98	0.99
			2	1000	0.8	0.76	0.94	0.76	0.93	0.96	1.00
			3	100	1.5	NA	NA	0.72	0.95	0.95	0.99
			4	1000	1.5	0.75	0.94	0.76	0.94	0.95	0.99
		$\rho = 0.8$	5	100	0.8	0.98	1.00	0.72	0.83	1.00	1.00
			6	1000	0.8	0.98	1.00	0.74	0.72	0.99	0.99
			7	100	1.5	0.96	1.00	0.91	0.96	NA	NA
			8	1000	1.5	0.98	1.00	0.98	0.98	1.00	1.00
	$s = 50$	$\rho = 0.1$	9	100	0.8	0.76	0.97	0.75	0.93	0.99	0.97
			10	1000	0.8	0.76	0.94	0.76	0.93	0.95	0.99
			11	100	1.5	0.62	0.98	0.71	0.95	0.96	0.99
			12	1000	1.5	0.75	0.94	0.76	0.94	0.94	1.00
		$\rho = 0.8$	13	100	0.8	0.98	1.00	0.72	0.84	1.00	1.00
			14	1000	0.8	0.98	1.00	0.90	0.90	0.96	0.99
			15	100	1.5	0.96	1.00	0.91	0.96	NA	NA
			16	1000	1.5	0.98	1.00	0.97	0.97	1.00	1.00
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	0.75	1.00	0.70	0.94	0.98	1.00
			34	1000	0.8	0.78	0.98	0.76	0.93	0.96	0.99
			35	100	1.5	0.68	1.00	0.66	0.99	0.98	1.00
		$\rho = 0.8$	37	100	0.8	0.78	0.99	0.77	0.97	NA	NA
			38	1000	0.8	0.97	1.00	0.75	0.73	NA	NA
			39	100	1.5	NA	NA	0.98	1.00	NA	NA
	$s = 50$	$\rho = 0.1$	41	100	0.8	0.75	1.00	0.71	0.95	0.99	1.00
			43	100	1.5	0.65	1.00	0.65	0.99	0.97	1.00
			44	1000	1.5	NA	NA	0.78	0.96	0.89	1.00
		$\rho = 0.8$	45	100	0.8	0.77	0.99	0.77	0.97	NA	NA
			47	100	1.5	NA	NA	0.98	1.00	NA	NA
			48	1000	1.5	NA	NA	0.98	1.00	NA	NA

Table 4.6.1: Average coverage on active and non active, computed on $R = 100$ repetitions. The correlation structure of the design matrix is the equicorrelated one.

					OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
		design	n	m	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	1	-0.13	0.07	-0.02	0.13	-0.48	0.01	-1.88	0.03	-0.59	0.01	-0.01	0.00
		2	100	1	-0.02	0.02	0.05	0.08	-0.32	0.00	-1.80	0.04	-0.36	0.01	-0.00	-0.00
		3	50	0.667	-0.20	0.10	-0.03	0.20	-0.73	0.01	-1.89	0.03	-0.85	0.02	-0.02	0.00
		4	100	0.667	-0.04	0.03	0.07	0.12	-0.48	0.00	-1.82	0.04	-0.54	0.01	-0.00	-0.00
	$s = 40$	5	50	1	-0.49	0.49	-0.21	0.73	-0.92	0.26	-1.48	0.36	-0.90	0.27	-1.08	0.26
		6	100	1	-0.06	0.06	-0.01	0.13	-0.16	0.06	-1.37	0.40	-0.17	0.06	-0.46	0.14
		7	50	0.667	-0.48	0.49	-0.20	0.74	-0.92	0.27	-1.48	0.36	-0.91	0.27	-1.07	0.26
		8	100	0.667	-0.09	0.08	-0.03	0.15	-0.21	0.08	-1.37	0.40	-0.22	0.09	-0.46	0.14
$p = 500$	$s = 5$	9	50	1	-0.31	0.08	-0.16	0.13	-0.71	0.00	-1.96	0.01	-0.96	0.00	0.00	0.00
		10	100	1	-0.08	0.04	0.01	0.08	-0.40	0.00	-1.93	0.01	-0.51	0.00	-0.00	0.00
		11	50	0.667	-0.44	0.14	-0.19	0.22	-1.08	0.00	-1.97	0.01	-1.27	0.01	-0.05	0.00
		12	100	0.667	-0.13	0.06	0.01	0.13	-0.60	0.00	-1.94	0.01	-0.73	0.00	-0.00	0.00
	$s = 40$	13	50	1	-0.75	0.66	-0.30	1.01	-1.67	0.07	-1.82	0.11	-1.68	0.07	-1.73	0.06
		14	100	1	-0.89	0.19	-0.73	0.31	-1.20	0.07	-1.77	0.12	-1.22	0.07	-1.45	0.06
		15	50	0.667	-0.74	0.67	-0.28	1.02	-1.67	0.07	-1.82	0.11	-1.68	0.07	-1.72	0.06
		16	100	0.667	-0.89	0.19	-0.73	0.32	-1.22	0.07	-1.77	0.12	-1.23	0.07	-1.45	0.06

Table 4.8.1: Bias estimation when the distribution of the noise is an Exponential with rate m equals to 1 or 0.667. The results are presented for 16 different designs, each repeated $R = 100$.

As previously, we measure the average bias on null and active variables in Table 4.8.1, the average MSE in Appendix, Table 7.5.12. The out of sample error is estimated on a new dataset of size corresponding to 15% of the original n in Table 4.8.2. The average coverage are shown in Table 4.8.3 and the average confidence interval's lengths are shown in Appendix, Table 7.5.11

The accuracy is not really impacted by the change in the noise distribution and the de-biased methods perform at the same scale as the other methods. As before, the bias and the MSE for the sparse methods are really small, and the non active variables, the SCAD has the smallest bias. These results agree with the theory, for the OD LASSO estimator, we have shown that the convergence holds in case of non Gaussian error and for the ND LASSO, even if the distribution of the estimator is not guaranteed, they have argued that the estimator stays close to its target.

		design	n	m	**OD**	**ND**	**LASSO**	**RIDGE**	**EN**	**SCAD**
$p = 100$	$s = 5$	1	50	1	12	36	3	10	3	2
		2	100	1	4	8	4	26	4	2
		3	50	0.667	27	76	7	13	7	5
		4	100	0.667	9	17	9	30	9	5
	$s = 40$	5	50	1	191	492	169	229	149	365
		6	100	1	8	16	9	116	9	142
		7	50	0.667	200	510	178	235	156	372
		8	100	0.667	16	23	17	124	17	146
$p = 500$	$s = 5$	9	50	1	288	726	3	10	4	1
		10	100	1	49	188	4	25	4	2
		11	50	0.667	721	1853	7	11	7	4
		12	100	0.667	108	414	8	28	9	5
	$s = 40$	13	50	1	13132	35931	332	201	309	469
		14	100	1	524	2016	109	109	101	236
		15	50	0.667	13562	37200	343	209	321	475
		16	100	0.667	547	2088	115	113	107	238

Table 4.8.2: Prediction error measured with a new test sample of size 15%n for an Exponential noise with parameter m for the 16 designs.

For the prediction also, the results under non Exponential noise are similar to the ones under Gaussian noise. The out-of-sample error explodes for the de-biased methods when the sparsity is low and the number of variables is larger than the number of observations.

Looking at the coverage, we see something interesting for the ND LASSO methods. In the cases where $p = 100$, the coverage under the null and the active variable is really high, around 98% but when the number of variables increases to 500, it fails below 70% for the active variables and below 80% for the null variables. For the OD LASSO method, the coverage under the active variables is not really satisfying for the cases with $p = 100$ and a high sparsity but for rest, the results are quite correct. For the EPSI methods, the confidence intervals have in average a coverage of 1 with a high value for the lengths, except for the design 13 where the method misses. The 'NA' values indicate the design where the methods were not able to reach the $R = 100$ repetitions for the comparison, even under 5000 simulations.

					OD LASSO		**ND LASSO**		**EPSI**	
		design	n	m	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	1	0.71	0.97	0.80	0.99	0.67	1.00
		2	100	1	0.74	0.96	0.80	0.99	1.00	1.00
		3	50	0.667	0.71	0.97	0.78	0.98	1.00	1.00
		4	100	0.667	0.74	0.96	0.79	0.98	1.00	1.00
	$s = 40$	5	50	1	0.84	0.95	0.97	0.99	1.00	1.00
		6	100	1	0.96	0.98	0.98	0.99	NA	NA
		7	50	0.667	0.85	0.95	0.97	0.99	1.00	1.00
		8	100	0.667	0.97	1.00	0.98	0.99	NA	NA
$p = 500$	$s = 5$	9	50	1	0.67	1.00	0.47	0.77	0.95	0.91
		10	100	1	0.75	0.99	0.67	0.83	0.97	0.97
		11	50	0.667	0.71	1.00	0.39	0.61	0.83	0.73
		12	100	0.667	0.75	0.99	0.55	0.69	0.88	0.90
	$s = 40$	13	50	1	0.93	0.99	0.74	0.79	0.57	0.35
		14	100	1	0.43	0.96	0.49	0.94	NA	NA
		15	50	0.667	0.94	0.99	0.74	0.78	NA	NA
		16	100	0.667	0.45	0.96	0.49	0.94	NA	NA

Table 4.8.3: Coverage for 95% confidence intervals in case of Exponential noise, on active variables the de-biased methods perfom badly, especially the ND LASSO one for which no theoretical results support the validity of its distribution.

We conclude that in Exponential error, when the number of variables is much higher than the number of observation, the ND LASSO does not estimate correctly the confidence interval. This is not something that contradicts the theory as there is no result for guaranteeing the normality of the ND LASSO. The OD LASSO presents confidence intervals which are slightly too large for the null variables but that are usable.

Chapter 5

Numerical results : Real data analysis.

After exploring the methods through simulations, we finish this first part by presenting the results of the estimators on some real datasets.

5.1 High-dimensional data set

The first data set is *riboflavin* which was provided by DSM (Switzerland), a health, nutrition and bioscience company and introduced by Bühlmann et al. (2014). It inspects the riboflavin production as a function of numerous genes. The response variable is the logarithm of the riboflavin production rate in Bacillus subtilis and the $p = 4088$ explanatory covariates consist in logarithms of the gene expressions level.

This dataset was already used by van de Geer et al. (2014) as well as Javanmard and Montanari (2014) to explore the performances of their method but here we chose the parameter λ by the $1se$ rule explained in the previous Chapter instead of the one find by cross validation for van de Geer et al. (2014) or $\lambda = 4\hat{\sigma}\sqrt{(2\log p)/n}$ for Javanmard and Montanari (2014). This parameter is the same for both methods for a better comparison. The original idea was to test the methods on another dataset, unfortunately, on all the tested datasets, the OD LASSO method fails to optimize the program 2.2.6 as explained in Chapter 4. We therefore fell back on the only high-dimensional dataset for which the method obtains results. For the same reason, we do not evaluate the prediction power of the method because when removing any of the observations, the OD LASSO also fails the estimation.

The LASSO estimator estimates that 30 genes are different than zero, the estimator from van de Geer et al. (2014), with this vector as initial value, finds 40 genes with confidence intervals that does not contain zero at level 95%. On the other hand, the alternative method of Javanmard and Montanari (2014) only selects 5 genes.

Figure 5.1.1 presents the results for the methods, the black lines are the confidence intervals for the significantly non null variables. For more clarity, we do not present the ones for the estimated null variables. We observe that for the OD LASSO estimator, the coefficients are estimated closer to zero.

The number of estimated coefficients is really different across the methods, the OD LASSO is the most parsimonious one, however, if we look to the corrected p-value using Hochberg adjustment (Benjamini and Hochberg, 1995) we find that the ND LASSO no longer selects any variable, as in van de Geer et al. (2014) and the number of selected variables by OD LASSO increases to 19.

The corrected p-values are used when we test the significance of several parameters with individual p-values. Indeed, the probability to make an error is no more controlled when multiple testing. The idea of the Hochberg correction is to modify the p-values in order to control the Familywise Error Rate FWER and this method is heavily used, especially in cases like this with data from genes. Other corrections are available as the well-known Bonferroni adjustment (Dunn, 1961) which is sometimes too restrictive.

Figure 7.5.3 in Appendix shows the boxplot of the confidence intervals' lengths. Even if the ND LASSO selects fewer variables, its confidence intervals have larger length and the OD LASSO estimators has some outliers for which the confidence intervals lengths are really small.

5.2 High-dimensional dataset with random permutations

The issue with real data is that we have no idea of what we are looking for, indeed we do not know the true model. The idea here is to destroy the correlation structure of a real dataset in order to have something to compare to, we expect that there are no selected variables after a random permutation of the data.

To proceed, we take again the riboflavin dataset with 71 observations and 4088 variables and we randomly permute all the observations in each column. These permutations are obviously independent and without replacement and the response variable is left unchanged. With this little trick, if there was a link between the response variable and a covariate in the original dataset, it is now completely destroyed. We then fit this new dataset with the LASSO and the desparsified methods.

First of all, the OD LASSO method returns an error, the algorithm failed to solve the minimization program 2.2.6. We have tested different permutations possible but the OD LASSO does not succeed even once.

The LASSO estimator selected only 1 variables in the model but if we look at Figure 5.2.1, we observe that increasing the penalization decreased always the parameter λ. The minimum is not unique and the parameter is so chosen at the edge where it remains only one variable.

Using the LASSO solution as initial value for the ND LASSO gives an estimator that selected 2678 variables on the 4088, i.e. 65% of the variables are each estimated significantly different from zero. This is a very surprising result given that we had expected to select no variable. If we look to the selected variable using the corrected p-values of Hochberg, we obtain still 1254 significant coefficients. We compare this result with the one where we take the default parameter for the ND LASSO and this case, the method selects 197 variables by looking to the individual p-values and finally none variables looking at the corrected p-value.

One thing that can explain these different results is the choice of the λ parameter. Then such results imply that the method is really sensitive to its choice. Thus when using the ND

LASSO, the choice of the regularization parameter needs to be discussed.

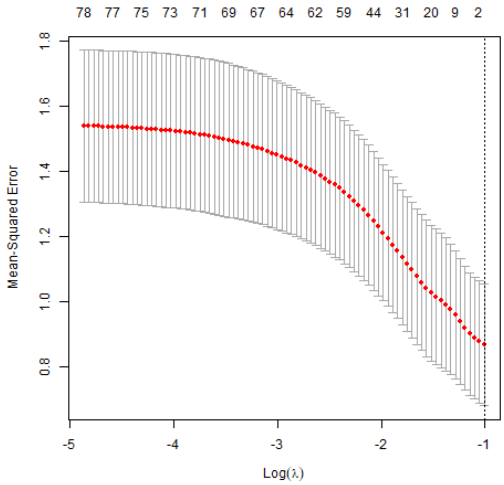

Figure 5.2.1: Mean Squared Error for the LASSO estimator in function of the regularization parameter, the function is decreasing.

5.3 Low dimensional dataset

To finish the chapter, we also look at the performances on a low-dimensional dataset containing $n = 442$ individuals. The goal is to analyse what are the factors that influence diabetes. We model a linear regression with the 10 covariates and their interactions, which, taking into account the binary variables, brings the number of variables to $p = 65$ if we add an intercept. This data set was used in Efron et al. (2004), the explanatory are composed of the age of the individuals, its sex, its body max index BMI and its average blood pressure BP, in addition there are 6 measurements on its blood. We scale the covariates and we exclude a sample of 15% of the observations to be able to calculate afterwards the prediction error on out-of-sample data. Since we are under a low-dimensional setting, the Ordinary Least Squares estimator is also used as competitor.

