Model selection in high-dimensional noisy data: a simulation study.

Giovanni Romeo ^a and Magne Thoresen ^a

^aOslo Centre for Biostatistics and Epidemiology, Department of Biostatistics, University of Oslo, Norway

ABSTRACT

In many practical applications, high-dimensional regression analyses have to take into account measurement error in the covariates. It is thus necessary to extend regularization methods, that can handle the situation where the number of covariates p largely exceed the sample size n , to the case in which covariates are also mismeasured. A variety of methods are available in this context, but many of them rely on knowledge about the measurement error and the structure of its covariance matrix. In this paper we set the goal to compare some of these methods, focusing on situations relevant for practical applications. In particular, we will evaluate these methods in setups in which the measurement error distribution and dependence structure are not known and have to be estimated from data. Our focus is on variable selection, and the evaluation is based on extensive simulations.

KEYWORDS

Measurement error; High-dimensional regression; Lasso; Matrix uncertainty selector; Convex conditional lasso; Variable selection

1. Introduction

Modern Statistics is facing problems due to the increase of dimensionality of the data in fields such as genomics, finance, climate science, network analysis, neuroimaging, etc. It is quite canonical in high-dimensional regression, where the number of variables p largely exceeds the sample size n , to assume that the number of covariates s that has an effect on the response variable y is much less than n . Hence, the vector of regression parameters is assumed to be s -sparse. A plethora of high-dimensional regression methods is available, among which we can mention Lasso [1], Dantzig selector (DS) [2], Smoothly Clipped Absolute Deviation Penalty (SCAD) [3]. These methods all allow model selection and parameter estimation through a penalization of the parameters.

These methods are developed for the case in which the covariates are fully observed and without errors. However, in many applications, our data are subject to at least some measurement error. Some examples are gene expression microarrays [4,5], sensor network data [6,7], surveys and questionnaires [8] and food frequency questionnaires [9].

In the classical regression context, when $p < n$ and standard methods can be applied, it is well known that measurement error in the covariates will lead to bias in the estimation of the parameters and to loss of power [10]. Many correction methods are available to reduce this bias, but they generally need information about the measurement error structure. Typically this is taken from validation data or from replicated measurements.

Applying high-dimensional regression methods that do not account for measurement

error on error-prone data will also result in misleading inference, as was demonstrated for the Lasso and the Dantzig Selector [11] [12]. Correction for measurement error in penalized regression has been studied by various authors recently. Examples include the non-convex lasso (NCL) of Loh and Wainwright [13] the CoCoLasso of Datta and Zou [14] that both use the covariance matrix of the measurement error in the model. Furthermore, the matrix uncertainty selector (MUS) of Rosenbaum and Tsybakov [11] is an evolution of the Dantzig selector with a supplementary tuning parameter for the measurement error. Further developments of this method include the Improved matrix uncertainty selector [15] and the Conic estimator of Belloni et al. [16] which both again use the covariance matrix of the measurement error, while Sørensen et al. [17] extended the matrix uncertainty selector to generalized linear models.

Correction of penalized regression is not the only method available to correct for measurement error. For example the two stage non-penalized corrected least squares [18] separates model selection and estimation in two different steps, using the corrected least squares for the final estimation; the Measurement Error Boosting [19], is an iterative functional gradient descent type algorithm that generates measurement error corrected variable selection paths. Chen and Caramanis [20] developed a modified version of the orthogonal matching pursuit algorithm, also to deal with measurement error in the covariates in high-dimensional regression.

Most of these techniques require knowledge of the measurement error distribution or at least the covariance matrix of the measurement errors, yielding estimators with good theoretical properties. However, in practice this will almost always be unknown, and estimation of the covariance matrix of the measurement error can be computationally expensive or even unfeasible when the number of variables p increases. One exception is the MUS [11] that can account for measurement error without additional information about the measurement error structure, but sacrifice some statistical properties to get this robustness.

Generally, modelling and estimation of measurement errors and associated parameters are often challenging. It requires some knowledge about the measurement error process, and additional data are typically necessary for estimation.

The main goal of this paper is to look into penalized regression with measurement error and to compare some of these methods in a practical context, where the distribution and the variance of the measurement error is unknown and need to be estimated from data. We are particularly interested in to what extent this will influence the behavior of methods that use such measurement error information compared to those that do not. We will focus on the non-convex lasso of Loh and Wainwright as this is in many ways the most natural way to include the measurement error effect in the lasso regression, the CoColasso of Datta and Zou because it is strictly connected to the non-convex lasso, and the MUS of Rosenbaum and Tsybakov, as this method does not require any information about the measurement error distribution. Our main interest is variable selection. However estimation error will also be considered in the comparison.

The comparison will be based on simulations, and the focus will be on the linear regression model with additive measurement error.

The paper is organized as follows. We start in Section 2 with a brief overview of the models and methods that we will investigate. In Section 3 we give a brief overview of some of their theoretical properties. Section 4 describes the simulation setups and results. Section 5 contains a real data example, and Section 6 is devoted to the interpretation of the results and some final conclusions.

2. Models and Methods

In this work we will focus on the linear regression model, that in matrix-vector notation can be defined as:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where only the random vector $\mathbf{y} \in \mathbb{R}_n$ is observed. The p -dimensional vector $\boldsymbol{\beta}$ is the unobserved set of parameters we are interested in and $\boldsymbol{\epsilon} \in \mathbb{R}_n$ is the residual error, assumed to be independent and identically distributed. The matrix of the covariates $\mathbf{X} \in \mathbb{R}_{n \times p}$ cannot be directly observed due to measurement error.

The measurement error interpretation and definition is not unique. Two of the possible ways of modeling measurement error are through additive and multiplicative models. As previously mentioned, in this work we will focus on the additive measurement error model.

In the additive measurement error model the unobservable variable $\mathbf{X}$ is altered by adding random measurement error $\mathbf{U}$, so what we observe is:

$$w_{ij} = x_{ij} + u_{ij} \quad \text{for } i = 1, \dots, n \quad \text{and} \quad j = 1, \dots, p$$

A key assumption is that the measurement error is centered on zero; $E(u_{ij}) = 0$, so that $E(w_{ij}|x_{ij}) = x_{ij} \forall i, j$. This can be re-formulated in matrix notation:

$$\mathbf{W} = \mathbf{X} + \mathbf{U}$$

where $\mathbf{U}$ is an $n \times p$ random noise matrix with covariance matrix $\boldsymbol{\Sigma}_U$. $\mathbf{U}$ is assumed to be independent of $\boldsymbol{\epsilon}$. When measurement error is present and $\boldsymbol{\Sigma}_U$ is not known, it can be estimated through replicated measurements of $\mathbf{W}$. We investigate the frequently studied situation where there are M replicated measurements of the covariates. For the i -th individual we observe $W_{i1}, W_{i2}, \dots, W_{iM}$ and $\bar{W}_i$ is their mean. Replication allows us to estimate $\boldsymbol{\Sigma}_U$ as:

$$\hat{\boldsymbol{\Sigma}}_U = \frac{1}{n} \frac{\sum_{i=1}^n \sum_{k=1}^M (W_{ik} - \bar{W}_i)(W_{ik} - \bar{W}_i)'}{M - 1} \quad (1)$$

In a high-dimensional context it is intuitively clear that the vector of parameters $\boldsymbol{\beta}$, given n observations, can be estimated reasonably well only if some sort of sparsity is present in $\boldsymbol{\beta}$. Assume the number of true non-zero coefficients to be s . As a general guideline one can say that high-dimensional statistical inference can lead to acceptable level of accuracy if:

$$\log(p) \times s \ll n$$

2.1. The lasso and the Dantzig selector

Shrinkage methods make estimation in high-dimensional regression possible by imposing a penalty on the size of the regression coefficients. Many different penalizations are possible, leading to different results. The Least Absolute Shrinkage and Selection Operator [1], in acronym lasso, owes its popularity to its prediction properties and

ability to perform variable selection along with its computational feasibility and common software implementation. Moreover the Lasso is a penalized likelihood approach, making it is easier to be generalized and extended.

