Structure Learning of Linear Gaussian Structural Equation Models with Weak Edges

Marco F. Eigenmann
Seminar für Statistik
ETH Zurich
Zurich, Switzerland

Preetam Nandy
Department of Biostatistics and Epidemiology
University of Pennsylvania
Philadelphia, PA 19104, USA

Marloes H. Maathuis
Seminar für Statistik
ETH Zurich
Zurich, Switzerland

Abstract

We consider structure learning of linear Gaussian structural equation models with weak edges. Since the presence of weak edges can lead to a loss of edge orientations in the true underlying CPDAG, we define a new graphical object that can contain more edge orientations. We show that this object can be recovered from observational data under a type of strong faithfulness assumption. We present a new algorithm for this purpose, called aggregated greedy equivalence search (AGES), that aggregates the solution path of the greedy equivalence search (GES) algorithm for varying values of the penalty parameter. We prove consistency of AGES and demonstrate its performance in a simulation study and on single cell data from Sachs et al. (2005). The algorithm will be made available in the R-package `pcalg`.

1 INTRODUCTION

We consider structure learning of linear Gaussian structural equation models (SEMs) (Bollen, 1989). A linear SEM is a set of equations of the form $X = B^T X + \varepsilon$, where $X = (X_1, \dots, X_p)^T$, B is a $p \times p$ strictly upper triangular matrix, $\varepsilon = (\varepsilon_1, \dots, \varepsilon_p)^T$, and ε is multivariate Gaussian with mean vector zero and a diagonal covariance matrix D (hence assuming no hidden confounders). Such SEMs can be represented by a directed acyclic graph (DAG) G, where a nonzero entry B_{ij} corresponds to an edge from X_i to X_j. By putting the coefficients B_{ij} along the corresponding edges, one obtains a weighted graph. This weighted graph and the distribution of ε fully determine the distribution of X. Example 1.1 shows a simple instance with $p = 3$, where

$$B = \begin{pmatrix} 0 & 0.1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

The weighted DAG is shown in Figure 1a.

Based on n i.i.d. observations from X, we aim to learn the underlying DAG G. However, since G is generally not identifiable from the distribution of X, we learn the so-called Markov equivalence class of G, which can be represented by a completed partially directed acyclic graph (CPDAG) (see Section 2.1). A CPDAG can contain both directed and undirected edges, where undirected edges represent uncertainty about the edge orientation.

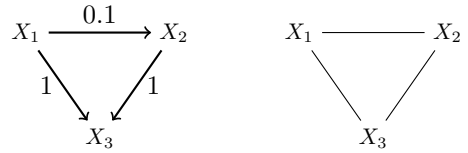
Several efficient algorithms have been developed to learn CPDAGs, such as for example the PC algorithm (Spirtes et al., 2000) and the greedy equivalence search algorithm (GES) (Chickering, 2002b). These algorithms have been proved to be sound and consistent (Spirtes et al., 2000; Kalisch and Bühlmann, 2007; Chickering, 2002b; Nandy et al., 2015).

Example 1.1 illustrates a somewhat counter-intuitive behaviour of these algorithms for varying sample size.

Example 1.1. *Consider the following SEM:*

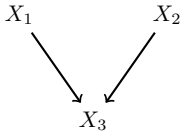
$$\begin{aligned} X_1 &= \varepsilon_1 \\ X_2 &= 0.1 \cdot X_1 + \varepsilon_2 \\ X_3 &= X_1 + X_2 + \varepsilon_3, \end{aligned}$$

where $\varepsilon \sim N(0, I)$. The corresponding CPDAG is the complete undirected graph in Figure 1b. When running PC or GES with a very large sample size, the algorithms will output this CPDAG with high probability. For a smaller sample size, however, the algorithms are likely to miss the weak edge $X_1 - X_2$, leading to the CPDAG in Figure 1c. Note that the latter CPDAG contains two edge orientations that are identical to the orientations in the

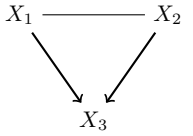


(a) True DAG G_0.

(b) CPDAG of G_0.

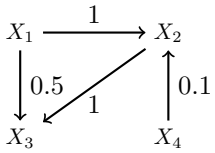


(c) CPDAG without the weak edge.

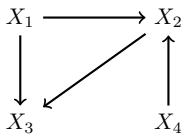


(d) Desired APDAG.

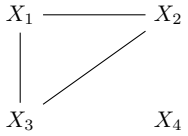
Figure 1: A simple case where the inclusion of a weak edge leads to a loss of edge orientations (see Example 1.1).



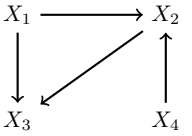
(a) True DAG G_0.



(b) CPDAG of G_0.



(c) CPDAG without the weak edge.



(d) Desired APDAG.

Figure 2: A simple case where the inclusion of a weak edge helps to obtain edge orientations (see Example 1.2).

underlying DAG G_0. Thus, both CPDAGs in Figures 1b and 1c contain some relevant information, one in terms of correct adjacencies and one in terms of correct edge orientations. For this example, GES outputs Figure 1c for a sample size smaller than 100 *and Figure 1b for a sample size larger than 1000, with high probabilities.*

One may think that a simple solution to the above problem is to omit weak edges either by using a strong penalty on model complexity or by truncating edges with small weights. In some cases, however, the inclusion of weak edges can also help to obtain edge orientations. This is illustrated in Example 1.2.

Example 1.2. *Consider the weighted DAG in Figure 2a with $\varepsilon \sim N(0, I)$. Figure 2b represents the corresponding CPDAG, which is fully oriented. For large sample sizes, PC and GES will output this CPDAG with high probability. For smaller sample sizes, however, they are likely to miss the weak edge $X_4 \to X_2$, leading to the CPDAG in Figure 2c, which is fully undirected.*

With a larger sample size we expect to gain more insight into a system, and the fact that we can lose correct edge orientations is undesirable. In the extreme case of a complete DAG with many weak edges, a small sample size yields informative output in terms of certain edge orientations, while a large sample size yields the asymptotically correct CPDAG, which is the uninformative complete undirected graph. This problem is relevant in practice for situations where the underlying system contains many weak effects and the sample size can be very large.