The LASSO estimates 2 non null variables and by using 95% confidence intervals we find that the ND LASSO selects 8 variables, the OD LASSO 5 and the sparest method is the OLS with only 4 significant coefficients. If we compare them, we observe that the models share some variables but there are also several variables which are only selected in one model and no variable is present in all the models. One variable is selected in all the models, it is the Body Mass Index BMI, then the Mean Arterial Pressure MAP variable is selected in all models except the LASSO. If we select the models based on the corrected p-value with the Hocheberg method, we observe that the ND LASSO and the OD LASSO only select 5 variables.

Figure 5.3.1 shows the estimated coefficients for the different methods with the confident intervals only display for the selected variables. We can observe that the ND LASSO method is very close to zero while the OD LASSO and the OLS are located further away.

Without surprise, the OLS estimator has the lowest out-of-sample error (4 058), shortly after,

Variables	**OLS**	**LASSO**	**ND LASSO**	**OD LASSO**
intercept	selected			selected
bmi	selected	selected	selected	selected
age			selected	
map	selected		selected	selected
ltg		selected		selected
tch			selected	selected
sex	selected		selected	
glu^2			selected	
hdl^2			selected	
tc*tch			selected	

Table 5.3.1: Selected variables by methods for the diabetes dataset, the displayed variables are the ones selected at least by one method.

the one of the OD LASSO is the second (4 879) and the ND LASSO as well as the LASSO present much larger value with respectively a prediction error of 25 252 and 29 191. Contrary to the simulation cases, the prediction error for the de-biased methods are better than the one for the LASSO and the result for OD LASSO is very encouraging.

In Figure 7.5.4 in Appendix, we can compare the length of the intervals for the different methods. What resort first is a huge difference between the OLS method and the two desparsified one. Even if the OLS has the best prediction error, the precision for the inference is really weak, the ND LASSO has slightly larger confidence intervals but the median is almost the same as for the OD LASSO.

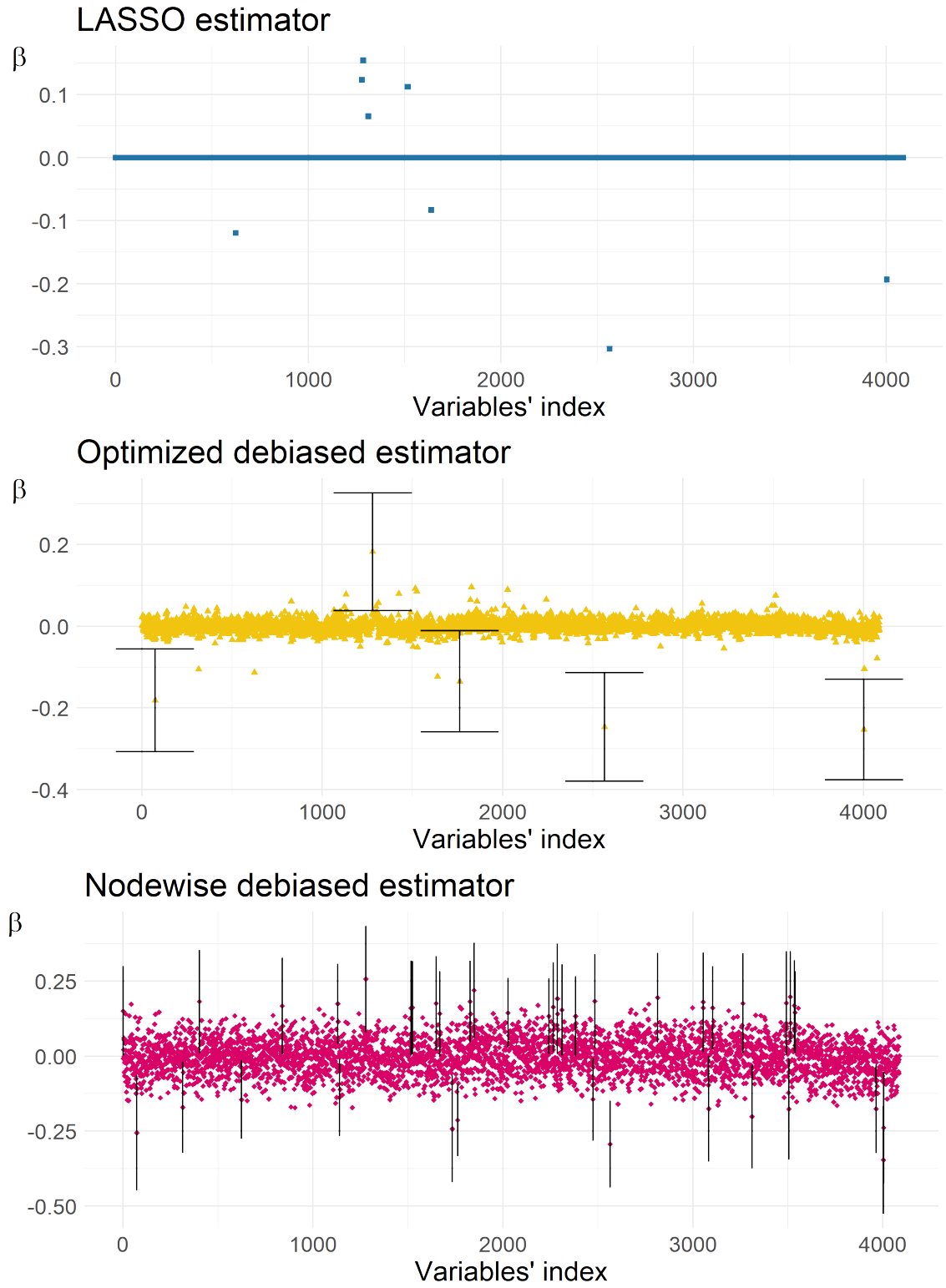

Figure 5.1.1: Estimated coefficients for the riboflavin dataset, for the LASSO estimators and the desparsified ones. The confidence intervals are shown only for the significant coefficients.

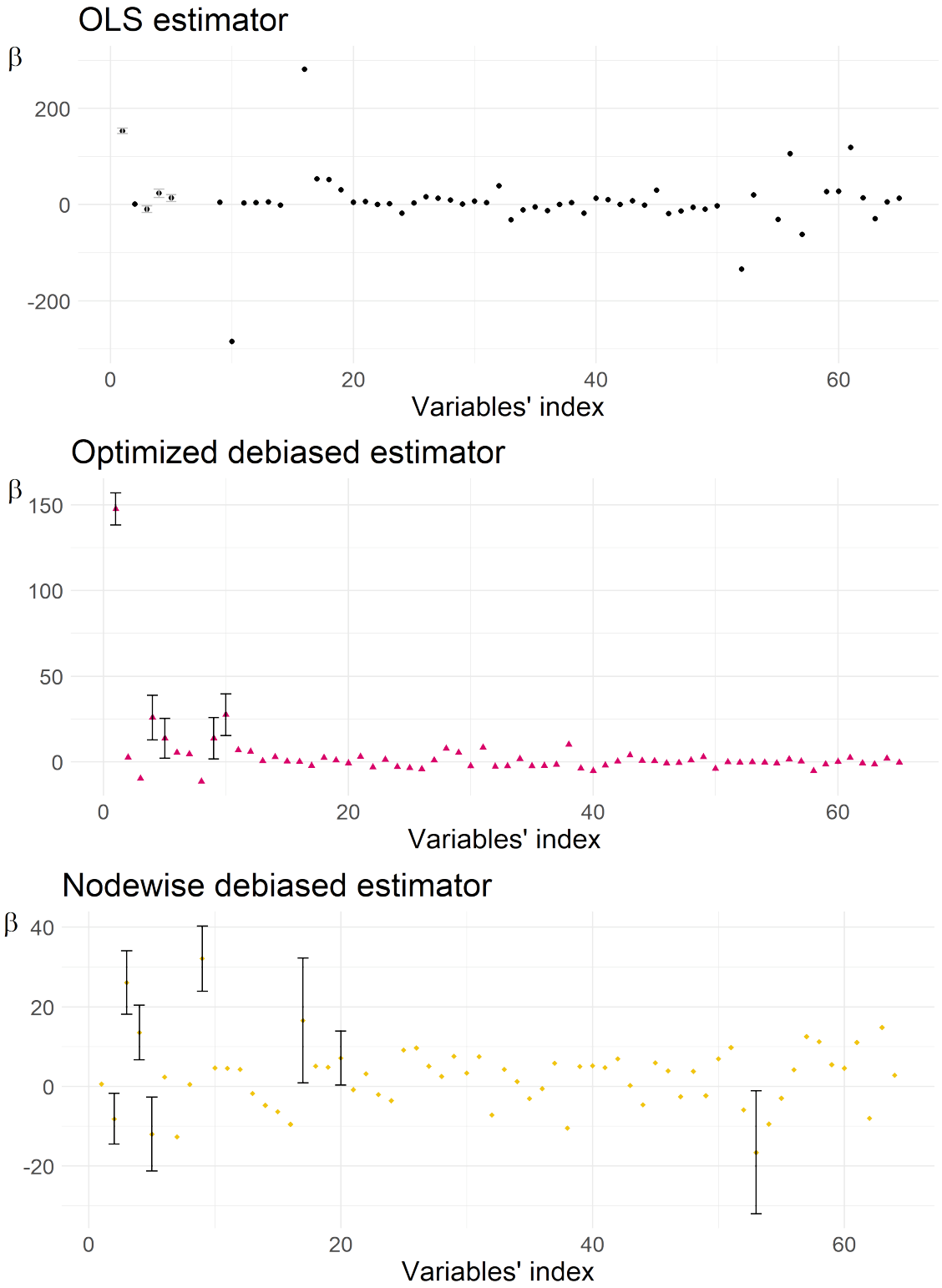

Figure 5.3.1: Estimated coefficients for the diabetes dataset, the confidence intervals are shown only for the estimated non null variables.

Part II

Extensions

Chapter 6

Model averaging.

The LASSO estimator depends on a penalty parameter : the λ in the equation 1.0.3. In Chapter 4 and Chapter 5, we have chosen this parameter following the $1se$ rule and cross-validation. However, the choice of this parameter is not that easy. There exist several methods that proposes a good parameter but nothing to guarantee that this parameter is the best. In some cases, the estimator can be really sensitive to the choice of the penalized parameter and choosing one instead of another gives different results. Moreover, the methods are often costly in terms of time and computation.

Recently, a method has been proposed to get rid of this challenging choice, the Model Averaging estimator. In this chapter, we will discuss on this method and look at how it can be applied to the de-biased LASSO estimators.

6.1 Model averaging methodology

Model averaging for the regression case has been proposed as a way to take into account the model selection uncertainty. Indeed, even in low dimensions, model selection can often lead to biased coefficients for finite sample settings and we can find too small standard errors (Buckland et al., 1997). This is so because the uncertainty regarding the choice of the model is not directly represented in the standard errors.

In model averaging, instead of choosing one particular model, one proposes a weighted mean of several models. That is, given K estimators $\hat{\beta}_k, \quad k = 1, \ldots, K$, the purpose is to combine these estimators to create a new estimator

$$\hat{\beta}_{\text{MA}} = \sum_{k=1}^{K} w_k \widehat{\beta}_k,$$

where the w_k represent a set of weights. To find appropriate weights, often we base ourselves on information criteria such as the Akaike Information Criterion (AIC, Akaike, 1973) and Bayesian Information Criterion (BIC, Schwarz, 1978) as in Palm and Zellner (1992).

The asymptotic risk of the averaging estimator was shown to be smaller than the non averaged one where the Least-Squares estimator is used and when the weights are selected based on a

penalized least squares criterion (Hansen, 2014).

Another goal of the method is to increase the prediction power of the estimator and to do so, the weights are chosen in order to minimize some risk function. For example, Hansen (2007) constructs an estimator based on Mallow's C_p criterion and they show it is asymptotically optimal in terms of prediction. However, this objective of optimal model averaging does not necessarily go hand in hand with the initial one, namely to account for model uncertainty, as shown in Schomaker and Heumann (2020).

In this chapter, we will present a method on how to adapt this technique to the desparsified LASSO estimator. Up to now, the method has been proposed for the classical LASSO estimator with good results (Feng and Liu, 2020) but there are no proposals yet to mix the advantages of the desparsified LASSO estimator and the Model Averaging method.

6.2 Model averaging adapted to LASSO estimator

Due to the ℓ_1 sparsity enforcing penalty, the LASSO estimator can also be viewed as a model selection tool. Indeed, the estimator selects some coefficients to be non-zero while the rest are shrunk to 0 and thus, correspond to variables excluded from the model. As for the post selection inference method described in the previous chapter, the model selected by the LASSO is

$$\widehat{\mathcal{M}}_\lambda = \{j = 1, \dots, p; \hat{\beta}_{LASSO,j}(\lambda) \neq 0\}.$$

Feng and Liu (2020) proposes to exploit the regularization parameter to construct nested models. Indeed, the number of variables included in the model is a decreasing monotonic function of λ. When the parameter is null, all variables are included in the model $\widehat{\mathcal{M}}_{\lambda=0}$ and when the parameter increases, the model contains fewer and fewer variables.

In the model averaging, we do not choose one model but we take instead a combination of the best models defined by λ. For the LASSO estimator, this method works as follows :

Step 1. We first estimate the regression coefficients with a LASSO method on a grid of, say, K different λ values.

Step 2. For each λ_k (with $k = 1, \dots, K$), we evaluate the 'relative importance' of the model using an information criterion (IC) and set weights associated to each penalty parameter. A popular choice is to use Exponential smoothing weights of the form

$$w_{\mathcal{M}} = \frac{\exp(-IC_{\mathcal{M}}/2)}{\sum_{k=1}^{K} \exp(-IC_k/2)},$$

where $IC_{\mathcal{M}}$ is the value of the IC for model $\mathcal{M}$. These weights come from Bayesian theory, where the probability that a model is true, conditionally on the data, is proportional to $\exp(-BIC/2)$. Other weights are feasible but we will focus on these for the moment.

Step 3. Finally, with the weights computed in step 2, we obtain the LASSO average estimator

$$\hat{\beta}_{\text{MA}} = \sum_{k=1}^{K} w_k \hat{\beta}_{LASSO}(\lambda_k).$$

Our proposal is to apply this idea to the desparsified LASSO estimator, as the proposal is supported by the result on Miolane and Montanari (2021) that states that the desparsified estimator convergence is uniform for the regularization parameter. This idea is novel and there is no theory yet that ensures that the model averaging applied to the desparsified LASSO will give a good estimator or that we keep its distribution.

The method follows the same principles as the one for the model averaging LASSO, only the last part of the method has to be modified. The following steps take place just after the second one where the weights are constructed using the information criteria from the LASSO models.

Step 2.5 For each of the K different models defined by the choices of λ_k, we debias the LASSO estimator next :

$$\hat{\beta_k^d} = \hat{\beta}_{LASSO}(\lambda_k) + \frac{1}{n} M \mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\hat{\beta}_{LASSO}(\lambda_k))$$

Step 3 Finally, we obtain the new model averaged estimator

$$\widetilde{\beta}_{MA} = \sum_{k=1}^{K} w_k \hat{\beta}_k^d$$

where w_k are the weights derived in Step 2 and $\hat{\beta}_k^d$ are the de-biased estimators from Step 3.

The final estimator $\widetilde{\beta}_{MA}$ does not depend on the penalization parameter anymore. Moreover, the cost in time is of the same order as the one from the desparsified LASSO method. Indeed, the step that requires the most computation time is the construction of the matrix M but it only depends on the design matrix $\mathbf{X}$, thus it can be computed once for all the estimators $\hat{\beta}_k^d$..

