An ensemble learning method for variable selection: application to high dimensional data and missing values

Avner Bar-Hen, Vincent Audigier

An ensemble learning method for variable selection: application to high dimensional data and missing values

Avner Bar-Hen*, Vincent Audigier†
CNAM, Laboratoire Cedric-MSDMA, 2 rue Conté, 75003 Paris, France

March 10, 2020

Abstract Standard approaches for variable selection in linear models are not tailored to deal properly with high dimensional and incomplete data. Currently, methods dedicated to high dimensional data handle missing values by ad-hoc strategies, like complete case analysis or single imputation, while methods dedicated to missing values, mainly based on multiple imputation, do not discuss the imputation method to use with high dimensional data. Consequently, both approaches appear to be limited for many modern applications.

With inspiration from ensemble methods, a new variable selection method is proposed. It extends classical variable selection methods such as stepwise, lasso or knockoff in the case of high dimensional data with or without missing data. Theoretical properties are studied and the practical interest is demonstrated through a simulation study.

In the low dimensional case, the procedure improves the control of the error risks, especially type I error, even without missing values. With missing values, the method performs better than reference selection methods based on multiple imputation. Similar performances are obtained in the high-dimensional case with or without missing values.

Keywords: Ensemble method; High Dimensional Data; Linear Regression;

*avner@cnam.fr
†vincent.audigier@cnam.fr

Missing Data; Variable Selection.

1 Introduction

Large scale data is challenging for data visualisation, data understanding, large measurement and storage requirements, training and utilisation times, or prediction. Variable selection, such as stepwise e.g., is one of the most common strategies to tackle the issue. Many procedures of variable selection are still proposed in the modern literature such as Lasso [1], Bolasso [2], knockoff [3, 4] among others (see for example [5] for a review).

In this article we focus on a classical linear model framework in which a Gaussian response Y is related to variables among a set of explanatory Gaussian variables X_j ($j = 1, \ldots, p$). In this context, variable selection consists in identifying explanatory variables which are significantly related to Y.

Issues commonly encountered in variable selection gather stability of the selected subset of variables, high dimensionality, or missing data for instance. Many methods have been developed to overcome each of them.

Ensemble learning methods provide a way to improve stability [6, 7]. Such methods consist in perturbing the data several times, applying the selection procedure on the perturbed data, and then, aggregating over all obtained subsets. For example, ensemble methods have been suggested for variable selection by random forests [8] or lasso [7]. As regards the high dimensionality, it can be tackled by techniques like shrinkage methods (e.g. ridge regression or lasso [1]), or by using preliminary screening steps [9, 4]. As regards the missing data issue, multiple imputation [10, 11, 12] appears the most intensively investigated. In particular, many methods have been proposed to pool several subsets of variables obtained from each imputed data set, independently to the way used to fill-in the data [13].

However, in practice, we potentially face to all challenges simultaneously, making difficult to perform variable selection in a suitable way. In this paper, we propose an original variable selection method based on an ensemble learning method allowing variable selection in various cases, notably for high dimensional data or missing data, while improving stability of the selection. To achieve this goal, the main idea is to perform variable selection on random subsets of variables and, then, to combine them to recover which variables X_j are related to the response Y. Note that ensemble learning methods for variable selection generally resample the individuals, but here, only variables are resampled. Performing variable selection on several subsets of variables solve the high-dimensional issue

and allows treatment of missing values by classical techniques. More precisely, the outline of the algorithm are as follows: let consider a random subset of size k among p variables. By choosing k small, this subset is low dimensional, allowing treatment of missing values by standard imputation method. Then, any selection variable scheme can be applied. We will focus on standard variable selection methods, such as stepwise, lasso, but also on a more recent method, named knockoff [4], which has the specific property to be consistent. By resampling B times, a sample of size k among the p variables, we may count how many times, a variable is considered as significantly related to the response variable Y and how many times it is not. We need to define a threshold to conclude if a given variable is significantly related to the response Y.

In the next section, we fully describe the proposed algorithm. Rules to tune its parameters are given and mathematically justified. We also derive some theoretical properties of the algorithm. In Section 3, we illustrate the relevance of the selection of variable method through a simulation study. Finally, a discussion about extensions closes the paper.

2 Algorithm

2.1 Notation and context

Let consider a classical linear regression model

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

where $X = (X_1, \ldots, X_p)$ denotes a set of p explanatory variables, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ denotes the vector of regression coefficients, ε is a Gaussian noise with variance σ^2 and null expectation, Y is the response variable. n independent realisations of (Y, X) are observed, leading to a data set with n rows and $p + 1$ columns.

We assume that missing values occur on covariates only, without loss of generalities [12].We note $R = (R_1, \ldots, R_p)$ the missing data mechanism so that $R_j = 1$ indicates variable X_j is missing, and $R_j = 0$ indicates variable is observed. The n realisations of R are assumed to be independent. We do not put any restrictions on the missing data mechanism, and any restrictions on the number of missing values in order to cover a large range of situations.

We intended to select the "best" subset of predictors, i.e. the subset of non-null coefficients of $\boldsymbol{\beta}$. The central premise is that the data contains many features

that are either redundant or irrelevant, and can thus be removed without incurring much loss of information. Successful procedures are characterized by high predictive accuracy, yielding interpretable models while retaining computational efficiency. Penalized methods that perform coefficient shrinkage (such as lasso) have been shown to be successful in many cases. Models with correlated predictors are particularly challenging to tackle and missing data are difficult to handle [14, 15]. Some alternative such as knockoff also provide statistical guarantees [3, 4] but have not been adapted to handle missing data. Stepwise regression is also very popular process of building a model by successively adding or removing variables based solely on the statistics such as AIC criterion or $t-$test of their estimated coefficients. Unfortunately, the model is fit using unconstrained least squares, therefore nothing can be said about the mathematical properties of the results. Furthermore, stepwise cannot be directly applied on high dimensional data or data with missing values.

2.2 The method

As for ensemble methods, our algorithm has two steps: one which creates many regression instances and one which aggregates instances into an overall regression. More precisely, each regression instance allows to test if the relationships between (part of) explanatory variables and the response variable is significant or not. Then, we aggregate tests of the instances to obtain a global test for each variable.

To create regression instance, we sample k variables among the p variables. Next, a variable selection procedure is applied on the k variables. If the method does not handle high-dimensional data, k is chosen less than n, so that the high dimensional issue is tackled. If the dataset has missing values, two cases can be considered: the first one is the number of individuals with missing is very small. For such a case, complete-case analysis can be a sufficient strategy to solve the missing data issue. Otherwise, single stochastic imputation by the multivariate Gaussian model can be performed. Note that because we do not aim to build confidence intervals for regression coefficients, multiple imputation is not required here. Imputation methods need accounting for the nature of the missing data mechanism [11, 10, 12]. We will consider a classical method dealing with missing at random (MAR) mechanisms [10], but methods dedicated to missing not at random (MNAR) mechanisms could also be used [16, e.g.].