The lasso is defined as the solution to the optimization problem:

$$\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \text{ subject to } \|\beta\|_1 \leq t$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm and $\|\cdot\|_2$ denotes the ℓ_2 norm. We can also rewrite the Lasso definition in the equivalent and most common Lagrangian form with regularization parameter λ :

$$\hat{\beta}(\lambda) = \arg \min_{\beta} \{\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_1\} \quad (2)$$

The lasso is not the only ℓ_1 penalization possible. The Dantzig selector (DS) [2] can be formulated as the lasso but instead of controlling the squared error loss, it controls the correlation of residuals with $\mathbf{X}$. The Dantzig selector was proposed by Candes and Tao as the solution to the minimization problem:

$$\min_{\beta} \|\mathbf{X}'(\mathbf{y} - \mathbf{X}\beta)\|_{\infty} \text{ subject to } \|\beta\|_1 \leq t$$

where $\|\cdot\|_{\infty}$ denotes the ℓ_{∞} norm.

Both models share the feature of setting some of the parameters to zero i.e. they perform variable selection. If $n < p$ both models lead to the same solution, the least squares solution. When $n > p$ and for small values of t the solution paths of the two models may differ [21].

2.2. Non-convex lasso (NCL)

So far these models were formulated under the assumption that $\mathbf{X}$ is observable. However, with measurement error we only observe the error-prone version $\mathbf{W}$. The naive lasso approach would be to just plug in $\mathbf{W}$ for $\mathbf{X}$ in the lasso estimator defined in (2):

$$\hat{\beta}(\lambda_n) = \arg \min_{\beta} \{\|\mathbf{y} - \mathbf{W}\beta\|_2^2 + \lambda_n \|\beta\|_1\}$$

However, this yields a biased loss function:

$$E(\|\mathbf{y} - \mathbf{W}\beta\|_2^2 | \mathbf{X}, \mathbf{y}) = \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + n\beta' \Sigma_U \beta$$

The most natural way of correcting for this bias leads to the constrained corrected lasso (CCL):

$$\hat{\beta}_{CCL} \in \arg \min_{\|\beta\|_1 \leq R} \frac{1}{n} \|\mathbf{y} - \mathbf{W}\beta\|_2^2 - \beta' \Sigma_U \beta \quad (3)$$

introduced by Loh and Wainwright [13]. The parameter space here has to be restricted to the ℓ_1 -ball $\{\beta : \|\beta\|_1 \leq R\}$ with some finite radius R , because the loss function of

the CCL is always non-convex when $p > n$. Notice that Σ_U is in practice unknown and must be estimated from data.

The estimator (3) can be reformulated as:

$$\hat{\beta}_{CCL} \in \arg \min_{\|\beta\|_1 \leq R} \frac{1}{2} \beta' \hat{\Sigma}_X \beta - \tilde{\rho}' \beta \quad (4)$$

for some constant R . Here $\hat{\Sigma}_X = \frac{1}{n} \mathbf{W}' \mathbf{W} - \Sigma_U$ and $\tilde{\rho} = \frac{1}{n} \mathbf{W}' \mathbf{y}$. The regularized version of this estimator (regularized corrected lasso, RCL) solves the corresponding problem:

$$\hat{\beta}_{RCL} \in \arg \min_{\|\beta\|_1 \leq b_0 \sqrt{s}} \frac{1}{2} \beta' \hat{\Sigma}_X \beta - \tilde{\rho}' \beta + \lambda_{RCL} \|\beta\|_1$$

for some constant b_0 . Note that ‘ $\in$ ’ and not ‘ $=$ ’ has been used, because there can be multiple local or global minimizers. Loh and Wainwright [13] demonstrated that if b_0 is properly chosen, the projected gradient descent algorithm will converge to a small neighborhood of the set of all global minimizers. From now on we will refer to both estimators as NCL, despite for computational reasons only the CCL has been considered in the simulation study.

2.3. Convex conditional Lasso (CoCoLasso)

A clear drawback of the previous method is that it leads to a non-convex optimization problem. The idea behind CoCoLasso is to intervene directly on $\hat{\Sigma}_X$, the estimated covariance matrix of $\mathbf{X}$, with a transformation that will provide a positive semi-definite matrix. First, we define the nearest positive semi-definite matrix projection operator:

$$(\mathbf{G})_+ = \arg \min_{\mathbf{G}_1 \geq 0} \|\mathbf{G} - \mathbf{G}_1\|_{max}$$

where $\|\mathbf{G}\|_{max} = \max_{i,k} |g_{ij}|$ denotes the elementwise maximum norm. This operator will project the matrix $\hat{\Sigma}_X$ into a space of positive semi-definite matrices, selecting the nearest one. Then by denoting $\tilde{\Sigma}_X = (\hat{\Sigma}_X)_+$ the CoCoLasso estimator is defined as:

$$\hat{\beta}_{coco} = \arg \min_{\beta} \frac{1}{2} \beta' \tilde{\Sigma}_X \beta - \tilde{\rho}' \beta + \lambda_{coco} \|\beta\|_1 \quad (5)$$

Note that the matrix $\tilde{\Sigma}_X$ is always positive semi-definite by construction while $\hat{\Sigma}_X$ is guaranteed to be positive semi-definite only for $p < n$. Consequently the optimization problem in (5) is guaranteed to be convex. To obtain the projection, the authors use the alternating direction method of multipliers (ADMM) algorithm [22]. The estimator can again be reformulated as:

$$\hat{\beta}_{coco} = \arg \min_{\beta} \frac{1}{n} \|\tilde{\mathbf{y}} - \tilde{\mathbf{X}} \beta\|_2^2 + \lambda_{coco} \|\beta\|_1 \quad (6)$$

where $\tilde{\mathbf{X}}/\sqrt{n}$ is the Cholesky factor of $\tilde{\Sigma}$ and $\tilde{\mathbf{y}}$ is such that $\tilde{\mathbf{X}}'\tilde{\mathbf{y}} = \tilde{\rho}$. This is the regular lasso regression of $\tilde{\mathbf{y}}$ on $\tilde{\mathbf{X}}$ with penalization parameter λ_{coco} . This is of great advantage for the practical implementation. We can apply any standard lasso algorithm, as the coordinate descent [23] or the least angle regression [24] to obtain a solution.

2.3.1. Calibrated cross-validation

The choice of the tuning parameter in penalized methods often relies on cross-validation. Cross-validation in the presence of measurement error has received little attention, although its naive application might lead to biased results. Consider the usual k -fold cross-validation and let $(\mathbf{X}_k, \mathbf{y}_k)$ denote the subset of $\mathbf{X}$ and $\mathbf{y}$ for the k -th fold, with $k = 1, \dots, K$. In the regular lasso the optimal λ is obtained by minimizing the total cross-validation error:

$$\hat{\lambda} = \arg \min_{\lambda} \frac{1}{K} \sum_{k=1}^K \frac{1}{n_k} \|\mathbf{y}_k - \mathbf{X}_k \hat{\beta}_k(\lambda)\|_2^2 \quad (7)$$

where n_k is the number of observations in the k -th fold and $\hat{\beta}_k(\lambda)$ is the lasso estimate based on $(\mathbf{X}_{-k}, \mathbf{y}_{-k})$ (the data after removing the k -th fold) and the tuning parameter λ .

When measurement error is present and $\mathbf{X}$ cannot be directly observed, we can naively use the observed data $(\mathbf{W}, \mathbf{y})$ and minimize:

$$\frac{1}{K} \sum_{k=1}^K \frac{1}{n_k} \|\mathbf{y}_k - \mathbf{W}_k \hat{\beta}_k(\lambda)\|_2^2 \quad (8)$$

where $\mathbf{W}_k$ is the subset of $\mathbf{W}$ for the k -th fold.

However, even if CoCoLasso is used to estimate $\hat{\beta}_k(\lambda)$ from $(\mathbf{W}_{-k}, \mathbf{y}_{-k})$ the minimization in (8) will be biased due to the measurement error.

