Controlling false positive selections in high-dimensional regression and causal inference

Peter Bühlmann[1] Philipp Rütimann[1] Markus Kalisch[1]

Abstract

Guarding against false positive selections is important in many applications. We discuss methods based on subsampling and sample splitting for controlling the expected number of false positives and assigning p-values. They are generic and especially useful for high-dimensional settings. We review encouraging results for regression, and we discuss new adaptations and remaining challenges for selecting relevant variables, based on observational data, having a causal or interventional effect on a response of interest.

Keywords

High-dimensional causal inference; High-dimensional regression; Lasso; Observational data; PC-algorithm; P-values; Stability selection

1 Introduction

Determining important variables for a response of interest is an often encountered problem in many applications. We consider it in the high-dimensional framework where the number of variables is much larger than sample size. Furthermore, we address variable importance for two very different targets, namely either in terms of regression effects or with respect to causal or intervention effects. A lot of work has been done in the past for variable selection in high-dimensional (generalized) regression [24, 7, 31, 18, 30, 29, 17, 32, 28, 2] while the counterpart for causal or interventional analysis based on observational data is much less developed [16].

All of these methods need to be complemented with measures of uncertainty, especially for controlling false positive selections (type I error) which is a prime concern in e.g. medical applications. The main difficulty is that the parameter corresponding to a variable is not a marginal effect but rather a functional of a multi-dimensional distribution: hence, statistical inference is much more challenging than for e.g. large-scale multiple testing of many marginal parameters [6] or global tests [10]. For non-marginal parameters such as regression coefficients or causal effects one can rely on subsampling or sample-splitting: we review and discuss stability selection [19] and construction of p-values [20] for regression problems (Section 3), and we adapt the techniques for the more ambitious task of variable selection for causal or interventional targets based on observational data (Section 4).

2 Variable selection in high-dimensional regression

Consider the situation with a univariate response variable Y and p-dimensional covariate X. The response and the covariates can be continuous or discrete and the latter are deterministic (fixed design) or random (random design). The data is assumed to be realizations from a generalized linear model of the form

$$\begin{aligned} &Y_1, \ldots, Y_n \text{ independent} \\ &g(\mathbb{E}[Y_i | X_i = x]) = \mu + \sum_{j=1}^{p} \beta_j x^{(j)}, \end{aligned} \tag{2.1}$$

[1]Seminar für Statistik, ETH Zürich, Switzerland

Corresponding author: Peter Bühlmann, Seminar für Statistik, ETH Zürich, CH-8092 Zürich, Switzerland. Email: buhlmann@stat.math.ethz.ch

where $g(\cdot)$ is a real-valued, known link function, μ denotes the intercept and β the generalized regression parameters. An implicit assumption of the model in (2.1) is that the (conditional) distribution of Y_i (given X_i) is depending on X_i only through the function $g(\mathbb{E}[Y_i|X_i])$. The linear model is a special case with identity link function $g(\eta) = \eta$ and we then write

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon, \tag{2.2}$$

with $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$, $(n \times p)$-design matrix $\mathbf{X}$ whose ith row equals X_i and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^T$ with independent components and $\mathbb{E}[\varepsilon_i|X_i] = 0$ for all i. With this notation, we always absorb a potential intercept term into the design matrix $\mathbf{X}$ (and we would then have $p + 1$ columns in $\mathbf{X}$ with an intercept and p covariates; for notational simplicity, we omit this detail in the sequel). When assuming that the model is true, we denote the true parameter by β^0.

We allow that the dimension of the covariate may be much larger than sample size n, i.e., high-dimensionality with $p \gg n$. To cope with this situation, we need to regularize the statistical estimation procedure.

2.1 The Lasso

The Lasso [24] is a very popular and powerful method to estimate the parameters in a high-dimensional generalized linear model. For the linear model in (2.2), the parameters are estimated by

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \Big(\|\mathbf{Y} - \mathbf{X}\beta\|_2^2/n + \lambda\|\beta\|_1 \Big), \tag{2.3}$$

where $\|\mathbf{Y} - \mathbf{X}\beta\|_2^2 = \sum_{i=1}^n (Y_i - (\mathbf{X}\beta)_i)^2$, $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$ is the ℓ_1-norm and where $\lambda \geq 0$ is a penalty parameter. The estimator has the property that it does variable selection in the sense that $\hat{\beta}_j(\lambda) = 0$ for some j's (depending on the choice of λ) and $\hat{\beta}_j(\lambda)$ can be thought as a shrunken least squares estimator; hence, the name Least Absolute Shrinkage and Selection Operator (LASSO).

For generalized linear models in (2.1), the Lasso estimator is defined by penalizing the negative log-likelihood with the ℓ_1-norm. The (conditional) probability or density of $Y|X = x$ is of the form $\mathrm{p}(y|x) = \mathrm{p}_{\mu,\beta}(y|x)$ and the negative log-likelihood equals $-\sum_{i=1}^n \log(\mathrm{p}_{\mu,\beta}(Y_i|X_i))$. The ℓ_1-norm penalized Lasso estimator is then defined as:

$$\hat{\mu}(\lambda), \hat{\beta}(\lambda) = \arg\min_{\mu,\beta} \left(-n^{-1} \sum_{i=1}^n \log(\mathrm{p}_{\mu,\beta}(Y_i|X_i)) + \lambda\|\beta\|_1 \right),$$

where we usually do not penalize the intercept term (and the same remark applies to the Lasso for a linear model above).

For linear and generalized linear models, the Lasso can be computed very efficiently since it involves convex optimization only. In comparison, an all subset selection method is not feasible due to the combinatorial complexity when p is large.

2.1.1 Variable selection with the Lasso

With the Lasso estimator $\hat{\beta}(\lambda)$, we can do prediction and variable selection. We are particularly interested in the latter. We implicitly assume that only a fraction of the variables in the generalized linear model is relevant: we denote the true active set of variables and its cardinality (the sparsity of the model) as

$$S_0 = \{j;\ \beta_j^0 \neq 0,\ j = 1, \ldots, p\} \text{ with } s_0 = |S_0|.$$

The simplest way to estimate S_0 with the Lasso is

$$\hat{S}(\lambda) = \{j;\ \hat{\beta}_j(\lambda) \neq 0,\ j = 1, \ldots, p\}. \tag{2.4}$$

We remark that the variables with corresponding non-zero estimated coefficients remain the same across different solutions $\hat{\beta}(\lambda)$ of the Lasso (different solutions occur when $p > n$ since the optimization, although convex, is not strictly convex). Furthermore, it is worth pointing out that no significance testing is involved.

The selection in (2.4) can be mathematically justified, assuming restrictive conditions. In a linear model, the main assumptions for consistent variable selection concern the (fixed or random) design matrix $\mathbf{X}$ and the magnitude of the non-zero regression coefficients. The condition on the design is called neighborhood stability [18] or irrepresentable condition [29, 30]. Under such a design condition, and assuming that the non-zero regression coefficients satisfy a "beta-min" condition of the form

$$\inf_{j \in S_0} |\beta_j^0| \gg \sqrt{s_0 \log(p)/n}, \tag{2.5}$$

The following is shown in [18]: for a suitable $\lambda = \lambda_n \gg \sqrt{\log(p_n)/n}$,

$$\mathbb{P}[\hat{S}(\lambda) = S_0] \to 1 \ (n \to \infty), \tag{2.6}$$

where the asymptotics allows that $p = p_n$ grows such that $\log(p_n)/n \to 0$.

In practice, we would choose the regularization parameter λ by cross-validation, e.g., 10-fold CV, leading to $\hat{\lambda}_{\mathrm{CV}}$. There is supporting theory for particular examples, and there is a lot of empirical evidence as well for many more examples, that

$$\mathbb{P}[\hat{S}(\hat{\lambda}_{\mathrm{CV}}) \supseteq S_0] \text{ is large}, \tag{2.7}$$

if the design behaves reasonably and condition (2.5) holds. For such a screening result, we need less restrictive design conditions (i.e. compatibility condition or restricted eigenvalue assumption) than for recovering the active set S_0: for a fixed $\lambda \asymp \sqrt{\log(p)/n}$, this is discussed in e.g. [3, Cor.7.6]. The screening result in (2.7) is very useful because $|\hat{S}(\lambda)| \leq \min(n, p)$ for any λ. Thus, when $p \gg n$, we achieve an immense dimensionality reduction (in terms of the original covariates) which contains with high probability the true active variables from S_0.

We will discuss below that this view point is rather optimistic in real applications. The main reason for it stems from the fact that the irrepresentable condition on the design and the "beta-min" condition in (2.5) are restrictive and (essentially) necessary for variable selection consistency of the Lasso, and also the screening result still needs condition (2.5).

2.2 Variable selection with other methods

There are many other proposals in the literature for inferring the active set S_0 in a high-dimensional generalized linear model [7, 31, 30, 17, 32, 28]. Some of them diminish the bias problems of the Lasso causing the necessity of the restrictive irrepresentable condition on the design. We will not go into details about this. All of them, nevertheless, require a "beta-min" condition as in (2.5) which seems almost necessary for any method. In practice, it may be often unlikely to believe that most of the coefficients are exactly zero and all the relevant variables have large coefficients satisfying (2.5). In view of uncheckable conditions on the design and the signal strength as in (2.5), we discuss next some methods for obtaining more reliable results in statistical practice.

3 Stability and p-values in high-dimensional regression

We illustrate here that a result as in (2.7) needs to be interpreted with care. Violation of certain conditions on the design or of the "beta-min" condition in (2.5) can easily lead to failure of (2.7).

3.1 Example about motif regression: subsampling and stability

We use the Lasso on a motif regression problem [5] for finding the binding sites in DNA sequences of the so-called HIF1α transcription factor. Such transcription factor binding sites, also called motifs, are short "words" of DNA letters denoted by $\{A, C, G, T\}$, typically 6-15 base pairs long.

The data consists of a univariate response variable Y measuring the binding intensity of the HIF1α transcription factor on coarse DNA segments which are a few thousands base pairs long. This data is collected using CHIP-chip experiments. In order to get to the exact short DNA "words" or motifs, short candidate DNA "words" of length 6–15 base pairs are generated and their abundance scores are measured within coarse DNA regions. This can be done using computational biology algorithms based on DNA sequence data only, and we use a variant of the MDScan algorithm [14]. In our particular application, we have the following data:

Y_i measures the binding intensity of HIF1α in coarse DNA segment i,
$X_i^{(j)}$ measures the abundance score of candidate motif j in DNA segment i,
$i = 1, \ldots, n = 287;\ j = 1, \ldots, p = 195$.

A linear model fits reasonably well (see below) for relating the response to the covariates:

$$Y_i = \mu + \sum_{j=1}^{195} \beta_j X_i^{(j)} + \varepsilon_i \ (i = 1, \ldots, n = 287).$$

The main goal in this application is variable selection to infer the relevant covariates and hence the relevant motifs (short DNA "words"). Having scaled the covariates to the same empirical variance, we use the Lasso with regularization parameter $\hat{\lambda}_{\mathrm{CV}}$ from 10-fold cross-validation. The fitted model has a (OLS re-fitted) $R^2 \approx 50\%$ which is rather high for this kind of application. There are $|\hat{S}(\hat{\lambda}_{\mathrm{CV}})| = 26$ non-zero coefficient estimates $\hat{\beta}_j(\hat{\lambda}_{\mathrm{CV}})$ which are plotted in the upper-left panel of Figure 1.