Thus, any variable selection procedure can be applied, leading to the regression instances among the k variables that are significantly related to Y (according

to a given threshold). We iterate the process B times, leading to B regression instances.

As a second step of the algorithm, the regression instances are aggregated. For each variable X_j, we count the ratio r_j between (i) the number of times the variable X_j is selected as significantly related to the response variable Y and (ii) the number of times the variable is present in the subsets. We conclude that a variable X_j is significantly related to Y if r_j is greater that a threshold r.

Sampling a subset of k variables implies that each variable is chosen a random number of times. Therefore, a direct sampling of the variables implies that variance of X_j is not constant among the variables. This behaviour is irrelevant since it required more iterations to bound the variance. Note that the sampling process of a given variable can be viewed as a Bernoulli distribution and a bound on the minimum of Bernoulli can be easily obtained through Chernoff inequality. An alternative to this sampling scheme is random partitioning of the variables. If p is a multiple of k, we have p/k subsets of variables by partition. Selection variable techniques is applied to each of the p/k subsets so that X_j is observed with the same proportion k/p over all subsets. We iterate the process by choosing random partitions.

Three questions arise: (i) how to choose k, (ii) how to choose B and finally (iii) how to choose the threshold r.

2.2.1 How many iterations?

To improve the stability of the procedure, the proportion of times that a variable is considered as significant (r_j) needs to be calculated from many iterations (B).

B has to be chosen, so that $\mathbb{V}(r_j)$ is small. If the number of times the variable X_j is significant follows a Binomial distribution, then for $\tilde{B}$ regression instances gathering X_j ($\tilde{B} = B \times k/p$), we have $\mathbb{V}(r_j)$ is less than $\frac{1}{4\tilde{B}}$. Thus, $\tilde{B} = 100$ can be chosen to obtain a standard error less than 5%. For k fixed, it provides a guideline to tune B.

We can note that B is related to p, meaning that the number of iterations of the algorithm needs to be chosen according to the number of variables.

2.2.2 Which value for the threshold r?

r_j can be seen as a variable importance measure or more precisely as an estimate of α, the risk of the test between the null hypothesis H_0 : "$\beta_j = 0$" versus the alternative H_1 : "$\beta_j \neq 0$" over all the B iterations. Following Neyman-Pearson

lemma, r_j needs to be chosen over than a threshold r (e.g. 95%) to control the α risk.

To tune the threshold r whatever the data set, theoretical properties of the selection method are needed. For instance, the false discovery rate is controlled by knockoff at each iteration. Thus, it could be preserved by choosing r accordingly. However, for many variables selection methods no such guaranties are available. For them, r can be chosen a posteriori by empirical methods like cross-validation. More precisely, the algorithm is applied on the train set, given a variable importance for each variable. Then, a grid of thresholds is fixed and a sequence of nested linear regression models can be derived. For each one, an error of prediction can be calculated. Note that in a context of regression, optimising r in terms of prediction error is equivalent to optimisation in terms of regression coefficient estimate, making cross-validation consistent with identification of non-null regression coefficients.

Note that high-dimensional data and missing values are tricky for cross-validation, since the linear models cannot be directly fit in both cases. The high-dimensional issue can be tackled by a screening step using the variable importance measure, while the missing data issue can be handled by imputing the test set and train set simultaneously, excluding the outcome on the test set as proposed in [17].

2.3 Some mathematical properties

2.3.1 Aggregation of regression coefficients

Even if our goal is only to identify the subset of variables related to the response Y, we investigate the performances of the aggregation of the regression coefficients estimates obtained by averaging of the B instances.

At first, let consider the sampling of variables for a regression instance and assume for the moment the absence of missing data. For a regression instance, let's define δ_j such that $\delta_j = 1$ if β_j is drawn and zero otherwise. Putting them in a diagonal matrix $\Delta = diag(\delta_1, \ldots, \delta_p)$, the regression model based on a sample of k variables can be rewritten as:

$$\begin{aligned} Y &= X\beta + \varepsilon \\ &= X\Delta\beta + X(I - \Delta)\beta + \varepsilon \\ &= X\Delta\beta + \varepsilon' \end{aligned}$$

$X\Delta$ corresponds to the design matrix constructed on selected variables and $\varepsilon' \sim \mathcal{N}(X(I-\Delta)\beta, \sigma^2 I)$. Since Δ is a projection matrix, then $\Delta^2 = \Delta$ and $(X\Delta)(\Delta\beta) = X\Delta\beta$.

We assume that X is inversible and, by convention, $0/0 = 0$. Therefore

$$\Delta\hat{\beta} = (\Delta X'X\Delta)^{-1}\Delta X'Y$$

and

$$\begin{aligned}
\mathbb{E}(\Delta\hat{\beta}) &= (\Delta X'X\Delta)^{-1}\Delta X'\mathbb{E}(Y) \\
&= (\Delta X'X\Delta)^{-1}\Delta X'(X\Delta\beta + X(I-\Delta)\beta) \\
&= \Delta\beta + (\Delta X'X\Delta)^{-1}\Delta X'X(I-\Delta)\beta \qquad (2) \\
\mathbb{V}(\Delta\hat{\beta}) &= (\Delta X'X\Delta)^{-1}\sigma^2 \qquad (3)
\end{aligned}$$

We see from Equation (2) that the bias of $\hat{\beta}$ is induced by the correlation between the subset of variables in the regression instance and the other variables that are not selected in the regression instance. Thus, aggregation of regression estimates by averaging is relevant if and only if the design is orthogonal and very tricky otherwise.

2.3.2 Relevance to use k variables instead of p

The practical usefulness to perform selection from a subset of k variables instead of p have been already explained. We now highlight how does this strategy influence the performances of a selection procedure.

Without loss of generality, consider that X gathers significant explanatory variables only (i.e. $\beta_j \neq 0$ for all $1 \leq j \leq p$). Then, by independence between ε and X

$$\begin{aligned}
\mathbb{V}(Y|X\Delta) &= \mathbb{V}(X(I-\Delta)\beta) + \mathbb{V}(\varepsilon) \qquad (4) \\
&= \mathbb{V}(X(I-\Delta)\beta) + \mathbb{V}(Y|X). \qquad (5)
\end{aligned}$$

The more higher the proportion of significant variables not present in the regression instances, the more $\mathbb{V}(X(I-\Delta)\beta)$ can be large (and $\mathbb{V}(Y|X\Delta)$ *a fortiori*). This implies that the regression scheme will be noised if relevant significant variables are missed. This situation arises when the variables are sampled through the algorithm, but identifying significant variables is more challenging on noisy

data. Thus, to limit this loss of power, it seems more relevant to consider a large value for the number of selected instances (k).