Datta and Zou [14] reformulated the estimator (7) as:

$$\hat{\lambda} = \arg \min_{\lambda} \frac{1}{K} \sum_{k=1}^K \hat{\beta}_k(\lambda)' \Sigma_{X_{(k)}} \hat{\beta}_k(\lambda) - 2 \rho'_k \hat{\beta}_k(\lambda)$$

and substituted Σ_X and ρ with their projected and estimated counterparts $\tilde{\Sigma}_X$ and $\tilde{\rho}$. The minimization of this loss function will lead to an unbiased estimator of λ_{coco} :

$$\hat{\lambda}_{coco} = \arg \min_{\lambda} \frac{1}{K} \sum_{k=1}^K \hat{\beta}_k(\lambda_{coco})' \tilde{\Sigma}_{X_{(k)}} \hat{\beta}_k(\lambda) - 2 \tilde{\rho}'_k \hat{\beta}_k(\lambda)$$

2.4. Matrix uncertainty selector (MUS)

Both of the previous methods assume the covariance matrix of the measurement errors to be known, or at least that some estimator is available. This is often not the case. Rosenbaum and Tsybakov [11] developed the MUS adding to the DS a new parameter δ that bounds the magnitude of the measurement error. They showed that when:

$$\frac{1}{n} \|\mathbf{W}'\boldsymbol{\epsilon}\|_{\infty} \leq \lambda_{MUS} \text{ and } \|\mathbf{U}\|_{\infty} \leq \delta$$

where $\|\cdot\|_{\infty}$ is the maximum component norm, the vector of the true parameters β_0 is a feasible solution of the MUS, defined as follows:

$$\hat{\beta}_{MUS} = \arg \min_{\beta} \|\beta\|_1 : \frac{1}{n} \|\mathbf{W}'(\mathbf{y} - \mathbf{W}\beta)\|_{\infty} \leq \lambda_{MUS} + \delta \|\beta\|_1 \quad (9)$$

It is worth noting that the estimation error bounds for the MUS do not go to zero when $n \rightarrow \infty$. This is the price to pay for not knowing the measurement error distribution.

2.4.1. Choice of tuning parameters

The choice of the unknown tuning parameters in the MUS is from a practical point of view not trivial. We chose to select the parameter λ through cross-validation. In particular we used the parameter selected by cross-validation of the naive lasso, i.e. the lasso regression of $\mathbf{y}$ on $\mathbf{W}$.

The choice of the parameter δ is critical. Low values of δ implies that the measurement error is not taken into account, while big values will overweight this effect. An objective rule for the selection of this parameter is not available, but the authors suggested a rule of thumb, using the ‘elbow point’. Through a graphical representation of the number of selected variables as a function of δ we will select the point where the plot, after a quick decrease, starts ‘to flatten out’ [11], see Fig. 1. This, however, is a very subjective criterion, and as will be seen from the simulations, does not necessarily work well in all situations.

3. Theoretical properties

Sign consistent variable selection is possible in the ordinary lasso under a beta-min condition which essentially says that the size of the non-zero coefficients cannot be too small, and an irrepresentable condition (IR) which says that the variables in the active set S cannot be too highly correlated with themselves and/or the remaining variables [25,26]. Formally, the beta-min condition states that the non-zero coefficients must satisfy the following condition:

$$\inf_{j \in S} |\beta_j| \gg \sqrt{s \log(p)/n}.$$

Furthermore, let $\mathbf{C} = \frac{1}{n} \mathbf{X}'\mathbf{X}$. Without loss of generality, assume that the active set consists of the first s variables. The matrix $\mathbf{C}$ can then be expressed in a block-wise form as

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix}$$

Figure 1. Elbow plots for six different datasets generated with a common set of covariates, of dimensions $n = 200$, $p = 1000$ and vector of coefficients $\beta_{(i \in S)} = 1$ (and zero outside the active set). The parameter λ has been selected by cross-validation on the naive model and the curves have been smoothed with polynomial regression for graphical purpose.

where $\mathbf{C}_{11}$ is an $s \times s$ matrix corresponding to the active set, $\mathbf{C}_{12} = \mathbf{C}_{21}'$ is $s \times (p - s)$ and $\mathbf{C}_{22}$ is $(p - s) \times (p - s)$. The IR is then defined as

$$\|\mathbf{C}_{21}\mathbf{C}_{11}^{-1}\text{sign}(\beta_1, \dots, \beta_s)\|_\infty \leq \theta \text{ for some } 0 < \theta < 1$$

where $\text{sign}(\beta_1, \dots, \beta_s) = [\text{sign}(\beta_1), \dots, \text{sign}(\beta_s)]'$.

The theoretical properties of the three correction methods when it comes to variable selection have been established in [11,12,14]. All three methods are shown to give sign consistent variable selection under certain conditions.

Sørensen et al. [12] showed that the NCL yields sign consistent variable selection under conditions corresponding to the ones for the ordinary lasso, as long as the local optimum $\hat{\beta}$ satisfies $\|\hat{\beta}\|_1 < R$, where R is the constant from (3). More formally, let

$$\hat{\Sigma}_X = \begin{bmatrix} \hat{\Sigma}_{X_{11}} & \hat{\Sigma}_{X_{12}} \\ \hat{\Sigma}_{X_{21}} & \hat{\Sigma}_{X_{22}} \end{bmatrix}.$$

Definition 4 in [12] states that the IR for the NCL holds if $\hat{\Sigma}_{X_{11}}$ (the part of $\hat{\Sigma}_X$ corresponding to the active set) is invertible and there exists a constant θ in $[0, 1)$ such that

$$\|\hat{\Sigma}_{X_{21}}\hat{\Sigma}_{X_{11}}^{-1}\text{sign}(\beta_1, \dots, \beta_s)\|_\infty \leq \theta$$

Correspondingly, Datta and Zou [14] established sign consistency for the CoCoLasso, essentially under the same conditions as for the ordinary lasso. However, they also needed a closeness condition; that the surrogates $\tilde{\Sigma}_X$ and $\tilde{\rho}$ are sufficiently close to the populations values Σ_X and ρ in terms of the elementwise maximum norm.

Finally, Rosenbaum and Tsybakov [11] showed that the thresholded version of the MUS also provides sign consistent variable selection under conditions similar to the above; a type beta-min condition stating that the non-zero coefficients must be larger than a given threshold and a coherence condition which essentially says that the explanatory variables $\mathbf{X}$ cannot be too highly correlated. Formally, they state that the off-diagonal elements of the normalized version of the matrix $\mathbf{C}$ defined above must be $\leq (3\alpha s)^{-1}$ for some $\alpha > 1$.

These are rather strict assumptions. They also assume all regularization parameters known, in addition to the covariance matrix of the measurement errors (for the first two methods). Hence, although interesting from a theoretical point of view, these properties should be considered with some care from a practical viewpoint.

4. Simulation study

The goal of this simulation study is to compare the three methods; the non-convex lasso, the convex conditional lasso and the matrix uncertainty selector, in high-dimensional regression problems where the measurement error and corresponding parameters are unknown and must be estimated from data. We will focus on variable selection, but will also look at estimation error. In addition to the three correction methods, we will also show the results of the naive lasso and the lasso that uses the true $\mathbf{X}$ (the "true" lasso).

4.1. Simulations design

We simulate data from the true model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(0, \sigma_\epsilon) \quad (10)$$

where $\mathbf{X}$ has been generated as $\mathbf{X} \sim N(0, \boldsymbol{\Sigma}_X)$. All the coefficients belonging to the active set S of dimension s are set to one: $\beta_i = 1$ if $i \in S$ and zero otherwise, the active set indexes have been chosen randomly for each simulated scenario. The model error variance is defined as $\sigma_\epsilon^2 = 9$.

The observed data were generated with two replicated measurements per observation:

$$w_{ijk} = x_{ij} + u_{ijk} \quad \text{for } i = 1, \dots, n; \quad j = 1, \dots, p; \quad k = 1, 2 \quad (11)$$

in matrix notation $\mathbf{W}_k = \mathbf{X} + \mathbf{U}_k$, $k = 1, 2$. The measurement error has been generated as $\mathbf{U}_k \sim N(0, \boldsymbol{\Sigma}_U)$. We indicate with $\bar{\mathbf{W}}$ the per-replica observed average with each element defined as: $\bar{w}_{ij} = (w_{ij1} + w_{ij2})/2$ and we refer to the lasso regression of $\mathbf{y}$ on $\bar{\mathbf{W}}$ as the naive lasso.