The main advantage and the purpose of the de-biased LASSO estimator was its distribution. As the weights for the mean of the estimators are random and not independent, we cannot conclude that the weighted mean keeps the Gaussian distribution anymore.

6.3 Simulation

In this section, we evaluate the performance of the model averaging de-biased LASSO estimator through some simulations. The idea is to find out whether it is more stable and whether we have a gain in accuracy compared to the simple de-biased LASSO. We compare the methods introduced with the LASSO, the two methods of de-biased LASSO, the Elastic Net estimator, the Ridge one and the SCAD.

As in Chapter 4 for the simulations, we compare the accuracy using the average bias and the MSE on null and non-null variables and for the prediction, we look at the out-of-sample

error. We do not evaluate the inference for this estimator since nothing can guarantee that the bound of the confidence interval are a weighted mean with the same weights, as these quantities are random.

The number of considered setting is 16, we vary the parameters p, n, s and ρ, the covariance structure for all these designs is the equicorrelated one.

$$p \in \{100, 500\}, \qquad n \in \{50, 100\},$$
$$s \in \{5, 40\}, \qquad \rho \in \{0.1, O.8\}.$$

We mainly compare the model averaging methods. In Table 7.5.15 in Appendix, which displays the bias, the differences between the methods are really small and no pattern appears. For the MSE error in Table 6.3.1, we observe a small difference, the averaging methods have lower MSE especially when the sparsity is low. For the design 13, with high-dimensional non sparse design $n = 50, p = 500$ and $s = 40$, this is the most remarkable with a twice as low MSE.

					ODLASSO		**NDLASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**		**OD MA**		**ND MA**	
		design	n	ρ	active	null	active	null	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	3.56	0.21	3.76	0.21	2.89	0.20	0.19	0.20	2.65	0.20	3.79	0.20	3.55	0.21	3.74	0.21
		2	100	0.1	3.75	0.20	3.88	0.20	3.20	0.20	0.20	0.19	3.07	0.20	3.82	0.20	3.73	0.20	3.84	0.20
		3	50	0.8	2.65	0.25	3.22	0.26	2.49	0.20	0.23	0.19	1.93	0.20	4.93	0.21	2.88	0.20	3.06	0.27
		4	100	0.8	3.01	0.22	3.49	0.23	2.84	0.20	0.26	0.19	2.40	0.20	4.75	0.20	3.14	0.19	3.40	0.23
	$s = 40$	5	50	0.1	3.32	1.74	4.03	1.84	2.77	1.68	1.06	1.18	2.47	1.62	5.99	2.55	3.14	1.69	3.72	1.70
		6	100	0.1	2.36	1.53	2.56	1.40	2.16	1.55	0.99	1.13	2.14	1.54	4.68	1.77	2.35	1.55	2.54	1.41
		7	50	0.8	7.62	8.40	4.32	4.01	3.76	2.08	0.97	0.98	2.11	1.56	12.15	5.11	7.11	7.86	4.35	3.90
		8	100	0.8	4.05	2.94	2.62	1.69	2.31	1.49	0.97	0.97	1.79	1.37	2.74	1.63	4.44	1.19	2.44	1.69
$p = 500$	$s = 5$	9	50	0.1	3.43	0.05	3.74	0.06	2.64	0.04	0.04	0.04	1.98	0.04	3.96	0.04	3.39	0.05	3.73	0.06
		10	100	0.1	3.79	0.05	3.99	0.05	3.13	0.04	0.04	0.04	2.87	0.04	3.98	0.04	3.77	0.04	3.96	0.05
		11	50	0.8	1.97	0.09	2.57	0.08	1.81	0.04	0.11	0.04	1.09	0.04	4.43	0.05	2.31	0.05	2.49	0.09
		12	100	0.8	2.88	0.06	3.54	0.06	2.66	0.04	0.08	0.04	2.14	0.04	4.73	0.04	2.99	0.04	3.42	0.07
	$s = 40$	13	50	0.1	2.63	1.11	4.05	1.79	1.05	0.45	0.30	0.30	0.87	0.42	1.37	0.60	1.57	0.56	2.10	0.77
		14	100	0.1	2.10	0.42	2.43	0.46	1.59	0.37	0.31	0.30	1.42	0.36	2.41	0.46	2.07	0.41	2.44	0.46
		15	50	0.8	4.16	3.89	2.11	1.60	1.49	0.66	0.29	0.30	0.70	0.44	3.40	1.64	5.16	5.29	2.11	1.85
		16	100	0.8	2.63	2.19	1.80	0.70	1.63	0.44	0.32	0.30	0.94	0.38	1.59	0.47	2.38	2.38	1.50	0.67

Table 6.3.1: Mean Squared Error for the estimators including the desparsified model averaging, the number of repetitions is $R = 100$.

		design	n	ρ	OD LASSO	ND LASSO	LASSO	RIDGE	EN	SCAD	OD MA	ND MA
$p = 100$	$s = 5$	1	50	0.1	4	11	1	10	1	1	3	10
		2	100	0.1	1	2	1	20	1	0	1	2
		3	50	0.8	167	36	1	5	1	2	39	66
		4	100	0.8	74	8	1	4	0	1	15	15
	$s = 40$	5	50	0.1	169	440	166	226	149	356	129	285
		6	100	0.1	4	15	4	108	5	148	3	13
		7	50	0.8	21582	4268	57.62	80	38	244	19632	4154
		8	100	0.8	7816	586	16	63	12	69	3631	588
$p = 500$	$s = 5$	9	50	0.1	85	214	1	8	2	1	64	229
		10	100	0.1	20	70	1	22	1	1	13	53
		11	50	0.8	5574	1346	1	5	2	3	1731	2135
		12	100	0.8	2417	199	1	5	1	2	493	432
	$s = 40$	13	50	0.1	10548	26724	293	176	272	461	1234	5686
		14	100	0.1	417	1386	101	106	96	233	290	1320
		15	50	0.8	358929	85148	102	44	75	318	523693	122323
		16	100	0.8	214294	18967	57	46	46	106	239477	21025

Table 6.3.2: Out-of-sample error computed with a test set size of size equals to 15% of the training dataset, the number of repetition for each design is $R = 100$.

Finally, the measure for the prediction power is given by the test error in Table 6.3.2, where the size of the test data corresponds to 15% of the training data. First of all, the test error strongly depends on the number of entries in $\mathbf{X}$ so we have high differences between the designs for all the estimators. As we have already noticed, the out of sample error is high for the de-biased methods compared to the other ones. However, for the optimized de-biased LASSO, the model averaging version of the estimator greatly reduces the errors for all the designs and it is emphasized when the sparsity is low.

Before concluding, we look at the case introduced at the end of Chapter 3 where the noise is non Gaussian. The following simulations are based on the same data as in Chapter 4, so we compare the result of the model averaging estimator with the result in Tables 4.8.1, 7.5.12 and 4.8.2 in Appendix.

Table 7.5.13 and 7.5.14 in Appendix present the results for the accuracy when the noise is Exponential with parameter m and the same remarks as for the case with Gaussian noise can be made : the accuracy of the model averaging method is very similar to that of the de-biased estimator unless for the MSE that is reduced when the sparsity is low. As before, the prediction for the model averaging estimators are greatly improved compared to the de-biased method. Compared to the Gaussian noise case, we have the same pattern, the distribution of the noise leads to a reduction of the test error.

To summarize, for the accuracy, the averaging method seems to perform as well as the de-biased methods despite the fact that they improve slightly the prediction power of the method, especially for the OD LASSO method. The prediction is not the purpose of the de-biased estimator and they were designed for it but it is interesting to see that the average method can greatly enhance this aspect and this conclusion holds also in cases where the noise is non Gaussian. This is in line with the position of Hansen (2007) that use the model averaging method to reduce the risk associated to the estimator. Pursuit this analysis with other type of weights, as the one based on the Mallows' Cp should be interesting, unfortunately, it is out of the scope of this work so we leave this question for the future.

Chapter 7

Generalized Linear Models.

The Generalized Linear Models (GLM) are really useful to describe more complex links between the response variable and the explanatory variables. The difference with the models considered up to now is the presence of a link function G between the response variable and the covariates, as

$$\mathbb{E}[\mathbf{Y} \mid \mathbf{X}] = G(\mathbf{X}\,\beta). \tag{7.0.1}$$

For instance, the logistic model corresponds to a link function $G(\mathbf{X}\,\beta) = \frac{1}{1+\exp(-\mathbf{X}\,\beta)}$ and it is used when the response variable is dichotomic. Indeed the G domain is the interval $[0, 1]$ and we then interpret the predicted value as the probability that the variable Y is equal to one. The major change in these settings is the form of the loss function.

The standard LASSO regression works for these models if we adapt slightly the estimator (Geer, 2008), the traditional $||\,\mathbf{Y} - \mathbf{X}\,\beta||_2^2$ will be now depend on the loss function.

In their article, van de Geer et al. (2014) extends their method of desparsified LASSO to GLM. They study the case where the loss function is strictly convex in β. They show that the convergence result of the de-biased LASSO estimator still holds under several assumptions on the matrix $\mathbf{X}$ in addition to the classical ones for the linear regression and the convergence of the LASSO estimator.

7.1 Method

In this section, we briefly remind the setting of GLM and construct the estimator for this case.

In GLM, the model 7.0.1 can be written as follows :

$$\begin{aligned} \mathbf{Y} &= \mu \\ g(\mu) &= \mathbf{X}\,\beta. \end{aligned}$$

Usually, the elements Y_i are assumed to come from some distribution in the exponential family, which means that the density of an observation Y can be written as

$$f(y, x\beta) = \exp\left(\frac{y(x\beta) - b(x\beta)}{a}\right) h(y),$$

where b and h are functions and a a parameter. This family of distribution includes the most common laws, such as the Gaussian, the Exponential, the Poisson and many others.

The loss function of the regression directly depends on the link function. When the log likelihood is used for a distribution among the Exponential family we obtain

$$l_\beta(y, x) \coloneqq l(y, x\beta) = \frac{y(x\beta) - b(x\beta)}{a} + cst$$

For more simplicity in the notation, we denote the loss function for n observations by $P_n l_\beta \coloneqq \frac{1}{n}\sum_{i=1}^n l_n(y_i, x_i)$. The LASSO estimator for the GLM is pretty much the same as in the Gaussian case, only the loss function is adapted. A such,

$$\hat{\beta}_{LASSO} \coloneqq \arg\min_\beta \frac{1}{n}\sum_{i=1}^n l_\beta(y_i, \mathbf{X}_i) + \lambda||\beta||_1 = \arg\min_\beta P_n l_\beta + \lambda||\beta||_1. \tag{7.1.1}$$

The construction for the de-biased estimator is also similar to 2.2.5. First, we need to define the variance covariance matrix in the GLM context and the derivative of the loss function.

$$\dot{l} \coloneqq \frac{\partial}{\partial\, \beta} l_\beta \qquad\qquad \ddot{l} \coloneqq \frac{\partial^2}{\partial\, \beta\, \partial \beta^\top} l_\beta$$
$$\Sigma \coloneqq \frac{1}{n}\sum_{i=1}^n \ddot{l}_\beta(y_i, \mathbf{X}_i) = P_n \ddot{l}_\beta.$$

The desparsified estimator is constructed with the same principle as for the simple case.

$$\hat{\beta}^d \coloneqq \hat{\beta}_{LASSO} - M P_n \dot{l}_{\hat{\beta}_{LASSO}} = \hat{\beta}_{LASSO} - M\left(\frac{1}{n}\sum_{i=1}^n \dot{l}_{\hat{\beta}_{LASSO}}(y_i, \mathbf{X}_i)\right). \tag{7.1.2}$$

If we replace the loss function with the one from the linear model with canonical link function, we find the estimator defined in 2.2.5.

Construction of the matrix M

The matrix M plays the role of the inverse covariance matrix, since this one depends on the loss function, the construction of the matrix M still uses the node-wise LASSO but it is slightly adapted.

$\Sigma_{A,B}$ denotes the Σ sub-matrix such that only the row $j \in A$ and the column $j \in B$ are kept. The index $-j$ indicates that we consider all the indices except the jth, $-j \coloneqq \{1, \dots, p\} \setminus \{j\}$. We then defined for each column j,

$$\begin{aligned}\hat{\gamma}_j &\coloneqq \arg\min_{\gamma \in \mathbb{R}^{p-1}} \left(\Sigma_{j,j}\gamma - 2\Sigma_{j,-j} + \gamma^\top \Sigma_{-j,-j}\gamma + 2\lambda_j ||\gamma||_1 \right), \\ \hat{\tau}_j^2 &\coloneqq \Sigma_{j,j} - \Sigma_{j,-j}\hat{\gamma}_j.\end{aligned}$$

Then, $M \coloneqq \hat{T}^{-2}\hat{C}$, with $\hat{T}$ and $\hat{C}$ defined as in Chapter 2.

In the Gaussian case, the loss function is given by the squared error $l_\beta(y, x) = (y - x^\top \beta)^2$ and the variance covariance matrix becomes $\Sigma = \frac{1}{n}\mathbf{X}^\top \mathbf{X}$. We then come back to the same estimators as in Chapter 2. Indeed,

$$\begin{aligned}\frac{1}{n}||X_j - \mathbf{X}_{-j}\gamma||_2^2 + 2\lambda_j||\gamma||_1 &= \frac{1}{n}(X_j - \mathbf{X}_{-j}\gamma)^\top (X_j - \mathbf{X}_{-j}\gamma) + 2\lambda_j||\gamma||_1 \\ &= \frac{1}{n}(X_j^\top - \gamma^\top \mathbf{X}_{-j}^\top)(X_j - \mathbf{X}_{-j}\gamma) + 2\lambda_j||\gamma||_1 \\ &= \frac{1}{n}(X_j^\top X_j - X_j^\top \mathbf{X}_{-j}\gamma - \gamma^\top \mathbf{X}_{-j}^\top X_j + \gamma^\top \mathbf{X}_{-j}^\top \mathbf{X}_{-j}\gamma) + 2\lambda_j||\gamma||_1 \\ &= \Sigma_{j,j} - 2\Sigma_{j,-j}\gamma + \gamma^\top \Sigma\gamma + 2\lambda_j||\gamma||_1.\end{aligned}$$

7.2 Example for the logit case

To make this theory more concrete, we illustrate it here with an example in logistic regression. This setting is used when the response variable is binary, we assume that each variable Y_i follows a Bernoulli distribution and we try to explain the probability of success with the explanatory variables stored in the matrix $\mathbf{X}$.