Previous results imply relationships between k and r. More precisely if k is small, error of the model often contains significant variables and the ratio signal/error is lower. Sensibility of r to k will be discussed in the simulations.

3 Simulations

3.1 Simulation design

To study the quality of the procedure we simulate various cases varying the number of variables (p), the correlation between covariates (ρ), the signal to noise ratio (snr), the nature of the missing data mechanism. For each configuration, $T = 100$ data sets are generated, and for each one, variable selection is performed according to the proposed algorithm and methods presented below (Section 3.1.3).

3.1.1 Data generation

For a given configuration, data sets are generated as follows. First, n observations for p covariates are generated according to a multivariate normal distribution with null expectation, and variance $\begin{pmatrix} 1 & \rho & \dots & \dots & \rho \\ \rho & 1 & \rho & \dots & \rho \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \rho & \dots & \rho & 1 & \rho \\ \rho & \dots & \rho & \dots & 1 \end{pmatrix}$ where $-1 \leq \rho \leq 1$. Then, the response Y is simulated according to a linear model $Y = X\boldsymbol{\beta} + \varepsilon$ where ε is Gaussian with null expectation and variance $\frac{1}{snr+1}$, $\boldsymbol{\beta}$ is the sparse vector of regression coefficients composed of zeros and a fixed value β only. This value is chosen so that Var $(Y) = 1$.

For all configurations, we keep:

- the number of individuals $n = 200$
- and the number of non-zero values in $\boldsymbol{\beta}$ is 8.

Before introducing missing values, configurations vary only by the values of p, ρ and snr:

- we consider $p = 100$ or $p = 300$ variables. Let note that for the second case, the number of variables is higher that the number of observations

- we test two cases for the correlation $\rho = 0$ and $\rho = 0.4$. High correlation among explanatory variables often generates spurious results for variable selection
- finally, we test $snr = 2$ and $snr = 4$, by tuning β and the variance of the noise under the constraint that the variance of Y is equal to one. Each case corresponding to high or low difficulty to select relevant variables.

3.1.2 Missing data mechanisms

Next, missing values are added on covariates of each data set according to several mechanisms. We consider a missing completely at random (MCAR) mechanism, so that $\mathbb{P}(R_j = 1) = a$ for all j $(1 \leq j \leq p)$ and a MAR mechanism, so that $\mathbb{P}(R_j = 1|Y) = \Phi(a + Y)$ with Φ the cumulative distribution function of the standard normal distribution. The coefficient a of those models is tuned to get (in expectation) 20% of missing values. The MCAR mechanism is a particular case of MAR mechanism, which is generally simpler to handle.

3.1.3 Methods

Parameters of the algorithm are tuned as follows: at each iteration $k = 6$ variables are drawn when $p = 100$, while $k = 10$ variables are drawn when $p = 300$; $B = 6000$ iterations are performed; variables that are selected at least $r = 95\%$ of the time are kept. Sensitivity to the parameters k, B and r is assessed in Section 3.3.

The investigated variable selection procedures are the knockoff, the lasso and the stepwise (with AIC). In any configuration, these methods can be used through the proposed algorithm, but not directly on the full data set because they have some lacks with high-dimensional data and or missing values. Thus, we make comparisons as follows: we first generate the data sets (without missing data) and apply knockoff, lasso as well as stepwise variable selection procedure. Two versions of the knockoff are available: the fixed-X knockoff and the model-X knockoff. According to recommendations [18], we use fixed-X knockoff for low dimensional data and model-X knockoff for high-dimensional data. Note that in the proposed algorithm, only fixed-X knockoff is used. High-dimensional setting is tackled by a screening step in stepwise.

Then, we generate the missing values according to a pre-defined missing data mechanism. If possible, knockoff, lasso and stepwise variable selection are applied using complete case analysis. Note that handling missing values by imputation would be challenging here because of the large number of variables compared to the number of individuals [19]. The proposed algorithm is also applied by using knockoff, lasso and stepwise variable selection where missing values are handled by single stochastic imputation according to the Gaussian model. In addition, we make comparison with a recent method combining multiple imputation and random lasso variable selection [20] named MIRL. This method consists in performing multiple imputation by chained equations to fill the data, then applying random lasso on imputed data sets and combining selected subsets of variables. Since multiple imputation by chained equations is too much time consuming for large data sets, we cannot apply it for high-dimensional data.

All computations were performed using R [21]. Lasso was computed using the library *glmnet*, knockoff using the library *knockoff* and stepwise using the library *stats*. The R code used for MIRL has been obtained from authors. Single stochastic imputation by the Gaussian model has been performed with the library *norm*. The R code used for simulations is available on demand.

