Journal of Machine Learning Research 7 (2006) 2003-2030 Submitted 3/06; Revised 7/06; Published 10/06

A Linear Non-Gaussian Acyclic Model for Causal Discovery

Shohei Shimizu* SHOHEIS@ISM.AC.JP
Patrik O. Hoyer PATRIK.HOYER@HELSINKI.FI
Aapo Hyvärinen AAPO.HYVARINEN@HELSINKI.FI
Antti Kerminen ANTTI.KERMINEN@HELSINKI.FI
Helsinki Institute for Information Technology, Basic Research Unit
Department of Computer Science
University of Helsinki
FIN-00014, Finland

Editor: Michael Jordan

Abstract

In recent years, several methods have been proposed for the discovery of causal structure from non-experimental data. Such methods make various assumptions on the data generating process to facilitate its identification from purely observational data. Continuing this line of research, we show how to discover the complete causal structure of continuous-valued data, under the assumptions that (a) the data generating process is linear, (b) there are no unobserved confounders, and (c) disturbance variables have non-Gaussian distributions of non-zero variances. The solution relies on the use of the statistical method known as independent component analysis, and does not require any pre-specified time-ordering of the variables. We provide a complete Matlab package for performing this LiNGAM analysis (short for Linear Non-Gaussian Acyclic Model), and demonstrate the effectiveness of the method using artificially generated data and real-world data.

Keywords: independent component analysis, non-Gaussianity, causal discovery, directed acyclic graph, non-experimental data

1. Introduction

Several authors (Spirtes et al., 2000; Pearl, 2000) have recently formalized concepts related to causality using probability distributions defined on directed acyclic graphs. This line of research emphasizes the importance of understanding the process which generated the data, rather than only characterizing the joint distribution of the observed variables. The reasoning is that a causal understanding of the data is essential to be able to predict the consequences of interventions, such as setting a given variable to some specified value.

One of the main questions one can answer using this kind of theoretical framework is: 'Under what circumstances and in what way can one determine causal structure on the basis of observational data alone?'. In many cases it is impossible or too expensive to perform controlled experiments, and hence methods for discovering likely causal relations from uncontrolled data would be very valuable.

Existing discovery algorithms (Spirtes et al., 2000; Pearl, 2000) generally work in one of two settings. In the case of discrete data, no functional form for the dependencies is usually assumed.

*. Current address: The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan

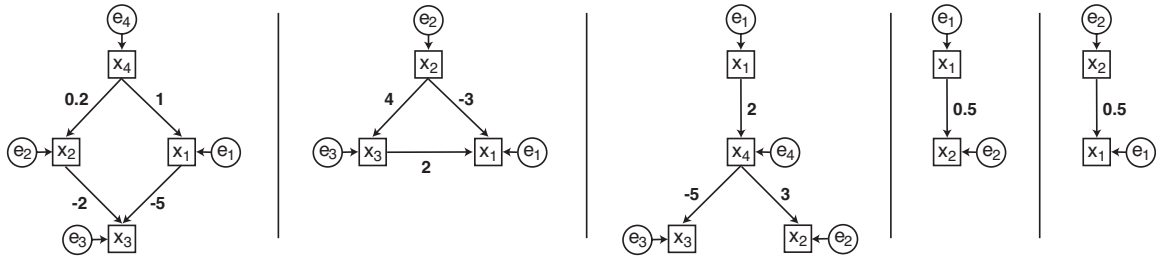

Figure 1: A few examples of data generating models satisfying our assumptions. For example, in the left-most model, the data is generated by first drawing the e_i independently from their respective non-Gaussian distributions, and subsequently setting (in this order) $x_4 = e_4$, $x_2 = 0.2x_4 + e_2$, $x_1 = x_4 + e_1$, and $x_3 = -2x_2 - 5x_1 + e_3$. (Here, we have assumed for simplicity that all the c_i are zero, but this may not be the case in general.) Note that the variables are not causally sorted (reflecting the fact that we usually do not know the causal ordering a priori), but that in each of the graphs they *can* be arranged in a causal order, as all graphs are directed acyclic graphs. In this paper we show that the full causal structure, including all parameters, are identifiable given a sufficient number of observed data vectors $\mathbf{x}$.