Next, we pursue subsampling to informally check stability of the estimated set of variables. We denote by $\hat{S}_\lambda(I)$ the estimated active set based on regularization parameter λ and a subsample $I \subset \{1, \ldots, n\}$ (note the slight switch in notation). Let I^* be a random subsample of $\{1, \ldots, n\}$ of size $\lfloor n/2 \rfloor$, drawn without replacement. For every set $K \subseteq \{1, \ldots, p\}$, the subsampling-probability of being in the selected set $\hat{S}_\lambda(\cdot)$ is

$$\hat{\Pi}_K(\lambda) = \mathbb{P}^*[K \subseteq \hat{S}_\lambda(I^*)]. \tag{3.1}$$

Most often, we use single variables as sets K, i.e., $|K| = 1$. The probability $\mathbb{P}^*$ in (3.1) is with respect to the random subsampling and it equals the relative frequency for $K \subseteq \hat{S}_\lambda(I_b)$ over all $\binom{n}{m}$ subsets I_b ($b = 1 \ldots, \binom{n}{m}$) of size $m = \lfloor n/2 \rfloor$. The expression in (3.1) can be approximated by B random subsamples $I^{*1}, \ldots, I^{*B}$ (B large): $B^{-1} \sum_{b=1}^{B} 1(K \subseteq \hat{S}_\lambda(I^{*b}))$. The subsample size of $\lfloor n/2 \rfloor$ is chosen as it resembles most closely the bootstrap [8, 4].

Figure 1 illustrates three summary statistics of the subsampled estimator. Besides the subsampling probabilities $\hat{\Pi}_j(\hat{\lambda}_{\mathrm{CV}})$ for $j = 1, \ldots, p$, we also show the mean and median of $\{\hat{\beta}_{\lambda_{\mathrm{CV}};j}(I^{*b}); b = 1, \ldots, B\}$. The mean and median aggregation qualitatively look rather similar to the original, non-subsampled estimated coefficients. However, the subsampled probabilities $\hat{\Pi}_j$ yield a rather different picture. We will argue in Section 3.2 that using $\hat{\Pi}_j$ leads to interesting methodology and theory.

3.2 Stability selection

The example in Section 3.1 illustrates that we should develop measures quantifying uncertainty and reliability regarding the selection of variables.

In the following, we consider any variable selection, denoted by

$$\hat{S}(\lambda) = \hat{S}_\lambda(I) \subseteq \{1, \ldots, p\}$$

based on a sample $I \subseteq \{1, \ldots, n\}$ (I may be the full sample with all n data) and with a tuning parameter $\lambda \in \Lambda \subseteq \mathbb{R}^+$. (Note the slight change of notation where we emphasize at some places with $\hat{S}_\lambda(I)$ the dependence on the sample I). A prime example is the Lasso as in (2.4).

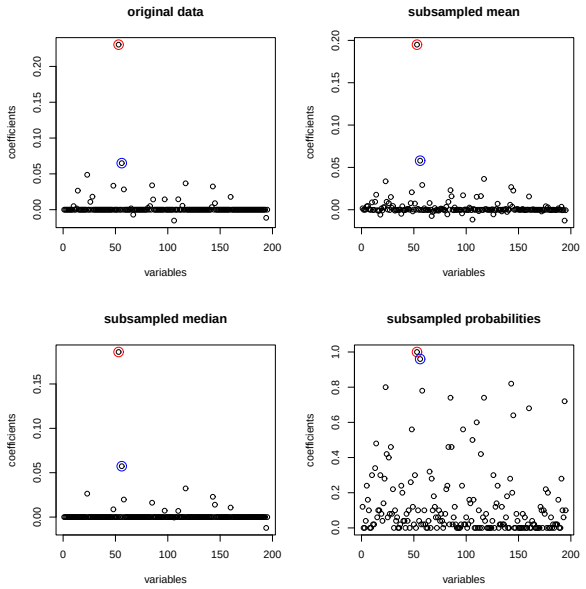


Figure 1: Motif regression example. Upper-left panel: Estimated regression parameters with $\hat{s} = 26$ non-zero coefficients, based on original data. Upper-right and lower-left panels: mean and median aggregated subsampled coefficient estimates. Lower-right panel: subsampled selection probabilities $\hat{\Pi}_j$. All estimates based on the Lasso with $\hat{\lambda}_{\mathrm{CV}}$ from 10-fold CV; number of subsampling runs is $B = 100$.

For a cutoff π_{thr} with $0 < \pi_{\mathrm{thr}} < 1$ and a set of regularization parameters Λ, the set of stable variables is defined as

$$\hat{S}_{\mathrm{stable}} = \{j;\ \max_{\lambda \in \Lambda} \hat{\Pi}_j(\lambda) \geq \pi_{\mathrm{thr}}\}, \tag{3.2}$$

where $\hat{\Pi}$ is as defined in (3.1). The procedure using (3.2) is called stability selection [19]. Of course, the problem of choosing an appropriate amount of regularization has now shifted to choose a good cutoff value $0 < \pi_{\mathrm{thr}} < 1$. We will discuss its choice below in Section 3.2.1. It is worthwhile to emphasize that empirical results do not depend very much on the choice of the initial regularization λ (if $\Lambda = \{\lambda\}$ is a singleton) or the region Λ for the regularization parameter: loosely speaking, as long as λ or Λ contain values leading to overestimation of the true active set of variables S_0, the results after the stability selection procedure are often useful.

3.2.1 Choice of regularization and error control

We focus here on the problem how to choose the regularization parameter π_{thr} in the stability selection procedure in (3.2). We address it by controlling the expected number of false positives (false selections), i.e., type I error control.

For such an error control, we introduce some additional notation. Let $\hat{S}_\Lambda = \cup_{\lambda \in \Lambda} \hat{S}(\lambda)$ be the set of selected variables when varying the regularization parameter $\lambda \in \Lambda$. Let q_Λ be the expected number of selected variables $q_\Lambda = \mathbb{E}|\hat{S}_\Lambda(I)|$. Define V to be the number of falsely selected variables (false positives) with stability selection,

$$V = \left|S_0^c \cap \hat{S}_{\mathrm{stable}}\right|.$$

The goal is to achieve control or an upper bound for $\mathbb{E}[V]$, the expected number of false positives.

Since the distribution of the underlying estimator $\hat{S}(\lambda)$ depends on many unknown quantities, exact finite-sample control of $\mathbb{E}[V]$ is difficult in general. We provide an answer under some simplifying assumptions.

Theorem 1. *[19] Consider data $Z_1, \ldots, Z_n$ i.i.d., where $Z_i = (X_i, Y_i)$. Assume that the distribution of $\{1(j \in \hat{S}(\lambda)), j \in S_0^c\}$ is exchangeable for all $\lambda \in \Lambda$. Furthermore, assume that the original selection procedure is not worse than random guessing, i.e.,*

$$\frac{\mathbb{E}(|S_0 \cap \hat{S}_\Lambda|)}{\mathbb{E}(|S_0^c \cap \hat{S}_\Lambda|)} \geq \frac{|S_0|}{|S_0^c|}. \tag{3.3}$$

Then, the expected number V of falsely selected variables is bounded for $\pi_{\text{thr}} \in (1/2, 1]$ by

$$\mathbb{E}[V] \quad \leq \quad \frac{1}{2\pi_{\text{thr}} - 1} \frac{q_\Lambda^2}{p}. \tag{3.4}$$

The exchangeability condition is a restrictive assumption, see for example [3, Ch.10]. The expected number of falsely selected variables is sometimes called the per-family error rate (PFER) or, if divided by p, $\mathbb{E}[V]/p$ is the per-comparison error rate (PCER) in multiple testing. Note that Theorem 1 does not require a "beta-min condition" as in (2.5) because the theorem only makes a statement about false positive selections (while a "beta-min" condition is needed to avoid false negatives).

For fixed Λ, the threshold value π_{thr} is the tuning parameter for stability selection. We propose to fix this value via the PFER control $\mathbb{E}[V] \leq \nu$ where ν is specified a-priori. Note that this fits into the commonly used framework of fixing type-I error control beforehand. Given ν, we can then solve for the tuning parameter:

$$\text{if } q_\Lambda^2 \leq p\nu: \ \pi_{\text{thr}} = (1 + \frac{q_\Lambda^2}{p\nu})/2, \tag{3.5}$$

and if $q_\Lambda^2 > p\nu$, we cannot control the error $\mathbb{E}[V] \leq \nu$, with an upper bound ν, based on the formula appearing in Theorem 1 (due to the restriction $\pi_{\text{thr}} \in (1/2, 1]$). For such cases where $q_\Lambda^2 > p\nu$, we could consider another range Λ' of regularization parameters leading to a smaller value $q_{\Lambda'}$ or using a larger upper bound ν' for $\mathbb{E}[V]$ such that $q_{\Lambda'}^2 \leq p\nu'$. To use (3.5), we need knowledge about q_Λ.

Trivially, q_Λ is known for variable selection procedures which select q variables: then $q_\Lambda = q$. We describe now examples of procedures which select q variables. Consider the Lasso in (2.3) for a range $\Lambda = [\lambda_{\min}, \lambda_{\max}]$ of regularization parameters. Define the Lasso-based procedure $\hat{S}_q$ selecting the q variables which enter first in the regularization path when varying from the maximal value $\lambda_{\max}$ to the minimal value $\lambda_{\min}$. Note that if there would be less than q active variables in the regularization path over the range Λ, we would select all active variables and this number is bounded by q which is sufficient for the error control in Theorem 1. Alternatively, consider the Lasso in (2.3) for a singleton $\Lambda = \{\lambda\}$. We then have an estimated active set $\hat{S}(\lambda)$. Define $\hat{S}_q$ as the procedure selecting the q variables from $\hat{S}(\lambda)$ whose regression coefficients are largest in absolute values. Typically, we would choose λ such that $|\hat{S}(\lambda)| \geq q$. If there would be less than q active variables in $\hat{S}(\lambda)$, we would select all active variables and this number is bounded by q which again is sufficient for the error control in Theorem 1. Other methods like forward selection or L_2Boosting [1] lead to selectors $\hat{S}_q$ which include the first q variables arising during the computational iterations.

Theorem 1 can be applied to any structure estimation method with a corresponding estimator $\hat{S}(\lambda)$: e.g. the method is used in [19] for estimating the edge set in a high-dimensional Gaussian graphical model, among other applications. The price to pay for this great generality is the restrictive exchangeability condition. However, it is empirically shown in [19] that the error rate $\mathbb{E}[V]$ is well under (conservative) control in situations, with real data design, where the exchangeability condition is likely to fail. Figure 2 is illustrating this fact. The number of falsely chosen variables is remarkably well controlled at the desired level, giving empirical evidence that the derived error control is useful beyond the discussed setting of exchangeability. Stability selection thus helps to select an appropriate amount of regularization such that false positive selections are under control.