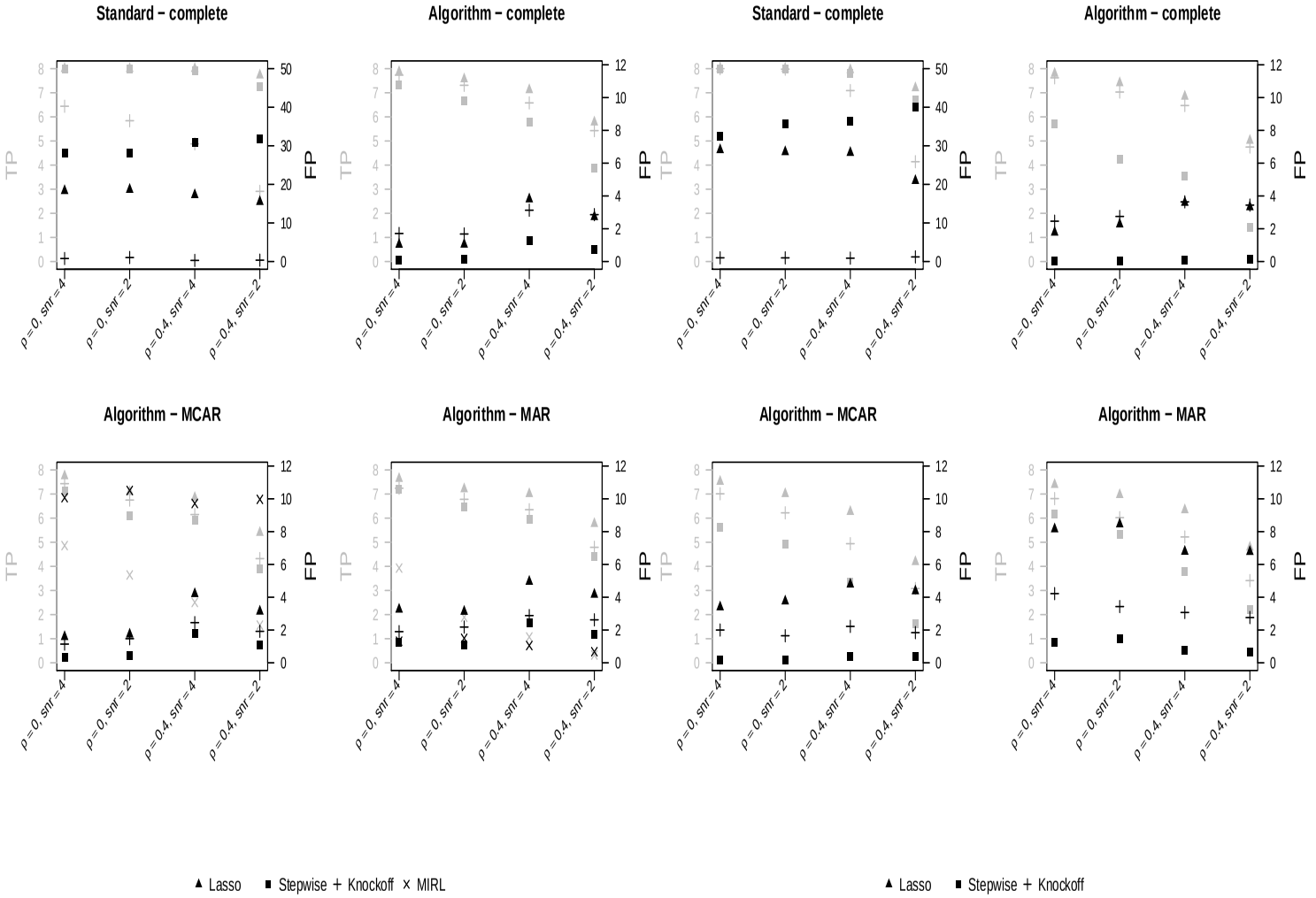
3.2 Results

We split results in four parts depending if we consider low/high dimension and complete or incomplete data (results shown in Figure 1). We successively present the results for the four ones. Finally, we study the robustness of our algorithm to the tuning parameters.

3.2.1 Low dimensional data without missing values

In the case $n > p$ without missing values, direct application of any standard selection variable procedures can be performed. The top-left of Figure 1a reports a very large number of false positives, FP, (over than 15) for lasso and stepwise. On the contrary, this number is well controlled by knockoff (close to 1), while having many true positives, TP, (over than 5) even when the signal to noise ratio is small or correlation between covariates is large.

With our ensemble method (see the top-right of Figure 1a), the selection based on knockoff shows larger number of TP and FP than its direct application on the data-set (see the top of Figure 1a). For stepwise and lasso, performances are much



(a) Low-dimensional setting

(b) High-dimensional setting

Figure 1: Assessment of the ensemble method according to the number of variables, the missing data mechanism and the data structure. Figure 1a reports results when $p = 100$ (low dimension) and Figure 1b reports results when $p = 300$ (high-dimensional setting). Assessment is based on data sets varying by the correlation between covariates (ρ), the signal to noise ratio (snr) and the missing data mechanism (complete, MCAR, MAR). For a given configuration, $T = 100$ data sets are generated. When data are complete (see the top of Figure 1a and Figure 1b), three variable selection methods (Knockoff, Lasso and Stepwise) are compared when they are directly applied on the data set (standard) or when they are iteratively applied on subsets of variables (algorithm). When data are incomplete, only the ensemble method can be applied. MIRL is also investigated when the dimension is low. Performances of the selection procedure are assessed by: the mean number of true positives (in grey) and the mean number of false positives (in black) over the $T = 100$ data sets.

better improved by our algorithm whatever the correlation and the signal to noise ratio. Indeed, the number of false positives becomes close to 0.

3.2.2 Low dimensional data with missing values

The bottom of Figure 1a and Table 1 in Appendix report simulation results when data are missing completely at random or missing at random. In such a case, lasso, knockoff and stepwise cannot be directly applied. Therefore, complete case analysis is used. Because of the decrease of the number of individuals, selection methods have less power, leading to very poor performances (results shown in Table 1 only). Indeed, the number of true positives is close to 0 and the number of false negatives close to 8 for each of them. When applying the MIRL method, selection is also quite bad. Indeed, the issue is that the predictive distribution of missing values is not well estimated because of the too large number of variables compared to the number of individuals [19].

On the contrary, by using our algorithm, the performances are globally similar to the case without missing values (cf bottom of Figure 1a).

3.2.3 High dimensional data without missing values

The top of Figure 1b summarizes simulation results in the case $n < p$ without missing values. In a similar way to the case where the dimensionality is low (at top of Figure 1a), our algorithm decreases the number of false positives for selection by lasso or stepwise, but does not improve performances of the knockoff. Note that the knockoff is well suited to handle high-dimensional data when data are complete [4].

3.2.4 High dimensional data with missing values

Bottom of Figure 1b and Table 2 report results for configurations where $n < p$ with missing values generated according to a MCAR or MAR mechanism. Because of the large number of variables, MIRL method cannot be applied since the imputation becomes too much time consuming. Direct application of variable selection methods by complete case analysis appears clearly irrelevant (see Table 2). Indeed, the number of false negatives is close to 8, whatever the selection variable method: like in the low dimensional case, complete case analysis decreases the power of the tests and selection variables methods rarely reject the null hypothesis.

On the opposite, our algorithm leads to a small number of FP, even if this number is a little higher than in the case without missing values (see the top-right of Figure 1b).

3.3 Influence of tuning parameters

To complete this simulation study, robustness to the tuning parameters is assessed. We focus on the number of variables sampled (k), the number of iterations (B) and the threshold (r).

3.3.1 Influence of k

Figure 2 reports the number of true positives and the number of false negatives according to the number of variables sampled in the algorithm (when $n > p$ without missing values). Surprisingly, the number of true positives is globally decreasing when k is increasing, like the false positive rate. More precisely, the increase is important for stepwise, while it remains moderate for lasso and knockoff. The opposite could be expected since the regression scheme is more noised if k is small (cf Section 2.3). The reason is that the counterpart to increasing k is to decrease the degrees of freedom attributed to the model selection process. However, by drawing k among p, the gain to increase k is small because the probability of selecting significant variables is small (here 8 over 300), while degrees of freedom are decreasing, implying a loss of power, but not a substantial decrease of noise on the regression scheme.

The behaviour is more severe for stepwise. Indeed, this procedure additionnaly often rejects the null if there are no significant variables in the subset. Such cases are more frequent when k is small and tends to disappear when k increases.

Note that similar results are observed for the high dimensional setting (Figure 8 in Appendix) or when data are incomplete (Figures 6 and 7 in Appendix).