On the other hand, when working with continuous variables, a linear-Gaussian approach is almost invariably taken.

In this paper, we show that when working with continuous-valued data, a significant advantage can be achieved by departing from the Gaussianity assumption. While the linear-Gaussian approach usually only leads to a *set* of possible models, equivalent in their conditional correlation structure, a linear-*non-Gaussian* setting allows the full causal model to be estimated, with no undetermined parameters.

The paper is structured as follows.[1] First, in Section 2, we describe our assumptions on the data generating process. These assumptions are essential for the application of our causal discovery method, detailed in Sections 3 through 5. Section 6 discusses how one can test whether the found model seems plausible and proposes a statistical method for pruning edges. In Sections 7 and 8, we conduct a simulation study and provide real data examples to verify that our algorithm works as stated. We conclude the paper in Section 9.

2. Linear Causal Networks

Assume that we observe data generated from a process with the following properties:

1. The observed variables x_i, $i \in \{1,\ldots,m\}$ can be arranged in a *causal order*, such that no later variable causes any earlier variable. We denote such a causal order by $k(i)$. That is, the generating process is *recursive* (Bollen, 1989), meaning it can be represented graphically by a *directed acyclic graph* (DAG) (Pearl, 2000; Spirtes et al., 2000).

1. Preliminary results of the paper were presented at UAI2005 and ICA2006 (Shimizu et al., 2005, 2006b; Hoyer et al., 2006a).

2. The value assigned to each variable x_i is a *linear function* of the values already assigned to the earlier variables, plus a 'disturbance' (noise) term e_i, and plus an optional constant term c_i, that is

$$x_i = \sum_{k(j)<k(i)} b_{ij} x_j + e_i + c_i.$$

3. The disturbances e_i are all continuous-valued random variables with *non-Gaussian* distributions of non-zero variances, and the e_i are independent of each other, that is, $p(e_1,\ldots,e_m) = \prod_i p_i(e_i)$.

A model with these three properties we call a *Linear, Non-Gaussian, Acyclic Model*, abbreviated LiNGAM.

We assume that we are able to observe a large number of data vectors $\mathbf{x}$ (which contain the components x_i), and each is generated according to the above-described process, with the same causal order $k(i)$, same coefficients b_{ij}, same constants c_i, and the disturbances e_i sampled independently from the same distributions.

Note that the above assumptions imply that there are *no unobserved confounders* (Pearl, 2000).[2] Spirtes et al. (2000) call this the *causally sufficient* case. Also note that we do not require 'stability' in the sense as described by Pearl (2000), that is, 'faithfulness' (Spirtes et al., 2000) of the generating model. See Figure 1 for a few examples of data models fulfilling the assumptions of our model.

A key difference to most earlier work on the linear, causally sufficient, case is the assumption of non-Gaussianity of the disturbances. In most work, an explicit or implicit assumption of Gaussianity has been made (Bollen, 1989; Geiger and Heckerman, 1994; Spirtes et al., 2000). An assumption of Gaussianity of disturbance variables makes the full joint distribution over the x_i Gaussian, and the covariance matrix of the data embodies all one could possibly learn from observing the variables. Hence, all conditional correlations can be computed from the covariance matrix, and discovery algorithms based on conditional independence can be easily applied.