We propose a solution for this problem by defining a new graphical target object that can contain more edge orientations than the CPDAG. This object is a partially directed acyclic graph (PDAG) obtained by aggregating several CPDAGs of sub-DAGs of the underlying DAG G_0, and is called an aggregated PDAG (APDAG). In Example 1.1, we intuitively overlay the CPDAGs of Figures 1b and 1c to obtain the APDAG in Figure 1d, that contains both the correct skeleton and some edge orientations that were present in G_0 but not in the CPDAG of G_0. The APDAG for Example 1.2 is given in Figure 2d, and is in this case identical to the CPDAG of G_0.

Our APDAG is a maximally oriented PDAG, as studied in Meek (1995). We will show that APDAGs can be learned from observational data under a type of strong faithfulness condition, namely strong faithfulness with respect to a sequence of sub-DAGs of the underlying DAG. In this sense, our work is related to other structure learning algorithms that output maximally oriented PDAGs or DAGs under certain restrictions on the model class (e.g., Shimizu et al., 2006; Hoyer et al., 2008; Peters and Bühlmann, 2014; Ernest et al., 2016; Peters et al., 2014). Perković et al. (2017) provide methods for causal reasoning with maximally oriented PDAGs.

We propose an algorithm to learn APDAGs, by aggregating the solution path of the greedy equivalence search (GES) algorithm for varying values of the penalty parameter. The algorithm is therefore called aggregated GES (AGES). We show that the entire solution path can basically be computed at once, similarly to the computation of the solution path of the Lasso (Tibshirani, 1996; Tibshirani and Taylor, 2011). We also prove consistency of the algorithm, and demonstrate its performance in a simulation study and on data from Sachs et al. (2005). All proofs are given in the supplementary material.

2 PRELIMINARIES

2.1 GRAPHICAL MODELS

We now introduce the main terminology for graphical models that we need. Further definitions can be found in Section 1 of the supplementary material.

A *graph* $G = (X, E)$ consists of a set of *vertices* $X = \{X_1, \ldots, X_p\}$ and a set of *edges* E. The edges can be either *directed* $X_i \to X_j$ or *undirected* $X_i - X_j$. A *directed graph* is a graph that contains only directed edges. A *partially directed graph* can contain both directed and undirected edges.

If $X_i \to X_j$, then X_i is a *parent* of X_j. The set of parents of X_i in a graph G is denoted by $\mathrm{Pa}_G(X_i)$. A triple (X_i, X_j, X_k) in a graph G is called a *v-structure* if $X_i \to X_j \leftarrow X_k$ and X_i and X_k are not adjacent in G.

A *directed acyclic graph* (DAG) is a directed graph that does not contain directed cycles. A partially directed graph that does not contain directed cycles is a *partially directed acyclic graph* (PDAG). A PDAG P is *extendible* to a DAG if the undirected edges of P can be oriented to obtain a DAG without additional v-structures. The *skeleton* of a partially directed graph G is the graph obtained by replacing all directed edges by undirected edges, and is denoted by $\mathrm{Skeleton}(G)$. The *directed part* of a partially directed graph G is the graph obtained by removing all undirected edges, and is denoted by $\mathrm{DirPart}(G)$. A DAG G restricted to a graph H is the DAG G' obtained by removing from G all adjacencies not present in H. A DAG $G' = (X, E')$ is a *sub-DAG* of a DAG $G = (X, E)$ if $E' \subseteq E$.

A DAG encodes conditional independence constraints via the concept of d-separation (Pearl, 2009). Several DAGs can encode the same set of d-separations. Such DAGs are called *Markov equivalent*. Markov equivalent DAGs have the same skeleton and the same v-structures (Verma and Pearl, 1990). A Markov equivalence class of DAGs can be represented by a *completed partially directed acyclic graph* (CPDAG) (Andersson et al., 1997; Chickering, 2002a). We denote by $\mathrm{CPDAG}(G)$ the CPDAG of a DAG G. A directed edge $X_i \to X_j$ in a CPDAG means that $X_i \to X_j$ occurs in all DAGs in the Markov equivalence class. An undirected edge $X_i - X_j$ in a CPDAG means that there is a DAG with $X_i \to X_j$ and a DAG with $X_i \leftarrow X_j$ in the Markov equivalence class.

We denote conditional independence of two variables X_i and X_j given a set $S \subseteq X \setminus \{X_i, X_j\}$ by $X_i \perp\!\!\!\perp X_j | S$, and the corresponding d-separation relation in a DAG G is denoted by $X_i \perp_G X_j | S$.

A DAG $G = (X, E)$ is a perfect map of the distribution of X if every conditional independence constraint in the distribution is also encoded by the DAG G via d-separation, and vice versa. The first direction is known as the *faithfulness condition* while the backward direction is known as the *Markov condition*. A multivariate Gaussian distribution is said to be δ-*strong faithful* to a DAG $G = (X, E)$ if for every $X_i, X_j \in X$ and for every $S \subseteq X \setminus \{X_i, X_j\}$ it holds that $X_i \not\perp_G X_j | S \Rightarrow |\rho_{X_i, X_j | S}| > \delta$, where $\rho_{X_i, X_j | S}$ is the partial correlation between X_i and X_j given S (cf., Zhang and Spirtes, 2003). Faithfulness is a special case of δ-strong faithfulness with $\delta = 0$.

Throughout the paper we consider distributions of X that allow a perfect map representation through a DAG $G_0 = (X, E)$. The density f of X then admits the following factorization based on G_0: $f(x) = \prod_{i=1}^{p} f(x_i | \mathrm{Pa}_{G_0}(x_i))$.

We denote n i.i.d. observations of $\tilde{X} \subseteq X$ by $\tilde{X}^{(n)}$. DAGs will be denoted with the letter G, PDAGs with P, CPDAGs with C, and APDAGs with A. We reserve the subscript 0 for graphs associated with the true underlying distribution.