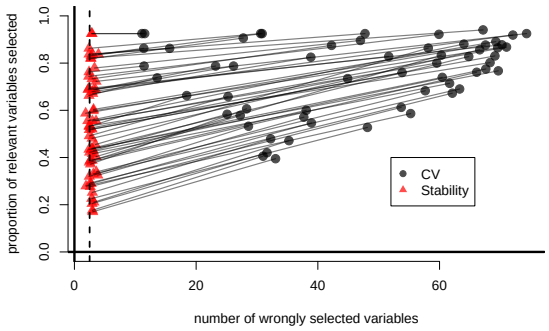


Figure 2: Comparison of stability selection with cross-validation for 64 linear models with different signal to noise ratios, sparsity and real data design matrices. The cross-validated solution (for standard Lasso) is indicated by a dot and the corresponding stability selection by a red triangle, showing the average proportion of correctly identified relevant variables versus the average number of falsely selected variables. Each pair consisting of a dot and triangle corresponds to one simulation setting. The broken vertical line indicates the value at which the number of wrongly selected variables is nominally controlled with stability selection, namely $\mathbb{E}(V) \leq 2.5$. Looking at stability selection, the error control holds up very well and the proportion of correctly identified relevant variables is very close to the CV-solution while the number of falsely selected variables is reduced dramatically. The figure is taken from [19]. A more detailed description of the simulation models is given in Appendix B.

Motif regression example
For the motif regression example in Section 3.1 we find two stable variables (motifs) when using stability selection with the Lasso and using the control $\mathbb{E}[V] \leq 1$. These variables are plotted with additional circles in Figure 1: one of them, with the second highest selection probability (lower-right panel of the figure), corresponds to a true, biologically known motif.

3.3 P-values

In many applications, particularly in the medical and health sciences, we want to assign p-values to individual tests from a generalized linear model. Furthermore, although the error control in Theorem 1 is useful, it doesn't easily translate to established measures from multiple testing like the familywise error or the false discovery rate. We will discuss in this section how this can be achieved, assuming also weaker conditions than the exchangeability condition from Theorem 1.

For ease of exposition, we consider the case of a linear model as in (2.2); the extension to generalized linear models is briefly mentioned at the end of the section. Our goal is to assign p-values for the null- and alternative hypotheses

$$H_{0,j} : \beta_j = 0; \quad H_{A,j} : \beta_j \neq 0,$$

for all $j = 1, \ldots, p$ (we could use one-sided alternatives instead).

3.3.1 Sample-splitting

An approach proposed by [26] is to split the data into two parts, reducing the dimensionality to a manageable size of predictors (keeping the important variables with high probability) using the first half of the data, and then to assign p-values and making a final selection using classical least squares estimation based on the second part of the data. Clearly, such a sample-splitting approach is related to subsampling with subsample size $\lfloor n/2 \rfloor$ used in Section 3.2.

The data are split randomly into two disjoint sets $I_1, I_2 \subset \{1, \ldots, n\}$ with $|I_1| = \lfloor n/2 \rfloor$, $I_2 = n - \lfloor n/2 \rfloor$, $I_1 \cap I_2 = \emptyset$ and hence $I_1 \cup I_2 = \{1, \ldots, n\}$. Thus, the corresponding

data sub-samples are $(\mathbf{X}_{I_1}, \mathbf{Y}_{I_1})$ and $(\mathbf{X}_{I_2}, \mathbf{Y}_{I_2})$. Let $\hat{S}$ be a variable selection or screening procedure. We denote by $\hat{S}(I_1)$ the set of selected predictors based on $(\mathbf{X}_{I_1}, \mathbf{Y}_{I_1})$ which may include the choice of potential tuning or regularization parameters. A prime example for variable selection or screening is the Lasso in (2.3). The regression coefficients and the corresponding p-values $\tilde{P}_1, \ldots, \tilde{P}_p$ of the selected predictors are determined based on the other half of the data $(\mathbf{X}_{I_2}, \mathbf{Y}_{I_2})$ by using ordinary least squares estimation and the corresponding t-tests on the set of variables from $\hat{S}(I_1)$, i.e.,

$$\tilde{P}_j = \begin{cases} P_{\text{raw},j} \text{ based on } \mathbf{Y}_{I_2}, \mathbf{X}_{I_2,\hat{S}(I_1)} & , \text{ if } j \in \hat{S}(I_1), \\ 1 & , \text{ if } j \notin \hat{S}(I_1), \end{cases} \tag{3.6}$$

where $P_{\text{raw},j}$ is the p-value from the two-sided t-test, using least squares estimation, for $H_{0,j}$ (based on the second half of the data I_2 and using only the variables in $\hat{S}(I_1)$). If the selected set of variables contains the true model S_0, i.e., $\hat{S}(I_1) \supseteq S_0$, the p-values $\tilde{P}_j$ are controlling the (single testing) type I error, assuming Gaussian errors ε_i and $\text{rank}(\mathbf{X}_{I_2,\hat{S}(I_1)}) = |\hat{S}(I_1)|$, where $\mathbf{X}_{I_2,\hat{S}(I_1)}$ is the design sub-matrix with rows corresponding to I_2 and columns corresponding to $\hat{S}(I_1)$. The assumption on the rank is most often fulfilled if $|\hat{S}(I_1)| < n/2$. Finally, each p-value $\tilde{P}_j$ is adjusted by a factor $|\hat{S}(I_1)|$ to correct for the multiplicity of the testing problem:

$$\tilde{P}_{\text{corr},j} = \min(\tilde{P}_j \cdot |\hat{S}(I_1)|, 1) \ \ (j = 1, \ldots p). \tag{3.7}$$

We make the following assumptions:

$$\lim_{n\to\infty} \mathbb{P}[\hat{S}_{\lfloor n/2 \rfloor} \supseteq S_0] = 1. \tag{3.8}$$

Furthermore, we assume

$$\lim_{n\to\infty} \mathbb{P}[|\hat{S}_{\lfloor n/2 \rfloor}| < n/2] = 1. \tag{3.9}$$

Here, $\hat{S}_m$ denotes any variable selection procedure based on m observations. For any subset $I_{(m)} \subset \{1, \ldots, n\}$ with $|I_{(m)}| = m = n - \lfloor n/2 \rfloor$, let $\hat{\Sigma}(I_{(m)}) = m^{-1}\mathbf{X}_{I_{(m)}}^T \mathbf{X}_{I_{(m)}}$ and $\hat{\Sigma}(I_{(m)})_{S,S}$ be the $|S| \times |S|$ sub-matrix corresponding to rows and columns of the subset $S \subset \{1, \ldots, p\}$. We assume:

$$\Lambda^2_{\min}(\hat{\Sigma}(I_{(m)})_{S,S}) > 0 \text{ for all } S \text{ with } |S| < n/2, \text{ for all } I_{(m)}, \tag{3.10}$$

where $\Lambda^2_{\min}(A)$ denotes the minimal eigenvalue of a symmetric matrix A. Under conditions (3.8), (3.9) and (3.10), one can easily show that the p-values in (3.7) control the familywise error rate (FWER) which is defined as $\mathbb{P}[V > 0]$, the probability of making at least one false rejection where V denotes the number of false selections (i.e. false positives). Such a result is implicitly contained in [26]. Some discussion about the conditions used is given below.

The single data-splitting method for the p-values in (3.7) is easy to implement. It relies, however, on an arbitrary split of the data into I_1 and I_2. Results, at least for finite samples, can change drastically if this split is chosen differently. This in itself is unsatisfactory since results are not reproducible, as illustrated in Figure 3.

3.3.2 Multi sample-splitting and familywise error control

An obvious alternative and improvement to a single arbitrary sample split is to divide the sample repeatedly [20]. In contrast to the "p-value lottery" phenomenon illustrated in Figure 3, the multi sample split method makes results reproducible, at least approximately if the number of random splits is chosen to be sufficiently large. Moreover, we will show empirically that, maybe unsurprisingly, the resulting procedure is more powerful than the single-split method, see Section 3.3.4. The multi sample split method is defined as follows.

For $b = 1, \ldots, B$:

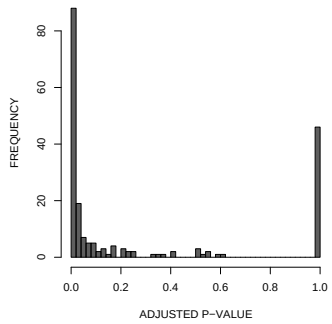


Figure 3: Histogram of adjusted p-values $\tilde{P}_{\mathrm{corr},j}$ for a single variable in the motif regression example of Section 3.1 with $n = 287$ and $p = 195$: the different p-values correspond to different random splits of the data into I_1 and I_2. Due to the high variability, we call the phenomenon a "p-value lottery". The figure is taken from [20].

1. Randomly split the original data into two disjoint groups $I_1^{[b]}$ and $I_2^{[b]}$ of (almost) equal size.
2. Using only $I_1^{[b]}$, estimate the set of active predictors $\hat{S}^{[b]} = \hat{S}(I_1^{[b]})$.
3. Compute the adjusted (non-aggregated) p-values as in (3.7), i.e.,

$$\tilde{P}_{\mathrm{corr},j}^{[b]} = \min(\tilde{P}_j^{[b]} \cdot |\hat{S}^{[b]}|, 1) \ \ (j = 1, \ldots, p)$$

   where $\tilde{P}_j^{[b]}$ is based on the two-sided t-test, as in (3.6), based on $I_2^{[b]}$ and $\hat{S}^{[b]} = \hat{S}(I_1^{[b]})$.

Finally, we aggregate over the B p-values $\tilde{P}_{\mathrm{corr},j}^{[b]}$, as discussed next.

3.3.3 Aggregation over multiple p-values

The procedure described above leads to a total of B p-values for each covariate $j = 1, \ldots, p$. For each $j = 1, \ldots, p$, aggregation of the p-values $\tilde{P}_{\mathrm{corr},j}^{[b]}$ over the indices $b = 1, \ldots, B$ can be done using empirical quantiles. For $\gamma \in (0, 1)$ define

$$Q_j(\gamma) = \min\left\{q_\gamma\big(\{\tilde{P}_{\mathrm{corr},j}^{[b]}/\gamma;\ b = 1, \ldots, B\}\big), 1\right\}, \tag{3.11}$$

where $q_\gamma(\cdot)$ is the empirical γ-quantile function.

A p-value for each variable $j = 1, \ldots, p$ is then given by $Q_j(\gamma)$, for any fixed $0 < \gamma < 1$. We will describe in Section 3.3.4 that this is an asymptotically correct p-value for controlling the familywise error rate.

A proper selection of γ may be difficult. Error control is not guaranteed anymore if we search for the best value of γ. But we can use instead an adaptive version which selects a suitable value of the quantile based on the data. Let $\gamma_{\min} \in (0, 1)$ be a lower bound for γ, typically 0.05, and define

$$P_j = \min\left\{\big(1 - \log\gamma_{\min}\big) \inf_{\gamma \in (\gamma_{\min}, 1)} Q_j(\gamma), 1\right\} \ \ (j = 1, \ldots, p). \tag{3.12}$$

The extra correction factor $1 - \log\gamma_{\min}$ ensures that the familywise error rate remains controlled despite of the adaptive search for the best quantile, as described in Theorem 2 in Section 3.3.4. For the recommended choice of $\gamma_{\min} = 0.05$, this factor equals $1 - \log(0.05) \approx 3.996$.