We investigated different choices of measurement error structures, different sizes of s , n and p , and different choices of $\boldsymbol{\Sigma}_X$. In particular, we considered:

- datasets of two different dimensions: $n = 200$ and $p = 1000$ (small set) or $n = 500$ and $p = 2000$ (big set). For some of the most interesting frameworks a dataset of medium dimension $n = 200$ and $p = 2000$ has also been simulated.
- two different sets of $\boldsymbol{\beta}$, with the number of signals s equal to 5 or 20 (with random positions over the p variables) and all values equal to one.
- the covariates $\mathbf{X}$ have been generated with two different covariance matrices: diagonal, where $\boldsymbol{\Sigma}_X = \mathbf{I}_p \sigma_X^2$ with $\sigma_X^2 = 1$, or autoregressive, where $\boldsymbol{\Sigma}_{X,ij} = 0.5^{|i-j|}$ and 1 on the diagonal.
- three different measurement error schemes: 1) independence, with $\boldsymbol{\Sigma}_U = \mathbf{I}_p \sigma_U^2$ with $\sigma_U^2 = 0.5$ chosen such that the reliability ratio of $\bar{\mathbf{W}}$ is equal to 0.8; 2) correlation of the measurement error of the variables belonging to the active set, updating $\boldsymbol{\Sigma}_U$ outside the diagonal such that $\boldsymbol{\Sigma}_{U,ij} = 0.125$ if i and j belongs to the active set; and 3) correlation among the measurement errors of the variables belonging to the active set (as in 2) and a subset of the other variables. Specifically, for all $X_i \in S$, we correlate also to all $X_{i \pm j}$, $j = 1, \dots, 4$. We use $\boldsymbol{\Sigma}_{U,ij} = 0.125$ for all relevant i, j . For each of these schemes we considered measurement errors generated as Normal, Chi-square distributed (with one degree of freedom) and Uniform.

Many of the previous settings have also been re-simulated with a small model error of $\sigma_\epsilon^2 = 0.01$. For each combination of these elements the estimation procedures were repeated in 100 independent Monte Carlo experiments.

It is worth noticing that in the situations when $\boldsymbol{\Sigma}_X$ is not diagonal, the criteria for consistent variable selection in the MUS discussed in Section 3, are not met.

4.2. Model implementation

In this section we will describe how the models described in section 2 have been implemented and how the choice of tuning parameters have been made. The models were implemented as suggested by the authors but with some adjustments to lighten the computational burden.

The NCL authors in their original formulation assumed $\|\beta\|_1$ known and consequently making the choice of R in (3) trivial, but in practice this quantity is not likely to be known. Sørensen et al. [12] and Datta and Zou [14] suggested to select the optimal R through a cross-validation procedure, choosing the optimal R such that it minimize the cross-validated loss in the interval $[R_{max}/500; R_{max}]$ where $R_{max} = 2 \cdot \|\hat{\beta}_{naive}\|_1$. We follow the same strategy, but monitor the selected values of R for the possibility of having $R = R_{max}$. We also added a supplementary correction for when $R_{max} = 0$, i.e. the naive method does not select any variable. In these cases R_{max} has been set equal to 10 for $s = 5$ and to 25 for $s = 20$.

The CoCoLasso has been implemented as the authors suggested, solving the minimization problem with the LARS algorithm and selecting the penalization parameter λ through the calibrated cross-validation procedure described in Section 2.3.1, but only with 5 folds.

Regarding the MUS, we decided to approximate the optimal λ with the penalization parameter selected by the cross-validation of the naive lasso. After that, the curve in Figure 1 has been estimated by taking the average number of non-zero coefficients over a subset of 10 simulations (5 for the bigger data setting) for each set-up. Based on this we selected two different values of δ without knowing the data-generating model. Before proceeding with the analyses, all the variables have been centered on zero and all the methods have been estimated without intercept.

Both the NCL and the CoCoLasso has been formulated assuming Σ_U known. We are also interested in investigating how these methods perform when an estimator of Σ_U (based on the replicates of $\mathbf{W}$) is being used instead. In all scenarios Σ_U has been estimated with the estimator in (1). Furthermore, we have also estimated Σ_U by assuming independent measurement errors ($\text{cov}(\mathbf{U}_i, \mathbf{U}_j) = 0, i \neq j$), as this will be the most relevant solution when p grows, leading to the estimator $\hat{\Sigma}_U^{diag}$.

The experiments have been carried out using R version 3.4.3 on the Abel computer cluster of the University of Oslo. For the estimation of the MUS we used the GLPK package (GNU Linear Programming Kit, <https://www.gnu.org/software/glpk/>) and the connected R package ‘Rglpk’.

4.3. Simulation results

We will describe the results dividing the simulations into three subsections according to the measurement error structure: in the first subsection the model assumptions about the measurement error are correct and $\hat{\Sigma}_U^{diag}$ is a valid estimator of Σ_U , while in the other two subsections the assumption of independent measurement errors is violated. We will present the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm ($\ell_2 = \|\hat{\beta} - \beta_{true}\|_2^2$). All results are reported as median values with interquartile range.

4.3.1. Independent measurement errors

When no correlation is present among measurement errors (tables 1 and 2), no correction method is overall better than the naive estimator.

When focusing on model selection, the regression of $\mathbf{y}$ on $\mathbf{W}$ always guarantees the highest sensitivity, together with the CoCoLasso. For $s = 5$ the naive lasso and the NCL perform about the same, while for $s = 20$, the NCL reduces the number of false positives compared to the naive lasso. The CoCoLasso generally selects far too many variables.

The MUS works really well in detecting a small number of signals in the bigger data setting ($n=500$, $p=2000$) where it is able to pick up the signals with very few false positives. However, generally the MUS selects too few variables. This is particularly pronounced for $s = 20$. The CoCoLasso, despite always being able to select the right variables, tends to select too many false positives, much more than the naive estimator.

Concerning estimation error both the NCL and the CoCoLasso perform better than the naive estimator; the NCL in some cases considerably better. The MUS is generally the worst method from an estimation point of view. The estimation of Σ_U does not seem to influence the performance of the NCL and the CoCoLasso a lot. Both models perform similarly from both the variable selection and estimation point of view, with ME variance known or estimated. The presence of some moderate correlation among the covariates does not seem to affect the results much (table S1 and S2 of supplementary material show results for simulations with uncorrelated $\mathbf{X}$). This is true also for the MUS, even if the additional correlation in $\mathbf{X}$ should be detrimental for the criteria for MUS consistent variable selection. For this reason all the remaining simulations presented in this work will be generated with the autoregressive version of Σ_X .

4.3.2. Measurement error correlated within the active set

When there is some correlation between the measurement errors of the variables in the active set, the situation is slightly changed (Tables 3 and 4). Regarding variable selection, the NCL performs about the same as the naive method for $s=5$ but considerably better for $s=20$ as long as Σ_U is known or estimated correctly. However, when Σ_U is estimated erroneously and the correlation among the measurement errors is not taken into account, the NCL selects too few variables. This is particularly pronounced for $s=20$. In particular, as shown in the boxplots in Figure 2, the number of correctly selected variables by the NCL decreases drastically when a wrong estimator is used for Σ_U . The MUS again performs well for $s=5$, but selects too few variables for $s=20$. The CoCoLasso generally selects far too many variables. All methods improve a bit going from the smaller to the larger data setting. When it comes to estimation error, the NCL does very well as long as Σ_U is known or correctly estimated. Otherwise, there are no major differences between the methods.

4.3.3. Measurement error correlated beyond the active set

When a wider correlation structure is present in the measurement error (tables 5 and 6) the NCL performs clearly better than the other correction methods when it comes to variable selection, as long as Σ_U is known or estimated correctly. For $s = 20$ it also performs better than the naive method. The NCL is able to recover the full active set for $s = 5$ and most of it for $s = 20$, while selecting a small number of false positives (smaller than the true lasso in some cases). However, when Σ_U is estimated assuming independence among the measurement errors, the NCL performs considerably worse,

Figure 2. Distributions of the correctly selected variables for naive, NCL and CoCoLasso estimators in simulations with $s = 20$ and correlation present in the measurement error covariance matrix Σ_U (simulations in tables 4 and 6).

in particular for $s = 20$. As earlier, the MUS performs reasonably well for $s = 5$, in the sense that it is able to select as many of the important variables as the other methods and at the same time reducing the number of false positives. For $s = 20$, the MUS selects far too few variables. Also as before, the CoCoLasso in general selects a large number of false positives. When it comes to estimation performance, the NCL clearly outperforms the other methods, including the naive method.