2.2 STRUCTURE LEARNING ALGORITHMS

We will make use of the Greedy Equivalence Search (GES) algorithm of Chickering (2002b). This algorithm is composed of two phases called the forward and the backward phase. Starting generally from the empty graph, the forward phase greedily adds edges, one at a time, minimizing each time a scoring criterion over the set of neighbouring CPDAGs. The forward phase stops when the score can no longer be improved by a single edge addition. At that point, the backward phase starts and removes edges, also one at a time, minimizing each time the same scoring criterion, until the score can no longer be improved.

GES operates on the space of CPDAGs. Conceptually, a move from one CPDAG to the next goes as follows: GES computes all DAGs belonging to the actual CPDAG. It then computes all possible edge additions (deletions) for each of the found DAGs. Among all possible edge additions (deletions) it chooses the one that leads to the maximum score improvement, and then computes the CPDAG of the resulting DAG. Chickering (2002b) presented an efficient way to move from one CPDAG to the next without computing the DAGs as described above.

GES has one tuning parameter which we call penalty parameter and denote by λ. As scoring criterion we take a penalized negative log-likelihood function of the follow-

ing form:

$$\mathcal{S}_\lambda(G, X^{(n)}) = -\sum_{i=1}^{p} \frac{1}{n} \log(L(X_i^{(n)}, \mathrm{Pa}_G(X_i)^{(n)})) + \lambda |E_G|$$

where L is the likelihood function (cf. Definition 5.1 in Nandy et al., 2015). As oracle version of this scoring criterion, we use the true covariance matrix to compute the expected log-likelihood (see Nandy et al., 2015). We denote the output of the oracle version of GES by $\mathrm{GES}_\lambda(f)$ and the output of the sample version of GES by $\mathrm{GES}_\lambda(X^{(n)})$.

Chickering (2002b) showed consistency for GES for a class of scoring criteria including the Bayesian Information Criterion (BIC), which corresponds to $\lambda = \log(n)/(2n)$. The oracle version of GES is sound for $\lambda = 0$, i.e., $\mathrm{GES}_0(f) = \mathrm{CPDAG}(G_0)$.

Given the density f of X, the *solution path* of the oracle version of GES is defined as the ordered set of CPDAGs $\mathrm{GES}_\lambda(f)$ for increasing values of the penalty parameter λ, $\lambda \geqslant 0$. Given n i.i.d. samples $X^{(n)}$, the solution path of the sample version of GES is defined as the ordered set of estimated CPDAGs $\mathrm{GES}_\lambda(X^{(n)})$ for increasing values of the penalty parameter λ, for $\lambda \geqslant \log(n)/(2n)$.

Nandy et al. (2015) showed that the difference in score between two DAGs $G = (X, E)$ and $G' = (X, E')$ that differ by a single edge, i.e., $E' = E \cup \{X_i \to X_j\}$, is given by

$$S_\lambda(G', X^{(n)}) - S_\lambda(G, X^{(n)}) = \frac{1}{2} \log(1 - \hat{\rho}^2_{X_i, X_j | \mathrm{Pa}_G(X_j)}) + \lambda \qquad (1)$$

(see Lemma 1.2 of the supplementary material). An edge is added (or deleted) in the forward (or backward) phase of GES only if this quantity is negative. To obtain the oracle version of Equation (1) we use the true covariance matrix to compute the partial correlation.

3 AGES

The main idea behind our new algorithm, Algorithm 2, is to consider a sequence of sub-DAGs of the underlying DAG G_0, to compute their CPDAGs, and finally to aggregate these CPDAGs. Considering only sub-DAGs of G_0 ensures that if an edge is oriented in one of these CPDAGs it has the same orientation as in G_0. This property makes the aggregation intuitive since all CPDAGs will have compatible edge orientations. To learn these CPDAGs we need to assume a special type of δ-strong faithfulness with respect to the sub-DAGs (see Theorem 3.2). The CPDAGs mentioned above can be computed efficiently using GES (see Section 3.5). Therefore, we base our new algorithm on GES and call it aggregated GES (AGES).

Algorithm 1: AggregateCPDAGs

input : Ordered set of CPDAGs $\mathcal{C} = \{C_0, \ldots, C_k\}$
output: APDAG A

```
A ← C_0
for i ∈ {1, ..., k} do
    Define P ← A
    for All edges in C_i do
        if an edge is oriented in C_i but not in P then
            Orient it in P as in C_i
        end
    end
    if P is extendible to a DAG then
        A ← P
    end
end
return MeekOrient(A) (Sec. 1 of the supp. material)
```

3.1 THE APDAG A_0

We construct our new target, the aggregated PDAG (APDAG) A_0, with the following four steps:

S.1 Given a multivariate density f of X, compute the solution path of the oracle version of GES for $\lambda \geqslant 0$ and keep the outputs whose skeletons are contained in the skeleton of $C_0 = \mathrm{GES}_0(f)$. This yields a set of CPDAGs $\mathcal{C} = \{C_0, \cdots, C_k\}$ with associated penalty parameters $\lambda_0 < \ldots < \lambda_k$.[1]

S.2 Construct the set of DAGs $\mathcal{G} = \{G_0, \ldots, G_k\}$ consisting of G_0 restricted to the skeletons of the CPDAGs in $\mathcal{C}$.

S.3 Construct the CPDAGs $\tilde{\mathcal{C}} = \{\tilde{C}_0, \ldots, \tilde{C}_k\}$ where $\tilde{C}_i = \mathrm{CPDAG}(G_i)$, $0 \leqslant i \leqslant k$.

S.4 Let $A_0 = \mathrm{AggregateCPDAGs}(\tilde{\mathcal{C}})$ (Algorithm 1).