3.3.4 Control of familywise error

The resulting adjusted p-values P_j $(j = 1, \ldots, p)$ from (3.12) can be used for both familywise error (FWER) and false discovery rate (FDR) control. For FWER control at level $\alpha \in (0, 1)$, simply all p-values below α are rejected and the selected subset is

$$\hat{S}_{\text{multi-split FWER}}(\alpha) = \{j : P_j \leq \alpha\}. \tag{3.13}$$

Denote by $V_{\text{multi-split FWER}}(\alpha) = |\hat{S}_{\text{multi-split FWER}}(\alpha) \cap S_0^c|$ the number of false positive selections.

Theorem 2. *[20] Consider a linear model as in (2.2) with fixed design and Gaussian errors and assume that (3.8), (3.9) and (3.10) hold. Furthermore, the number B of random splits in the multi-split method is fixed. Then, for any $\gamma_{\min} \in (0, 1)$ (see (3.12)),*

$$\limsup_{n\to\infty} \mathbb{P}[V_{\text{multi-split FWER}}(\alpha) > 0] \leq \alpha.$$

For the Lasso, the screening property in (3.8) and (3.9) hold assuming sparsity, a compatibility or restricted eigenvalue condition on the design (which is weaker than the irrepresentable condition, cf. [25]), and a "beta-min" condition as in (2.5). More details can be found in [3, Cor.7.6,Chs.6-7&11]. Furthermore, (3.10) is a very weak assumption on the design. It is worth pointing out that for assigning p-values controlling the FWER, we require (much) weaker assumptions than the exchangeability condition in Theorem 1; the reason is due to the fact that we focus here on a specific model, i.e., a linear model, whereas Theorem 1 applies to much more general settings.

Simulation study [20]
We simulate data from 16 different linear models as in (2.2) with $n = 100$ and $p = 1000$. A more detailed description is given in Appendix B. As initial variable selection or screening method $\hat{S}$ we use three approaches which are all based on the Lasso. The first one, denoted by $\hat{S}_{\text{fixed}}$, uses the Lasso and selects those $\lfloor n/6 \rfloor$ variables which appear most often in the regularization path when varying the penalty parameter. The constant number of $\lfloor n/6 \rfloor$ variables is chosen, somewhat arbitrarily, to ensure a reasonably large set of selected coefficients on the one hand and to ensure, on the other hand, that least squares estimation will work reasonably well on the second half of the data with sample size $n - \lfloor n/2 \rfloor$. The second method $\hat{S}_{\text{CV}}$ is more data-driven: it uses the Lasso with penalty parameter chosen by 10-fold cross-validation and selecting the variables whose corresponding estimated regression coefficients are different from zero. The third method, $\hat{S}_{\text{adapt}}$ is the adaptive Lasso [30], with the Lasso solution used as initial estimator for the adaptive Lasso, and where the regularization parameters are chosen based on 10-fold cross-validation. The selected variables are again the ones whose corresponding estimated regression parameters are different from zero. The number of random splits in the multi sample split method is always chosen as $B = 100$.

Results are shown in Figure 4 with the default value $\gamma_{\min} = 0.05$ in (3.12). Using the multi sample split method, the average number of true positives (the variables in S_0 which are selected) is typically slightly increased while the FWER (the probability of selecting variables in S_0^c) is reduced sharply. The asymptotic control seems to give a good control in finite sample settings with the multi sample split method. The single sample split method, in contrast, selects in nearly all cases too many noise variables, exceeding the desired FWER sometimes substantially. This suggests that the error control for finite sample sizes works much better for the multi sample split method yet with a larger number of true discoveries. Furthermore, the multi sample split p-value in (3.12) typically leads to *conservative* error control: this fact has to do with the p-value aggregation in (3.11) and (3.12) which asymptotically guarantees that the FWER is bounded by (but typically not equal to) a pre-specified level α.

Motif regression example
We apply the multi sample split method to the real data example in Section 3.1. We use the multi sample split method with the adaptive Lasso $\hat{S}_{\text{adapt}}$ as described above. The multi sample split

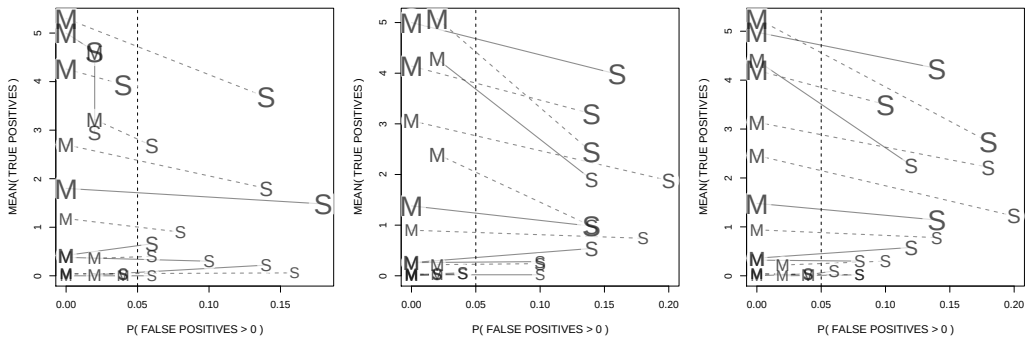


Figure 4: Simulation results for 16 linear models with $n = 100$ and $p = 1000$ (see Appendix B). Average number of true positives vs. the familywise error rate (FWER) for the single split method ("S") against the multi-split version ("M"). FWER is controlled (asymptotically) at $\alpha = 0.05$ for both methods and this value is indicated by a broken vertical line. From left to right are results for $\hat{S}_{\text{fixed}}$, $\hat{S}_{\text{CV}}$ and $\hat{S}_{\text{adapt}}$. Results of a single scenario are joined by a line, which is solid if the regression coefficients follow the "uniform" sampling and broken otherwise (see Appendix B). Increasing signal to noise ratio is indicated by increasing symbol size. The figure is taken from [20].

method identifies one variable at the 5% significance level with an adjusted p-value of 0.0059. The single sample split method is not able to identify a single significant predictor. In view of the asymptotic error control in Theorem 2, and empirical results shown in Figure 4, there is substantial evidence that the single selected variable is a truly relevant variable. The significant variable is the one with second largest absolute value for the estimated regression coefficient (upper-left panel in Figure 1) and second largest value of the subsampling probability (lower-right panel in Figure 1). Interestingly, the variable corresponding to the largest values did not turn out to be significant: this may be due to the fact that the proposed p-value method is conservative. For this specific application though, it seems desirable to pursue a conservative approach with stringent FWER control as biological follow-up experiments are laborious and expensive.

3.3.5 Complementary remarks

The multi sample split method for p-values can be adapted to control the false discovery rate instead of the FWER. Details and empirical results are given in [20]. Finally, the conceptual methodology can be extended for constructing p-values in generalized linear models: the p-values based on the second half of the data I_2 then rely on classical (e.g. likelihood ratio) tests applied to the selected submodel.

4 Causal effects: a different target for variable selection

The set of active variables S_0 in a linear or generalized linear model describes the variables with corresponding non-zero parameters in the model: thus, in a (generalized) regression model, these are the variables which have an association with the response variables Y. For many applications, we would like to infer the variables which have a causal effect on the response variable Y, based on observational data. This is the topic of this section.

4.1 A brief introduction to causal analysis

We consider the setting where we have a continuous response variable Y and p continuous covariates $X = (X^{(1)}, \ldots, X^{(p)})$. The model is

$$\begin{aligned} &X, Y \sim P^0 = \mathcal{N}_{p+1}(\mu^0, \Sigma^0), \\ &P^0 \text{ is faithful with respect to a causal DAG } G^0. \end{aligned} \tag{4.1}$$

In words, the assumption means that the variables $X^{(1)}, \ldots, X^{(p)}, Y$ are related to each other with a true underlying "causal influence diagram" which is here formalized as a directed acyclic graph (DAG) G^0. Furthermore, these variables have a joint Gaussian distribution which satisfies the Markov property with respect to the DAG G^0 and all marginal and conditional independencies can be read-off from the graph G^0: the latter is the faithfulness assumption, cf. [23]. We assume that the observational data are realizations

$$(X_1, Y_1), \ldots, (X_n, Y_n) \text{ i.i.d. from model (4.1).} \tag{4.2}$$

The notion of intervention is useful to describe and define causal effects. We can think of doing an intervention at a variable, say $X^{(j)}$, by setting it to a (deterministic) value u: using the notation and calculus developed by [21], we then write $\mathrm{do}(X^{(j)} = u)$. We are interested in the distribution of the response Y when doing an intervention at a single variable (for simplicity, we only consider here single variable interventions): $P(Y|\mathrm{do}(X^{(j)} = u))$. When emphasizing the true intervention distribution, we write $P^0(Y|\mathrm{do}(X^{(j)} = u))$. Using the so-called truncated factorization, such an intervention distribution can be computed from the distribution P^0 *and* knowledge of the true graph G^0, that is

$$P^0(Y|\mathrm{do}(X^{(j)} = u)) = P^0_{G^0}(Y|\mathrm{do}(X^{(j)} = u))$$

is a function of the true underlying P^0 and G^0. For details, we refer to [21].

The causal effect, or intervention effect, is then defined as

$$\frac{\partial}{\partial u}\mathbb{E}_{P^0,G^0}[Y|\mathrm{do}(X^{(j)} = u)] \equiv \theta^0_j \;\; (j = 1, \ldots, p),$$

where we used the fact that in the Gaussian case, the derivative is constant and hence, the causal effect is a real-valued parameter.

The following characterization of the causal effect is useful. In the true underlying causal DAG G^0, denote by $\mathrm{pa}(j)$ the parental set of the vertex j which corresponds to the variable $X^{(j)}$, i.e., $\mathrm{pa}(j) = \{k;\ \text{there is a directed edge from } k \to j,\ k = 1, \ldots, p+1 \setminus j\}$. Here and in the sequel, the response variable Y corresponds to the index $p+1$. Then, the causal effect of $X^{(j)}$ on Y can be characterized as a regression coefficient when conditioning on the right variables. Consider the linear regression

$$Y = \mu_j + \beta_j X^{(j)} + \sum_{k \in \mathrm{pa}(j)} \beta_k X^{(k)} + V_j \;\; (j = 1, \ldots, p)$$

with intercept μ_j and error term $\mathbb{E}[V_j] = 0$. Then,

$$\theta^0_j = \begin{cases} \beta_j & , \text{ if } p+1 \notin \mathrm{pa}(j) \\ 0 & , \text{ if } p+1 \in \mathrm{pa}(j) \end{cases} \quad (j = 1, \ldots, p). \tag{4.3}$$

That is, the causal effect equals the regression parameter when conditioning on $X^{(\mathrm{pa}(j))}$. We see explicitly from (4.3) that the causal effect θ^0_j depends on P^0 and the underlying DAG G^0; the dependence on the latter arises since the parental set $\mathrm{pa}(j)$ is a function of the DAG G^0.