4.4. Non-normal measurement error distribution

The results reported in tables 1 to 6 refer to the cases where the measurement errors were generated as normally distributed. We also explored some non-normal frameworks, with measurement errors generated as chi-square with one degree of freedom or as uniform. These simulation results show the same overall picture as with normally

distributed errors and are presented in the Supplementary material, tables S6 to S11.

4.5. *Small model error*

The result shown so far try to mimic a real application context. Here we observed some incongruence with what has been shown earlier, in particular with the naive model: it has been shown in simulations [12] that in presence of ME the naive estimator should generally select more variables than the true lasso. The model error might play a key role in this process, and for this reason we re-simulated some of the cases above (tables 7-10). As expected, the naive method over-selects compared to the true lasso. The NCL with Σ_U correctly estimated improves a lot on the naive: it decreases the number of variables selected and drastically reduces the estimation error compared to what can be observed for the corresponding simulations with a larger model error. As observed in the previous simulations the CoCoLasso tends to select too many variables, in particular for $s=20$. The MUS, despite being particularly efficient in selecting the active set for $s = 5$, falls short in all the $s = 20$ cases.

For most of the simulated situations, also the dimension $n = 200$ and $p = 2000$ has been explored. These results (shown in supplementary material) are comparable to what we observed for the smaller data settings, leading us to conclude that the differences we observed between the small and the big sets are mainly due to an increased n rather than increased p .

5. Analysis of a Microarray Data Set

In this section we show an application to a microarray data set containing 54675 probe sets and measurements of bone mineral density (BMD), for 84 Norwegian women [27] (data available at the European Bioinformatics Institute ArrayExpress repository, ID: E-MEXP-1618). Microarray measurements are known to be noisy [4,28,29], consequently this is an appropriate context for comparing the NCL, the CoCoLasso and the MUS. In the Affymetrix microarray data, multiple probes are used to measure each gene expression. These replicated measurements are analyzed with the Bayesian Gene Expression (BGX) Bioconductor package [30] that provides estimates of the posterior distributions of the mean gene expression (measured on the log scale) for each sample.

Due to the computational burden we decided to pre-select a subset of variables with simpler methods. First the Robust Multi-array Average (RMA) [31] (which compiles affymetrix chip data into gene expression data) has been used to obtain estimated gene expressions. Then the 800 genes with the highest correlation with the response have been chosen by iterative sure independence screening [32]. On top, we added the genes that were selected in [27], leading to a total of 811 genes, or probe sets.

We denote with $\hat{\mu}_{ij}$ the BGX posterior mean gene expression for $i = 1, \dots, n$ and $j = 1, \dots, p$. The standardized design matrix $\mathbf{W}$ is defined:

$$w_{ij} = \frac{\hat{\mu}_{ij} - \bar{\mu}_j}{\hat{\sigma}_j^2}, \quad i = 1, \dots, n \quad j = 1, \dots, p$$

with $\bar{\mu}_j = \sum_{i=1}^n \hat{\mu}_{ij}/n$ and $\hat{\sigma}_j^2 = \sum_{i=1}^n (\hat{\mu}_{ij} - \bar{\mu}_j)^2/n$. Then assuming constant measurement error variance among the samples we estimate the measurement error variance

for the j -th gene as $\hat{\sigma}_{U_j}^2 = \sum_{i=1}^n \text{var}(\hat{\mu}_{ij})/n$, where $\text{var}(\hat{\mu}_{ij})$ is the estimated BGX variance of the j -th gene for the i -th individual. The final estimate $\hat{\Sigma}_U$, under the assumption of independence of the measurement error among genes, on the scale of the standardized $\mathbf{W}$ can be defined as:

$$\hat{\Sigma}_{U,jj} = \frac{\hat{\sigma}_{U_j}^2}{\hat{\sigma}_j^2}, \quad \text{for } j = 1, \dots, p$$

and zero outside the diagonal. For some cases the estimated measurement error variance $\hat{\sigma}_{U_j}^2$ is larger than the between-sample variance of the means $\hat{\sigma}_j^2$. To avoid to consider methodologies that can correct for negative estimates of the variance of $\mathbf{X}$ we decided to analyze only the 713 genes (or probe sets) with a positive estimate of $(\Sigma_X)_{jj}$. Next, we ran the three correction methods in addition to the naive lasso, using the total hip T-score BMD as our outcome.

In table 11 is reported the number of selected variables, the ℓ_2 norm of the estimated coefficients (Norm) and the Residual Sum of Squares (RSS) $n^{-1} \sum_{i=1}^n (y_i - \sum_{j=1}^p w_{ij} \beta_j)$. The naive estimator is the method that selects more variables and the one with the smaller residual variance. All the other methods considered were able to reduce the number of explanatory variables quite substantially.

The NCL and the CoCoLasso tend to agree more in variable selection: nine of the 14 variables selected from the NCL are also selected from the CoCoLasso. The MUS selects eleven variables but only six of them are selected by the NCL or the CoCoLasso. Only three variables are selected by all methods.

6. Discussion

To our knowledge, this is the first systematic simulation study that investigates the performance of methods for measurement error correction in a truly high-dimensional situation. Based on our results, none of the correction methods are globally preferable to the naive estimator. The results are varying, depending on the measurement error structure and on the size of the active set: if the measurement error has an uncorrelated structure, generally the naive method is the best choice from a variable selection perspective, but the MUS is an interesting option if it is necessary to drastically reduce the dimension of the selected set. However, it does not seem to work very well for larger active sets. When the measurement errors are correlated, the NCL can give good results, but with incorrect estimation of Σ_U the NCL performance can become even worse than the naive method. The simulation study in [20] has also shown that if the estimator for Σ_U deteriorates, the NCL and the improved MUS fall short in sign recovery; unfortunately the regular MUS was not included and just a small active set was considered in the simulations.

Datta and Zou showed by simulations [14] that the CoCoLasso may have better performances than the NCL, both for variable selection and estimation error. However, the framework in which the two models were compared ($n=100$, $p=250$, $s=3$) is smaller than the dimensions considered in our simulations and no correlation structure was considered for the measurement error, which may explain the differing results. Brown et al. [19] included the CoCoLasso in their investigation of the MEBoost, and showed that the CoCoLasso had consistently lower sensitivity but higher specificity as compared to the naive lasso. Our simulations show consistently lower specificity. However, their

simulations were based on $n = 80$ and $p = 100$.

In this paper we investigated a practical approach to analysis of a high-dimensional noisy data without knowing the true measurement error covariance Σ_U , comparing a method that do not use any information about the measurement error, the MUS, with methods that include Σ_U , the NCL and the CoCoLasso. The simulation showed that when no information about Σ_U is available, the number of observations n is big enough and the number of variables in the active set is limited, the MUS (that is the only correction method available) can achieve good results. When Σ_U is estimated under correct assumptions the NCL showed to be able to correct naive variable selection and estimation, but not when Σ_U is incorrectly estimated. The CoCoLasso showed that, in particular for larger active set, it tends to over-select both compared to the naive method and the true lasso.

The choice of the tuning parameters may be crucial for the MUS and the NCL. As shown in Figure 1, the “elbow rule” for the selection of δ might be the cause of the poor performance of the MUS when a larger active set is present since the suggested elbow point is generally far below the actual size of the active set. It would be beneficial to come up with some other criterion for the selection of δ .

Also the criterion for choosing R_{max} suggested in section 3.2 might be questionable. Although $2 * \|\hat{\beta}_{naive}\|_1$ should generally include the ideal value $\|\beta\|_1$, if the naive method does not select any variables it will lead to a 0-length cross-validation interval; furthermore when Σ_U is non-diagonal, the cross-validation procedure tends to select R_{max} as optimal point too often. We re-simulated several of these scenarios using an interval five time larger than the original one, but the results were similar and still a large portion of the cases ended up at the new R_{max} .

In our simulations, there is a striking difference between the behavior of the MUS, which is built on the Dantzig selector and the two methods that are built on the lasso. The Lasso and The Dantzig selector have been demonstrated in [33] to be equivalent as long as the sparsity condition is present, so with the increase of the dimension of the active set, the MUS becomes less comparable to the NCL and the CoCoLasso. However, it is not likely that this can explain the relatively large observed differences between the MUS and the other two correction methods. Moreover, the idea behind both the NCL and the CoCoLasso correction is to recover the behavior of the lasso we would have seen without measurement error, which we term the “true lasso”. However, this will not necessarily imply good results with regard to variable selection as the true lasso itself might suffer from over-selection, in particular when p and s are large (as shown in tables 2, 4 and 6).