We emphasize that A_0 is a theoretical object, since its construction involves the oracle version of GES and the orientations of the true underlying DAG G_0. The construction ensures that $G_1, \ldots, G_k$ are sub-DAGs of G_0. Hence, any oriented edges in the corresponding CPDAGs $\tilde{C}_1, \ldots, \tilde{C}_k$ also correspond to those in G_0. As a result, the APDAG A_0 has the same skeleton as C_0 and

$$\mathrm{DirPart}(C_0) \subseteq \mathrm{DirPart}(A_0) \subseteq \mathrm{DirPart}(G_0).$$

[1] Throughout, we use the convention that any CPDAG computed by GES is associated with the smallest possible value of the penalty parameter λ for which this output can be obtained.

Algorithm 2: AGES (oracle)

input : Distribution of X
output: APDAG A

1 Compute the solution path of the oracle version of GES for $\lambda \geqslant 0$
2 Discard all outputs whose skeletons are not contained in the skeleton of the output when $\lambda = 0$. Denote the remaining set of CPDAGs associated with $\lambda_0 < \ldots < \lambda_k$ by $\mathcal{C} = \{C_0, \ldots, C_k\}$
3 **return** $\mathrm{AggregateCPDAGs}(\mathcal{C})$

This makes A_0 an interesting object to investigate.[2]

3.2 ORACLE VERSION OF AGES AND SOUNDNESS

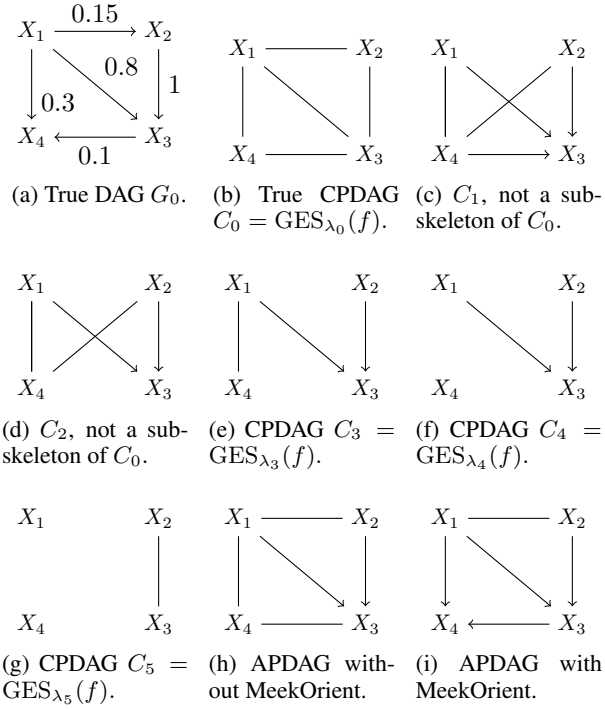
The oracle version of AGES is given in pseudocode as Algorithm 2. We use Example 3.1 to illustrate it. Soundness of the algorithm is shown in Theorem 3.2.

Example 3.1. *Consider the density f generated by the weighted DAG in Figure 3a with $\varepsilon \sim N(0, D)$, where D is a diagonal matrix with entries $(0.3, 0.4, 0.3, 0.4)$. We compute the solution path of the oracle version of GES, shown in the six CPDAGs in Figures 3b - 3g, corresponding to $\lambda_0 < \cdots < \lambda_5$. We discard C_1 and C_2 since their skeletons are not contained in the skeleton of C_0. We then aggregate the remaining CPDAGs C_0, C_3, C_4, and C_5, using lines 1-12 of Algorithm 1. The result shown in Figure 3h contains additional orientations, coming from the v-structure $X_1 \to X_3 \leftarrow X_2$ in C_3 . The final output in Figure 3i shows two further oriented edges due to MeekOrient.*

Theorem 3.2. *Given a multivariate Gaussian distribution of X with a perfect map $G_0 = (X, E)$, let $\mathcal{G}$ be the set of DAGs constructed in Step S.2, and let $\tilde{\mathcal{C}}$ be the corresponding set of CPDAGs of Step S.3. Assume that for all $1 \leqslant i \leqslant k$ the distribution of X is δ_i-strong faithful with respect to $G_i \in \mathcal{G}$, where δ_i is such that $\lambda_i = -1/2 \log(1 - \delta_i^2)$. Then $\mathrm{GES}_{\lambda_i}(f) = C_i = \tilde{C}_i$ for all $1 \leqslant i \leqslant k$, and the oracle version of AGES returns the APDAG A_0.*

Since the above δ_i-strong faithfulness assumption with respect to G_i for $1 \leqslant i \leqslant k$ is related to the solution path of GES, we refer to it as *path strong faithfulness*.

[2]We note that the if-clause on line 9 of Algorithm 1 is not needed when applying the algorithm to $\tilde{\mathcal{C}}$; it is needed in the context of Algorithms 2 and 3.



(a) True DAG G_0. (b) True CPDAG $C_0 = \mathrm{GES}_{\lambda_0}(f)$. (c) C_1, not a sub-skeleton of C_0.

(d) C_2, not a sub-skeleton of C_0. (e) CPDAG $C_3 = \mathrm{GES}_{\lambda_3}(f)$. (f) CPDAG $C_4 = \mathrm{GES}_{\lambda_4}(f)$.

(g) CPDAG $C_5 = \mathrm{GES}_{\lambda_5}(f)$. (h) APDAG without MeekOrient. (i) APDAG with MeekOrient.

Figure 3: Illustration of the oracle AGES algorithm (see Example 3.1).

3.3 SAMPLE VERSION OF AGES AND CONSISTENCY

The sample version of AGES is given in Algorithm 3. We see that the algorithm considers the output of the sample version of GES for all $\lambda \geqslant \log(n)/(2n)$, i.e., by penalizing equally strong or stronger than BIC for model complexity.

In line 2 of Algorithm 3, we may obtain CPDAGs with conflicting orientations. Because of such possible conflicts, we need the if-clause on line 9 of Algorithm 1. The aggregation algorithm is constructed so that orientations in $\hat{C}_\ell$ are only taken into account if they are compatible with the aggregated graph based on $\hat{C}_0, \ldots, \hat{C}_{\ell-1}$. In particular, the algorithm ensures that we stay within the Markov equivalence class defined by $\hat{C}_0 = \mathrm{GES}_{\log(n)/(2n)}(X^{(n)})$, i.e., the output of GES.