4.1.1 Identifiability from observational data

Observational data as in (4.2) do not involve any interventions. It is well known that it is impossible to infer the true underlying DAG G^0 from observational data. Instead, one can only identify the Markov equivalence class,

$$\mathcal{M}(G^0) = \mathcal{M}(P^0) = \{G;\ G \text{ a DAG which is Markov equivalent to } G^0\}.$$

The notation $\mathcal{M}(G^0) = \mathcal{M}(P^0)$ indicates that the Markov equivalence class depends on either G^0 or P^0 only, assuming faithfulness of P^0 w.r.t. G^0 (the set of conditional (in-)dependencies

among the variables is then described by G^0 or by P^0), cf. [21, 23]. This equivalence class can be encoded by a partially directed graph, the so-called CPDAG, which we denote by $\mathcal{E}(G^0) = \mathcal{E}(P^0)$; the letter "$\mathcal{E}$" indicates that the CPDAG is sometimes also referred to as essential graph. Consequently, one cannot infer in general all causal effects θ_j^0 $(j = 1, \ldots, p)$ from observational data.

However, it is possible to identify good lower bounds for causal effects from observational data. Suppose that $\mathcal{M}(G^0) = \mathcal{M}(P^0)$ consists of m_0 different DAG members $G_1, \ldots, G_{m_0}$: $\mathcal{M}(G^0) = \mathcal{M}(P^0) = \{G_1, \ldots, G_{m_0}\}$ (m_0 depends on G^0 or P^0). Consider the set of potential causal effects among all members in $\mathcal{M}(G^0) = \mathcal{M}(P^0)$:

$$\{\theta^0_{G_r;j};\ \theta^0_{G_r;j} \text{ a causal effect of } X^{(j)} \text{ on } Y \text{ in the DAG } G_r,\ r = 1, \ldots, m_0,\ j = 1, \ldots, p\}.$$

We denote the whole set by

$$\Theta^0 = \{\theta^0_{G_r;j};\ r = 1, \ldots, m_0,\ j = 1, \ldots, p\} \tag{4.4}$$

which can be written as a function of the observational distribution P^0 only (using the Markov equivalence class $\mathcal{M}(P^0)$ and P^0 for regression coefficients). A lower bound for the absolute value of the true causal effect is then given by

$$\ell_j^0 = \min_{r=1,\ldots,m_0} |\theta^0_{G_r,j}| \ \ (j = 1, \ldots, p). \tag{4.5}$$

Again, these lower bounds are identifiable from observational data, i.e., from the distribution P^0. In addition, if the true graph G^0 is sparse with many so-called protected edges, many of the $\theta^0_{G_r;j}$ $(r = 1, \ldots, m)$ are the same and the bound in (4.5) is rather tight and good. We will show in Section 4.3 a real data example where the estimated version of (4.5) leads to clearly better results than using regression techniques.

4.2 Estimation from data

A procedure for consistently estimating the lower bound values ℓ_j^0 in (4.5) based on observational data only is given in [16]. The method is called IDA, standing for **I**ntervention-calculus when the **D**AG is **A**bsent [15] which reflects that we do the estimation without knowledge of the underlying causal DAG G^0. It consists of two main steps: first, estimating the CPDAG $\mathcal{E}(G^0) = \mathcal{E}(P^0)$ which encodes the Markov equivalence class, and it then proceeds to estimate the lower bounds.

4.2.1 The PC-algorithm for the CPDAG

The PC-algorithm is named after its inventors **P**eter Spirtes and **C**larke Glymour [23]. It is a clever hierarchical scheme for testing conditional independencies among variables $X^{(j)}, X^{(k)}$ (all $j \neq k$) and among $X^{(j)}, Y$ (all j) in the DAG. The first level in the hierarchy are marginal correlations, then partial correlations of low and then higher order are tested to be zero or not. Due to the faithfulness assumption in model (4.1) and assuming sparsity of the DAG (in terms of maximal neighborhood size of the nodes), the algorithm is computationally feasible for problems where p is in the thousands. It is interesting to note that we can use a simplified version of the PC-algorithm for estimating the active set S_0 in a linear model (2.2) and that this estimator is quite comparable in accuracy with the Lasso and versions of it [2]. Furthermore, [27] presents another variant looking only at marginal and partial correlations of order one.

The PC-algorithm involves one tuning parameter, denoted by α, which can be interpreted as the significance level of a single partial correlation test. The output of the algorithm is an estimated CPDAG, denoted by $\hat{\mathcal{E}}_n(\alpha)$. The following result holds.

Theorem 3. *[12] Consider data as in (4.2) from model (4.1) where the dimension $p = p_n$ is allowed to grow much faster than sample size as $n \to \infty$. Under some assumptions described in Appendix A on sparsity, on incoherence of partial correlations and a "beta-min" analogue for the size of non-zero partial correlations,*

$$\mathbb{P}[\hat{\mathcal{E}}_n(\alpha_n) = \mathcal{E}(G^0) = \mathcal{E}(P^0)] \to 1 \ (n \to \infty),$$

where $\alpha_n \to 0$ at a suitable rate.

The rate of $\alpha_n \to 0$ in Theorem 3 depends on unknown quantities like the sparsity and the lower bound in the "beta-min" condition. And thus, choosing an appropriate value for α is not straightforward at all (since e.g. cross-validation with the Gaussian log-likelihood is primarily tailored for estimating the distribution P^0 and not for inferring the structure of the CPDAG).

4.2.2 The local algorithm for the lower bound values

Once we have an estimate $\hat{\mathcal{E}}(\alpha)$ of the CPDAG, we can in principle enumerate all its DAG members $\hat{G}_1, \ldots, \hat{G}_{\hat{m}}$ and then estimate the values $\theta^0_{\hat{G}_r;j}$ in (4.4) using estimated coefficients from linear regression as in (4.3). In short, we obtain estimates

$$\hat{\Theta} = \{\hat{\theta}_{\hat{G}_r,j};\ r = 1, \ldots, \hat{m},\ j = 1, \ldots, p\}, \tag{4.6}$$

and on the event $\hat{\mathcal{E}}(\alpha) = \mathcal{E}(G^0) = \mathcal{E}(P^0)$ (see Theorem 3), these estimates are for the true values Θ^0 in (4.4). In general, the set Θ^0 in (4.4) and $\hat{\Theta}$ in (4.6) are multi-sets, where some elements may take exactly the same values. We denote the set of distinct values in such multi-sets by ${}^{\text{set}}\Theta^0$ and ${}^{\text{set}}\hat{\Theta}$, respectively.

It can be a computationally horrendous and infeasible task to enumerate all DAG members in $\hat{\mathcal{E}}(\alpha)$. An algorithm is presented in [16] which works on local aspects of $\hat{\mathcal{E}}(\alpha)$, exploiting also the fact that a causal effect is only a function of the parental set instead of the whole graph, see (4.3). Such a procedure is computationally feasible to obtain all the estimated potential causal effects in $\hat{\Theta}$, and one can then obtain estimated lower bounds $\hat{\ell}_j$. As we show in Theorem 4, the local algorithm asymptotically finds the correct set ${}^{\text{set}}\Theta^0$. Of course, ${}^{\text{set}}\hat{\Theta} = {}^{\text{set}}\hat{\Theta}(\alpha)$ and $\hat{\ell}_j = \hat{\ell}_j(\alpha)$ depend on the tuning parameter α used in the PC-algorithm for an estimate of the CPDAG $\hat{\mathcal{E}}(\alpha)$. We note that $\hat{\theta}_{\hat{G}_r,j}$ and $\hat{\ell}_j$ may be zero, if the second case in (4.3) applies (for the estimated DAG $\hat{G}_r$).

Theorem 4. *[16] Consider data as in (4.2) from model (4.1) where the dimension $p = p_n$ is allowed to grow much faster than sample size as $n \to \infty$. Under some assumptions described in Appendix A on sparsity, on incoherence of partial correlations and a "beta-min" analogue for the size of non-zero partial correlations,*

$$\mathbb{P}[{}^{\text{set}}\hat{\Theta}(\alpha_n) = {}^{\text{set}}\Theta^0] \to 1\ (n \to \infty),$$

where $\alpha_n \to 0$ at a suitable rate. Furthermore, we also have

$$\sup_{j=1,\ldots,p} |\hat{\ell}_j(\alpha_n) - \ell^0_j| = o_P(1)\ (n \to \infty).$$

4.3 Effects of single gene knock-downs on all other genes in yeast

The example described in this section has been presented in [15]. We consider a data set from [11] on gene expressions from yeast. There are 5360 genes and there is a set of $n = 63$ observational gene expression data. Furthermore, there are 234 gene knock down experiments, each of them involving an intervention at a single gene and measuring the expression of all the genes. This fits into the framework presented above: we can think of responses where $Y = X^{(1)}$ with covariates $(X^{(2)}, \ldots, X^{(5360)})$, and then of another response $Y = X^{(2)}$ and covariates $(X^{(1)}, X^{(3)}, \ldots, X^{(5360)})$, and so on. Thus, $p = 5359$ and we have 5360 different response settings. For all these response settings, we can quantify the intervention effect of a covariate j (gene j) to the response under consideration. We simply use the estimated lower bounds $\hat{\ell}_{Y=k;j}$ (of the covariate $X^{(j)}$ for response $Y = X^{(k)}$). For comparison, we use high-dimensional regression techniques: although conceptually wrong, one can try them to "quantify the importance" of variables. Thanks to the available 234 intervention experiments, which play the role of a test set,

we know in good approximation the true intervention effects of 234 variables (genes) on all other genes: with a threshold, we can assign whether the variable has an effect or not. And therefore, we can measure the performance of the methods with an ROC curve, presented in Figure 5. We see that the IDA method described above is substantially better than the (conceptually wrong) Lasso or elastic net [31] regression method. Furthermore, the regression methods do hardly better than random guessing. Thus, we see for this example, that regression doesn't extract any useful information for selecting variables which have an intervention or causal effect; and on the other hand, the strong effects can be much better estimated with IDA (i.e. in the lower left corner of the ROC plot).

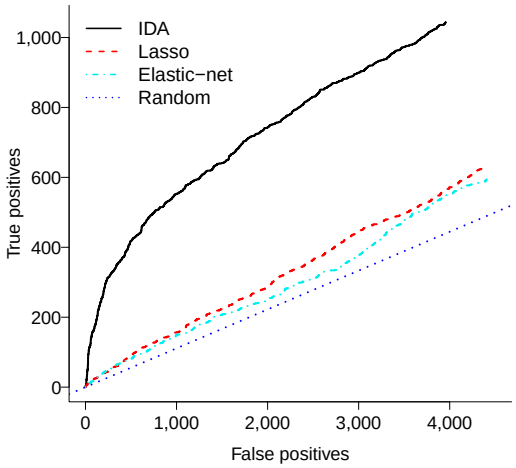


Figure 5: Intervention effects among 5360 genes in yeast. ROC-type curve with false positives (x-axis) and true positives (y-axis). IDA graphical modeling based method (black solid line), Lasso (dashed red line), elastic net (dash-dotted light blue line) and random guessing (fine dotted dark blue line). Observational data used for training has sample size $n = 63$, and there are 234 intervention experiments to validate the methods. The figure is taken from [15].