3.3.2 Influence of B

B controls the uncertainty on the proportion r_j: for low values of B, the subset of selected variables is expected to be unstable. To assess the robustness of the results to the number of iterations, we inspect the standard deviation of the number of false positives and true positives over the $T = 100$ generated data set according to the number of iterations. For simplicity, we only inspect 2 configurations: in the first one, data are complete with a signal to noise ratio of 4, null correlation

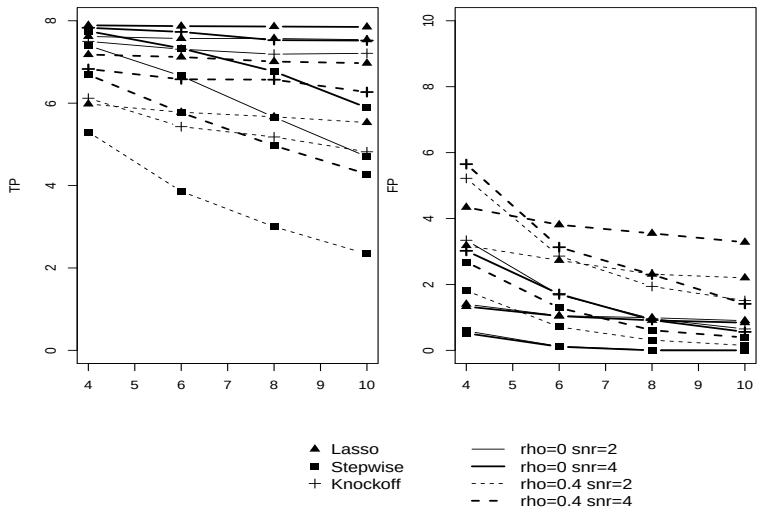


Figure 2: Influence of k for low dimensional setting without missing values: number of true positives (on the left) and false positives (on the right) according to the number of variables sampled in the algorithm (k) for the 4 configurations varying by the signal to noise ratio (snr) and the correlation between covariates (ρ). Three variable selection methods are reported (lasso, stepwise and knockoff).

between covariate and $n < p$ (Figure 3), while data are missing according to a MAR mechanism for the second one (Figure 9 in Appendix). As expected, in both cases, the variability of the TP and FP is decreasing and reaches convergence before 1000 iterations. This result is directly related to the variance of a proportion as mentioned in Section 2.3.2. Furthermore, the number of true positives is more stable than the false negative one, which is directly related to the larger number of negatives than positives in the data. For comparison, when selection variables methods are directly applied, standard deviation for TP is 0.14 for knockoff (1.29 for FP), 0 for lasso (13.35 for FP) and 0 for stepwise (6.34 for FP).

3.3.3 Influence of r

Of course, the threshold r allows a control on the FP and the TP since it is in bijection with the number of positives. According to the selection method used,

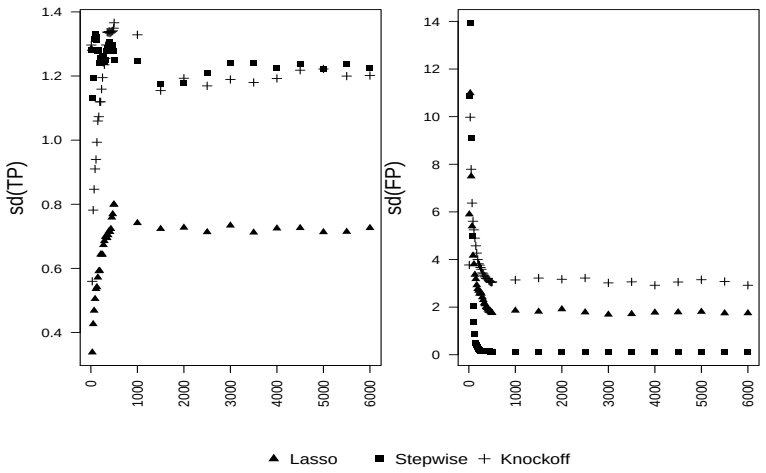


Figure 3: Influence of B for high dimensional setting without missing values: standard deviation of the number of true positives over the 100 generated data sets (on the left) and false positives (on the right) according to the number of iterations of the algorithm (B) for the configuration with signal to noise ratio equal to 4 and null correlation between covariates (ρ). Three variable selection methods are reported (lasso, stepwise and knockoff).

r can sometimes be tuned a priori, but in many cases, it should be driven by data. For achieving this goal, we use cross-validation. Figure 4 highlights performances of the algorithm when r is data driven (by using stepwise or lasso in the complete case, in the low or high-dimensional setting). In all configurations, cross-validation leads to choose a higher value of r than 0.95, which decreases the number of positives. The gain in terms of FP is more substantial than the lost in terms of TP since the number of real false positive (92 in the low dimensional setting and 292 in the high dimensional setting) is much higher than the number of real positives (8).

Finally, we illustrate the relationship between r and k, by investigating the robustness of the procedure to k when r is chosen by cross-validation. Results are shown in Figure 5. Compared to Figure 2 error rates are stable whatever the choice of k. Thus, cross-validation for r tuning makes the procedure robust to the

choice of k.

4 Discussion

High dimensional data as well as missing data are two of the main challenges for applied statistician at the digital era. In this article we proposed an algorithm for variable selection in the framework of linear models. This algorithm improves the performances of many selection methods (in terms of true positive and false positive rates) and provides a measure of importance for the explanatory variables. Furthermore, it allows handling missing values (MAR or MCAR) and/or high dimensional settings for any variable selection method. From a practical point of view, the method has the advantage to allow parallel calculation, solving some potential calculation time issues. In addition, its parameters can be easily tuned: the number of iterations B can be checked by inspecting stability of proportions of selection, while the number of variables drawn (k) can be chosen a priori (since the method is robust to this parameter) and the threshold r can be chosen by cross-validation.

Various extensions of the algorithm can be proposed. First, the algorithm can be easily adapted in the case of Generalized Linear Models (GLM) or mixed models, but additional statistical work has to be done to tune the parameters.

We did not explore the specific case of data missing not at random, but the algorithm could be adapted to accounting for such mechanisms by using suitable imputation method [16, e.g.].

Refinements of the algorithm could also be possible. In particular, accounting for the variation around r_j in the threshold could be quite easy. This could be useful for high time-consuming configurations, where the number of iterations need to be limited, since this variability could not be ignored anymore.

Outliers is also a classical problem in data analysis. While robust estimates can be considered (see [22] for example), it is also possible to remove them by replacing them with missing values. Therefore, this algorithm provides a way to handle outliers in variable selection.

Moreover, in this article we fixed a threshold to include (or not) a variable in the model for a given instance, but we could also aggregate the probabilities (under the null hypothesis) that $\beta_j = 0$. A natural aggregation over all instances is given by the empirical mean, that can be seen as the mean of the estimates of $\mathbb{P}(\beta_j = 0)$. Then, for each variable, this mean would be thresholded, as proposed.