Let $\mathrm{AGES}(X^{(n)})$ denote the output of AGES based on a sample $X^{(n)}$. Theorem 3.3 shows consistency of AGES.

Theorem 3.3. *Under the conditions of Theorem 3.2, we have*

$$\lim_{n\to\infty} \mathbb{P}\left(\mathrm{AGES}(X^{(n)}) = A_0\right) \to 1.$$

Algorithm 3: AGES (sample)

input : $X^{(n)}$, containing n i.i.d. observations of X
output: Estimated APDAG $\hat{A}$

1 Compute the solution path of the sample version of GES for $\lambda \geqslant \log(n)/(2n)$
2 Discard all outputs whose skeletons are not contained in the skeleton of the output when $\lambda = \log(n)/(2n)$. Denote the remaining CPDAGs, ordered according to increasing penalty parameter λ, by $\hat{\mathcal{C}} = \{\hat{C}_0, \ldots, \hat{C}_k\}$
3 **return** AggregateCPDAGs($\hat{\mathcal{C}}$)

3.4 THE PATH STRONG FAITHFULNESS ASSUMPTION

The δ-strong faithfulness assumption has been used before, for example to prove uniform consistency and high-dimensional consistency of structure learning methods (Kalisch and Bühlmann, 2007; Zhang and Spirtes, 2003). On the other hand, it has been criticised for being too strong (Uhler et al., 2013).

We do not assume the classical δ-strong faithfulness for the underlying distribution with respect to G_0. Instead, we assume δ_i-strong faithfulness of the distribution of X with respect to the sequence of sub-DAGs $G_1, \ldots, G_k$ as defined in Step S.2, with corresponding $\lambda_1 < \cdots < \lambda_k$. Hence, the corresponding δ_is satisfy $\delta_1 < \cdots < \delta_k$. Since smaller values of λ typically yield denser graphs, it follows that for smaller values of δ_i, the assumption has to hold with respect to a denser graph, while for larger values of δ_i, the assumption has to hold with respect to a sparser graph.

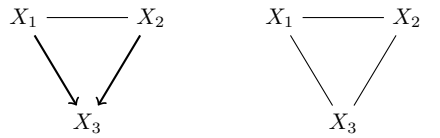
Example 3.4. *We first analyse the path strong faithfulness assumption by considering the SEM given in Example 1.1, but with unspecified edge weights B_{13} and B_{23}:*

$$\begin{aligned} X_1 &= \varepsilon_1 \\ X_2 &= 0.1 \cdot X_1 + \varepsilon_2 \\ X_3 &= B_{13}X_1 + B_{23}X_2 + \varepsilon_3, \end{aligned}$$

and $\varepsilon \sim N(0, I)$.

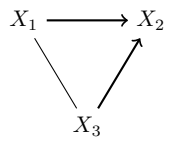
Depending on the edge weights, A_0 can be either the APDAG in Figure 4a or in Figure 4b. Figure 5 illustrates how A_0 and the path strong faithfulness assumption are related to the edge weights $B_{13} \in [-2, 2]$ and $B_{23} \in [-2, 2]$. We split the $[-2, 2] \times [-2, 2]$ rectangle into the following three regions:

White region: *A_0 equals the APDAG in Figure 4a and the path strong faithfulness assumption is satisfied.*

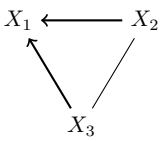


(a) Informative APDAG.

(b) Uninformative APDAG.



(c) APDAG with one wrong orientation.



(d) APDAG with two wrong orientations.

Figure 4: The possible outputs of AGES in Example 3.4.

Grey region: *A_0 equals the APDAG in Figure 4b and the path strong faithfulness assumption is satisfied.*

Black region: *A_0 equals the APDAG in Figure 4b and the path strong faithfulness assumption is violated.*

The output A of the oracle version of AGES can be one of the four APDAGs in Figure 4. Theorem 3.2 guarantees that $A = A_0$ when path strong faithfulness is satisfied, i.e., outside of the black region. Further, for this example, A equals one of the APDAGs in Figures 4c and 4d on the black region. This demonstrates that, in this simple example with $p = 3$, our strong faithfulness assumption is, in fact, a necessary and sufficient condition for having $A = A_0$. We emphasize that for $p > 3$, we may have $A = A_0$ even when the strong faithfulness assumption is violated.

Figure 5 shows that in a large fraction of the plane we gain structural information (white region), on a smaller part we perform as GES (grey region), and on another smaller part we make some errors when orienting edges (black region). Details about the construction of Figure 5 are given in Section 5 of the supplementary material.

The path strong faithfulness assumption is sufficient but not necessary for Theorem 3.2. In Section 6 of the supplementary material we provide a weaker version of the assumption that is necessary and sufficient for equality of $\tilde{\mathcal{C}}$ (as defined in Step S.3) and $\mathcal{C}$ (as defined in line 2 of Algorithm 2). This weaker version is only sufficient for equality of the true APDAG A_0 and the oracle output A of AGES ($\mathrm{AggregateCPDAGs}(\mathcal{C})$), since not all orientations of the CPDAGs in $\mathcal{C}$ are used in the aggregation process. The supplementary material also contains empirical results where we evaluated equality of $\tilde{\mathcal{C}}$ and $\mathcal{C}$,

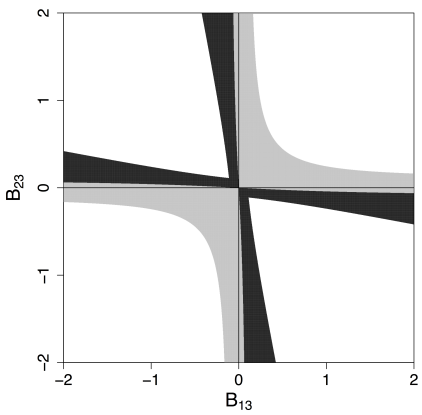


Figure 5: Visual representation of the dependence of A_0 and the path strong faithfulness assumption on the edge weights in Example 3.4.

as well A_0 and A, for the simulation setting described in Section 4.