However, it turns out, as we will show below, that an assumption of *non*-Gaussianity may actually be more useful. In particular, it turns out that when this assumption is valid, the complete causal structure can in fact be estimated, without any prior information on a causal ordering of the variables. This is in stark contrast to what can be done in the Gaussian case: algorithms based only on second-order statistics (i.e., the covariance matrix) are generally not able to discern the full causal structure in most cases. The simplest such case is that of two variables, x_1 and x_2. A method based only on the covariance matrix has no way of preferring $x_1 \to x_2$ over the reverse model $x_1 \leftarrow x_2$; indeed the two are indistinguishable in terms of the covariance matrix (Spirtes et al., 2000). However, assuming non-Gaussianity, one can actually discover the direction of causality, as shown by Dodge and Rousson (2001) and Shimizu and Kano (2006). This result can be extended to several variables (Shimizu et al., 2006a). Here, we further develop the method so as to estimate the full model including all parameters, and we propose a number of tests to prune the graph and to see whether the estimated model fits the data.

2. A simple explanation is as follows: Denote by f hidden common causes and by $\mathbf{G}$ its connection strength matrix. Then a new model with hidden common causes f can be written as $\mathbf{x} = \mathbf{B}\mathbf{x} + \mathbf{G}f + e'$. Since common causes f introduce some dependency between $e = \mathbf{G}f + e'$, the new model is different from the LiNGAM model with independent (not merely uncorrelated) disturbances e. See Hoyer et al. (2006b) for details.

3. Model Identification Using Independent Component Analysis

The key to the solution to the linear discovery problem is to realize that the observed variables are linear functions of the disturbance variables, and the disturbance variables are mutually independent and non-Gaussian. If we as preprocessing subtract out the mean of each variable x_i, we are left with the following system of equations:

$$\mathbf{x} = \mathbf{B}\mathbf{x} + \mathbf{e}, \tag{1}$$

where $\mathbf{B}$ is a matrix that could be permuted (by simultaneous equal row and column permutations) to strict lower triangularity if one knew a causal ordering $k(i)$ of the variables (Bollen, 1989). (Strict lower triangularity is here defined as lower triangular with all zeros on the diagonal.) Solving for $\mathbf{x}$ one obtains

$$\mathbf{x} = \mathbf{A}\mathbf{e}, \tag{2}$$

where $\mathbf{A} = (\mathbf{I} - \mathbf{B})^{-1}$. Again, $\mathbf{A}$ could be permuted to lower triangularity (although not *strict* lower triangularity, actually in this case all diagonal elements will be *non-zero*) with an appropriate permutation $k(i)$. Taken together, Equation (2) and the independence and non-Gaussianity of the components of $\mathbf{e}$ define the standard linear *independent component analysis* model.

Independent component analysis (ICA) (Comon, 1994; Hyvärinen et al., 2001) is a fairly recent statistical technique for identifying a linear model such as that given in Equation (2). If the observed data is a linear, invertible mixture of non-Gaussian independent components, it can be shown (Comon, 1994) that the mixing matrix $\mathbf{A}$ is identifiable (up to scaling and permutation of the columns, as discussed below) given enough observed data vectors $\mathbf{x}$. Furthermore, efficient algorithms for estimating the mixing matrix are available (Hyvärinen, 1999).

We again want to emphasize that ICA uses non-Gaussianity (that is, more than covariance information) to estimate the mixing matrix $\mathbf{A}$ (or equivalently its inverse $\mathbf{W} = \mathbf{A}^{-1}$). For Gaussian disturbance variables e_i, ICA cannot in general find the correct mixing matrix because many different mixing matrices yield the same covariance matrix, which in turn implies the exact same Gaussian joint density (Hyvärinen et al., 2001). Our requirement for non-Gaussianity of disturbance variables stems from the same requirement in ICA.

While ICA is essentially able to estimate $\mathbf{A}$ (and $\mathbf{W}$), there are two important indeterminacies that ICA cannot solve: First and foremost, the order of the independent components is in no way defined or fixed (Comon, 1994). Thus, we could reorder the independent components and, correspondingly, the columns of $\mathbf{A}$ (and rows of $\mathbf{W}$) and get an equivalent ICA model (the same probability density for the data). In most applications of ICA, this indeterminacy is of no significance and can be ignored, but in LiNGAM, we can and we have to find the correct permutation as described in Section 4 below.