Finally, we focused on variable selection, but one may notice that each in-

stance gives estimates of β and we can also aggregate these estimates. However, such an extension is not straightforward since estimates are generally biased on all instances. Further research on aggregation of those biased estimates could lead to the development of a robust estimator of regression coefficients in a high dimensional setting with missing values.

References

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

[2] Francis R Bach. Bolasso: model consistent lasso estimation through the bootstrap. In *Proceedings of the 25th international conference on Machine learning*, pages 33–40. ACM, 2008.

[3] R. F. Barber and E. J. Candès. Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085, 2015.

[4] R.F. Barber and E. J. Candes. A knockoff filter for high-dimensional selective inference. *ArXiv e-prints*, 2016.

[5] F.E. Harrell. *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. Springer Series in Statistics. Springer International Publishing, 2015.

[6] L. Breiman. Heuristics of instability and stabilization in model selection. *The annals of statistics*, 24(6):2350–2383, 1996.

[7] Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

[8] Robin Genuer, Jean-Michel Poggi, and Christine Tuleau-Malot. Variable selection using random forests. *Pattern Recognition Letters*, 31(14):2225–2236, 2010.

[9] Larry Wasserman and Kathryn Roeder. High dimensional variable selection. *Annals of statistics*, 37(5A):2178, 2009.

[10] J. Schafer. *Analysis of Incomplete Multivariate Data*. Chapman & Hall/CRC, London, 1997.

[11] D. Rubin. *Multiple Imputation for Non-Response in Survey*. Wiley, New-York, 1987.

[12] R. Little and D. Rubin. *Statistical Analysis with Missing Data*. Wiley series in probability and statistics, New-York, 2002.

[13] Yize Zhao and Qi Long. Variable selection in the presence of missing data: imputation-based methods. *Wiley Interdisciplinary Reviews: Computational Statistics*, 9(5):e1402–n/a, 2017. e1402.

[14] Nicolas Städler and Peter Bühlmann. Missing values: sparse inverse covariance estimation and an extension to sparse regression. *Statistics and Computing*, 22(1):219–235, 2012.

[15] Po-Ling Loh and Martin J Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. In *Advances in Neural Information Processing Systems*, pages 2726–2734, 2011.

[16] JacquesEmmanuel Galimard, Sylvie Chevret, Camelia Protopopescu, and Matthieu Resche-Rigon. A multiple imputation approach for mnar mechanisms compatible with heckman's model. *Statistics in Medicine*, 35(17):2907–2920, 2016.

[17] A. Kapelner and J. Bleich. Prediction with missing data via bayesian additive regression trees. *Canadian Journal of Statistics*, 43(2):224–239, 2015.

[18] Evan Patterson and Matteo Sesia. *knockoff: The Knockoff Filter for Controlled Variable Selection*, 2017. R package version 0.3.0.

[19] Vincent Audigier, Franois Husson, and Julie Josse. Multiple imputation for continuous variables using a bayesian principal component analysis. *Journal of Statistical Computation and Simulation*, 86(11):2140–2156, 2016.

[20] Ying Liu, Yuanjia Wang, Yang Feng, and Melanie M Wall. Variable selection and prediction with incomplete high-dimensional data. *The annals of applied statistics*, 10(1):418, 2016.

[21] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019.

[22] Andreas Alfons, Christophe Croux, and Sarah Gelper. Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *The Annals of Applied Statistics*, pages 226–248, 2013.

				Algorithm			Standard		
ρ	snr	mech	method	TP	FN	FP	TP	FN	FP
0	2	MCAR	Knockoff	6.75	1.25	1.47			
0	2	MAR	Knockoff	6.78	1.22	2.18	0.18	7.82	0.44
0	4	MCAR	Knockoff	7.43	0.57	1.14			
0	4	MAR	Knockoff	7.24	0.76	1.90	0.11	7.89	0.49
0.4	2	MCAR	Knockoff	4.32	3.68	1.92			
0.4	2	MAR	Knockoff	4.79	3.21	2.62	0.07	7.93	0.47
0.4	4	MCAR	Knockoff	6.15	1.85	2.45			
0.4	4	MAR	Knockoff	6.35	1.65	2.88	0.04	7.96	0.52
0	2	MCAR	Lasso	7.09	0.91	1.74			
0	2	MAR	Lasso	7.21	0.79	3.13	0.45	7.55	2.62
0	4	MCAR	Lasso	7.75	0.25	1.59			
0	4	MAR	Lasso	7.64	0.36	3.26	0.58	7.42	3.49
0.4	2	MCAR	Lasso	5.40	2.60	3.14			
0.4	2	MAR	Lasso	5.77	2.23	4.17	0.21	7.79	1.83
0.4	4	MCAR	Lasso	6.85	1.15	4.21			
0.4	4	MAR	Lasso	7.01	0.99	4.96	0.18	7.82	2.10
0	2	MCAR	Stepwise	6.11	1.89	0.46			
0	2	MAR	Stepwise	6.48	1.52	1.07			
0	4	MCAR	Stepwise	7.14	0.86	0.31			
0	4	MAR	Stepwise	7.18	0.82	1.25			
0.4	2	MCAR	Stepwise	3.88	4.12	1.07			
0.4	2	MAR	Stepwise	4.42	3.58	1.76			
0.4	4	MCAR	Stepwise	5.91	2.09	1.77			
0.4	4	MAR	Stepwise	5.95	2.05	2.44			

Table 1: Low dimensional setting with missing values: performances of three variable selection methods (Knockoff, Lasso and Stepwise) when they are iteratively applied on imputed subsets of variables (Algorithm) or when they are applied on the complete individuals of the data set (Standard). Missing values are related to failure because of a too low number of complete-cases. Data sets varying by the correlation between covariates (ρ) and the signal to noise ratio (snr). For a given configuration, $T = 100$ data sets are generated and performances of the selection procedure are assessed by: the mean number of true positives (TP), the mean number of false negatives (FN) and the mean number of false positives (FP) (the number of real positives is 8 and the number of real negatives is 92).