4.4 Stability selection and p-values for bounds of causal effects

As discussed above, we can identify the lower bound parameters ℓ_j^0 from the observational distribution P^0. Our goal is to infer stable causal variables and assign p-values.

4.4.1 Stability selection for causal variables

We consider the IDA algorithm yielding the estimated values $\hat{\ell}_j(\alpha)$ $(j = 1, \ldots, p)$. Then, define the top q variables as

$$\hat{S}_{\mathrm{IDA}}(q, \alpha) = \{j;\ \hat{\ell}_j(\alpha) \text{ is among the largest } q \text{ from all } \{\hat{\ell}_1(\alpha), \ldots, \hat{\ell}_p(\alpha)\}\}.$$

Thus $\hat{S}_{\mathrm{IDA}}(q, \alpha)$ is a variable selection procedure, targeted for causal variables. The set of stable variables is then defined as in (3.2), using subsampling and the selection method $\hat{S}_{\mathrm{IDA}}(q, \alpha)$. We can apply Theorem 1, implicitly assuming that $\hat{S}_{\mathrm{IDA}}(q, \alpha)$ is better than random guessing and that the exchangeability condition holds (which is a very restrictive assumption, but see the discussion in Section 3.2.1).

4.4.2 P-values for causal variables

Here, we are interested in assigning p-values for the null- and alternative hypotheses

$$H_{0,j} : \ell_j^0 = 0; \quad H_{A,j} : \ell_j^0 > 0.$$

The strategy is analogous to Section 3.3 using sample splitting. For a single sample split, we have the first and second half-samples I_1 and I_2. Based on I_1, we use the PC-algorithm to estimate the CPDAG $\hat{\mathcal{E}}_\alpha(I_1)$ with its corresponding Markov equivalence class containing the DAGs $\{\hat{G}_1(I_1), \ldots, \hat{G}_{\hat{m}}(I_1)\}$. Using the local algorithm on the estimated DAG structures $\hat{G}_r(I_1)$, we obtain estimates $\hat{\theta}_{\alpha,\hat{G}_r,j}(I_2)$ and corresponding p-values $P_{\mathrm{raw};r,j}$ from two-sided t-tests in Gaussian least squares regressions, see (4.3), for the null-hypotheses that $\theta^0_{G^0_r,j} = 0$ (on the event where $\hat{\mathcal{E}}_\alpha(I_1) = \mathcal{E}(G^0) = \mathcal{E}(P^0)$). From these, we obtain raw p-values $P_{\mathrm{raw};j}$ for the null-hypotheses that $\ell^0_j = \min_r |\theta^0_{G^0,r,j}| = 0$: we simply take

$$P_{\mathrm{raw};j} = \max_r P_{\mathrm{raw};r,j}.$$

As described in (4.3) and before Theorem 4, the lower bound estimates $\hat{\ell}_j(\alpha)$ (which depend on I_1 due to the estimated CPDAG $\hat{\mathcal{E}}_\alpha(I_1)$ and on I_2 due to estimated regression coefficients) can be exactly zero: the zeroes depend on I_1, the structure of the estimated CPDAG $\hat{\mathcal{E}}_\alpha(I_1)$, only. We define the set

$$\hat{S}(I_1) = \hat{S}_\alpha(I_1) = \{j;\ Y \notin \widehat{\mathrm{pa}}(j)\} = \{j;\ \hat{\ell}_j(I_1, I_2) \neq 0\}.$$

Now, we can proceed as in Section 3.3. Define, analogously as in (3.6),

$$\tilde{P}_j = \begin{cases} P_{\mathrm{raw},j} & , \text{ if } j \in \hat{S}(I_1), \\ 1 & , \text{ if } j \notin \hat{S}(I_1). \end{cases}$$

As in (3.7), we build

$$\tilde{P}_{\mathrm{corr},j} = \min(\tilde{P}_j \cdot |\hat{S}(I_1)|, 1) \ \ (j = 1, \ldots p).$$

And then, we can use multi sample-splitting and aggregation of p-values for deriving final p-values P_j $(j = 1, \ldots, p)$, exactly as described in Section 3.3.

From an asymptotic point of view, assuming the conditions in Theorem 4 (for sample size $\lfloor n/2 \rfloor$), these p-values control the familywise error rate. However, we point out that such a rough analysis assumes substantially more than for regression: the estimated CPDAG $\hat{\mathcal{E}}_\alpha(I_1)$ should be equal to the true $\mathcal{E}(G^0) = \mathcal{E}(P^0)$ which is a very ambitious task with finite data. This is in contrast to generalized regression, where the structure estimation (based on the first half-sample) only needs to satisfy the screening property (3.8) where the estimated active set $\hat{S} \supseteq S^0$ contains the true relevant variables. Such a requirement is much more realistic to hold (approximately) with finite data.

4.5 Empirical results

In the sequel, we evaluate the performance of methods with respect to the true causal effects of a covariate $X^{(j)}$ on a response Y, denoted by θ^0_j. Although the true causal effects are not identifiable from the observational distribution P^0, it is interesting to see how well we can infer the true effects (but we cannot separate any more the effect of a method for error control and power from the effect of non-identifiability of a causal effect). As a consequence, the number of positives $|\{\theta^0_j \neq 0;\ j\}|$ will be smaller than for the lower bounds $|\{\ell^0_j \neq 0;\ j\}|$ and hence, the number of false and true positives w.r.t. to the θ^0_j's will be smaller than for the lower bounds.

4.5.1 Simulations

We simulate Gaussian data according to a DAG: the corresponding structural linear equations are

$$\begin{aligned} X^{(1)} &= \epsilon^{(1)} \sim \mathcal{N}(0, 1), \\ X^{(j)} &= \sum_{r=1}^{j-1} B_{jr} X^{(r)} + \varepsilon^{(j)} \ \ (j = 2, \ldots, d), \end{aligned} \tag{4.7}$$

where $\varepsilon^{(1)}, \ldots, \varepsilon^{(d)}$ i.i.d. $\sim \mathcal{N}(0,1)$, independent of $\{X^{(r)};\ r \leq i-1\}$ and B is a matrix generated with non-zero elements according to a DAG with (fixed) values from realizations of a Uniform($[0.5, 0.6]$) distribution. We consider sparse and fairly dense graphs, as shown in Figures 10 and 11 in Appendix B, and we vary n and d as follows:

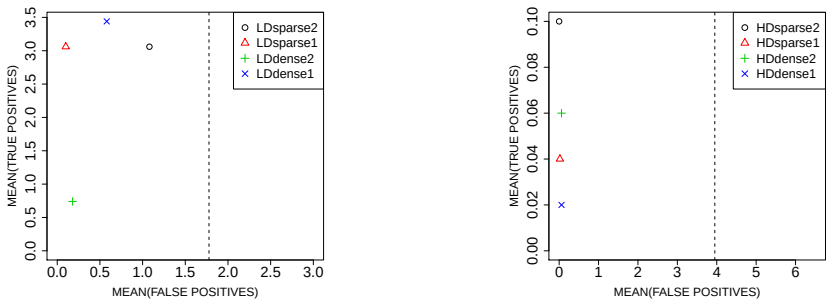
(LD) Data simulated according to the DAGs in Figure 10 with $d = 10$ and $n = 500$.

(HD) Data simulated according to the DAGs in Figure 11 with $d = 50$ and $n = 50$.

Stability selection for control of expected number of false positives

For the model in (4.7), every variable $X^{(k)}$ $(k = 1, \ldots, d)$ serves once as a response variable and all others $\{X^{(j)};\ j \neq k\}$ as covariates (and hence $p = d - 1$). We consider the entire estimation for the causal effect $\theta^0_{Y=k,j}$ of $X^{(j)}$ on $Y = X^{(k)}$. The active set is defined as $S_0 = \{(j,k);\ \theta^0_{Y=k,j} \neq 0,\ j \neq k\}$ and a false positive selection is defined with respect to this S_0 (i.e. selecting a causal effect which is an element of S_0^c). The estimated active set is $\hat{S}_{\mathrm{IDA}}(q, \alpha) = \{(j \neq k);\ \hat{\ell}_{Y=k,j}(\alpha)$ is among the largest q from all $\{\hat{\ell}_{Y=k',j'}(\alpha); j' \neq k'\}\}$, analogous as in Section 4.4.1. We choose $q = \lfloor \sqrt{0.8d(d-1)} \rfloor$ and $\alpha = 0.05$; furthermore, we use as stability threshold $\pi_{\mathrm{thr}} = 0.7$ and $\pi_{\mathrm{thr}} = 0.7$, see (3.2), resulting in bounds for the expected number of false positive selections $\mathbb{E}[V] \leq 1.7$ (for low-dimensional case) and $\mathbb{E}[V] \leq 4.0$ (for high-dimensional case), respectively.

Results are displayed in Figure 6. We see that stability selection works well for conservatively controlling the expected number of false positives. This finding is consistent with Theorem 1, although the exchangeability condition presumably does not hold for the model in (4.7). In terms of power for detecting true positives, stability selection is rather poor, especially for the high-dimensional case where stable selections occur only rarely (but if there is a stably selected variable, it is "almost surely" a true positive in the considered high-dimensional scenarios).



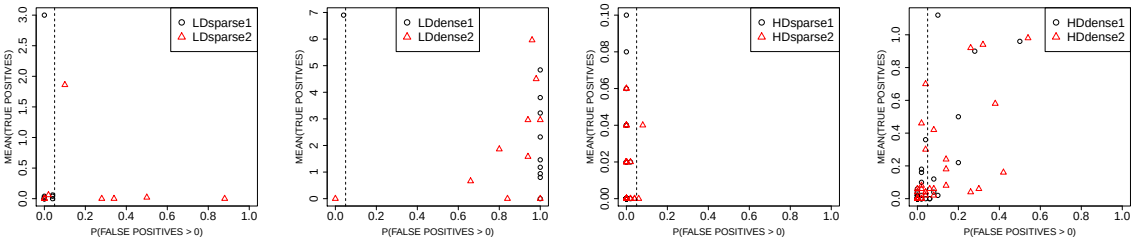
(a) Low-dimens. (LD): $d = 10$ (b) High-dimens. (HD): $d = 50$

Figure 6: Stability selection for true causal effects (with active set S_0 as described in the text). The dashed vertical line indicates the bound for $\mathbb{E}[V]$. The different model specifications in the legend are specified in Appendix B, Figures 10 and 11.

P-values for familywise error control

Here, we consider the problem of assigning p-values for every setting where $Y = X^{(k)}$ and the covariates are $\{X^{(j)};\ j \neq k\}$, $k = 1, \ldots, d$, see above. Note that for stability selection above, we used a "global" view among all these p different settings. We then use the approach described in Section 4.4.2 with $\alpha = 0.05$ for the PC-algorithm.

Results about FWER control are summarized in Figure 7. They represent a rather negative result. Only if the graph is sparse and $d = 10$, the method works. Although one can prove that the p-values are asymptotically controlling the FWER, under assumptions which are fulfilled by the simulation model, the considered sample sizes are much too small: when pushing sample size to $n \geq 5'000$ for the cases with $d = 10$, the method is reliably controlling the FWER (not shown here).