The second indeterminacy of ICA concerns the scaling of the independent components. In ICA, this is usually handled by assuming all independent components to have unit variance, and scaling $\mathbf{W}$ and $\mathbf{A}$ appropriately. On the other hand, in LiNGAM (as in SEM) we allow the disturbance variables to have arbitrary (non-zero) variances, but fix their weight (connection strength) to their corresponding observed variable to unity. This requires us to re-normalize the rows of $\mathbf{W}$ so that all the diagonal elements equal unity, before computing $\mathbf{B}$, as described in the LiNGAM algorithm below.

Our discovery algorithm, detailed in the next section, can be briefly summarized as follows: First, use a standard ICA algorithm to obtain an estimate of the mixing matrix $\mathbf{A}$ (or equivalently

of $\mathbf{W}$), and subsequently permute it and normalize it appropriately before using it to compute $\mathbf{B}$ containing the sought connection strengths b_{ij}.[3]

4. LiNGAM Discovery Algorithm

Based on the observations given in Sections 2 and 3, we propose the following causal discovery algorithm:

Algorithm A: LiNGAM discovery algorithm

1. Given an $m \times n$ data matrix $\mathbf{X}$ ($m \ll n$), where each column contains one sample vector $\mathbf{x}$, first subtract the mean from each row of $\mathbf{X}$, then apply an ICA algorithm to obtain a decomposition $\mathbf{X} = \mathbf{AS}$ where $\mathbf{S}$ has the same size as $\mathbf{X}$ and contains in its rows the independent components. From here on, we will exclusively work with $\mathbf{W} = \mathbf{A}^{-1}$.

2. Find the one and only permutation of rows of $\mathbf{W}$ which yields a matrix $\widetilde{\mathbf{W}}$ without any zeros on the main diagonal. In practice, small estimation errors will cause all elements of $\mathbf{W}$ to be non-zero, and hence the permutation is sought which minimizes $\sum_i 1/|\widetilde{\mathbf{W}}_{ii}|$.

3. Divide each row of $\widetilde{\mathbf{W}}$ by its corresponding diagonal element, to yield a new matrix $\widetilde{\mathbf{W}}'$ with all ones on the diagonal.

4. Compute an estimate $\widehat{\mathbf{B}}$ of $\mathbf{B}$ using $\widehat{\mathbf{B}} = \mathbf{I} - \widetilde{\mathbf{W}}'$.

5. Finally, to find a causal order, find the permutation matrix $\mathbf{P}$ (applied equally to both rows and columns) of $\widehat{\mathbf{B}}$ which yields a matrix $\widetilde{\mathbf{B}} = \mathbf{P}\widehat{\mathbf{B}}\mathbf{P}^T$ which is as close as possible to strictly lower triangular. This can be measured for instance using $\sum_{i \le j} \widetilde{\mathbf{B}}_{ij}^2$.

A complete Matlab code package implementing this algorithm is available online at our LiNGAM homepage: `http://www.cs.helsinki.fi/group/neuroinf/lingam/`

We now describe each of these steps in more detail.

In the first step of the algorithm, the ICA decomposition of the data is computed. Here, any standard ICA algorithm can be used. Although our implementation uses the FastICA algorithm (Hyvärinen, 1999), one could equally well use one of the many other algorithms available (see e.g., Hyvärinen et al., 2001). However, it is important to select an algorithm which can estimate independent components of many different distributions, as in general the distributions of the disturbance variables will not be known in advance. For example, FastICA can estimate both super-Gaussian and sub-Gaussian independent components, and we don't need to know the actual functional form of the non-Gaussian distributions (Hyvärinen, 1999).