The dimensions of the datasets considered in the simulations were chosen with the specific goal of testing the methods in high-dimensional situations, and are larger than in previous simulation studies. This may explain the different results, but it also indicates that our conclusions cannot be directly extended to even higher dimensions.

Funding sources

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of interest

Declaration of interest: none.

References

- [1] Tibshirani R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*. 1996;58(1):267–288.
- [2] Candès E, Tao T, et al. The dantzig selector: Statistical estimation when p is much larger than n . *The Annals of Statistics*. 2007;35(6):2313–2351.
- [3] Fan J, Li R. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*. 2001;96(456):1348–1360.
- [4] Rocke DM, Durbin B. A model for measurement error for gene expression arrays. *Journal of computational biology*. 2001;8(6):557–569.
- [5] Purdom E, Holmes SP. Error distribution for gene expression data. *Statistical applications in genetics and molecular biology*. 2005;4(1).
- [6] Bertrand A, Moonen M. Consensus-based distributed total least squares estimation in ad hoc wireless sensor networks. *IEEE Transactions on Signal Processing*. 2011;59(5):2320–2330.
- [7] Slijepcevic S, Megerian S, Potkonjak M. Location errors in wireless embedded sensor networks: sources, models, and effects on applications. *ACM SIGMOBILE Mobile Computing and Communications Review*. 2002;6(3):67–78.
- [8] Bound J, Brown C, Mathiowetz N. Measurement error in survey data. In: *Handbook of econometrics*. Vol. 5. Elsevier; 2001. p. 3705–3843.
- [9] Kipnis V, Subar AF, Midthune D, et al. Structure of dietary measurement error: results of the open biomarker study. *American journal of epidemiology*. 2003;158(1):14–21.
- [10] Carroll RJ, Ruppert D, Stefanski LA, et al. Measurement error in nonlinear models: a modern perspective. Chapter 1.8, Loss of power. New York, NY, USA: Chapman and Hall/CRC; 2006. *Monographs on Statistics and Applied Probability* 105; p. 18–22.
- [11] Rosenbaum M, Tsybakov AB. Sparse recovery under matrix uncertainty. *The Annals of Statistics*. 2010;38(5):2620–2651.
- [12] Sørensen Ø, Frigessi A, Thoresen M. Measurement error in lasso: Impact and likelihood bias correction. *Statistica Sinica*. 2015;25(2):809–829.
- [13] Loh BPL, Wainwright MJ. High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. *The Annals of Statistics*. 2012;40(3):1637–1664.
- [14] Datta A, Zou H, et al. Cocolasso for high-dimensional error-in-variables regression. *The Annals of Statistics*. 2017;45(6):2400–2426.
- [15] Rosenbaum M, Tsybakov AB, et al. Improved matrix uncertainty selector. In: *From probability to statistics and back: High-dimensional models and processes—a festschrift in honor of jon a. wellner*. Institute of Mathematical Statistics; 2013. p. 276–290.
- [16] Belloni A, Rosenbaum M, Tsybakov AB. Linear and conic programming estimators in high dimensional errors-in-variables models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 2017;79(3):939–956.
- [17] Sørensen Ø, Hellton KH, Frigessi A, et al. Covariate selection in high-dimensional generalized linear models with measurement error. *Journal of Computational and Graphical Statistics*. 2018;(to appear).
- [18] Kaul A, Koul HL, Chawla A, et al. Two stage non-penalized corrected least squares for high dimensional linear models with measurement error or missing covariates. *arXiv preprint:160503154*. 2016;.
- [19] Brown B, Weaver T, Wolfson J. Meboost: Variable selection in the presence of measurement error. *arXiv preprint:170102349*. 2017;.
- [20] Chen Y, Caramanis C. Noisy and missing data regression: Distribution-oblivious support recovery. In: *International Conference on Machine Learning*; 2013. p. 383–391.
- [21] Hastie T, Tibshirani R, Friedman J. The elements of statistical learning. Chapter 3.8.3, The Dantzig Selector. New York, NY, USA: Springer New York; 2009. *Springer Series in Statistics*; p. 89–90.
- [22] Boyd S, Parikh N, Chu E, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine*

- Learning. 2011;3(1):1–122.
- [23] Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*. 2010;33(1):1–22.
 - [24] Efron B, Hastie T, Johnstone I, et al. Least angle regression. *The Annals of statistics*. 2004;32(2):407–499.
 - [25] Zhao P, Yu B. On model selection consistency of lasso. *Journal of Machine learning research*. 2006;7(Nov):2541–2563.
 - [26] Zou H. The adaptive lasso and its oracle properties. *Journal of the American statistical association*. 2006;101(476):1418–1429.
 - [27] Reppe S, Refvem H, Gautvik VT, et al. Eight genes are highly associated with bmd variation in postmenopausal caucasian women. *Bone*. 2010;46(3):604–612.
 - [28] Boulesteix AL, Strobl C, Augustin T, et al. Evaluating microarray-based classifiers: an overview. *Cancer informatics*. 2008;6:77–97.
 - [29] Tadesse MG, Ibrahim JG, Gentleman R, et al. Bayesian error-in-variable survival model for the analysis of genechip arrays. *Biometrics*. 2005;61(2):488–497.
 - [30] Hein AMK, Richardson S, Causton HC, et al. Bgx: a fully bayesian integrated approach to the analysis of affymetrix genechip data. *Biostatistics*. 2005;6(3):349–373.
 - [31] Irizarry RA, Hobbs B, Collin F, et al. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*. 2003;4(2):249–264.
 - [32] Fan J, Lv J. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 2008;70(5):849–911.
 - [33] Bickel PJ, Ritov Y, Tsybakov AB, et al. Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*. 2009;37(4):1705–1732.

Table 1. Simulations for $s = 5$ and uncorrelated measurement error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	34,00 (26,25)	5,00 (0,00)	2,17 (0,99)	33,00 (20,25)	5,00 (0,00)	0,88 (0,31)
Naive lasso	26,00 (27,75)	5,00 (1,00)	3,02 (0,98)	37,50 (30,00)	5,00 (0,00)	1,66 (0,44)
MU(δ_1)	11,00 (6,50)	4,00 (2,00)	3,59 (0,95)	10,00 (5,00)	5,00 (0,00)	2,26 (0,40)
MU(δ_2)	8,00 (4,25)	4,00 (1,00)	3,84 (0,82)	6,00 (1,00)	5,00 (0,00)	2,78 (0,40)
NCL	24,00 (17,50)	5,00 (1,00)	2,77 (1,03)	38,00 (24,50)	5,00 (0,00)	1,13 (0,50)
NCL($\hat{\Sigma}_U$)	25,50 (18,25)	5,00 (1,00)	2,80 (1,30)	44,50 (27,75)	5,00 (0,00)	1,17 (0,42)
NCL($\hat{\Sigma}_U^{diag}$)	25,00 (17,00)	5,00 (1,00)	2,77 (1,01)	41,00 (24,50)	5,00 (0,00)	1,13 (0,50)
CoCoLasso	37,50 (27,00)	5,00 (1,00)	2,93 (1,07)	58,00 (23,50)	5,00 (0,00)	1,51 (0,46)
CoCoLasso($\hat{\Sigma}_U$)	38,00 (22,25)	5,00 (1,00)	2,87 (0,94)	59,00 (25,00)	5,00 (0,00)	1,52 (0,43)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	37,00 (22,25)	5,00 (1,00)	2,86 (0,92)	57,00 (29,00)	5,00 (0,00)	1,52 (0,44)