The model is

$$\begin{aligned}&Y_i \sim \text{Bernoulli}(\mu) \text{ for } i = 1, \dots, n, \\ &\mathbb{E}[\mathbf{Y} \mid \mathbf{X}] = \mu = G(\mathbf{X}\beta) = \frac{1}{1 + \exp(-\mathbf{X}\beta)}.\end{aligned}$$

The Bernoulli distribution belongs to the exponential family, its density is given by the following function where $\pi \coloneqq g(\mu) = G^{-1}(\mu)$.

$$f(y, \pi) = \exp(y\pi - \log(1 + e^\pi)).$$

We then find the following log-likelihood function where *cst* is a constant that does not depend on β.

$$l_\beta(y,x) = \Big(y(x\,\beta) - \log(1+e^{x\,\beta})\Big) + cst. \tag{7.2.1}$$

For logit regression, as for the most common distributions, the matrix Σ defined above can be written as $\frac{1}{n}\mathbf{X}_w^\top \mathbf{X}_w$, where $\mathbf{X}_w = W\,\mathbf{X}$ with W a diagonal matrix with entries $w_\beta(y_i, x_i)$. Indeed, the first and seconds derivatives of the log-likelihood are

$$\begin{aligned}
\dot{l}_\beta &= \frac{\partial}{\partial\beta}\Big(yx\beta - \log(1+e^{x\beta})\Big) = yx - \frac{xe^{x\beta}}{1+e^{x\beta}} \\
\ddot{l}_\beta &= \frac{\partial}{\partial\beta}\left(yx - \frac{xe^{x\beta}}{1+e^{x\beta}}\right) = \frac{-x^2e^{x\beta}}{(1+e^{x\beta})^2}.
\end{aligned}$$

Hence by using the sum over all the observations of the minus log-likelihood as loss function,

$$\begin{aligned}
\Sigma = \frac{1}{n}\sum_{i=1}^{n}\frac{X_i^2 e^{(\mathbf{X}\,\beta)_i}}{(1+e^{(\mathbf{X}\,\beta)_i})^2} &= \frac{1}{n}\mathbf{X}_w^\top\mathbf{X}_w \\
\text{with } w_\beta(y_i,\mathbf{X}_i) &= \frac{e^{\frac{1}{2}\mathbf{X}_i\,\beta}}{1+e^{\mathbf{X}_i\,\beta}}.
\end{aligned}$$

7.3 Theoretical results

In this section, we summarize the results of van de Geer et al. (2014) to demonstrate the validity of the estimator and of its distribution under GLM. The idea is to obtain a first result for any model with a stricly convex loss function that respects a certain number of assumptions, then the result is restrained to the standard GLM case.

The following hypotheses for some $K, \lambda, s, \lambda' \propto \lambda_j$ and $s' \geq s_j \forall j$ are needed to obtain the convergence of the estimator and especially its asymptotic distribution.

(C1) The first and second derivatives of the loss function exit, and for a neighbours δ, the second derivative is Lipschitz in the sense that :

$$\max_i \sup_{\max(|a-x_i^\top\,\beta\,|,|a'-x_i^\top\,\beta\,|)\leq\delta} \sup_y \frac{|\ddot{l}_a(y,a) - \ddot{l}_{a'}(y,a')|}{|a-a'|} \leq 1$$

Moreover, $\max_i \sup_y |\dot{l}_\beta(y,x_i)| = \mathcal{O}(1)$ and $\max_i \sup_{|a-x_i^\top\beta|\leq\delta}\sup_y |\ddot{l}_\beta(y,x_i)| = \mathcal{O}(1)$.

(C2) $||\hat{\beta}_{LASSO} - \beta||_1 = \mathcal{O}_{\mathbb{P}}(s\lambda)$, $||\,\mathbf{X}(\hat{\beta}_{LASSO} - \beta)||^2 = \mathcal{O}_{\mathbb{P}}(s\lambda^2)$.

(C3) $||\,\mathbf{X}\,||_\infty \coloneqq \max_{i,j}|\,\mathbf{X}_{i,j}\,| = \mathcal{O}(K)$.

(C4) $||\Sigma M_j^\top - e_j||_\infty = \mathcal{O}_{\mathbb{P}}(\lambda')$.

(C5) $||\,\mathbf{X}\,M_j^\top||_\infty = \mathcal{O}_{\mathbb{P}}(K)$ and $||M_j||_1 = \mathcal{O}_{\mathbb{P}}(\sqrt{s'})$.

(C6)

$$||P_n\dot{l}_\beta(\dot{l}_\beta)^\top\,\mathbb{E}[P_n(\dot{l}_\beta)(\dot{l}_\beta)^\top]||_\infty = \mathcal{O}_{\mathbb{P}}(K^2\lambda),$$

and

$$\max_j \frac{1}{\left(M\,\mathbb{E}[P_n\dot{l}_\beta(\dot{l}_\beta)^\top]M^\top\right)_{j,j}} = \mathcal{O}(1).$$

(C7) For every j, the following quantity converges weakly to a standardized Normal distribution.

$$\frac{\sqrt{n}(MP_n\dot{l}_\beta)_j}{\sqrt{\left(M\,\mathbb{E}[\sum_i^n \dot{l}_\beta(\dot{l}_\beta)^\top]M^\top\right)_{j,j}}}.$$

(C8) $Ks\lambda^2 = o(n^{-1/2}), \quad \lambda'\lambda s = o(n^{-1/2}), \quad K^2 s'\lambda + K^2\sqrt{s}\lambda = o(1)$.

(C1) is an assumption on the regularity of the loss function, (C2) requests that the LASSO estimator is close to its target and (C4) requests that M is not to far away from Σ^{-1}, (C3) bounds the matrix $\mathbf{X}$ while (C5) bounds M. Assumption (C8) is for the sparsity of $\mathbf{X}$ and of Σ^{-1}. It is the assumption (C7) that brings the normality result.

Theorem 7.3.1 (van de Geer et al. (2014), Theorem 3.1)**.** *Assume $C1 - C8$, then for the estimator 7.1.2 and for all variables j,*

$$\frac{\sqrt{n}(\hat{\beta}_j - \beta_j)}{\hat{\sigma}} = V_j + o_{\mathbb{P}}(1)$$

where V_j converges weakly to a standardized normal distribution. The variance is estimated by

$$\hat{\sigma}^2 = \left(MP_n\dot{l}_\beta(\dot{l}_\beta)^\top M^\top\right)_{j,j}.$$

For the complete proof, we refer to van de Geer et al. (2014). Here, we will give the main steps and highlight the necessity of all these assumptions.

Sketch of the proof. For the first part of the proof, we justify the following identity :

$$\hat{\beta}_j^d - \beta_j = -M_jP_n\dot{l}_\beta - k,$$

where $k \leq o_{\mathbb{P}}(n^{-1/2})$.

By condition (C1),

$$M_j P_n \dot{l}_{\hat{\beta}_{LASSO}} = M_j P_n \dot{l}_\beta + M_j P_n \ddot{l}_{\hat{\beta}_{LASSO}} (\hat{\beta}_{LASSO} - \beta) + k',$$

where the term k' is bounded

$$\begin{aligned} k' &= \mathcal{O}_{\mathbb{P}}(K) \frac{1}{n} \sum_{i=1}^{n} |x_i^\top (\hat{\beta}_{LASSO} - \beta)|^2 \text{ by (C1) and (C5)}, \\ &= \mathcal{O}_{\mathbb{P}}(K) \frac{1}{n} || \mathbf{X}(\hat{\beta}_{LASSO} - \beta)||_n^2 = \mathcal{O}_{\mathbb{P}}(K s \lambda^2) \text{ by (C2)}, \\ &= o_{\mathbb{P}}(n^{1/2}) \text{ by (C8)}. \end{aligned}$$

Then we have

$$\begin{aligned} \hat{\beta}_j^d - \beta_j &= \hat{\beta}_{j,LASSO} - \beta_j - M_j P_n \dot{l}_{\hat{\beta}_{LASSO}} \\ &= \hat{\beta}_{j,LASSO} - \beta_j - M_j P_n \dot{l}_\beta - M_j P_n \ddot{l}_{\hat{\beta}_{LASSO}} (\hat{\beta}_{LASSO} - \beta) - k' \\ &= -M_j P_n \dot{l}_\beta - (M_j P_n \ddot{l}_{\hat{\beta}_{LASSO}} - e_j)(\hat{\beta}_{LASSO} - \beta) - k' \\ &= -M_j P_n \dot{l}_\beta - k, \end{aligned}$$

where

$$|k| \leq |k'| + \mathcal{O}(\lambda') || \hat{be}_{LASSO} - \beta \, ||_1 = o_{\mathbb{P}}(n^{1/2}) + \mathcal{O}(\lambda' s \lambda) = o_{\mathbb{P}}(n^{1/2}).$$

The first inequality comes from (C4), then (C2) is used and finally (C8).

Up to now, we have used (C1), (C2), (C4), (C5) and (C8). If we show that the estimator $\hat{\sigma}^2$ is consistent, the theorem follows by condition (C7).

$$\begin{aligned} | \left(M P \dot{l}_\beta (\dot{l}_\beta)^\top M^\top \right)_{j,j} - \left(M P_n \dot{l}_{\hat{\beta}} (\dot{l}_{\hat{\beta}})^\top M^\top \right)_{j,j} | \leq \\ |(M(P_n - P)\dot{l}_\beta (\dot{l}_\beta)^\top M^\top)_{j,j}| + | \left(M P \dot{l}_\beta (\dot{l}_\beta)^\top M^\top \right)_{j,j} - \left(M P \dot{l}_{\hat{\beta}} (\dot{l}_{\hat{\beta}})^\top M^\top \right)_{j,j} |. \end{aligned}$$

The first term is $\mathcal{O}_{\mathbb{P}}(s' K^2 \lambda)$ by (C5) and (C6). For the second term, we need (C1), (C2), (C5) to find that it is an $\mathcal{O}_{\mathbb{P}}(K^2 \sqrt{s} \lambda)$. Condition (C8) allows us to bound the sum by $o_{\mathbb{P}}(1)$. □

The assumptions for the general case of the convex loss function are somewhat strong and not easy to verify, however in the case of the standard GLM they are relaxed. More precisely, the following explanations hold when the matrix Σ is parametrized by β, this is the case in standard GLM. Indeed $\Sigma = \frac{1}{n} \mathbf{X}_w^\top \mathbf{X}_w$, with $\mathbf{X}_w = W \mathbf{X}$ where $w_i = w_\beta(y_i, x_i)$ and when the loss function is convex for all y.

(D1) Condition (C1).

(D2) $||\frac{1}{\dot{l}_\beta}||_\infty = \mathcal{O}(1)$.

(D3) $||\,\mathbf{X}\,||_\infty = \mathcal{O}(1)$.

(D4) $||\,\mathbf{X}\,\beta||_\infty = \mathcal{O}(1)$ and for each j, $||\,\mathbf{X}_{w,-j}\,\gamma_j||_\infty = \mathcal{O}(1)$.

(D5) The smallest eigenvalue of Σ is bounded away from zero.

(D6) For each variable, $\frac{1}{\Sigma_j^{-1}\,\mathbb{E}[P_n \dot{l}_\beta (\dot{l}_\beta)^\top](\Sigma_j^{-1})^\top} = \mathcal{O}(1)$.

(D7) $s = o(\frac{\sqrt{n}}{\log p})$ and for each variables, $s_j = \sqrt{n/\log p}$.

The assumptions (D1) and (D2) concern the loss function l, the last assumption is about the sparsity of β and of the column of Σ^{-1}. When l is the minus log likelihood, (D6) is implied by (D1) - (D3), as $\Sigma = \mathbb{E}[P_n \dot{l}_\beta (\dot{l}_\beta)^\top]$ and the denominator is equal to $\Sigma_{j,j}^{-1} \leq \Sigma_{j,j}$ which is bounded by (D3).

It is clear than these conditions correspond to the one from the Gaussian model or is implied by the form of the loss function of this model.

Theorem 7.3.2 (van de Geer et al. (2014), Theorem 3.3)**.** *For 7.1.2 with $\lambda \propto \sqrt{(\log p)/n}$ and also with $\lambda_j \propto \sqrt{(\log p)/n}$ in the construction of M, then for each variable*

$$\frac{\sqrt{n}(\hat{\beta}_j - \beta_j)}{\hat{\sigma}} = V_j + o_{\mathbb{P}}(1)$$

and V_j converges weakly to a standardized normal distribution. The variance is estimated by

$$\hat{\sigma}^2 = \left(M \sum_{i=1}^{n} (\frac{\partial l}{\partial \beta})(\frac{\partial l}{\partial \beta})^\top M^\top \right)_{j,j}.$$

To prove this theorem, it is sufficient to show that the conditions (D1) - (D7) imply the (C1) - (C8). For the proof, the authors show that under (D1) - (D7), the matrix M is sufficiently close to its target and from that, we can obtain (C4) - (C8). The condition (D5) implies (C2) since under this setting the compatibility condition holds. As the proof is slightly long with several intermediate results, we omit it but it can be found in the supplementary content of van de Geer et al. (2014).

7.4 Simulations

We evaluate the performance of the method using simulations. We have selected 16 settings for a logistic regression and we compare the de-biased LASSO method with the LASSO, the Elastic Net, the Ridge, the SCAD and the selective inference method to obtain confidence intervals, these methods have already been presented in Chapter 4. First of all, the logistic regression satisfies the regularity condition (D1), thus Theorem 7.3.2 can be applied.

For these simulations, we vary the parameters p, n, ρ and s while the correlation structure of the matrix $\mathbf{X}$ is equicorrelated. The cases observed are all the combinations of

$$p \in \{100, 500\}, \qquad n \in \{50, 100\},$$
$$\rho \in \{0.1, 0.8\}, \qquad s \in \{5, 40\}.$$

The number of repetitions is set to $R = 100$ and we inspect the average bias, average Mean Squared Error, the out of sample error, average coverage and average length for the 95% confidence intervals.

					ND LASSO		**LASSO**		**RIDGE**		**EN**		**SCAD**	
		design	n	ρ	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	-0.95	0.18	-1.75	0.01	-1.97	0.01	-1.79	0.01	-1.46	0.01
		2	100	0.1	-0.93	0.10	-1.46	0.01	-1.96	0.01	-1.60	0.01	-0.74	0.01
		3	50	0.8	-0.43	0.71	-1.87	0.03	-1.98	0.02	-1.90	0.03	-1.79	0.04
		4	100	0.8	-0.44	0.43	-1.75	0.02	-1.98	0.02	-1.84	0.03	-1.61	0.03
	$s = 40$	5	50	0.1	-1.37	0.44	-1.96	0.02	-1.98	0.02	-1.94	0.03	-1.93	0.03
		6	100	0.1	-1.47	0.30	-1.91	0.03	-1.97	0.02	-1.90	0.04	-1.91	0.03
		7	50	0.8	-0.87	1.01	-1.94	0.05	-1.98	0.02	-1.93	0.06	-1.94	0.05
		8	100	0.8	-0.86	0.92	-1.92	0.06	-1.98	0.02	-1.91	0.07	-1.93	0.05
$p = 500$	$s = 5$	9	50	0.1	-1.06	0.22	-1.91	0.00	-1.99	0.00	-1.92	0.00	-1.80	0.00
		10	100	0.1	-1.01	0.10	-1.66	0.00	-1.98	0.00	-1.76	0.00	-1.18	0.00
		11	50	0.8	-0.60	0.62	-1.97	0.01	-1.99	0.01	-1.97	0.01	-1.94	0.01
		12	100	0.8	-0.71	0.38	-1.92	0.01	-1.99	0.01	-1.94	0.01	-1.86	0.01
	$s = 40$	13	50	0.1	-1.33	0.47	-1.99	0.00	-1.99	0.01	-1.98	0.01	-1.98	0.01
		14	100	0.1	-1.46	0.33	-1.97	0.01	-1.99	0.01	-1.97	0.01	-1.96	0.01
		15	50	0.8	-1.07	0.81	-1.99	0.01	-1.99	0.01	-1.98	0.01	-1.99	0.01
		16	100	0.8	-1.15	0.69	-1.98	0.01	-1.99	0.01	-1.98	0.01	-1.98	0.01

Table 7.4.1: Average bias for the logistics regressions on $R = 100$ repetitions.

In Table 7.4.1, we observe that the bias for the desparsified estimator is lower than the OLS for the active variable. The bias is the worst for designs $5, 6$ and $13, 14$ which present a high-dimensional matrix with low sparsity. Surprisingly, when the correlation is higher as in $7, 8$ and $15, 16$, the estimator is more accurate. The bias on null variables is, as expected greater for the desparsified estimator since the method is not sparse. The MSE, in Table 7.4.2, is greater for designs $5 - 8$ and $13 - 16$ which corresponds to the low sparsity case, however, even when the sparsity is high, the de-biased estimator presents a high MSE relatively to the other methods for the active variable.