Because of the permutation indeterminacy of ICA, the rows of $\mathbf{W}$ will be in random order. This means that we do not yet have the correct correspondence between the disturbance variables e_i and the observed variables x_i. The former correspond to the rows of $\mathbf{W}$ while the latter correspond to the columns of $\mathbf{W}$. Thus, our first task is to permute the rows to obtain a correspondence between the rows and columns. If $\mathbf{W}$ were estimated exactly, there would be only a single row permutation

3. It would be extremely difficult to estimate $\mathbf{B}$ directly using a variant of ICA algorithms, because we don't know the correct order of the variables, that is, the matrix $\mathbf{B}$ should be restricted to 'permutable to lower triangularity' not 'lower triangular' directly. This is due to the permutation problem illustrated in Appendix B.

that would give a matrix with no zeros on the diagonal, and this permutation would give the correct correspondence. This is because of the assumption of DAG structure, which is the key to solving the permutation indeterminacy of ICA. (A proof of this is given in Appendix A, and an example of the permutation problem is provided in Appendix B.)

In practice, however, ICA algorithms applied on finite data sets will yield estimates which are only approximately zero for those elements which should be exactly zero, and the model is only approximately correct for real data. Thus, our algorithm searches for the permutation using a cost function which heavily penalizes small absolute values in the diagonal, as specified in step 2. In addition to being intuitively sensible, this cost function can also be derived from a maximum-likelihood framework; for details, see Appendix C.

When the number of observed variables x_i is relatively small (less than eight or so) then finding the best permutation is easy, since a simple exhaustive search can be performed. However, for higher dimensionalities a more sophisticated method is required. We also provide such a permutation method for large dimensions; for details, see Section 5.

Once we have obtained the correct correspondence between rows and columns of the ICA decomposition, calculating our estimates of the b_{ij} is straightforward. First, we normalize the rows of the permuted matrix to yield a diagonal with all ones, and then remove this diagonal and flip the sign of the remaining coefficients, as specified in steps 3 and 4.

Although we now have estimates of all coefficients b_{ij} we do not yet have available a causal ordering $k(i)$ of the variables. Such an ordering (in general there may exist many if the generating network is not fully connected) is important for visualizing the resulting graph. A causal ordering can be found by permuting both rows and columns (using the same permutation) of the matrix $\widehat{\mathbf{B}}$ (containing the estimated connection strengths) to yield a strictly lower triangular matrix. If the estimates were exact, this would be a trivial task. However, since our estimates will not contain exact zeros, we will have to settle for approximate strict lower triangularity, measured for instance as described in step 5.[4]

It has to be noted that the computational stability of our method cannot be guaranteed. This is because ICA estimation is typically based on optimization of non-quadratic, possibly non-convex functions, and the algorithm might get stuck in local minima. Thus, for different random initial points used in the optimization algorithm, we might get different estimates of $\mathbf{W}$. An empirical observation is that typically ICA algorithms are relatively stable when the model holds, and unstable when the model does not hold. For a computational method addressing this issue, based on rerunning the ICA estimation part with different initial points, see Himberg et al. (2004).

5. Permutation Algorithms for Large Dimensions

In this section, we describe efficient algorithms for finding the permutations in steps 2 and 5 of the LiNGAM algorithm.

4. A reviewer pointed out that from a Bayesian viewpoint, the non-zero entries of the matrix $\widehat{\mathbf{B}}$ that would be zero in the infinite data case manifest a more general concept: the data cannot identify 'the' DAG structure, they can only help assign posterior probabilities to different structures.

5.1 Permuting the Rows of W

An exhaustive search over all possible row-permutations is feasible only in relatively small dimensions. For larger problems other optimization methods are needed. Fortunately, it turns out that the optimization problem can be written in the form of the classical *linear assignment problem*. To see this set $C_{ij} = 1/|\widetilde{\mathbf{W}}_{ij}|$, in which case the problem can be written as the minimization of

$$\sum_{i=1}^{m} C_{\phi(i),i},$$

where ϕ denotes the permutation to be optimized over. A great number of algorithms exist for this problem, with the best achieving worst-case complexity of $O(m^3)$ where m is the number of variables (see e.g., Burkard and Cela, 1999).