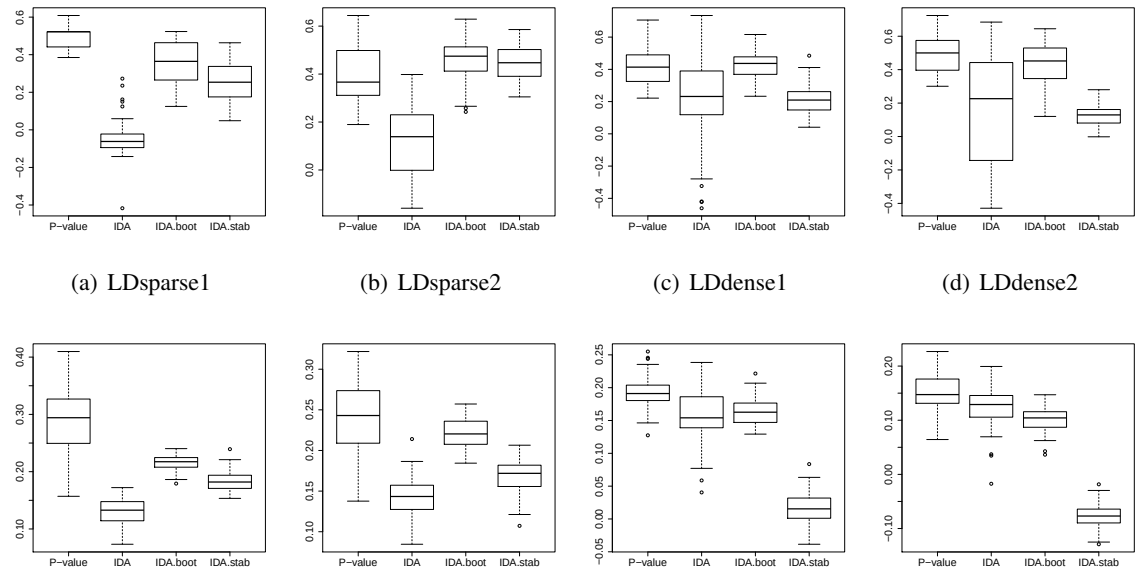
(a) LDsparse: $d = 10$ (b) LDdense: $d = 10$ (c) HDsparse: $d = 50$ (d) HDdense: $d = 50$

Figure 7: P-values for true causal effects $\{\theta^0_{Y=k,j};\ j \neq k\}$, for every $k = 1, \ldots d$. The different model specifications in the legend are specified in Appendix B, Figures 10 and 11.

Reasons for markedly different reliability of error control

Apparently, stability selection works much more reliable for type I error control than the p-values constructed from sample splitting. Reasons for such findings include the following. From a theory point of view, stability selection is justified using a weak "better than random guessing" assumption and a restrictive exchangeability condition, see Theorem 1. The latter seems rather far from necessary, as discussed also in Section 3.2.1. On the other hand, the p-value method relies on the fact that the true underlying CPDAG $\mathcal{E}(G^0) = \mathcal{E}(P^0)$ is estimated correctly using the first half of the sample, which seems far from approximately true. If the estimator $\hat{\mathcal{E}}_\alpha(I_1)$ is different from $\mathcal{E}(G^0) = \mathcal{E}(P^0)$, the resulting lower bounds and p-values are affected. This is very different from regression: there, instead of exact recovery of the active set, we only need the screening property (3.8) for constructing p-values which control the familywise error rate.

Besides these aspects from theory, control of the familywise error rate seems a more ambitious task than the less stringent control of the expected number of false positive.



(a) LDsparse1 (b) LDsparse2 (c) LDdense1 (d) LDdense2

(e) HDsparse1 (f) HDsparse2 (g) HDdense1 (h) HDdense2

Figure 8: Accuracy of ranking. Rank-correlation between true rank among $\{\theta^0_{Y=k,j};\ (j \neq k)\}$ and estimated rank. The different ranking methods are based on p-values (P-value), raw estimates from IDA (IDA), mean aggregation from bootstrapping (IDA.boot) and selection frequencies from subsampling (IDA.stab). The different model specifications in the legend are specified in Appendix B, Figures 10 and 11.

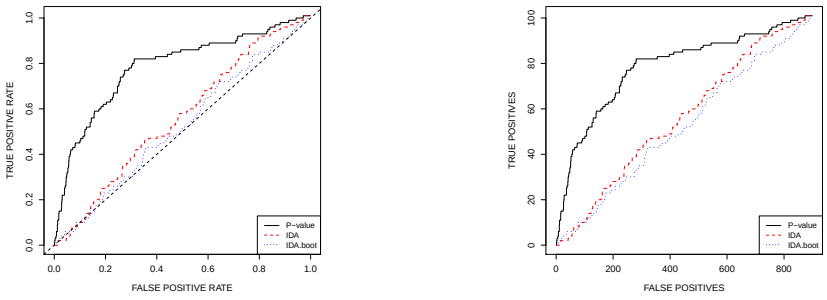
Ranking using p-values and subsampling
Since control of the familywise error rate (or the false discovery rate) seems to be unreliable for finite data, we wondered whether the p-values exhibit at least a better behavior for ranking. We rank the importance of the variables (in terms of their causal effects) by p-values, where the best variable with rank one has the smallest p-value. We do this over all settings where $Y = X^{(k)}$ and covariates $\{X^{(j)};\ j \neq k\}$. We then compare such a p-value based ranking with bootstrapping and mean aggregation over the bootstrapped lower bound estimates, and also with the selection frequencies $\hat{\pi}_{Y=k,j}$ from stability selection (using the top $q = \lfloor\sqrt{0.8d(d-1)}\rfloor$ largest lower bound estimates as selection method). Furthermore, we rank according to the (non-subsampled) estimated lower bound values $\hat{\ell}_{Y=k,j}$, where the best variable has largest value. The parameter in the PC-algorithm is chosen as $\alpha = 0.05$.

The results are summarized in Figure 8. We see that the p-values provide the most powerful ranking scheme, followed by mean-aggregated bootstrapping and selection frequencies from subsampling. All of them improve upon the non-subsampled lower bound estimates. Although familywise error control is very unreliable using the p-values described in Section 4.4.2, they are improving the ranking.

4.5.2 Gene interactions in yeast

We consider here the problem about inferring intervention or causal effects among many genes in yeast, see Section 4.3. Mainly for computational reasons, we look at a subset of 1000 true intervention effects (from 234 intervention experiments): we choose the 100 largest and 900 smallest true intervention effects, and we then assign them as true (100 in total) and false (900 in total). Figure 9 shows an ROC-type curve where the curves arise according to the rankings from the p-values, the mean-aggregated subsampling and the lower bound estimates obtained directly from the original data (we do not compare with the ranking based on subsampled selection frequencies as this was inferior than mean aggregation in the simulation study).

We see that the p-value based ranking improves the ROC-type curve while here, mean aggregation from subsampling doesn't improve upon the non-subsampled estimated lower bounds. We note that the latter is not better than random guessing, in contrast to the result shown in Figure 5 where we consider the whole and much larger set of intervention effects from 234 intervened to all other genes.



(a) ROC (b) True vs. false positives

Figure 9: Gene interactions in yeast (focusing on 1000 interactions). Different methods for ranking with abbreviations as in Figure 8.

5 Conclusions

We have reviewed stability selection and construction of p-values for high-dimensional (generalized) regression. They work reliably for conservatively controlling the expected number of false positive or familywise error rate, respectively.

We then adapted these techniques to the much more challenging problem of high-dimensional variable selection for causal or interventional targets based on observational data. We have found that stability selection continues to reliably control the expected number of false positive selections although the power to detect true positives can be poor (which seems mainly due to the complexity and difficulty of the problem). This is in contrast to assigning p-values: unless the underlying causal influence diagram is very sparse, controlling the familywise error is far from being trustworthy. Potential reasons for such different findings are briefly discussed (Section 4.5.1). In terms of ranking different variables according to their strength for causal influence, p-values and also re-/subsampling-based methods are found again to improve the non-subsampled estimates. Overall, re-/subsampling and sample splitting hold premise to yield more accurate estimates, and to some extent also for type I error control, in the challenging area of inferring (bounds of) intervention or causal effects based on observational data.

5.1 Software

For all computations the statistical programming language `R` was used [22]. Methods for stability selection and sample splitting were implemented by ourselves. For the Lasso or Elastic Net we used the R-package `glmnet` [9]. For estimating causal structures using the PC-algorithm or estimating causal effects using IDA the R-package `pcalg` is available [13].

Acknowledgments

We thank the Guest Editor Hui Zou and a referee for valuable comments and feedback.

References

[1] P. Bühlmann. Boosting for high-dimensional linear models. *Annals of Statistics*, 34:559–583, 2006.

[2] P. Bühlmann, M. Kalisch, and M.H. Maathuis. Variable selection in high-dimensional linear models: partially faithful distributions and the PC-simple algorithm. *Biometrika*, 97:261–278, 2010.

[3] P. Bühlmann and S. van de Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, 2011.

[4] P. Bühlmann and B. Yu. Analyzing bagging. *Annals of Statistics*, 30:927–961, 2002.

[5] E.M. Conlon, X.S. Liu, J.D. Lieb, and J.S. Liu. Integrating regulatory motif discovery and genome-wide expression analysis. *Proceedings of the National Academy of Sciences*, 100:3339–3344, 2003.

[6] B. Efron. *Large-Scale Inference: Empirical Bayes Methods for Estimation, Testing, and Prediction*. Institute of Mathematical Statistics Monographs. Cambridge University Press, second edition, 2010.

[7] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, 2001.

[8] D. Freedman. A remark on the difference between sampling with and without replacement. *Journal of the American Statistical Association*, 72:681, 1977.

[9] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[10] J.J. Goeman, S.A. van de Geer, and H.C. van Houwelingen. Testing against a high dimensional alternative. *Journal of the Royal Statistical Society Series B*, 68(3):477–493, 2006.

[11] T.R. Hughes, M.J. Marton, A.R. Jones, C.J. Roberts, R. Stoughton, C.D. Armour, H.A. Bennett, E. Coffey, H. Dai, Y.D. He, M.J. Kidd, A.M. King, M.R. Meyer, D. Slade, P.Y. Lum, S.B. Stepaniants, D.D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S.H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102:109–126, 2000.

[12] M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007.

[13] M. Kalisch, M. Mächler, D. Colombo, M.H. Maathuis, and P. Bühlmann. Causal inference using graphical models with the R package pcalg. *Preprint, available at http://cran.r-project.org/web/packages/pcalg/vignettes/pcalgDoc.pdf*, 2010.

[14] X.S. Liu, D.L. Brutlag, and J.S. Liu. An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature Biotechnology*, 20:835–839, 2002.

[15] M.H. Maathuis, D. Colombo, M. Kalisch, and P. Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7:247–248, 2010.

[16] M.H. Maathuis, M. Kalisch, and P. Bühlmann. Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37:3133–3164, 2009.

[17] N. Meinshausen. Relaxed Lasso. *Computational Statistics & Data Analysis*, 52:374–393, 2007.

[18] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 34:1436–1462, 2006.

[19] N. Meinshausen and P. Bühlmann. Stability selection (with discussion). *Journal of the Royal Statistical Society Series B*, 72:417–473, 2010.