For the prediction power, we look to the average on the $R = 100$ repetitions of the True Positive Ratio (TPR), True Negative Ratio (TNR) and False Postive Ratio (FPR). These quantities are defined as based on the number of True Positive (TP), True Negative (TN), False Postive (FP) and False Negative (FN) :

$$TP \coloneqq |\{j : \hat{\beta}_j = 0 \qquad \beta_j = 0\}|, TN \coloneqq |\{j : \hat{\beta}_j \neq 0 \qquad \beta_j \neq 0\}|$$

					ND LASSO		LASSO		RIDGE		EN		SCAD	
		design	n	ρ	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	1.31	0.35	0.30	0.20	0.20	0.20	0.25	0.20	0.89	0.21
		2	100	0.1	1.21	0.25	0.45	0.20	0.19	0.20	0.31	0.20	2.06	0.20
		3	50	0.8	3.92	1.75	0.30	0.21	0.20	0.20	0.22	0.20	0.46	0.23
		4	100	0.8	3.21	0.87	0.34	0.21	0.20	0.20	0.22	0.20	0.60	0.21
	$s = 40$	5	50	0.1	1.27	1.36	1.55	1.58	1.56	1.57	1.52	1.56	1.54	1.58
		6	100	0.1	1.13	1.29	1.51	1.56	1.56	1.57	1.47	1.55	1.50	1.56
		7	50	0.8	2.17	2.18	1.56	1.57	1.57	1.57	1.51	1.52	1.56	1.57
		8	100	0.8	2.02	1.89	1.53	1.56	1.57	1.57	1.48	1.52	1.53	1.55
$p = 500$	$s = 5$	9	50	0.1	1.05	0.22	0.07	0.04	0.04	0.04	0.06	0.04	0.21	0.04
		10	100	0.1	1.06	0.11	0.19	0.04	0.04	0.04	0.11	0.04	1.16	0.04
		11	50	0.8	2.94	1.41	0.06	0.04	0.04	0.04	0.04	0.04	0.11	0.05
		12	100	0.8	2.25	0.68	0.08	0.04	0.04	0.04	0.05	0.04	0.14	0.04
	$s = 40$	13	50	0.1	0.80	0.63	0.32	0.32	0.32	0.32	0.32	0.32	0.32	0.32
		14	100	0.1	0.55	0.42	0.32	0.32	0.32	0.32	0.32	0.32	0.33	0.32
		15	50	0.8	1.84	1.63	0.32	0.32	0.32	0.32	0.32	0.32	0.33	0.33
		16	100	0.8	1.32	1.11	0.33	0.32	0.32	0.32	0.32	0.32	0.33	0.32

Table 7.4.2: Mean Squared Error for the logistic regression, computed on $R = 100$ repetitions.

$$FP := |\{j : \hat{\beta}_j \neq 0 \qquad \beta_j = 0\}, FN := \{j : \hat{\beta}_j = 0 \qquad \beta_j \neq 0\}.$$

Then we defined

$$TPR := \frac{TP}{s} = \frac{TP}{TP + FN}, \qquad TNR := \frac{TN}{p - s} = \frac{TN}{TN + FP},$$
$$FPR := \frac{FP}{p - s} = \frac{FP}{TN + FP}.$$

In Table 7.4.3 we look at the average of these quantities on the $R = 100$ repetitions. The situation where all the individuals are correctly classed corresponds to a $TPR = 1, TNR = 1$ and $FPR = 0$. We see an interesting behaviour : for all the methods, the FPR is lower when the correlation ρ and the sparsity are high ($s = 5, \rho = 0.8$) and for almost all the designs, the FPR of the de-biased method is lower than the one for the LASSO. The method with the lowest FPR is the Ridge. For the TPR, the ND LASSO performs quite well, there are only two designs where the TPR is below 70%, it happens when $p = 500$ and $s = 5$. In general, the ND LASSO method presents very similar results with those of the LASSO and the other sparse methods, so we have no loss in the prediction power as it is the case for the Gaussian regression : when the number of variables was high, the prediction of the ND LASSO explodes relatively to the performances of the sparse methods.

					ND LASSO			**LASSO**			**RIDGE**			**EN**			**SCAD**		
		design	n	ρ	TPR	TNR	FPR	TPR	TNR	FPR	TPR	TNR	FPR	TPR	TNR	FPR	TPR	TNR	FPR
$p = 100$	$s = 5$	1	50	0.1	0.85	0.76	0.24	0.67	0.74	0.26	0.84	0.77	0.23	0.76	0.74	0.26	0.71	0.72	0.28
		2	100	0.1	0.78	0.73	0.27	0.87	0.77	0.23	0.76	0.71	0.29	0.87	0.78	0.22	0.88	0.79	0.21
		3	50	0.8	0.99	0.99	0.01	0.92	0.94	0.06	1.00	1.00	0.00	1.00	0.98	0.01	0.93	0.95	0.05
		4	100	0.8	0.87	0.98	0.02	0.85	0.96	0.04	0.86	1.00	0.00	0.88	0.98	0.01	0.86	0.96	0.04
	$s = 40$	5	50	0.1	0.90	0.95	0.05	0.76	0.82	0.18	0.92	0.95	0.05	0.84	0.89	0.11	0.80	0.82	0.18
		6	100	0.1	0.83	0.86	0.14	0.79	0.78	0.22	0.84	0.90	0.10	0.81	0.81	0.19	0.79	0.77	0.23
		7	50	0.8	0.99	1.00	0.00	0.97	0.97	0.03	1.00	1.00	0.00	1.00	1.00	0.00	0.96	0.96	0.04
		8	100	0.8	0.99	0.98	0.02	0.96	0.93	0.07	1.00	1.00	0.00	1.00	0.95	0.05	0.95	0.93	0.07
$p = 500$	$s = 5$	9	50	0.1	0.93	0.68	0.32	0.55	0.66	0.34	0.96	0.69	0.31	0.72	0.60	0.40	0.67	0.58	0.42
		10	100	0.1	0.64	0.57	0.43	0.76	0.74	0.26	0.62	0.49	0.51	0.75	0.73	0.27	0.75	0.75	0.25
		11	50	0.8	1.00	0.80	0.20	0.93	0.77	0.23	1.00	0.80	0.20	0.99	0.79	0.21	0.94	0.76	0.24
		12	100	0.8	0.68	1.00	0.00	0.68	0.97	0.03	0.67	1.00	0.00	0.68	0.99	0.01	0.68	0.98	0.02
	$s = 40$	13	50	0.1	0.77	1.00	0.00	0.74	0.79	0.21	0.77	1.00	0.00	0.80	0.88	0.12	0.76	0.84	0.16
		14	100	0.1	0.88	0.72	0.28	0.75	0.75	0.25	0.89	0.72	0.28	0.78	0.77	0.23	0.75	0.75	0.25
		15	50	0.8	0.99	0.99	0.01	0.98	0.96	0.04	1.00	1.00	0.00	1.00	1.00	0.00	0.96	0.96	0.04
		16	100	0.8	1.00	1.00	0.00	0.95	0.93	0.07	1.00	1.00	0.00	1.00	0.97	0.03	0.94	0.93	0.07

Table 7.4.3: Average of True Positive Ratio (TPR), True Negative Ratio (TNR) and False Postive Ratio (FPR) computed on a new sample of size $n15\%$ for the logistic regressions, the number of repetitions for each design is $R = 100$.

Finally, for the inference we compare the desparsified estimator with the confidence interval from the Post Selection Inference. Let us recall that the interpretation for this inference is different because the confidence interval is the one knowing that set of variables selected by the LASSO estimator. For example, a variable estimated to zero by the LASSO is then excluded from the coverage calculation.

Table 7.4.4 indicates the average coverage and 7.4.5 the average length of the 95% confidence intervals. The result of the de-biased estimator is disturbing, for the active variables the coverage sometimes falls under 50% and even to 0 for designs 1 and 2, if we look at the p-value we observe that the majority of the active variables are estimated significantly different to zero but the interval does not reach the true value. These poor coverages appear in the designs where the correlation between the variable is small ($\rho = 0.1$). If we do not consider these cases, the coverage is pretty good for the estimator, when the sparsity is low the true coefficients for the active variable are recovered by the confidence interval at 80% and at 97% for low sparsity. For the non active variable, there is no design when the coverage is under 90%. If we inspect the average length, we obtain that for designs $7, 11$ and 15 (few observations $n = 50$ and high correlation $\rho = 0.8$), the length explodes. However, for the other designs it is reasonable and the differences between the active and non active variables are negligible.

					ND LASSO		**EPSI**	
		design	n	ρ	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	0.00	0.91	0.70	1.00
		2	100	0.1	0.20	0.98	0.59	1.00
		3	50	0.8	0.80	1.00	0.99	1.00
		4	100	0.8	0.80	0.97	0.95	1.00
	$s = 40$	5	50	0.1	0.05	0.97	0.75	1.00
		6	100	0.1	0.42	1.00	0.68	1.00
		7	50	0.8	0.97	1.00	1.00	1.00
		8	100	0.8	0.97	1.00	0.99	1.00
$p = 500$	$s = 5$	9	50	0.1	0.00	0.95	0.67	1.00
		10	100	0.1	0.40	0.99	0.52	1.00
		11	50	0.8	0.80	1.00	1.00	1.00
		12	100	0.8	0.80	0.97	0.97	1.00
	$s = 40$	13	50	0.1	0.05	0.94	0.77	1
		14	100	0.1	0.00	0.97	0.77	1.00
		15	50	0.8	0.97	1.00	1.00	1.00
		16	100	0.8	0.97	1.00	1.00	1.00

Table 7.4.4: Coverage for logistic regression, for the de-biased estimator we have very law coverage under some designs but for the non active variable, the coverage is well around its target of 95%.

How can we justify the failure of the coverage for some designs ? A possible explanation is that one of the conditions (D1) - (D7) does not hold when the correlation is weak. In a recent work, Xia et al. (2020) show that in a majority of cases, the condition on the sparsity of the inverse variance covariance matrix is not met and the variance covariance matrix is the common point of our problematic designs. The sparsity in the inverse variance covariance

matrix was proven only for the Gaussian case.

					ND LASSO		EPSI	
		design	n	ρ	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	1.05	1.33	6.45	9.94
		2	100	0.1	1.29	1.44	2.98	7.83
		3	50	0.8	5.87	7.77	50.66	42.15
		4	100	0.8	3.14	3.44	19.33	23.44
	$s = 40$	5	50	0.1	1.91	1.99	12.28	11.46
		6	100	0.1	2.48	2.47	9.56	9.80
		7	50	0.8	11.86	12.52	73.04	68.33
		8	100	0.8	7.33	7.55	36.96	41.86
$p = 500$	$s = 5$	9	50	0.1	0.99	1.23	8.94	13.27
		10	100	0.1	1.38	1.60	3.58	7.88
		11	50	0.8	11.02	13.08	31.96	43.90
		12	100	0.8	2.53	3.16	26.19	24.05
	$s = 40$	13	50	0.1	1.70	1.77	9.75	11.51
		14	100	0.1	1.43	1.49	9.43	10.91
		15	50	0.8	13.29	13.73	80.30	70.77
		16	100	0.8	7.47	7.68	47.59	67.70

Table 7.4.5: Average length of the confidence intervals for the logistic regression, the mean is taken over $R = 100$ repetitions.

7.5 Real data examples

7.5.1 High-dimensional data

In this section, we apply the method for a logistic regression setting and we compare it with the LASSO estimator for binomial variables. The data we used are about prostate tumor, we have 52 observations from people with a tumor and 50 from healthy people. We next explore the link between having a prostate tumor and some gene expression. Since the covariates are composed of 6033 standardized logarithm gene expressions, we are clearly in a high-dimensional setting.

This dataset comes from Singh et al. (2002) and it can be found on the `R` package `spls`. A more complete analyse is given in Chung and Keles (2010) and in Dettling (2004).

The LASSO estimator selects 8 non null variables while the ND LASSO method estimates that 25 variables are significantly different from zero with a 95% degree of confidence and there are only 4 common selected variables. However, if we correct the p-value using Hochberg adjustment, the ND LASSO selects no more variables. Figure 7.5.1 shows the estimated coefficients for the two methods. The confidence intervals are only shown for significant ND LASSO coefficients. We observe that these coefficients are much higher than the ones estimated by the LASSO.

In Table 7.5.1, we observe the TPR, TNR and FPR of the two methods computed on a test

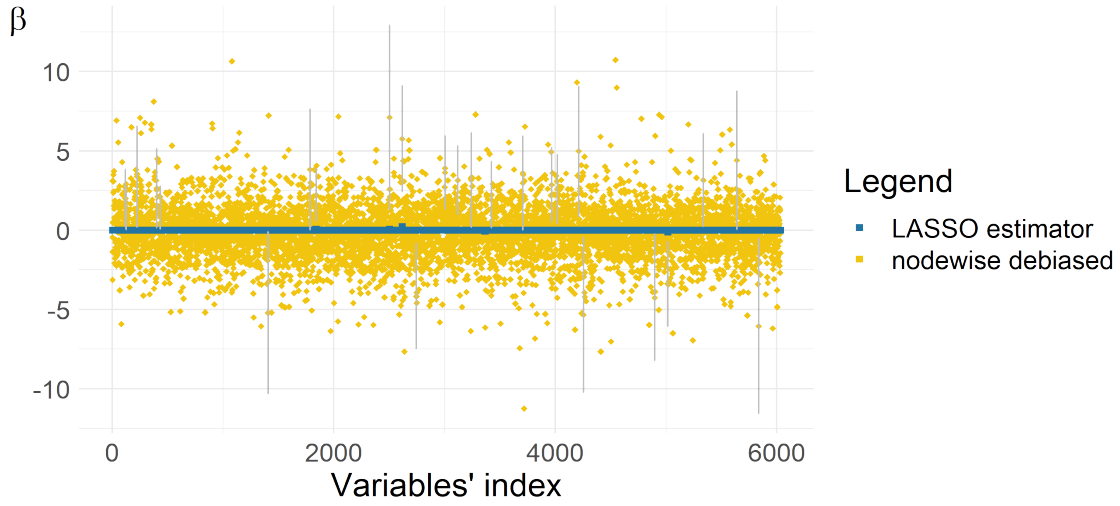

Figure 7.5.1: Estimated coefficients for the prostate dataset, the confidence intervals are only shown for the de-biased LASSO method.

	TPR	**TNR**	**FPR**
LASSO	1	0.14	0.86
ND LASSO	0.75	0.43	0.57

Table 7.5.1: True Positive Rate, True Negative Rate and False Positive Rate for the LASSO and the ND LASSO on a set of 15 new observations from the prostate data.

set excluded from the estimation. The LASSO method obtains a TPR of one, meaning that all the positive observations have been correctly predicted but, on the other hand, its TNR is really small. The estimator tends to predict too much positive cases and we also observe this phenomenon with its high FPR. The ND LASSO presents more moderate results, 75% of positive cases and 43% of negative cases are well predicted and the FPR is only at 57%.

7.5.2 Low dimensional data

The dataset `lungcap` from Kahn (2005) contains the smoking habits of 645 individuals, the response variable, *smoke*, indicates if one smokes and the explicative variables are *age*, fev the forced expiratory volume in litres, *ht* the height and *gender* a binary variable with 1 for male. In addition, we integrate to the model, the two order interactions and the squared variables, thus the total number of covariates is increased to 13. We then model 85% of the data using the LASSO adapted to Bernoulli distribution, the ND LASSO method and the Iteratively Reweighed Least Squares (IWLS) methods since we are in a low-dimensional framework. The remaining 15% observations will be used to obtain the out-of-sample prediction measures.

variables	**IWLS**	**LASSO**	**ND LASSO**
age	selected		
ht	selected		
age2		selected	
age*gender	selected	selected	

Table 7.5.2: Selected variables by the different methods for the logistic regression on lungcap dataset.