3.5 COMPUTATION

The forward phase of GES (of both the oracle version and the sample version) can be computed at once for all $\lambda \geqslant 0$. This follows from Equation (1). At each step in the forward phase, GES conceptually searches for the in absolute value largest partial correlation $|\rho_{X_i,X_j|\,\mathrm{Pa}_G(X_j)}|$ among all DAGs G in the current Markov equivalence class, and all pairs X_i and X_j that are not adjacent in G and where X_i is a non-descendant of X_j in G. The algorithm then adds the corresponding edge $X_i \to X_j$ to G if the score is improved, that is, if $1/2\log(1-\rho^2_{X_i,X_j|Pa_G(X_j)}) + \lambda < 0$, and then constructs the CPDAG the resulting DAG.

Thus, starting the forward phase with the empty graph and a very large λ, no edge is added. By decreasing λ so that $\lambda < \max_{i,j} -1/2\log(1-\rho^2_{X_i,X_j})$, the first edge is added. By decreasing λ further, one can compute the entire solution path of the forward phase in one go, analogously to the computation of the solution path of the lasso (Tibshirani, 1996; Tibshirani and Taylor, 2011).

For each distinct output of the forward phase, obtained for a given λ, one has to run the backward phase with this λ. Since the backward phase of GES usually only conducts very few steps, this does not cause a large computational burden.

The fast computation of the entire solution path of GES is one of the reasons for basing our approach on GES, rather than, for example, on the PC-algorithm for a range of different tuning parameters α.

4 EMPIRICAL RESULTS

4.1 SIMULATION SETUP

We simulate data from SEMs of the following form:

$$X = B^T X + \varepsilon,$$

with $\varepsilon \sim \mathcal{N}(0, D)$, where D is a $p \times p$ diagonal matrix whose diagonal entries are drawn independently from a Unif(0.5,1.5) distribution.

In order to vary the concentration of strong and weak edge weights as well as the sparsity of the models, we consider all combinations of pairs $(q_s, q_w) \in \{0.1, 0.3, 0.5, 0.7\}$ such that $q_s + q_w \leqslant 1$. Each entry of the matrix B has a probability of q_s of being strong, of q_w of being weak, and of $(1 - q_s - q_w)$ of being 0. The nonzero edge weights in the B matrix are drawn independently as follows: the absolute values of the weak and the strong edge weights are drawn from Unif(0.1,0.3) and Unif(0.8,1.2), respectively. The sign of each edge weight is chosen to be positive or negative with equal probabilities. Finally, in order to investigate whether our algorithm performs at least as good as GES when we do not encourage the presence of weak edges, we also simulate from SEMs with $q_s \in \{0.1, 0.2, \ldots, 1\}$ and $q_w = 0$.

We simulate from SEMs with $p = 10$ variables. The sample size used in the plots in the main paper is 10000. The number of simulations for each settings is 500.

In Section 3 of the supplementary material we show additional plots corresponding to sample sizes 100 and 1000. Those plots show a similar pattern as the ones in the main paper, but the ability to gain additional edge orientations diminishes for smaller n. Section 4 of the supplementary material also shows simulation results for $p = 100$ and varying sample sizes.

4.2 SIMULATION RESULTS

Since AGES always outputs the same skeleton as GES by construction, we analyse the performance of GES and AGES by comparing their precision and recall in estimating the directed part of the true DAG. The recall is the ratio of the number of correctly oriented edges in the estimated graph and the total number of oriented edges in the true DAG. The precision is the ratio of the number of correctly oriented edges in the estimated graph and the total number of oriented edges in the estimated graph.

Figure 6 summarizes the performance of GES and AGES (with $\lambda = \log(n)/(2n)$) for all combinations of $(q_s, q_w) \in \{0.1, 0.3, 0.5, 0.7\}$ such that $q_s + q_w \leqslant 1$. In each setting, AGES outperforms GES in recall, while achieving a roughly similar performance as GES in pre-

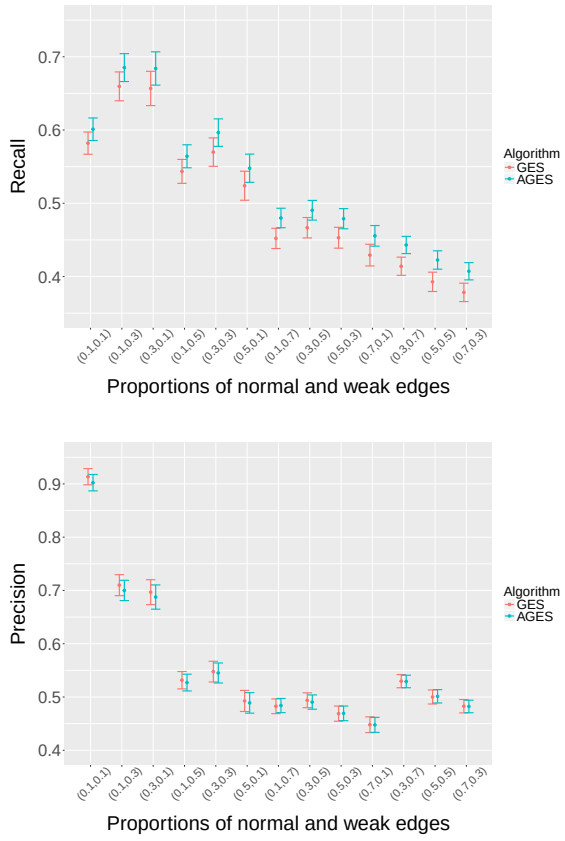


Figure 6: Mean precision and recall of GES and AGES over 500 simulations for all combinations of $(q_s, q_w) \in \{0.1, 0.3, 0.5, 0.7\}$ such that $q_s + q_w \leqslant 1$, using $\lambda = \log(n)/(2n)$ and $n = 10000$ (see Section 4.1). The bars in the plots correspond to $\pm$ twice the standard error of the mean.