5.2 Permuting B to Get a Causal Order

It would be trivial to permute both rows and columns (using the same permutation) of $\widehat{\mathbf{B}}$ to yield a strictly lower triangular matrix if the estimates were exact, because one could use the following algorithm:

Algorithm B: Testing for DAGness, and returning a causal order if true

1. Initialize the permutation p to be an empty list
2. Repeat until $\widehat{\mathbf{B}}$ contains no more elements:
   (a) Find a row i of $\widehat{\mathbf{B}}$ containing all zeros, if not possible return **false**
   (b) Append i to the end of the list p
   (c) Remove the i-th row and the i-th column from $\widehat{\mathbf{B}}$
3. Return **true** and the found permutation p

However, since our estimates will not contain exact zeros, we will have to find a permutation such that setting the upper triangular elements to zero changes the matrix as little as possible. For instance, we could define our objective to be to minimize the sum of squares of elements on and above the diagonal, that is $\sum_{i\leq j} \widetilde{\mathbf{B}}_{ij}^2$ where $\widetilde{\mathbf{B}} = \mathbf{P}\widehat{\mathbf{B}}\mathbf{P}^T$ denotes the permuted $\widehat{\mathbf{B}}$, and $\mathbf{P}$ denotes the permutation matrix representing the sought permutation. In low dimensions, the optimal permutation can be found by exhaustive search. However, for larger problems this is obviously infeasible. Since we are not aware of any efficient method for exactly solving this combinatorial problem, we have taken another approach to handling the high-dimensional case.

Our approach is based on setting small (absolute) valued elements to zero, and testing whether the resulting matrix can be permuted to strict lower triangularity. Thus, the algorithm is:

Algorithm C: Finding a permutation of $\widehat{\mathbf{B}}$ by iterative pruning and testing

1. Set the $m(m+1)/2$ smallest (in absolute value) elements of $\widehat{\mathbf{B}}$ to zero

2. Repeat

   (a) Test if $\widehat{\mathbf{B}}$ can be permuted to strict lower triangularity (using Algorithm B above). If the answer is yes, stop and return the permuted $\widehat{\mathbf{B}}$, that is, $\widetilde{\mathbf{B}}$.

   (b) Additionally set the next smallest (in absolute value) element of $\widehat{\mathbf{B}}$ to zero

---

If in the estimated $\widehat{\mathbf{B}}$, all the true zeros resulted in estimates smaller than all of the true non-zeros, this algorithm finds the optimal permutation. In general, however, the result is not optimal in terms of the above proposed objective. However, simulations below show that the approximation works quite well.

6. Statistical Tests for Pruning Edges

The LiNGAM algorithm consistently estimates the connection strengths (and a causal order) if the model assumptions hold and the amount of data is sufficient. But what if our assumptions do not in fact hold? In such a case there is of course no guarantee that the proposed discovery algorithm will find true causal relationships between the variables.

The good news is that, in some cases, it is possible to detect violations of the model assumptions. In the following sections, we provide three statistical tests: i) testing significance of b_{ij} for pruning edges; ii) examining an overall fit of the model assumptions including estimated structure and connection strengths to data; iii) comparing two nested models. Then we propose a method for pruning edges of an estimated network using these statistical tests.

Unfortunately, however, it is never possible to completely confirm the assumptions (and hence the found causal model) purely from observational data. Controlled experiments, where the individual variables are explicitly manipulated (often by random assignment) and their effects monitored, are the only way to verify any causal model. Nevertheless, by testing the fit of the estimated model to the data we can recognize situations in which the assumptions clearly do not hold and reject models (e.g., Bollen, 1989). Only pathological cases constructed by mischievous data designers seem likely to be problematic for our framework. Thus, we think that a LiNGAM analysis will prove a useful first step in many cases for providing educated guesses of causal models, which might subsequently be verified in systematic experiments.