				Algorithm			Standard		
ρ	snr	mech	method	TP	FN	FP	TP	FN	FP
0	2	MCAR	Knockoff	6.58	1.42	3.57			
0	2	MAR	Knockoff	6.32	1.68	6.22	0.02	7.98	0.77
0	4	MCAR	Knockoff	7.20	0.80	4.21			
0	4	MAR	Knockoff	7.19	0.81	6.69	0.04	7.96	0.52
0.4	2	MCAR	Knockoff	4.04	3.96	4.76			
0.4	2	MAR	Knockoff	4.32	3.68	6.52	0.03	7.97	0.53
0.4	4	MCAR	Knockoff	5.51	2.49	4.75			
0.4	4	MAR	Knockoff	5.78	2.22	6.19	0.00	8.00	0.47
0	2	MCAR	Lasso	7.11	0.89	4.39			
0	2	MAR	Lasso	6.96	1.04	8.63	0.14	7.86	3.07
0	4	MCAR	Lasso	7.58	0.42	4.02			
0	4	MAR	Lasso	7.45	0.55	8.48	0.08	7.92	2.66
0.4	2	MCAR	Lasso	4.42	3.58	5.22			
0.4	2	MAR	Lasso	4.77	3.23	7.33	0.03	7.97	1.83
0.4	4	MCAR	Lasso	6.33	1.67	5.59			
0.4	4	MAR	Lasso	6.40	1.60	7.53	0.05	7.95	1.88
0	2	MCAR	Stepwise	6.01	1.99	0.71			
0	2	MAR	Stepwise	6.18	1.82	2.92			
0	4	MCAR	Stepwise	6.77	1.23	0.80			
0	4	MAR	Stepwise	6.79	1.21	2.79			
0.4	2	MCAR	Stepwise	2.62	5.38	1.08			
0.4	2	MAR	Stepwise	3.22	4.78	2.13			
0.4	4	MCAR	Stepwise	4.65	3.35	1.36			
0.4	4	MAR	Stepwise	5.04	2.96	2.05			

Table 2: High dimensional setting with missing values: performances of three variable selection methods (Knockoff, Lasso and Stepwise) when they are iteratively applied on imputed subsets of variables (Algorithm) or when they are applied on the complete individuals of the data set (Standard). Missing values are related to failure because of a too low number of complete-cases. Data sets varying by the correlation between covariates (ρ) and the signal to noise ratio (snr). For a given configuration, $T = 100$ data sets are generated and performances of the selection procedure are assessed by: the mean number of true positives (TP), the mean number of false negatives (FN) and the mean number of false positives (FP) (the number of real positives is 8 and the number of real negatives is 92).

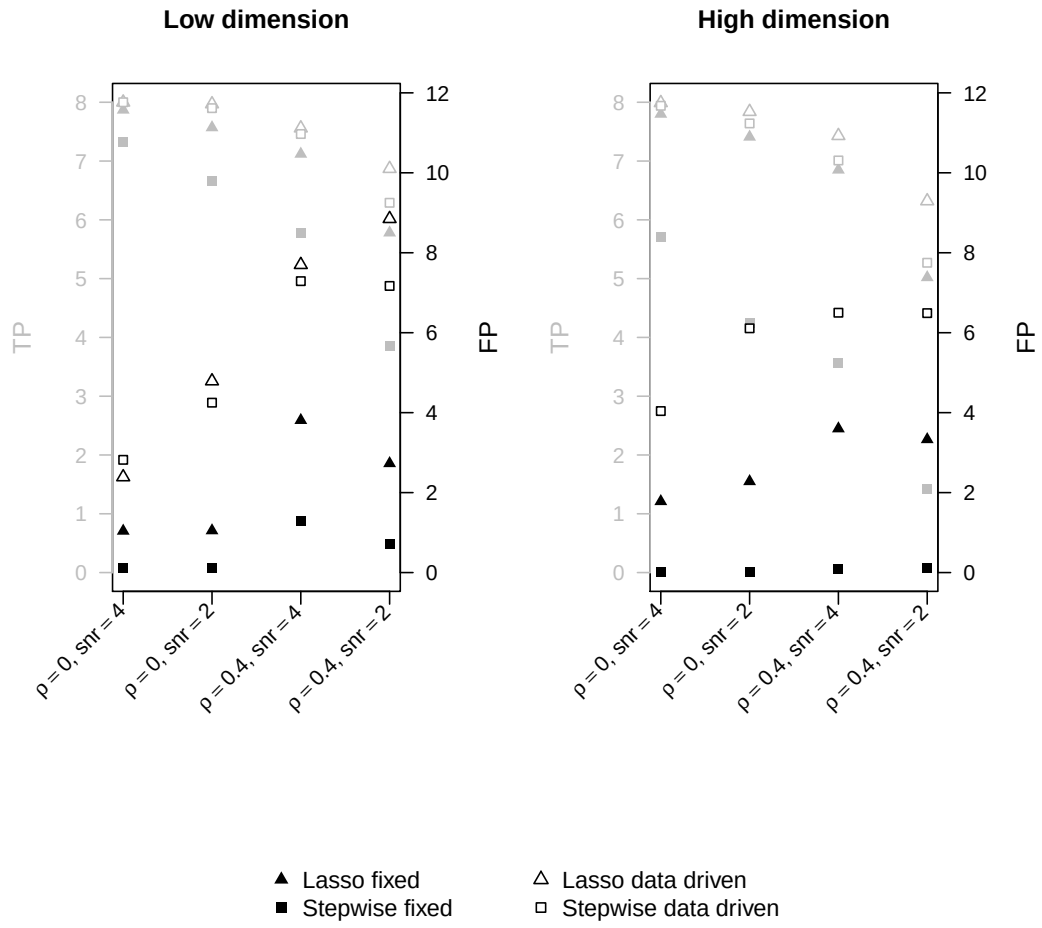


Figure 4: Influence of the thresholding in the low dimensional (left) and high (right) setting without missing values: illustration for Lasso and Stepwise by tuning the threshold by cross-validation or by fixing it to 0.95. Data sets vary by the correlation between covariates (ρ) and the signal to noise ratio (snr). For a given configuration, $T = 100$ data sets are generated and performances of the selection procedure are assessed by: the number of true positives and the false positive (the number of real positive is 8 and the number of real negative is 92 or 292).

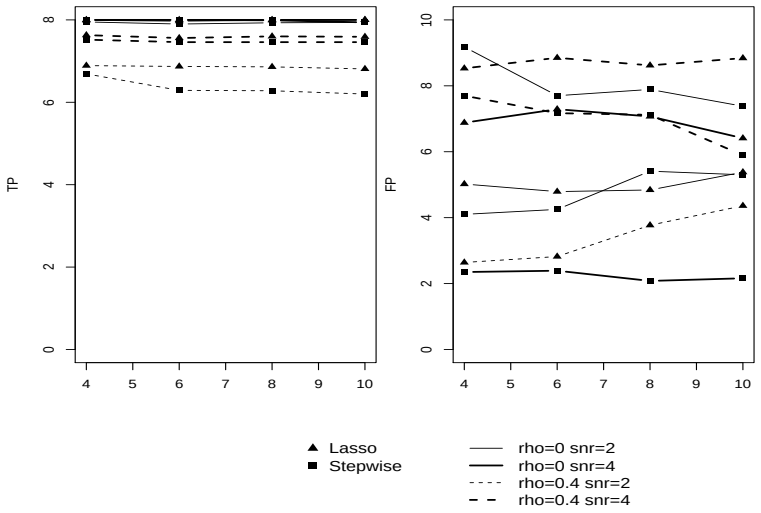


Figure 5: Influence of k for low dimensional setting without missing values when r is chosen by cross-validation: number of true positives (on the left) and false positives (on the right) according to the number of variables sampled in the algorithm (k) for the 4 configurations varying by the signal to noise ratio (snr) and the correlation between covariates (ρ). Two variable selection methods are reported (lasso and stepwise).