Table 2. Simulations for $s = 20$ and uncorrelated measurement error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	96,00 (29,00)	19,00 (2,00)	9,09 (2,82)	117,00 (37,75)	20,00 (0,00)	3,12 (0,72)
Naive lasso	74,50 (42,25)	16,00 (3,00)	13,86 (3,05)	127,50 (52,25)	20,00 (0,00)	7,15 (1,41)
MU(δ_1)	27,00 (11,00)	11,00 (4,00)	16,06 (1,50)	23,00 (4,00)	17,00 (2,00)	13,28 (0,84)
MU(δ_2)	20,00 (8,25)	9,00 (4,00)	16,75 (1,23)	19,00 (4,00)	15,00 (3,00)	14,65 (0,73)
NCL	45,50 (13,00)	13,00 (3,25)	14,37 (3,16)	104,00 (25,25)	20,00 (0,00)	4,69 (1,79)
NCL($\hat{\Sigma}_U$)	43,00 (10,25)	12,00 (3,00)	14,67 (3,67)	97,50 (20,25)	20,00 (1,00)	5,34 (2,14)
NCL($\hat{\Sigma}_U^{diag}$)	46,00 (13,50)	13,00 (4,00)	13,97 (3,44)	103,00 (25,00)	20,00 (0,00)	4,70 (1,63)
CoCoLasso	77,50 (35,25)	15,00 (3,00)	13,69 (3,06)	149,00 (47,00)	20,00 (0,00)	6,58 (1,35)
CoCoLasso($\hat{\Sigma}_U$)	73,00 (35,25)	15,00 (3,00)	14,00 (3,11)	154,00 (42,75)	20,00 (0,00)	6,79 (1,38)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	71,50 (35,25)	15,00 (3,25)	14,08 (2,96)	154,00 (39,25)	20,00 (0,00)	6,57 (1,33)

Table 3. Simulations for $s = 5$ and measurement error correlated within the active set. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	34,00 (26,25)	5,00 (0,00)	2,17 (0,99)	33,00 (20,25)	5,00 (0,00)	0,88 (0,31)
Naive lasso	22,50 (28,50)	4,00 (2,00)	3,49 (0,96)	33,00 (26,50)	5,00 (0,00)	2,20 (0,51)
MU(δ_1)	12,00 (11,00)	4,00 (1,00)	3,72 (0,91)	11,00 (5,00)	5,00 (0,00)	2,63 (0,45)
MU(δ_2)	10,00 (8,00)	3,50 (1,25)	3,85 (0,88)	8,00 (3,25)	5,00 (0,00)	2,85 (0,42)
NCL	26,00 (21,00)	5,00 (1,00)	2,50 (1,37)	39,00 (25,25)	5,00 (0,00)	1,11 (0,46)
NCL($\hat{\Sigma}_U$)	26,00 (24,00)	5,00 (1,00)	2,68 (1,24)	41,50 (30,25)	5,00 (0,00)	1,09 (0,43)
NCL($\hat{\Sigma}_U^{diag}$)	19,00 (19,00)	4,00 (2,00)	3,47 (1,14)	32,50 (26,25)	5,00 (0,00)	1,86 (0,57)
CoCoLasso	32,00 (27,50)	5,00 (1,00)	3,21 (1,01)	56,50 (26,25)	5,00 (0,00)	1,80 (0,54)
CoCoLasso($\hat{\Sigma}_U$)	35,00 (22,00)	5,00 (1,00)	3,16 (0,90)	57,00 (29,50)	5,00 (0,00)	1,87 (0,50)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	30,50 (21,25)	4,00 (1,00)	3,34 (0,79)	54,00 (28,00)	5,00 (0,00)	2,07 (0,52)

Table 4. Simulations for $s = 20$ and measurement error correlated within the active set. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	96,00 (29,00)	19,00 (2,00)	9,09 (2,82)	117,00 (37,75)	20,00 (0,00)	3,12 (0,72)
Naive lasso	46,50 (42,25)	10,00 (3,25)	17,28 (2,22)	89,50 (48,00)	16,00 (3,00)	14,48 (1,11)
MU(δ_1)	36,00 (30,25)	9,00 (4,00)	17,17 (1,62)	34,00 (12,25)	14,00 (3,00)	15,19 (1,11)
MU(δ_2)	19,50 (14,00)	7,00 (4,00)	17,67 (1,36)	17,00 (5,25)	11,00 (2,00)	16,49 (0,80)
NCL	41,00 (20,00)	17,00 (3,00)	9,98 (4,04)	77,50 (27,25)	20,00 (0,00)	4,80 (1,84)
NCL($\hat{\Sigma}_U$)	40,00 (16,50)	16,00 (4,00)	11,51 (4,82)	76,00 (23,25)	20,00 (1,00)	5,66 (2,44)
NCL($\hat{\Sigma}_U^{diag}$)	30,00 (20,00)	8,00 (3,00)	17,56 (2,56)	69,00 (24,75)	14,00 (3,00)	14,61 (1,49)
CoCoLasso	60,00 (53,25)	13,00 (5,00)	15,33 (2,98)	121,00 (42,25)	19,00 (2,00)	11,19 (1,82)
CoCoLasso($\hat{\Sigma}_U$)	56,00 (46,25)	13,00 (5,25)	15,75 (3,35)	128,00 (41,00)	18,50 (1,00)	11,64 (1,75)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	47,00 (35,00)	10,00 (3,00)	17,21 (2,19)	108,00 (33,75)	16,00 (2,25)	14,33 (1,30)

Table 5. Simulations for $s = 5$ and measurement error correlated beyond the active set. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	32,00 (22,25)	5,00 (0,00)	2,10 (1,19)	34,50 (28,25)	5,00 (0,00)	0,85 (0,33)
Naive lasso	23,00 (20,25)	4,00 (1,00)	3,53 (1,11)	28,50 (24,75)	5,00 (0,00)	2,14 (0,52)
MU(δ_1)	12,00 (8,25)	4,00 (1,00)	3,63 (1,12)	11,00 (5,00)	5,00 (0,00)	2,65 (0,46)
MU(δ_2)	8,00 (6,00)	3,00 (1,00)	3,81 (1,01)	6,00 (1,00)	5,00 (0,00)	3,03 (0,46)
NCL	26,00 (20,25)	5,00 (1,00)	2,62 (1,52)	38,50 (24,50)	5,00 (0,00)	1,00 (0,54)
NCL($\hat{\Sigma}_U$)	26,00 (21,25)	5,00 (1,00)	2,70 (1,46)	41,00 (27,25)	5,00 (0,00)	1,05 (0,46)
NCL($\hat{\Sigma}_U^{diag}$)	19,50 (16,00)	4,00 (2,00)	3,40 (1,29)	32,00 (21,25)	5,00 (0,00)	1,72 (0,62)
CoCoLasso	33,50 (20,25)	4,00 (1,00)	3,25 (1,04)	56,00 (26,25)	5,00 (0,00)	1,72 (0,54)
CoCoLasso($\hat{\Sigma}_U$)	34,00 (25,25)	4,00 (1,00)	3,26 (1,06)	53,00 (29,75)	5,00 (0,00)	1,80 (0,55)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	31,50 (25,00)	4,00 (1,00)	3,34 (1,06)	51,00 (30,25)	5,00 (0,00)	1,95 (0,52)

Table 6. Simulations for $s = 20$ and measurement error correlated beyond the active set. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	93,50 (30,00)	18,00 (2,00)	10,08 (3,11)	117,00 (39,25)	20,00 (0,00)	2,99 (0,78)
Naive lasso	45,00 (48,25)	9,00 (3,00)	17,84 (1,97)	79,50 (50,50)	16,00 (2,00)	14,63 (1,54)
MU(δ_1)	20,50 (17,00)	6,00 (4,00)	18,15 (1,45)	22,00 (7,00)	13,00 (3,00)	16,21 (1,06)
MU(δ_2)	14,00 (9,00)	6,00 (3,00)	18,50 (1,28)	14,00 (3,00)	11,00 (3,00)	17,12 (0,81)
NCL	44,00 (21,25)	16,00 (6,00)	11,29 (5,33)	66,50 (29,50)	20,00 (0,00)	5,12 (2,50)
NCL($\hat{\Sigma}_U$)	41,00 (19,25)	14,00 (6,00)	12,97 (5,16)	69,00 (30,00)	20,00 (1,00)	5,83 (2,39)
NCL($\hat{\Sigma}_U^{diag}$)	30,50 (24,50)	7,00 (3,00)	18,14 (2,00)	67,50 (38,25)	14,00 (3,00)	14,88 (1,55)
CoCoLasso	63,00 (58,00)	12,00 (6,00)	16,38 (2,22)	125,00 (55,50)	19,00 (1,25)	11,32 (2,11)
CoCoLasso($\hat{\Sigma}_U$)	63,00 (40,00)	12,00 (4,00)	16,74 (2,18)	121,50 (45,25)	18,00 (2,00)	11,90 (1,71)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	46,00 (39,25)	9,00 (4,25)	17,87 (1,95)	99,50 (41,00)	16,00 (2,00)	14,53 (1,44)