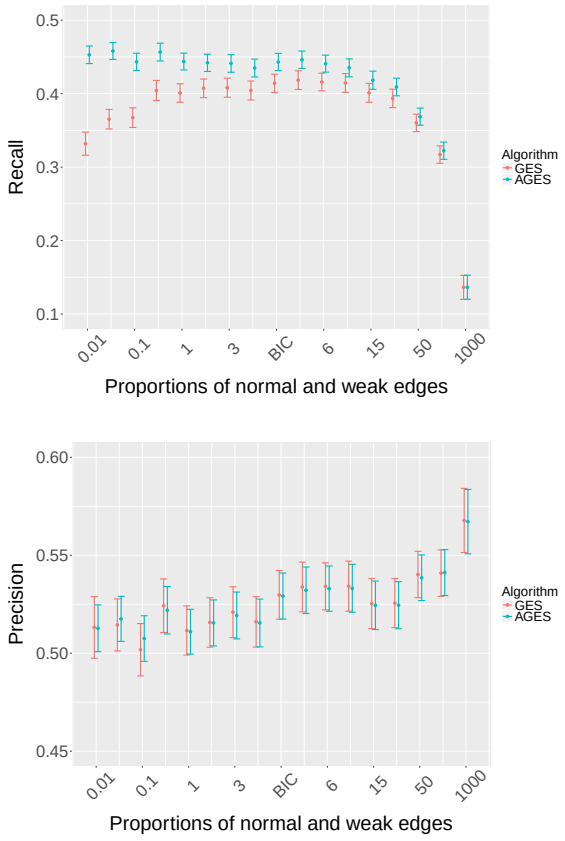


Figure 7: Mean precision and recall of GES and AGES over 500 simulations for $(q_s, q_w) = (0.3, 0.7)$, using $n = 10000$ and varying values of λ (see Section 4.1). The bars in the plots correspond to $\pm$ twice the standard error of the mean.

cision. This demonstrates that AGES is able to orient more edges than GES without increasing the false discovery rate.

Figure 7 compares the performance of GES and AGES for various choices of the penalty parameter λ when $(q_s, q_w) = (0.3, 0.7)$. In each case, we use the chosen penalty of GES as the minimum penalty of AGES, so that the skeletons of both outputs are identical. We see that AGES outperforms GES for all penalty parameters, and that AGES is less sensitive to the choice of the penalty parameter.

Figure 8 compares GES and AGES for $q_s \in \{0.1, 0.2, \ldots, 1\}$ and $q_w = 0$, using again $\lambda = \log(n)(2n)$. We see that AGES outperforms GES in recall for all values of q_s. There tends to be a small loss in precision for the sparser graphs.

4.3 APPLICATION TO SINGLE CELL DATA

We apply AGES to the well-known single cell data of Sachs et al. (2005), consisting of quantitative amounts of 11 proteins in human T-cells that were measured under 14 experimental conditions. In each experimental condition, different interventions were made, concerning the abundance or the activity of the molecules[3] (Sachs et al., 2005; Mooij and Heskes, 2013). We analyze each experimental condition separately, yielding 14 data sets with sample sizes between 700 and 1000.

Sachs et al. (2005) presented a conventionally accepted signalling network for these proteins (Sachs et al., 2005, Figures 2 and 3). We use this to determine a ground truth for each experimental condition (see Section 7 of the supplementary material), so that we can assess the performance of AGES in comparison to GES on these data.

[3] An activity intervention can either activate or inhibit the molecule

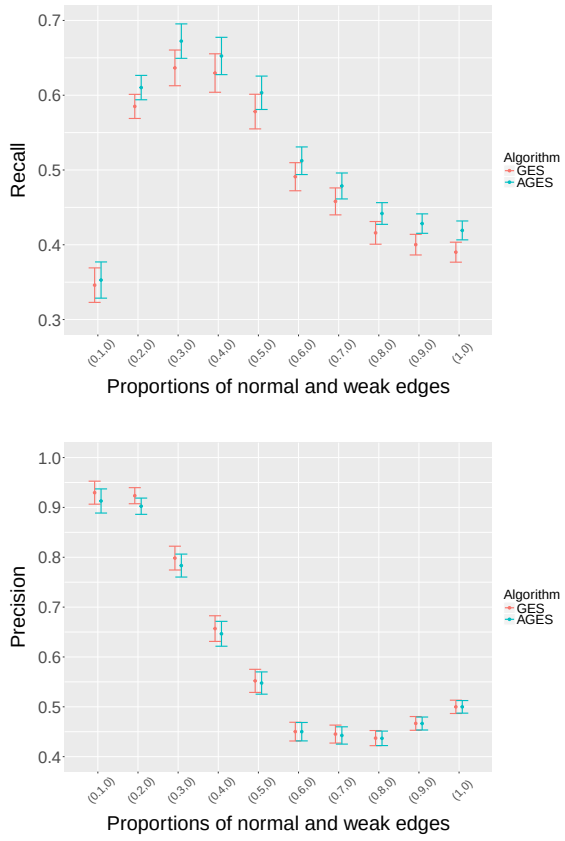


Figure 8: Mean precision and recall over 500 simulations with $q_s \in \{0.1, 0.2, \ldots, 1\}$ and $q_w = 0$, using $\lambda = \log(n)/(2n)$ and $n = 10000$ (see Section 4.1). The bars in the plots correspond to $\pm$ twice the standard error of the mean.

Again, since the skeletons of the outputs of GES and AGES are identical by construction, we only evaluate the directed edges. Moreover, we limit ourselves to adjacencies that are present in the true network. Considering these adjacencies, AGES found additional edge orientations in 6 experimental conditions. Table 1 summarizes the results. In experimental conditions 8 and 9, AGES was able to substantially improve the output, while in the other 4 conditions (4, 5, 13 and 14), AGES and GES had roughly similar performances. Thus, although these data almost certainly violate various assumptions of our methods (acyclicity, Gaussianity, path strong faithfulness, hidden confounders), we obtain encouraging results.