[20] N. Meinshausen, L. Meier, and P. Bühlmann. P-values for high-dimensional regression. *Journal of the American Statistical Association*, 104:1671–1681, 2009.

[21] J. Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2000.

[22] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[23] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, second edition, 2000.

[24] R. Tibshirani. Regression analysis and selection via the Lasso. *Journal of the Royal Statistical Society Series B*, 58:267–288, 1996.

[25] S.A. van de Geer and P. Bühlmann. On the conditions used to prove oracle results for the Lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.

[26] L. Wasserman and K. Roeder. High dimensional variable selection. *Annals of Statistics*, 37:2178–2201, 2009.

[27] A. Wille and P. Bühlmann. Low-order conditional independence graphs for inferring genetic networks. *Statistical Applications in Genetics and Molecular Biology*, 5:1–32, 2006.

[28] C.H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, 38:894–942, 2010.

[29] P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

[30] H. Zou. The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, 101:1418–1429, 2006.

[31] H. Zou and T. Hastie. Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society Series B*, 67:301–320, 2005.

[32] H. Zou and R. Li. One-step sparse estimates in nonconcave penalized likelihood models (with discussion). *Annals of Statistics*, 36:1509–1566, 2008.

Appendix A

We describe here the assumptions used for the theorems in Section 4.

Assumptions for Theorem 3.
We consider a triangular scheme of observations from model (4.2):

$$X_{n,1}, \ldots, X_{n,n} \text{ i.i.d. } \sim P^{(n)},\ n = 1, 2, 3, \ldots,$$

where $X_n = (X_n^{(1)}, \ldots X_n^{(p_n)}, X_n^{(p_n+1)})$ with $X_n^{(p_n+1)} = Y_n$. Our assumptions are as follows.

(A1) The distribution $P^{(n)}$ is multivariate Gaussian and faithful to a DAG $G^{(n)}$ for all $n \in \mathbb{N}$.

(A2) The dimension $p_n = O(n^a)$ for some $0 \leq a < \infty$.

(A3) The maximal number of neighbors in the undirected graph $G^{(n)}$, denoted by $q_n = \max_{1 \leq j \leq p_n+1} |\text{adj}(G_n, j)|$, satisfies $q_n = O(n^{1-b})$ for some $0 < b \leq 1$.

(A4) The partial correlations satisfy:

$$\inf\{|\rho_{jk|C}|;\ \rho_{jk|C} \neq 0,$$
$$j, k = 1, \ldots, p_n + 1\ (j \neq k), C \subseteq \{1, \ldots, p_n + 1\} \setminus \{j, k\},\ |C| \leq q_n\} \geq c_n,$$

where $c_n^{-1} = O(n^d)$ $(n \to \infty)$ for some $0 < d < b/2$ and $0 < b \leq 1$ as in (A3);

$$\sup_n\{|\rho_{jk|C}|;$$
$$j, k = 1, \ldots, p_n + 1\ (j \neq k), C \subseteq \{1, \ldots, p_n + 1\} \setminus \{j, k\}, |C| \leq q_n\} \leq M < 1.$$

Assumptions for Theorem 4.
We consider the same setting as above. We assume (A1)-(A4) and in addition:

(A5) The conditional variances satisfy the following bound:

$$\inf\left\{\frac{\text{Var}(X_n^{(j)}|X_n^{(S)})}{\text{Var}(X_n^{(p_n+1)}|X_n^{(j)}, X_n^{(S)})};\ S \subseteq \text{adj}(G_n, j),\ j = 1, \ldots, p_n\right\} \geq v^2,$$

for some $v > 0$.

Appendix B

We describe here the details of the simulation models used.

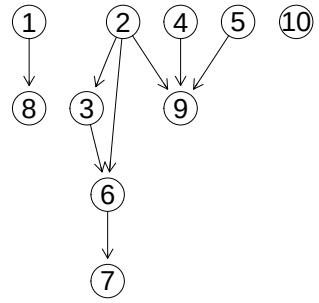
Simulation in Section 3.2.1 underlying Figure 2.
We consider semi-synthetic data from a linear model as in (2.2) having designs from real data-sets, one with $p = 660$ and $n = 2587$ and another one with $p = 4088$ and $n = 158$. Sparse regression coefficients β_j^0 i.i.d. Uniform($[0, 1]$) are generated and the size of the active set is varied with s_0

taking 16 different values between 4 and 50. We choose Gaussian errors with variances σ^2 to achieve signal to noise ratios (SNRs) in $\{0.5, 2\}$. In total, there are 64 scenarios and we run each of them 100 times.

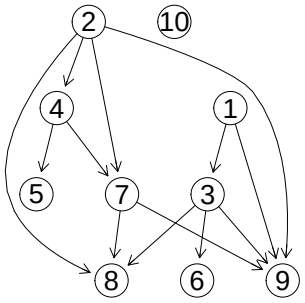
We then test how well the error control of Theorem 1 holds up for these semi-synthetic data-sets. We are interested in the comparison between the 10-fold cross-validated solution for the Lasso (without stability selection) and stability selection using the Lasso. For stability selection, we chose $q = \lfloor\sqrt{0.8p}\rfloor$ (the first q variables entering the regularization path) and a threshold of $\pi_{\text{thr}} = 0.6$, corresponding to a control of $\mathbb{E}[V] \leq 2.5$, where V is the number of wrongly selected variables. The control is mathematically derived under the assumption of exchangeability as described in Theorem 1. This assumption is most likely not fulfilled for the given real data designs and it is of interest to see how well the error bound holds up for our semi-synthetic data-sets. The results are shown in Figure 2.

Simulation in Section 3.3.4 underlying Figure 4.
We simulate data from 16 different linear models as in (2.2) with $n = 100$ and $p = 1000$. The design matrix is obtained from realizations of $X_1, \ldots, X_n$ i.i.d $\mathcal{N}_p(0, \Sigma)$, $\Sigma_{j,k} = 0.5^{|j-k|}$, and we use sparse β^0-vectors with active set S_0 and $s_0 = |S_0|$. In each simulation run, a new parameter vector β^0 is created by either "uniform" or "varying-strength" sampling. Under "uniform" sampling, s_0 randomly chosen components of β are set to 1 and the remaining $p - s_0$ components to 0. Under "varying-strength" sampling, s_0 randomly chosen components of β are set to values $1, \ldots, s_0$. The error variance σ^2 is adjusted such that the signal to noise ratio (SNR) $\in \{0.25, 1, 4, 16\}$ and the number s_0 of active variables is either 5 or 10.Thus, we consider 16 different scenarios (4 different SNRs, 2 different sparsity values s_0 and 2 different sampling schemes for β). We perform 50 simulations for each scenario.

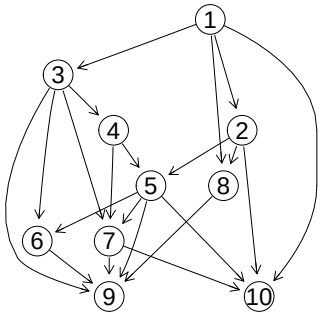
Simulation in Section 4.5.1 underlying Figures 6, 7 and 8.
The DAGs underlying the simulation model in Section 4.5.1 are given below in Figures 10 and 11.



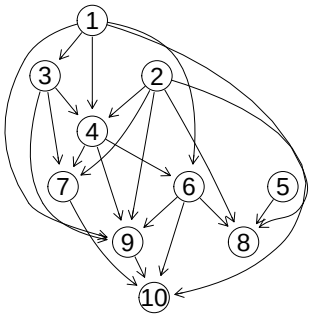
(a) $d = 10$ and sparse (LDsparse1)



(b) $d = 10$ and sparse (LDsparse2)

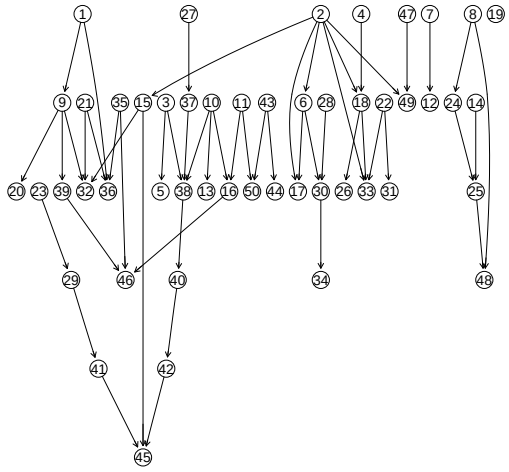


(c) $d = 10$ and dense (LDdense1)

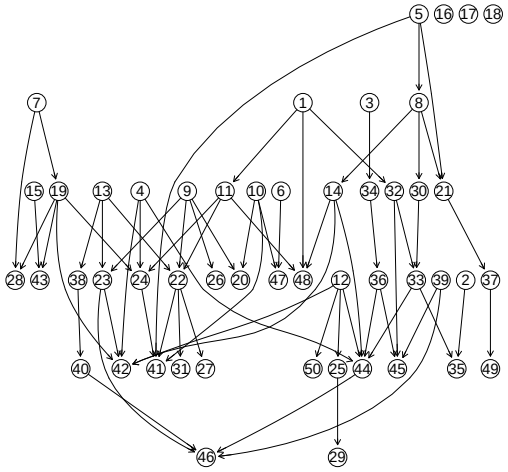


(d) $d = 10$ and dense (LDdense2)

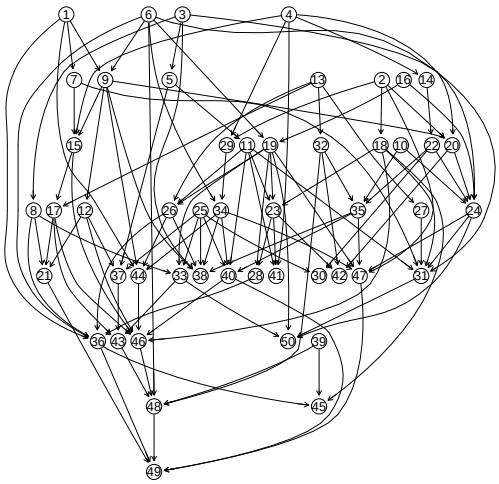
Figure 10: Low-dimensional DAGs.



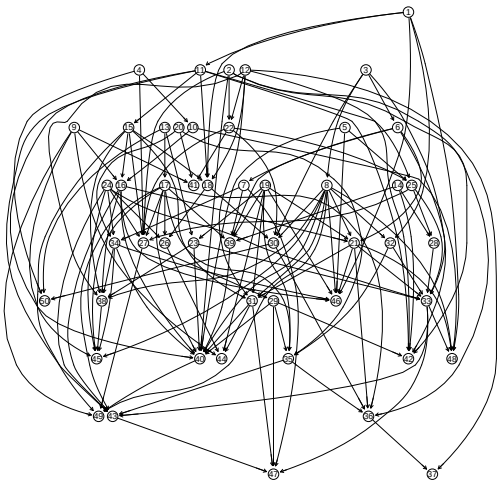
(a) $d = 50$ and sparse (HDsparse1)



(b) $d = 50$ and sparse (HDsparse2)



(c) $d = 50$ and dense (HDdense1)



(d) $d = 50$ and dense (HDdense2)

Figure 11: High-dimensional DAGs.