Table 7. Simulations for $s = 5$, uncorrelated measurement error and small model error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	5,00 (3,25)	5,00 (0,00)	0,01 (0,01)	15,00 (14,50)	5,00 (0,00)	0,00 (0,00)
Naive lasso	39,00 (36,25)	5,00 (0,00)	0,75 (0,17)	48,50 (35,25)	5,00 (0,00)	0,57 (0,10)
MU(δ_1)	10,00 (4,00)	5,00 (0,00)	1,35 (0,27)	9,00 (4,00)	5,00 (0,00)	1,24 (0,16)
MU(δ_2)	8,00 (3,00)	5,00 (0,00)	1,70 (0,28)	6,00 (0,00)	5,00 (0,00)	1,88 (0,17)
NCL	39,50 (13,00)	5,00 (0,00)	0,19 (0,08)	54,00 (34,50)	5,00 (0,00)	0,10 (0,04)
NCL($\hat{\Sigma}_U$)	45,00 (7,25)	5,00 (0,00)	0,33 (0,23)	96,50 (19,00)	5,00 (0,00)	0,19 (0,11)
NCL($\hat{\Sigma}_U^{diag}$)	39,00 (11,25)	5,00 (0,00)	0,19 (0,09)	53,50 (32,75)	5,00 (0,00)	0,10 (0,04)
CoCoLasso	48,00 (24,00)	5,00 (0,00)	0,56 (0,16)	68,00 (32,25)	5,00 (0,00)	0,41 (0,09)
CoCoLasso($\hat{\Sigma}_U$)	50,50 (21,50)	5,00 (0,00)	0,59 (0,17)	68,50 (38,00)	5,00 (0,00)	0,45 (0,11)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	49,50 (22,25)	5,00 (0,00)	0,56 (0,16)	68,50 (27,50)	5,00 (0,00)	0,40 (0,09)

Table 8. Simulations for $s = 20$, uncorrelated measurement error and small model error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	27,00 (5,00)	20,00 (0,00)	0,07 (0,03)	20,00 (1,00)	20,00 (0,00)	0,03 (0,01)
Naive lasso	108,50 (31,50)	20,00 (0,00)	6,99 (2,25)	134,00 (44,25)	20,00 (0,00)	3,60 (0,54)
MU(δ_1)	30,00 (6,00)	15,00 (3,00)	13,71 (1,72)	22,00 (2,00)	19,00 (1,00)	11,68 (0,77)
MU(δ_2)	18,00 (4,00)	12,00 (2,00)	15,58 (1,27)	18,00 (2,00)	17,00 (1,00)	14,59 (0,50)
NCL	51,00 (7,00)	20,00 (1,00)	3,59 (2,28)	101,00 (16,50)	20,00 (0,00)	1,07 (0,28)
NCL($\hat{\Sigma}_U$)	43,00 (7,00)	19,00 (2,25)	6,29 (3,34)	103,00 (15,00)	20,00 (0,00)	1,75 (0,39)
NCL($\hat{\Sigma}_U^{diag}$)	50,00 (6,25)	20,00 (1,00)	3,86 (2,11)	101,00 (20,25)	20,00 (0,00)	1,10 (0,26)
CoCoLasso	115,00 (26,25)	20,00 (0,00)	6,58 (2,13)	158,50 (39,25)	20,00 (0,00)	2,93 (0,50)
CoCoLasso($\hat{\Sigma}_U$)	103,50 (17,25)	20,00 (0,00)	7,00 (2,43)	159,50 (35,25)	20,00 (0,00)	3,11 (0,62)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	114,00 (24,50)	20,00 (0,00)	6,68 (1,94)	158,00 (37,50)	20,00 (0,00)	2,89 (0,57)

Table 9. Simulations for $s = 20$, measurement error correlated within the active set and small model error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	27,00 (5,00)	20,00 (0,00)	0,07 (0,03)	20,00 (1,00)	20,00 (0,00)	0,03 (0,01)
Naive lasso	70,00 (42,50)	13,00 (3,00)	15,73 (1,96)	120,00 (46,50)	19,00 (1,25)	12,86 (1,11)
MU(δ_1)	52,50 (27,00)	12,00 (3,00)	15,66 (1,77)	37,00 (10,25)	18,00 (1,25)	13,67 (0,82)
MU(δ_2)	25,00 (11,25)	10,00 (3,00)	16,56 (1,38)	19,00 (3,00)	15,00 (2,00)	15,57 (0,66)
NCL	42,50 (10,25)	19,00 (2,00)	6,44 (2,26)	72,00 (13,00)	20,00 (0,00)	2,94 (0,94)
NCL($\hat{\Sigma}_U$)	41,50 (12,25)	18,00 (2,00)	8,26 (4,09)	81,00 (17,50)	20,00 (0,00)	3,79 (1,27)
NCL($\hat{\Sigma}_U^{diag}$)	41,00 (17,00)	10,00 (3,25)	15,95 (1,90)	86,00 (25,25)	16,00 (1,00)	12,87 (1,36)
CoCoLasso	100,50 (40,75)	18,00 (3,00)	11,40 (3,21)	140,50 (40,50)	20,00 (0,00)	8,43 (1,17)
CoCoLasso($\hat{\Sigma}_U$)	92,50 (43,25)	17,00 (3,00)	12,63 (3,02)	152,50 (39,50)	20,00 (0,00)	9,23 (1,33)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	67,50 (32,25)	13,00 (3,00)	15,53 (1,80)	139,50 (44,50)	19,00 (1,00)	12,73 (1,21)

Table 10. Simulations for $s = 20$, measurement error correlated beyond the active set and small model error. The table reports the number of variables selected (S), the number of correct variables selected (C) and the estimation error as ℓ_2 norm. All results are reported as median values and (interquartile range).

models	n=200, p=1000			n=500, p=2000		
	S	C	ℓ_2	S	C	ℓ_2
True lasso	28,00 (6,00)	20,00 (0,00)	0,07 (0,03)	20,00 (1,00)	20,00 (0,00)	0,03 (0,01)
Naive lasso	64,00 (51,50)	13,00 (3,00)	15,99 (1,71)	109,00 (46,25)	19,00 (1,00)	13,00 (1,10)
MU(δ_1)	24,50 (13,00)	10,00 (4,00)	16,86 (1,39)	23,00 (5,00)	16,00 (2,00)	14,94 (0,93)
MU(δ_2)	16,00 (6,25)	8,00 (4,00)	17,48 (1,22)	16,00 (4,00)	13,00 (3,00)	16,29 (0,62)
NCL	44,00 (12,00)	19,00 (2,00)	6,94 (2,65)	69,50 (13,00)	20,00 (0,00)	3,01 (0,84)
NCL($\hat{\Sigma}_U$)	42,00 (13,25)	18,00 (2,00)	8,86 (2,85)	78,00 (18,50)	20,00 (0,00)	3,76 (1,07)
NCL($\hat{\Sigma}_U^{diag}$)	40,00 (18,00)	10,00 (3,00)	16,30 (1,74)	79,00 (23,25)	16,00 (3,00)	13,07 (1,39)
CoCoLasso	98,50 (59,00)	18,00 (3,00)	12,37 (3,09)	138,00 (45,25)	20,00 (0,00)	8,72 (1,27)
CoCoLasso($\hat{\Sigma}_U$)	94,00 (55,50)	17,00 (4,00)	13,33 (2,53)	145,00 (48,25)	20,00 (0,00)	9,53 (1,24)
CoCoLasso($\hat{\Sigma}_U^{diag}$)	66,50 (27,75)	13,00 (3,00)	16,10 (1,77)	127,00 (33,25)	19,00 (1,00)	12,82 (1,16)

Table 11. The table shows the results of the analysis of the microarray data in the example.

	selected	Norm	RSS
Naive	46,00	0,87	0,29
MUS	11,00	0,56	1,41
NCL	14,00	0,60	1,10
Coco	30,00	0,75	0,57