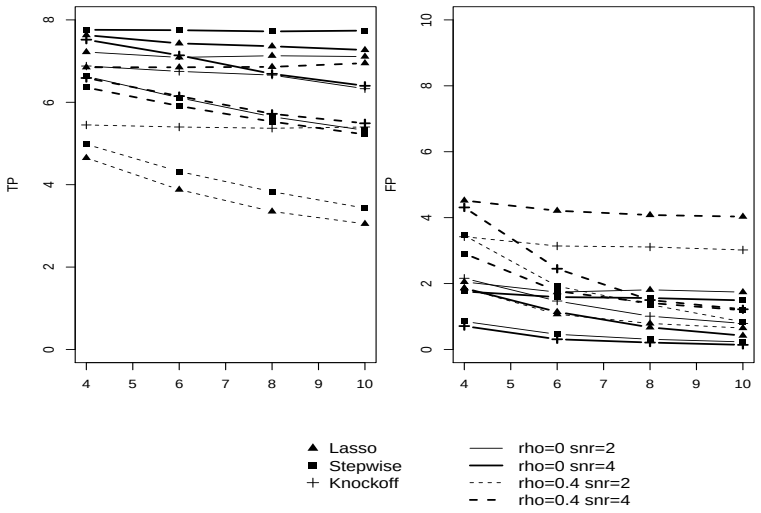


Figure 6: Influence of k in the low dimensional setting with values missing completely at random: true positive rate (on the left) and false positive rate (on the right) according to the number of variables sampled in the algorithm (k) for the 4 configurations varying by the signal to noise ratio (snr), the correlation between covariates (ρ). Three variable selection methods are reported (lasso, stepwise and knockoff).

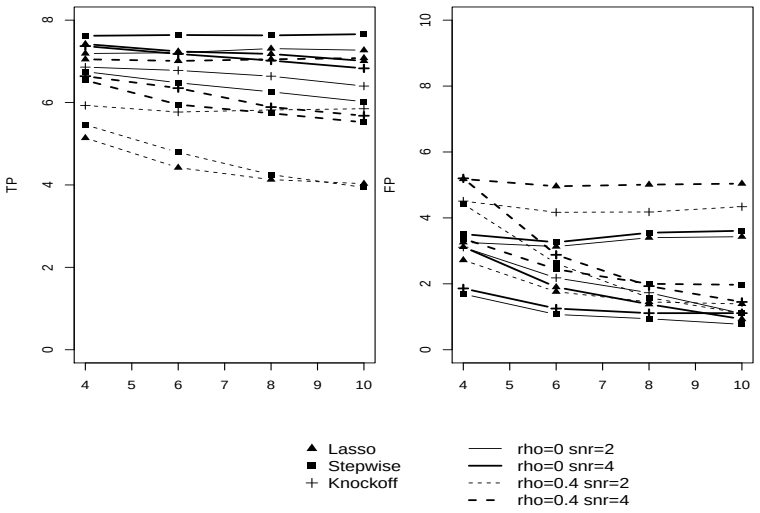


Figure 7: Influence of k in the low dimensional setting with values missing at random: true positive rate (on the left) and false positive rate (on the right) according to the number of variables sampled in the algorithm (k) for the 4 configurations varying by the signal to noise ratio (snr), the correlation between covariates (ρ). Three variable selection methods are reported (lasso, stepwise and knockoff).

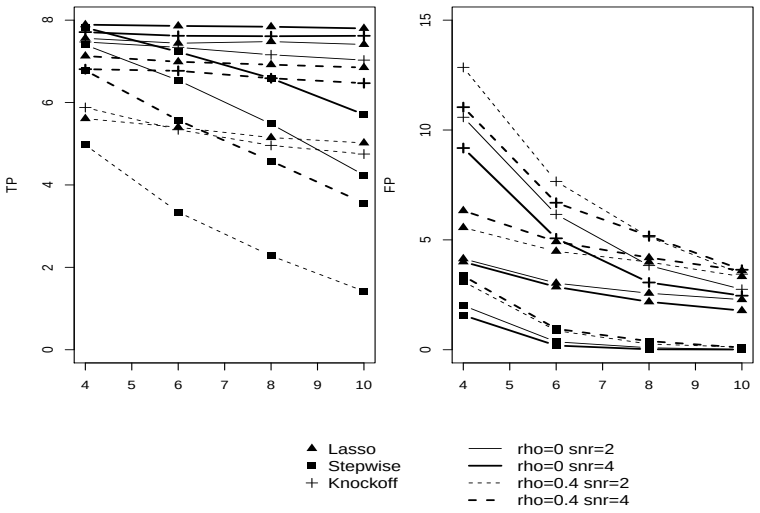


Figure 8: Influence of k in the high dimensional setting without missing values: true positive rate (on the left) and false positive rate (on the right) according to the number of variables sampled in the algorithm (k) for the 4 configurations varying by the signal to noise ratio (snr), the correlation between covariates (ρ). Three variable selection methods are reported (lasso, stepwise and knockoff).

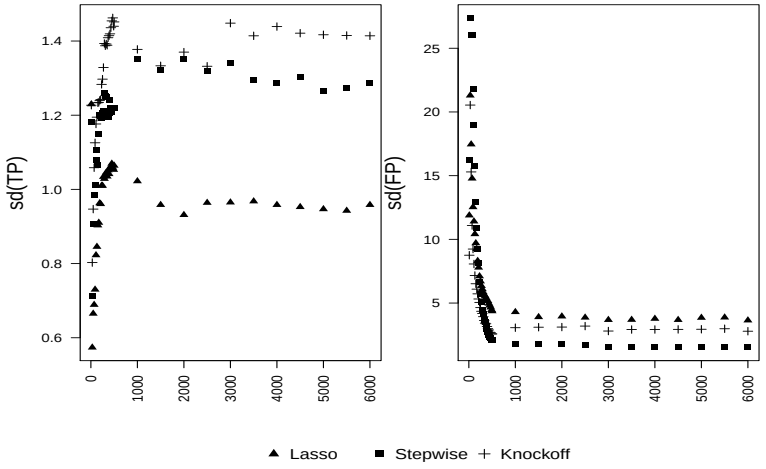


Figure 9: Influence of B for high dimensional setting with missing values: standard deviation of the true positive rate over the 100 generated data sets (on the left) and false positive rate (on the right) according to the number of iterations of the algorithm (B) for the configuration with signal to noise ratio equal to 4, null correlation between covariates (ρ) and values missing at random. 3 variable selection methods are reported (lasso, stepwise and knockoff).