Table 7.5.2 presents the variables selected by the different methods, the ones selected by no model are not displayed. We first observe that the de-biased methods have estimated no variable significantly different from zero, then the LASSO model and the IWLS model only share the interaction between age and gender.

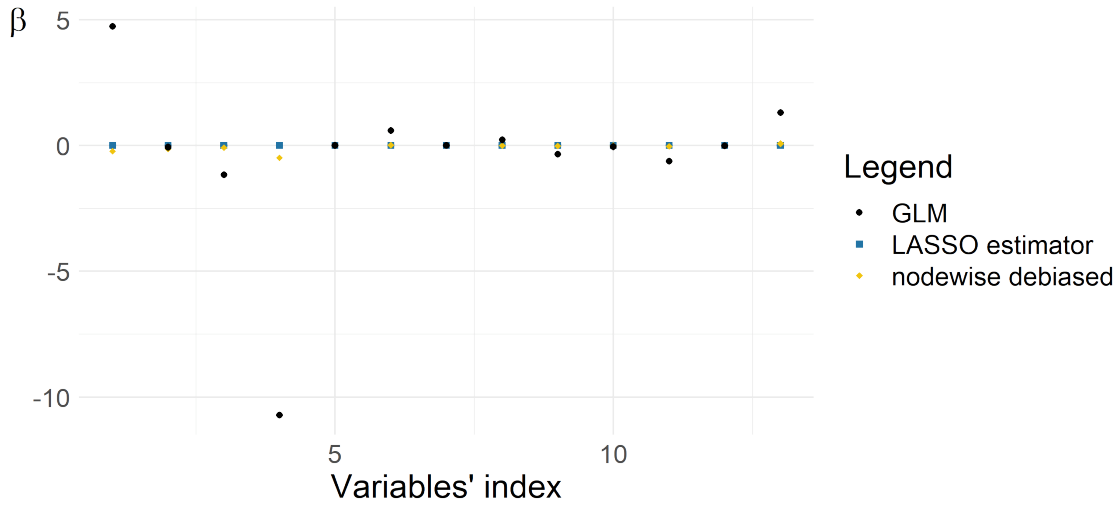

Figure 7.5.2: Estimated coefficients for the lungcap dataset, the confidence intervals are only shown for the de-biased LASSO method.

Figure 7.5.2 shows the estimated coefficients for all the variables, the confidence intervals are shown for the ND LASSO method only and in Figure 7.5.5 in Appendix, we observe their lengths for the two methods that provide inferences. The median for the de-biased LASSO is nearly the same as that for the other method but the box is shifted, indicating a higher mean.

models	**TPR**	**TNR**	**FPR**
LASSO	1	0	1
ND LASSO	0	1	0
IWLS	0.09	0.99	0.01

Table 7.5.3: True Positive Rate, True Negative Rate and False Postive Rate for the different models on a set of 98 (15% of the total sample size) observations excluded from the estimation.

In table 7.5.3, we observe the prediction results for the different methods, they are computed on a sample excluded from the estimation, which contains 98 observations. The LASSO method predicts only positive cases and obtains a TPR and a FPR equal to one and in contrary, the ND LASSO classifies all the values to zero. Only the GLM method has positive and negative predicted values. These results can be explained by the fact that the data contains few positive cases, only 65 on 654 observations and in this particular test set, there are only 2 positive cases. In this context, the result for the LASSO are really surprising but the ND LASSO as well as the GLM methods do not perform so badly.

Conclusion

In this work, we have presented the desparsified LASSO methods that answer to the long-awaited question of the inference for the LASSO estimator and they allow us to obtain tools for hypothesis testing in high-dimensional models. The two main methods, the Node-wise De-biased LASSO and the Optimized De-biased LASSO, are based on the same principle but the construction differs slightly. First, we have explored these differences through the theoretical results that support each of the estimators, both need a sparsity of order $o(\sqrt{n}/\log p)$ for converging to the true parameter, which is stronger than the condition for the LASSO convergence. In a random design, their distribution is valid if the matrix $\mathbf{X}$ has strictly positive eigenvalues, is Gaussian or sub-Gaussian and if the number of observations n is large enough.

We compared then the performances of the estimators on simulated and real data. It highlights that the methods correct well the LASSO's bias on active variables and that the cost in terms of Mean Squared Error is not too high. For the prediction, the poor performances of the estimators are disappointing despite the fact that they are not intended for this task. We recommend being careful when p is really large compared to n and even more if the signal-to-noise ratio is low. For the inference, the coverage for the active variable is sometimes below 70% when the sparsity is high and we suggest the OD LASSO method rather than the ND LASSO but the results are really satisfying for the null variables.

The OD LASSO is robust when the noise is not normally-distributed. However, the implemented method presents convergence issues under some high-dimensional designs. In particular, for the analyse on real data, the method did not converge for numerous datasets.For both methods, the analyse on real data shows notably that the choice of the regularization parameter impact strongly the inference.

In the second part of this work, we have explored this impact through a proposal estimator that joins the desparsified methods and the model averaging theory. We show, through simulations, that this proposal has similar performances for the accuracy of the estimation but it seems slightly to reduce the issues for the prediction. In conclusion, the proposal for model averaging desparsified estimators seems to be a good direction to explore, it would be interesting to develop their distribution and to verify if the normality is preserved.

Finally, we have presented the extended method of ND LASSO for the GLMs. The construction of the estimator is based on the same idea as for the Gaussian case but the theory needs a supplementary assumption on the sparsity of the variance-covariance matrix. We have inspected the numerical results of the method for the logistic regression and we found that the method has really good coverage for the non active variables, for the active variables the

measure is satisfying only in case of high correlation.

The high-dimensional models are more and more present nowadays and the desparsified LASSO estimators complete well the theory of high-dimensional linear models by providing interesting tools for the inference in these challenging setting. This work shows that the methods present correct result for the estimation and the inference.

Bibliography

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In B. Petrov and F. Csáki (Eds.), *Second International Symposium on Information Theory*, pp. 267–281. Akadémiai Kiadó, Budapest.

Benjamini, Y. and Y. Hochberg (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological) 57*(1), 289–300.

Benjamini, Y. and D. Yekutieli (2005). False Discovery Rate–Adjusted Multiple Confidence Intervals for Selected Parameters. *Journal of the American Statistical Association 100*(469), 71–81.

Berk, R., L. Brown, A. Buja, K. Zhang, and L. Zhao (2013). Valid post-selection inference. *The Annals of Statistics 41*(2), 802–837.

Bickel, P. J., Y. Ritov, and A. B. Tsybakov (2009). Simultaneous analysis of Lasso and Dantzig selector. *The Annals of Statistics 37*(4), 1705–1732.

Buckland, S., K. Burnham, and N. Augustin (1997). Model selection: An integral part of inference.

Bühlmann, P., M. Kalisch, and L. Meier (2014). High-Dimensional Statistics with a View Toward Applications in Biology. *Annual Review of Statistics and Its Application 1*(1), 255–278.

Bühlmann, P. and S. van de Geer (2011). *Statistics for High-Dimensional Data*. Springer Series in Statistics. Berlin, Heidelberg: Springer Berlin Heidelberg.

Bühlmann, P. and S. van de Geer (2015). High-dimensional inference in misspecified linear models. *Electronic Journal of Statistics 9*(1), 1449–1473.

Chung, D. and S. Keles (2010). Sparse partial least squares classification for high dimensional data. *Statistical Applications in Genetics and Molecular Biology 9*, Article17.

Dettling, M. (2004). BagBoosting for tumor classification with gene expression data. *Bioinformatics 20*(18), 3583–3593.

Dunn, O. J. (1961). Multiple Comparisons among Means. *Journal of the American Statistical Association 56*(293), 52–64.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least Angle Regression. *The Annals of Statistics 32*(2), 407–451.

Fan, J. and R. Li (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association 96*(456), 1348–1360.

Feng, Y. and Q. Liu (2020). Nested model averaging on solution path for high-dimensional linear regression. *Stat 9*(1), e317.

Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software 33*(1), 1–22.

Geer, S. A. v. d. (2008). High-dimensional generalized linear models and the lasso. *The Annals of Statistics 36*(2), 614–645.

Hansen, B. E. (2007). Leats Squares Model Averaging. *Econometrica 75*(4), 1175–1189.

Hansen, B. E. (2014). Model averaging, asymptotic risk, and regressor groups. *Quantitative Economics 5*(3), 495–530.

Hoerl, A. E. and R. W. Kennard (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics 12*(1), 55–67.

Javanmard, A. and A. Montanari (2013). Nearly optimal sample size in hypothesis testing for high-dimensional regression. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1427–1434.

Javanmard, A. and A. Montanari (2014). Confidence intervals and hypothesis testing for high-dimensional regression. *Journal of Machine Learning Research 15*(1), 2869–2909.

Javanmard, A. and A. Montanari (2018). Debiasing the lasso: Optimal sample size for Gaussian designs. *The Annals of Statistics 46*(6A), 2593–2622.

Kahn, M. (2005). An Exhalent Problem for Teaching Statistics. *Journal of Statistics Education 13*(2).

Lee, J. D., D. L. Sun, Y. Sun, and J. E. Taylor (2016). Exact post-selection inference, with application to the lasso. *The Annals of Statistics 44*(3), 907–927.

Miolane, L. and A. Montanari (2021). The distribution of the Lasso: Uniform control over sparse balls and adaptive parameter tuning. *The Annals of Statistics 49*(4), 2313 – 2335.

Palm, F. C. and A. Zellner (1992). To combine or not to combine? issues of combining forecasts. *Journal of Forecasting 11*(8), 687–701.

Raskutti, G., M. Wainwright, and B. Yu (2010). Restricted Eigenvalue Properties for Correlated Gaussian Designs. *Journal of Machine Learning Research 11*, 2241–2259.

Rudelson, M. and S. Zhou (2013). Reconstruction From Anisotropic Random Measurements. *IEEE Transactions on Information Theory 59*(6), 3434–3447.

Schomaker, M. and C. Heumann (2020). When and when not to use optimal model averaging. *Statistical Papers 61*(5), 2221–2240.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics 6*(2), 461–464.

Singh, D., P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub,

and W. R. Sellers (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell 1*(2), 203–209.

Sun, T. and C.-H. Zhang (2012). Scaled sparse linear regression. *Biometrika 99*(4), 879–898.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological) 58*(1), 267–288.

van de Geer, S., P. Bühlmann, Y. Ritov, and R. Dezeure (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics 42*(3), 1166–1202.

Vershynin, R. (2012). Introduction to the non-asymptotic analysis of random matrices. In G. Kutyniok and Y. C. Eldar (Eds.), *Compressed Sensing: Theory and Applications*, pp. 210–268. Cambridge: Cambridge University Press.

Xia, L., B. Nan, and Y. Li (2020). A revisit to de-biased lasso for generalized linear models. arXiv.

Zhang, C.-H. and S. S. Zhang (2014). Confidence Intervals for Low-Dimensional Parameters in High-Dimensional Linear Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 76*, 217–242.

Zou, H. (2006). The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association 101*(476), 1418–1429.

Zou, H. and T. Hastie (2005). Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology) 67*(2), 301–320.

Appendix

Numerical results: supplementary tables and figures.

			design	n	sd	**OD LASSO**	**ND LASSO**	**LASSO**	**RIDGE**	**EN**	**SCAD**
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	1	1	1	22	1	1
			18	1000	0.8	1	1	1	1	1	1
			19	100	1.5	2	2	1	31	1	2
			20	1000	1.5	1	1	1	1	1	1
		$\rho = 0.8$	21	100	0.8	12	13	13	154	14	151
			22	1000	0.8	1	1	1	1	1	1
			23	100	1.5	20	17	8	243	6	134
			24	1000	1.5	1	2	2	2	2	1
	$s = 50$	$\rho = 0.1$	25	100	0.8	4	5	2	23	3	2
			26	1000	0.8	2	2	2	3	2	2
			27	100	1.5	6	6	2	32	2	3
			28	1000	1.5	2	2	2	3	2	2
		$\rho = 0.8$	29	100	0.8	23	24	25	165	24	160
			30	1000	0.8	2	2	2	3	2	2
			31	100	1.5	21	19	11	257	9	134
			32	1000	1.5	2	3	2	3	2	2
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	6	10	1	26	1	0
			51	100	1.5	12	17	1	47	1	1
			52	1000	1.5	NA	1	1	33	1	1
		$\rho = 0.1$	53	100	0.8	1258	2015	199	202	197	197
			54	1000	0.8	1	1	1	147	1	1
			55	100	1.5	76	108	12	568	8	191
	$s = 50$	$\rho = 0.8$	57	100	0.8	22	33	3	26	4	2
			59	100	1.5	34	52	2	48	2	3
			60	1000	1.5	4	5	2	34	2	3
		$\rho = 0.1$	61	100	0.8	1290	2067	209	211	206	205
			63	100	1.5	84	116	19	614	14	201
			64	1000	1.5	NA	5	2	280	2	60

Table 7.5.4: Prediction error computed with a test set of size 15%n for the desparsified estimators and the competitors, the number of repetitions is $R = 100$ and the correlation structure of the design matrix is the Toeplitz one.

						OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
			design	n	sd	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	-0.00	0.00	0.02	0.00	-0.17	0.00	-1.82	0.00	-0.20	0.00	0.00	0.00
			18	1000	0.8	0.00	0.00	0.01	0.00	-0.08	0.00	-0.11	0.00	-0.08	0.00	-0.00	0.00
			19	100	1.5	0.07	0.01	0.08	0.00	-0.08	0.00	-1.55	0.01	-0.09	0.00	-0.02	0.00
			20	1000	1.5	0.00	0.00	0.01	0.00	-0.04	0.00	-0.12	0.00	-0.04	0.00	-0.00	-0.00
		$\rho = 0.8$	21	100	0.8	-0.15	0.00	-0.15	0.00	-0.23	0.00	-1.70	0.00	-0.26	0.00	-0.82	0.00
			22	1000	0.8	0.00	0.00	0.00	-0.00	-0.03	0.00	-0.05	-0.00	-0.03	0.00	0.00	-0.00
			23	100	1.5	0.07	0.01	0.05	0.01	-0.08	0.00	-0.92	0.04	-0.08	0.00	-0.18	0.00
			24	1000	1.5	0.00	0.00	0.04	0.00	-0.05	0.00	-0.06	0.01	-0.05	0.00	0.00	-0.00
	$s = 50$	$\rho = 0.1$	25	100	0.8	-0.00	0.00	0.04	0.00	-0.32	0.00	-1.83	0.00	-0.37	0.00	0.01	0.00
			26	1000	0.8	0.00	0.00	0.01	0.00	-0.14	0.00	-0.16	0.00	-0.15	0.00	-0.00	0.00
			27	100	1.5	0.12	0.01	0.14	0.01	-0.15	0.00	-1.56	0.01	-0.17	0.00	-0.03	0.00
			28	1000	1.5	0.00	0.00	0.02	0.00	-0.08	0.00	-0.16	0.00	-0.08	0.00	-0.00	-0.00
		$\rho = 0.8$	29	100	0.8	-0.19	0.00	-0.20	0.01	-0.30	0.01	-1.70	0.00	-0.31	0.01	-0.88	0.00
			30	1000	0.8	0.00	0.00	0.00	-0.00	-0.05	0.00	-0.06	-0.00	-0.05	0.00	0.00	-0.00
			31	100	1.5	0.06	0.01	0.04	0.01	-0.08	0.00	-0.92	0.04	-0.07	0.00	-0.18	0.00
			32	1000	1.5	0.00	0.00	0.02	0.00	-0.02	0.00	-0.03	0.01	-0.02	0.00	0.00	-0.00
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	-0.07	0.03	-0.01	0.05	-0.26	0.00	-1.97	0.01	-0.36	0.00	-0.00	0.00
			51	100	1.5	0.07	0.00	0.09	0.00	-0.09	0.00	-1.81	0.00	-0.11	0.00	-0.03	0.00
			52	1000	1.5	NA	NA	0.00	0.00	-0.05	0.00	-1.51	0.00	-0.06	0.00	-0.01	0.00
		$\rho = 0.8$	53	100	0.8	-0.28	0.00	0.15	0.00	-1.91	-0.00	-1.98	0.00	-1.89	0.00	-1.82	0.00
			54	1000	0.8	-0.00	-0.00	0.00	-0.00	-0.05	0.00	-1.45	0.00	-0.05	0.00	0.00	0.00
			55	100	1.5	0.07	0.00	0.08	0.00	-0.12	0.00	-1.37	0.00	-0.10	0.00	-0.24	-0.00
		$\rho = 0.1$	57	100	0.8	-0.09	0.00	0.01	0.00	-0.47	0.00	-1.99	0.00	-0.60	0.00	0.00	0.00
	$s = 50$		59	100	1.5	0.12	0.00	0.17	0.00	-0.19	0.00	-1.82	0.00	-0.22	0.00	-0.05	0.00
			60	1000	1.5	0.03	0.00	0.03	0.00	-0.08	0.00	-1.51	0.00	-0.09	0.00	-0.01	0.00
		$\rho = 0.8$	61	100	0.8	-0.26	0.00	0.17	0.00	-1.92	-0.00	-1.98	0.00	-1.89	0.00	-1.82	0.00
			63	100	1.5	0.07	0.00	0.08	0.00	-0.13	0.00	-1.39	0.00	-0.12	0.00	-0.25	0.00
			64	1000	1.5	NA	NA	0.04	0.00	-0.02	0.00	-0.79	0.00	-0.02	0.00	-0.08	0.00

Table 7.5.5: Average bias on active and non active variables on $R = 100$ repetitions for the desparsified LASSO methods and all the competitors, the correlation structure of the design matrix is the Toeplitz one.

						OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
			design	n	sd	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	3.80	0.21	3.90	0.21	3.19	0.20	0.20	0.20	3.09	0.20	3.82	0.20
			18	1000	0.8	3.80	0.20	3.83	0.20	3.52	0.20	3.38	0.20	3.50	0.20	3.80	0.20
			19	100	1.5	4.11	0.21	4.14	0.22	3.53	0.20	0.31	0.20	3.48	0.20	5.35	0.20
			20	1000	1.5	3.80	0.20	3.85	0.20	3.65	0.20	3.38	0.20	3.64	0.20	3.80	0.20
		$\rho = 0.8$	21	100	0.8	1.88	2.04	1.88	2.04	1.74	2.02	1.52	2.02	1.70	2.02	3.08	2.31
			22	1000	0.8	2.00	2.00	2.00	2.00	1.95	2.00	1.91	2.00	1.95	2.00	2.00	2.00
			23	100	1.5	2.57	2.01	2.51	2.07	2.21	2.00	1.05	1.97	2.06	2.00	11.60	2.00
			24	1000	1.5	2.00	2.00	2.08	2.00	1.91	2.00	1.89	1.99	1.91	2.00	2.00	2.00
	$s = 50$	$\rho = 0.1$	25	100	0.8	3.82	0.22	4.00	0.23	2.72	0.20	0.20	0.20	2.57	0.20	3.86	0.20
			26	1000	0.8	3.80	0.20	3.85	0.20	3.28	0.20	3.21	0.20	3.25	0.20	3.80	0.20
			27	100	1.5	4.39	0.22	4.46	0.27	3.37	0.20	0.31	0.20	3.25	0.20	5.58	0.20
			28	1000	1.5	3.81	0.21	3.89	0.21	3.53	0.20	3.25	0.20	3.51	0.20	3.81	0.20
		$\rho = 0.8$	29	100	0.8	1.91	2.08	1.91	2.08	1.76	2.05	1.52	2.02	1.71	2.04	3.16	2.30
			30	1000	0.8	2.00	2.00	2.01	2.01	1.90	2.00	1.89	2.00	1.91	2.00	2.00	2.00
			31	100	1.5	2.63	2.01	2.59	2.08	2.31	2.00	1.06	1.96	2.15	1.99	11.40	2.01
			32	1000	1.5	2.01	2.01	2.04	2.01	1.97	2.00	1.95	2.00	1.97	2.00	2.01	2.00
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	3.80	0.03	4.00	0.03	3.03	0.02	0.04	0.02	2.75	0.02	3.97	0.02
			51	100	1.5	4.27	0.03	4.39	0.04	3.63	0.02	0.07	0.02	3.56	0.02	5.41	0.02
			52	1000	1.5	NA	NA	3.98	0.02	3.78	0.02	0.27	0.02	3.72	0.02	4.95	0.02
		$\rho = 0.8$	53	100	0.8	4.18	1.42	6.48	2.14	0.32	0.20	0.20	0.20	0.32	0.20	0.51	0.21
			54	1000	0.8	3.80	0.20	3.81	0.20	3.63	0.20	0.40	0.21	3.62	0.20	3.80	0.20
			55	100	1.5	4.73	0.23	4.76	0.30	3.96	0.20	0.49	0.22	3.72	0.20	14.64	0.21
	$s = 50$	$\rho = 0.1$	57	100	0.8	3.66	0.04	4.05	0.05	2.35	0.02	0.04	0.02	1.99	0.02	3.98	0.02
			59	100	1.5	4.55	0.04	4.77	0.08	3.38	0.02	0.07	0.02	3.22	0.02	6.18	0.02
			60	1000	1.5	4.08	0.02	4.10	0.03	3.65	0.02	0.27	0.02	3.63	0.02	5.24	0.02
		$\rho = 0.8$	61	100	0.8	4.26	1.46	6.62	2.20	0.31	0.20	0.20	0.20	0.32	0.20	0.51	0.21
			63	100	1.5	4.95	0.24	5.00	0.32	4.15	0.20	0.48	0.22	3.82	0.20	14.72	0.21
			64	1000	1.5	NA	NA	3.96	0.21	3.72	0.20	1.44	0.20	3.72	0.20	10.06	0.20

Table 7.5.6: Average Mean Squared Error on active and non active variables computed on $R = 100$ repetitions, the correlation structure is the Toeplitz one.

						OD LASSO		**ND LASSO**		**EPSI**	
			design	n	sd	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	1	100	0.8	0.31	0.39	0.29	0.36	0.54	1.72
			2	1000	0.8	0.09	0.11	0.08	0.11	0.11	0.54
			3	100	1.5	NA	NA	0.57	0.71	1.36	3.33
			4	1000	1.5	0.18	0.23	0.18	0.22	0.30	0.85
		$\rho = 0.8$	5	100	0.8	8.59	8.74	0.82	0.84	4.11	6.92
			6	1000	0.8	5.44	5.55	0.14	0.14	0.16	0.59
			7	100	1.5	33.51	34.16	4.15	4.22	NA	NA
			8	1000	1.5	5.44	5.55	0.75	0.77	0.84	2.31
	$s = 50$	$\rho = 0.1$	9	100	0.8	0.57	0.71	0.54	0.67	0.95	2.79
			10	1000	0.8	0.16	0.20	0.16	0.20	0.20	0.86
			11	100	1.5	1.50	1.87	1.05	1.31	2.66	6.29
			12	1000	1.5	0.34	0.43	0.33	0.41	0.58	1.67
		$\rho = 0.8$	13	100	0.8	8.55	8.69	0.95	0.97	4.86	7.57
			14	1000	0.8	5.44	5.55	0.19	0.20	0.23	0.75
			15	100	1.5	32.77	33.41	4.14	4.22	NA	NA
			16	1000	1.5	5.45	5.56	0.76	0.77	0.87	2.50
$p = 1000$	$s = 5$	$\rho = 0.1$	33	100	0.8	0.43	0.54	0.31	0.38	0.66	2.06
			34	1000	0.8	0.10	0.12	0.09	0.11	0.12	0.59
			35	100	1.5	2.30	2.87	0.59	0.74	2.24	5.15
		$\rho = 0.8$	37	100	0.8	4.25	4.33	3.45	3.52	NA	NA
			38	1000	0.8	0.25	0.25	0.14	0.15	NA	NA
			39	100	1.5	NA	NA	17.11	17.45	NA	NA
	$s = 50$	$\rho = 0.1$	41	100	0.8	0.80	0.99	0.58	0.72	1.25	3.67
			43	100	1.5	4.23	5.27	1.09	1.36	3.53	7.61
			44	1000	1.5	NA	NA	0.33	0.42	0.64	2.08
		$\rho = 0.8$	45	100	0.8	4.23	4.32	3.47	3.54	NA	NA
			47	100	1.5	NA	NA	8.02	8.13	NA	NA
			48	1000	1.5	NA	NA	0.79	0.80	NA	NA

Table 7.5.7: Average length of the confidence intervals for the desparsified LASSO methods and the EPSI, the correlation structure is the equicorrelated and the number of repetition is $R = 100$.

						OD LASSO		ND LASSO		EPSI	
			design	n	sd	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	0.74	0.97	0.71	0.93	0.98	0.98
			18	1000	0.8	0.75	0.94	0.76	0.94	0.95	1.00
			19	100	1.5	0.59	0.99	0.61	0.92	0.97	1.00
			20	1000	1.5	0.75	0.94	0.72	0.94	0.94	0.60
		$\rho = 0.8$	21	100	0.8	0.98	1.00	0.63	0.87	1.00	1.00
			22	1000	0.8	0.98	1.00	0.91	0.92	0.97	0.99
			23	100	1.5	0.98	1.00	0.65	0.93	1.00	1.00
			24	1000	1.5	0.98	1.00	0.89	0.95	1.00	
	$s = 50$	$\rho = 0.1$	25	100	0.8	0.75	0.97	0.73	0.95	0.97	0.97
			26	1000	0.8	0.75	0.94	0.76	0.94	0.95	1.00
			27	100	1.5	0.61	0.99	0.64	0.94	0.96	1.00
			28	1000	1.5	0.75	0.94	0.72	0.94	0.94	0.67
		$\rho = 0.8$	29	100	0.8	0.98	1.00	0.63	0.86	1.00	1.00
			30	1000	0.8	0.98	1.00	0.92	0.93	0.97	0.99
			31	100	1.5	0.98	1.00	0.65	0.94	0.99	1.00
			32	1000	1.5	0.98	1.00	0.89	0.93	0.96	1.00
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	0.77	1.00	0.74	0.96	0.99	1.00
			51	100	1.5	0.64	1.00	0.61	0.96	0.99	1.00
			52	1000	1.5	NA	NA	0.80	0.90	0.95	1.00
		$\rho = 0.1$	53	100	0.8	0.97	0.99	0.93	0.96	0.97	0.99
			54	1000	0.8	NA	NA	0.89	0.94	0.99	0.99
			55	100	1.5	0.74	1.00	0.59	0.96	1.00	1.00
		$\rho = 0.8$	57	100	0.8	0.76	1.00	0.73	0.96	0.99	0.99
			59	100	1.5	0.64	1.00	0.62	0.96	0.98	0.96
			60	1000	1.5	0.70	0.98	0.71	0.95	0.95	1.00
	$s = 50$	$\rho = 0.1$	61	100	0.8	0.97	0.99	0.93	0.96	0.95	1.00
			63	100	1.5	0.68	1.00	0.86	1.00	1.00	1.00
			64	1000	1.5	NA	NA	0.98	1.00	0.98	1.00

Table 7.5.8: Average coverage on active and non active, computed on $R = 100$ repetitions. The correlation structure of the design matrix is the Toeplitz one.

						OD LASSO		**ND LASSO**		**EPSI**	
			design	n	sd	active	null	active	null	active	null
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	0.31	0.39	0.28	0.36	0.45	1.75
			18	1000	0.8	0.08	0.10	0.08	0.10	0.10	0.56
			19	100	1.5	0.42	0.54	0.43	0.55	0.75	1.84
			20	1000	1.5	0.16	0.22	0.15	0.19	0.20	0.32
		$\rho = 0.8$	21	100	0.8	8.27	8.42	0.66	0.67	4.13	6.39
			22	1000	0.8	5.44	5.55	0.10	0.10	0.12	0.48
			23	100	1.5	8.27	8.42	1.18	1.20	2.86	9.12
			24	1000	1.5	5.44	5.57	0.30	0.30	0.35	
	$s = 50$	$\rho = 0.1$	25	100	0.8	0.57	0.71	0.53	0.67	0.86	3.05
			26	1000	0.8	0.15	0.19	0.15	0.19	0.19	0.91
			27	100	1.5	0.74	0.95	0.80	1.03	1.43	4.04
			28	1000	1.5	0.31	0.41	0.28	0.36	0.38	0.70
		$\rho = 0.8$	29	100	0.8	8.05	8.20	0.90	0.92	6.28	9.22
			30	1000	0.8	5.44	5.55	0.19	0.19	0.22	0.87
			31	100	1.5	8.21	8.36	1.32	1.35	3.67	8.37
			32	1000	1.5	5.45	5.58	0.36	0.37	0.42	2.04
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	0.38	0.48	0.31	0.38	0.56	2.25
			51	100	1.5	0.45	0.57	0.44	0.56	0.80	1.83
			52	1000	1.5	NA	NA	0.12	0.16	0.22	1.17
		$\rho = 0.1$	53	100	0.8	6.80	6.93	5.54	5.65	25.76	35.67
			54	1000	0.8	NA	NA	0.10	0.11	0.17	0.58
			55	100	1.5	1.85	1.89	1.31	1.34	4.68	8.18
	$s = 50$	$\rho = 0.8$	57	100	0.8	0.71	0.88	0.58	0.72	1.05	4.03
			59	100	1.5	0.78	0.99	0.75	0.95	1.44	4.17
			60	1000	1.5	0.27	0.35	0.27	0.35	0.38	1.20
		$\rho = 0.1$	61	100	0.8	6.73	6.86	5.61	5.71	22.08	27.01
			63	100	1.5	1.95	2.00	2.92	2.99	22.90	32.98
			64	1000	1.5	NA	NA	0.72	0.74	0.73	1.90

Table 7.5.9: Average length of the confidence intervals for the desparsified LASSO methods and the EPSI, the correlation structure is the Toeplitz and the number of repetition is $R = 100$.

			design	n	sd	**OD LASSO**	**EPSI**
$p = 100$	$s = 5$	$\rho = 0.1$	17	100	0.8	1.00	0.60
			18	1000	0.8	0.47	0.46
			19	100	1.5	1.00	0.89
			20	1000	1.5	1.00	0.99
		$\rho = 0.8$	21	100	0.8	1.00	0.08
			22	1000	0.8	1.00	0.43
			23	100	1.5	1.00	0.73
			24	1000	1.5	1.00	1.00
	$s = 50$	$\rho = 0.1$	25	100	0.8	1.00	0.58
			26	1000	0.8	1.00	0.98
			27	100	1.5	1.00	0.89
			28	1000	1.5	1.00	0.99
		$\rho = 0.8$	29	100	0.8	1.00	0.07
			30	1000	0.8	1.00	0.47
			31	100	1.5	1.00	0.53
			32	1000	1.5	1.00	0.95
$p = 1000$	$s = 5$	$\rho = 0.1$	49	100	0.8	1	0.39
			51	100	1.5	1.00	1.00
			52	1000	1.5	1.00	1.00
		$\rho = 0.1$	53	100	0.8	1.00	0.11
			54	1000	0.8	1.00	0.04
			55	100	1.5	1.00	0.15
		$\rho = 0.8$	57	100	0.8	1.00	0.37
			59	100	1.5	1.00	0.82
			60	1000	1.5	1.00	0.99
	$s = 50$	$\rho = 0.1$	61	100	0.8	0.01	0.08
			63	100	1.5	1.00	1.00
			64	1000	1.5	1.00	0.98

Table 7.5.10: Success rate of the EPSI and OD LASSO methods for the design with a Toeplitz correlation structure computed on $R = 100$ repetitions.