Experimental condition	4	5	8	9	13	14
Correct	0	1	8	5	1	1
Wrong	1	0	0	0	2	0

Table 1: For each of the listed experimental conditions, we report the number of correct and wrong edge orientations among edge orientations that were found by AGES but not by GES. The results are limited to adjacencies that are present in the true network (see Figure 9 of the supplementary material), and correctness of edge orientations was evaluated with respect to this network.

5 DISCUSSION

We considered structure learning of linear Gaussian SEMs with weak edges. We presented a new graphical object, called APDAG, that aggregates the structural information of many CPDAGs, yielding additional orientation information. We proposed a structure learning algorithm that uses the solution path of GES to learn this new object and gave sufficient conditions for its soundness and consistency. The algorithm will be made available in the R-package `pcalg` (Kalisch et al., 2012).

We applied AGES in a simulation study and on data from Sachs et al. (2005). Despite the fact that in both cases the assumptions of Theorem 3.2 are likely violated, we obtained promising results.

Our work can be easily extended to the so called nonparanormal distributions (Liu et al., 2009; Harris and Drton, 2013). In this setting we assume that there is a latent linear Gaussian SEM and that each observed variable is a strictly increasing (or strictly decreasing) transformation of the corresponding latent variable. In this case, the weakness of an edge can be connected to its edge weight in the latent linear Gaussian SEM and we can use AGES with a rank correlation based scoring criterion as defined in Nandy et al. (2015).

Moreover, the Gaussian error assumption can be dropped, i.e., we can consider linear SEMs with arbitrary error distributions. This is due to a one-to-one correspondence between zero partial correlations in a linear SEM with arbitrary error distributions and d-separations in its corresponding DAG (e.g., Hoyer et al., 2008). When all error variables are non-Gaussian, one can use the LiNGAM algorithm (Shimizu et al., 2006) to recover the data generating DAG uniquely. In this case, one would therefore not run GES or AGES. If some error variables are Gaussian and others are non-Gaussian, Hoyer et al. (2008) proposed a combination of PC and LiNGAM. It would be an interesting direction for future work to combine (A)GES with LiNGAM for a mixture of Gaussian and non-Gaussian error variables.

5.1 Acknowledgements

This work was supported in part by the Swiss NSF Grant 200021_172603.

References

Andersson, S. A., Madigan, D., and Perlman, M. D. (1997). A characterization of Markov equivalence classes for acyclic digraphs. *Ann. Stat.*, 25:505–541.

Bollen, K. (1989). *Structural Equations with Latent Variables*. Wiley, New York.

Chickering, D. M. (2002a). Learning equivalence classes of Bayesian-network structures. *J. Mach. Learn. Res.*, 2:445–498.

Chickering, D. M. (2002b). Optimal structure identification with greedy search. *J. Mach. Learn. Res.*, 3:507–554.

Ernest, J., Rothenhäusler, D., and Bühlmann, P. (2016). Causal inference in partially linear structural equation models: identifiability and estimation. arXiv:1607.05980.

Harris, N. and Drton, M. (2013). PC algorithm for nonparanormal graphical models. *J. Mach. Learn. Res.*, 14:3365–3383.

Hoyer, P. O., Hyvarinen, A., Scheines, R., Spirtes, P. L., Ramsey, J., Lacerda, G., and Shimizu, S. (2008). Causal discovery of linear acyclic models with arbitrary distributions. In *Proceedings of UAI 2008*, pages 282–289.

Kalisch, M. and Bühlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *J. Mach. Learn. Res.*, 8:613–636.

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *J. Statist. Software*, 47(11):1–26.

Liu, H., Lafferty, J., and Wasserman, L. (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *J. Mach. Learn. Research*, 10:2295–2328.

Meek, C. (1995). Causal inference and causal explanation with background knowledge. In *Proceedings of UAI 1995*, pages 403–410.

Mooij, J. M. and Heskes, T. (2013). Cyclic causal discovery from continuous equilibrium data. In *Proceedings of UAI 2013*, pages 431–439.

Nandy, P., Hauser, A., and Maathuis, M. H. (2015). High-dimensional consistency in score-based and hybrid structure learning. arXiv:1507.02608v4.

Pearl, J. (2009). *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, 2nd edition.

Perković, E., Kalisch, M., and Maathuis, M. H. (2017). Interpreting and using CPDAGs with background knowledge. In *Proceedings of UAI 2017*. To appear; arXiv:1707.02171.

Peters, J. and Bühlmann, P. (2014). Identifiability of Gaussian structural equation models with equal error variances. *Biometrika*, 101:219–228.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models. *J. Mach. Learn. Res.*, 15:2009–2053.

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A., and Nolan, G. P. (2005). Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308:523–529.

Shimizu, S., Hoyer, P., Hyvärinen, A., and Kerminen, A. (2006). A linear non-Gaussian acyclic model for causal discovery. *J. Mach. Learn. Res.*, 7:2003–2030.

Spirtes, P., Glymour, C., and Scheines, R. (2000). *Causation, Prediction, and Search*. MIT Press, Cambridge, 2nd edition.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58:267–288.

Tibshirani, R. J. and Taylor, J. (2011). The solution path of the generalized lasso. *Ann. Statist.*, 39:1335–1371.

Uhler, C., Raskutti, G., Bhlmann, P., and Yu, B. (2013). Geometry of the faithfulness assumption in causal inference. *Ann. Statist.*, 41:436–463.

Verma, T. and Pearl, J. (1990). Equivalence and synthesis of causal models. In *Proceedings of UAI 1990*, pages 255–270.

Zhang, J. and Spirtes, P. (2003). Strong faithfulness and uniform consistency in causal inference. In *Proceedings of UAI 2003*, pages 632–639.