Non Gaussian noise

					OD LASSO		**ND LASSO**		**EPSI**	
		design	n	m	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	1	0.86	1.09	1.32	1.67	5.60	11.30
		2	100	1	0.55	0.68	0.76	0.95	1.09	5.58
		3	50	0.667	1.28	1.61	1.54	1.94	3.47	11.01
		4	100	0.667	0.81	1.02	0.91	1.14	1.28	5.79
	$s = 40$	5	50	1	4.56	4.68	9.92	10.18	161.27	190.18
		6	100	1	2.20	2.25	3.79	3.88	NA	NA
		7	50	0.667	4.71	4.84	9.97	10.24	218.32	281.60
		8	100	0.667	4.31	4.41	3.99	4.08	NA	NA
$p = 500$	$s = 5$	9	50	1	1.16	1.43	0.48	0.60	1.24	2.53
		10	100	1	0.62	0.78	0.34	0.42	0.53	2.15
		11	50	0.667	1.79	2.22	0.52	0.65	1.15	2.20
		12	100	0.667	0.93	1.16	0.37	0.47	0.55	1.72
	$s = 40$	13	50	1	7.62	7.80	2.94	3.01	4.51	5.75
		14	100	1	2.01	2.06	2.12	2.17	NA	NA
		15	50	0.667	7.75	7.93	2.95	3.02	NA	NA
		16	100	0.667	2.10	2.15	2.13	2.18	NA	NA

Table 7.5.11: Confidence interval's length for linear regression with Exponential noise of parameter m.

					OD LASSO		**ND LASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**	
		design	n	m	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	1	3.45	0.23	3.85	0.24	2.32	0.20	0.19	0.19	2.06	0.20	3.78	0.20
		2	100	1	3.73	0.22	3.98	0.20	2.73	0.20	0.20	0.19	2.58	0.20	3.80	0.20
		3	50	0.667	3.34	0.27	3.93	0.29	1.82	0.20	0.19	0.19	1.54	0.20	3.79	0.20
		4	100	0.667	3.72	0.24	4.09	0.22	2.29	0.20	0.20	0.19	2.11	0.20	3.80	0.20
	$s = 40$	5	50	1	3.29	1.76	3.96	1.83	2.70	1.70	1.05	1.18	2.38	1.62	5.68	2.65
		6	100	1	2.37	1.53	2.50	1.43	2.16	1.52	1.01	1.14	2.14	1.51	4.75	1.76
		7	50	0.667	3.34	1.78	4.02	1.86	2.72	1.71	1.05	1.18	2.40	1.63	5.72	2.60
		8	100	0.667	2.41	1.54	2.53	1.41	2.16	1.50	1.02	1.14	2.12	1.49	4.68	1.76
$p = 500$	$s = 5$	9	50	1	2.96	0.08	3.47	0.10	1.79	0.04	0.04	0.04	1.23	0.04	3.99	0.04
		10	100	1	3.67	0.06	4.01	0.06	2.56	0.04	0.04	0.04	2.24	0.04	3.96	0.04
		11	50	0.667	2.67	0.13	3.49	0.20	1.13	0.04	0.04	0.04	0.76	0.04	3.91	0.04
		12	100	0.667	3.55	0.08	4.06	0.09	2.02	0.04	0.04	0.04	1.67	0.04	3.97	0.04
	$s = 40$	13	50	1	2.93	1.22	4.73	2.16	1.10	0.44	0.30	0.30	0.89	0.41	1.95	0.56
		14	100	1	2.03	0.44	2.42	0.52	1.47	0.37	0.31	0.30	1.33	0.37	2.19	0.45
		15	50	0.667	2.98	1.25	4.82	2.22	1.10	0.44	0.30	0.30	0.89	0.42	2.00	0.58
		16	100	0.667	2.04	0.45	2.44	0.53	1.45	0.38	0.31	0.30	1.31	0.37	2.22	0.45

Table 7.5.12: Mean Squared Error estimation when the noise follows an Exponential law with parameter $m = 1$ or 0.667.

Real data analysis

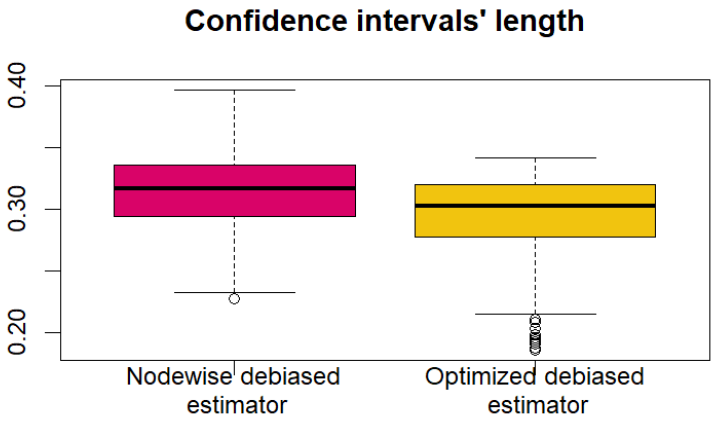

Figure 7.5.3: Comparison of the confidence intervals' lengths for the two methods on the riboflavin dataset.

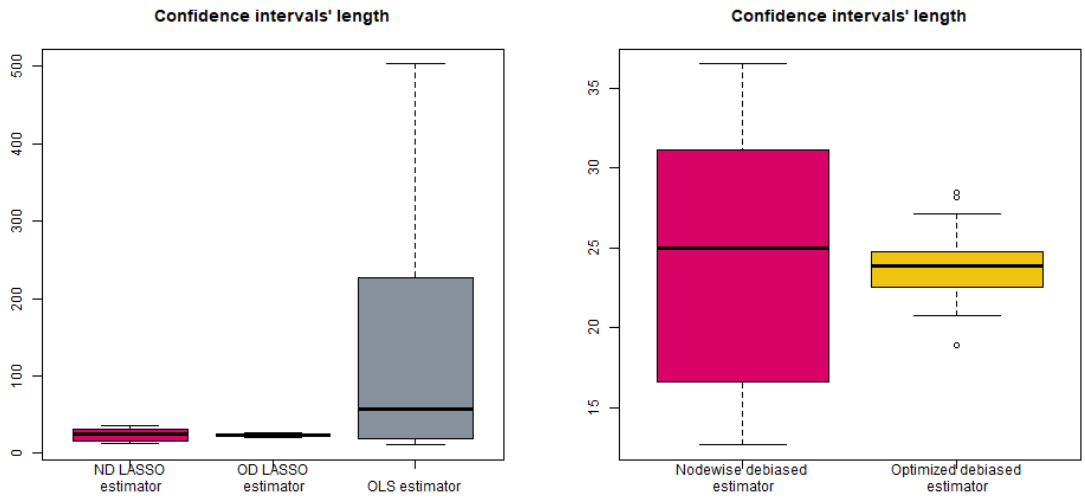

Figure 7.5.4: Comparison of the length for the confidence intervals on the diabetes dataset.

Model averaging

					OD MA		**ND MA**	
		design	n	m	active	null	active	null
$p = 100$	$s = 5$	1	50	1	-0.16	0.03	-0.04	0.07
		2	100	1	-0.04	0.01	0.00	0.03
		3	50	0.667	-0.27	0.03	-0.12	0.06
		4	100	0.667	-0.07	0.01	-0.03	0.02
	$s = 40$	5	50	1	-0.57	0.40	-0.34	0.64
		6	100	1	-0.06	0.05	-0.00	0.10
		7	50	0.667	-0.57	0.40	-0.35	0.64
		8	100	0.667	-0.09	0.07	-0.01	0.13
$p = 500$	$s = 5$	9	50	1	-0.38	0.04	-0.16	0.07
		10	100	1	-0.13	0.01	-0.02	0.04
		11	50	0.667	-0.64	0.03	-0.36	0.06
		12	100	0.667	-0.23	0.01	-0.09	0.03
	$s = 40$	13	50	1	NA	NA	-0.94	0.54
		14	100	1	NA	NA	-0.73	0.30
		15	50	0.667	-1.26	0.27	-0.97	0.51
		16	100	0.667	NA	NA	-0.91	0.30

Table 7.5.13: Bias for the model averaging estimators in a setting with non Gaussian noise. The simulated data are the same as the ones for 4.8.1 and the results can be compared.

					OD MA		**ND MA**	
		design	n	m	active	null	active	null
$p = 100$	$s = 5$	1	50	1	3.33	0.22	3.71	0.21
		2	100	1	3.68	0.22	3.82	0.20
		3	50	0.667	3.08	0.25	3.50	0.23
		4	100	0.667	3.61	0.24	3.72	0.21
	$s = 40$	5	50	1	3.09	1.70	3.51	1.66
		6	100	1	2.35	1.54	2.47	1.46
		7	50	0.667	3.13	1.73	3.54	1.72
		8	100	0.667	2.39	1.55	2.52	1.43
$p = 500$	$s = 5$	9	50	1	2.75	0.06	3.43	0.07
		10	100	1	3.51	0.05	3.88	0.05
		11	50	0.667	2.10	0.06	2.82	0.07
		12	100	0.667	3.16	0.06	3.65	0.06
	$s = 40$	13	50	1	NA	NA	2.31	0.85
		14	100	1	NA	NA	2.44	0.47
		15	50	0.667	1.70	0.57	2.30	0.81
		16	100	0.667	NA	NA	2.03	0.47

Table 7.5.14: Mean Squared Error for the model averaging desparsified estimators under the case where the distribution of the noise is Exponential with parameter m.

					ODLASSO		**NDLASSO**		**LASSO**		**RIDGE**		**EN**		**SCAD**		**OD MA**		**ND MA**	
		design	n	ρ	active	null	active	null	active	null	active	null	active	null	active	null	active	null	active	null
$p = 100$	$s = 5$	1	50	0.1	-0.07	0.03	-0.02	0.06	-0.26	0.00	-1.87	0.03	-0.34	0.01	-0.01	0.00	-0.07	0.03	-0.02	0.06
		2	100	0.1	-0.02	0.01	0.02	0.04	-0.17	0.00	-1.80	0.04	-0.21	0.00	0.00	0.00	-0.02	0.01	0.01	0.03
		3	50	0.8	-0.36	-0.09	-0.18	-0.04	-0.40	0.02	-1.89	0.08	-0.61	0.03	-0.22	0.01	-0.27	0.09	-0.23	-0.08
		4	100	0.8	-0.22	-0.05	-0.07	-0.01	-0.26	0.01	-1.88	0.08	-0.41	0.02	-0.03	0.00	-0.17	0.06	-0.10	-0.04
	$s = 40$	5	50	0.1	-0.49	0.46	-0.22	0.71	-0.91	0.26	-1.47	0.36	-0.91	0.27	-1.03	0.25	-0.58	0.39	-0.32	0.62
		6	100	0.1	-0.03	0.05	0.05	0.14	-0.13	0.04	-1.36	0.40	-0.14	0.04	-0.45	0.14	-0.04	0.04	0.04	0.13
		7	50	0.8	-2.10	-1.00	-1.24	-0.24	-0.77	0.42	-1.27	0.68	-0.78	0.44	-0.97	0.43	-2.16	-0.97	-1.24	-0.19
		8	100	0.8	-0.19	0.34	-0.20	0.17	-0.38	0.20	-1.25	0.69	-0.44	0.25	-0.66	0.28	0.52	1.05	-0.21	0.12
$p = 500$	$s = 5$	9	50	0.1	-0.15	0.04	-0.07	0.07	-0.38	0.00	-1.96	0.01	-0.62	0.00	-0.00	0.00	-0.16	0.04	-0.07	0.07
		10	100	0.1	-0.04	0.02	0.01	0.05	-0.23	0.00	-1.93	0.01	-0.30	0.00	0.00	-0.00	-0.05	0.02	-0.00	0.04
		11	50	0.8	-0.67	-0.10	-0.45	-0.04	-0.71	0.01	-1.97	0.02	-1.04	0.01	-0.42	0.00	-0.52	0.12	-0.46	-0.03
		12	100	0.8	-0.31	-0.05	-0.12	-0.01	-0.37	0.00	-1.97	0.02	-0.54	0.01	-0.02	0.00	-0.27	0.06	-0.15	-0.03
	$s = 40$	13	50	0.1	-0.82	0.61	-0.40	0.93	-1.68	0.07	-1.82	0.11	-1.67	0.07	-1.78	0.06	-1.29	0.27	-1.07	0.44
		14	100	0.1	-0.88	0.17	-0.74	0.28	-1.17	0.07	-1.77	0.12	-1.19	0.07	-1.42	0.06	-0.89	0.15	-0.73	0.28
		15	50	0.8	-2.80	-1.09	-2.08	-0.45	-1.63	0.12	-1.85	0.15	-1.67	0.13	-1.72	0.12	-3.31	-1.56	-2.41	-0.67
		16	100	0.8	-1.96	-0.60	-1.32	-0.11	-1.32	0.10	-1.84	0.15	-1.41	0.11	-1.43	0.10	-2.24	-0.88	-1.46	-0.19

Table 7.5.15: Comparisons of the bias for the different methods including the model averaging ones, the number of repetitions is $R = 100$.

		design	n	m	**OD MA**	**ND MA**
$p = 100$	$s = 5$	1	50	1	6	15
		2	100	1	4	4
		3	50	0.667	12	17
		4	100	0.667	9	7
	$s = 40$	5	50	1	136	299
		6	100	1	7	12
		7	50	0.667	144	304
		8	100	0.667	15	19
$p = 500$	$s = 5$	9	50	1	56	225
		10	100	1	10	46
		11	50	0.667	44	154
		12	100	0.667	14	36
	$s = 40$	13	50	1	NA	1568
		14	100	1	NA	1028
		15	50	0.667	1284	6685
		16	100	0.667	NA	966

Table 7.5.16: Out of sample prediction error for the model averaging estimators in case where the noise is Exponential with parameter m. These metrics should be compared to the ones in 4.8.2 since the generated data by design are the same.

Generalized Linear Models

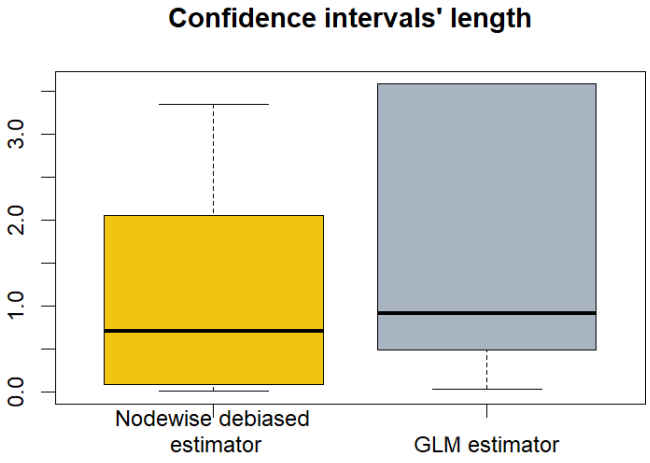

Figure 7.5.5: Comparison of the confidence intervals' length between the ND LASSO estimator and the GLM regression for the lungcap dataset.