6.1 Wald Test for Examining Significance of Edges

After finding a causal ordering $k(i)$, we set to zero the coefficients of $\widehat{\mathbf{B}}$ which are implied zero by the order (i.e., those corresponding to the upper triangular part of the causally permuted connection matrix $\widetilde{\mathbf{B}}$). However, all remaining connections are in general non-zero. Even estimated connection strengths which are exceedingly weak (and hence probably zero in the generating model) remain and the network is fully connected. Both for achieving an intuitive understanding of the data, and especially for visualization purposes, a pruned network would be desirable. The Wald statistics provided below can be used to test which remaining connections should be pruned.

We would like to test if the coefficients of $\mathbf{B}$ are zero or not, which is equivalent to testing the coefficients of $\widetilde{\mathbf{W}}$ (see steps 3 and 4 in the LiNGAM algorithm above). Such tests are conducted to answer the fundamental question: Does the observed variable x_j have a statistically significant

effect on x_i? Here, the null and alternative hypotheses H_0 and H_1 are as follows:

$$H_0 : \widetilde{w}_{ij} = 0 \qquad \text{versus} \qquad H_1 : \widetilde{w}_{ij} \neq 0,$$

equivalently

$$H_0 : b_{ij} = 0 \qquad \text{versus} \qquad H_1 : b_{ij} \neq 0.$$

One can use the following Wald statistics

$$\frac{\widetilde{w}_{ij}^2}{\text{avar}(\widetilde{w}_{ij})},$$

to test significance of $\widetilde{w}_{ij}$ (or b_{ij}), where $\text{avar}(\widetilde{w}_{ij})$ denote the asymptotic variances of $\widetilde{w}_{ij}$ (see Appendix D for the complete formulas). The Wald statistics can be used to test the null hypothesis H_0. Under H_0, the Wald statistic asymptotically approximates to a chi-square variate with one degree of freedom (Bollen, 1989). Then we can obtain the probability of having a value of the Wald statistic larger than or equal to the empirical one computed from data. We reject H_0 if the probability is smaller than a significance level, and otherwise we accept H_0. Acceptance of H_0 implies that the assumption $\widetilde{w}_{ij} = 0$ (or b_{ij}) fits data. Rejection of H_0 suggests that the assumption is in error so that H_1 holds (Bollen, 1989). Thus, we can test significance of remaining edges using Wald statistics above.

6.2 A Chi-Square Test for Evaluating the Overall Fit of the Estimated Model

Next we propose a statistical measure using the model-based second-order moment structure to evaluate an overall fit of the model, for example, linearity, lower-triangularity (acyclicity), estimated structure and connection strengths, to data.

6.2.1 MOMENT STRUCTURES OF MODELS

First, we introduce some notations. For simplicity, assume $\mathbf{x}$ to have zero mean. Let us denote by $\sigma_2(\tau)$ the vector that consists of elements of the covariance matrix based on the model where any duplicates due to symmetry have been removed and by τ the vector of statistics of disturbances and coefficients of $\mathbf{B}$ that uniquely determines the second-order moment structures of the model $\sigma_2(\tau)$. Then the $\sigma_2(\tau)$ can be written as

$$\sigma_2(\tau) \quad = \quad \text{vec}^+\{E(\mathbf{x}\mathbf{x}^T)\}, \tag{3}$$

where $\text{vec}^+(\cdot)$ denotes the vectorization operator which transforms a symmetric matrix to a column vector by taking its non-duplicate elements. The parameter vector τ consists of free parameters of $\mathbf{B}$ and $E(e_i^2)$.

Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be a random sample from a LiNGAM model in (1), and define the sample counterparts to the moments in (3) as

$$m_2 \quad = \quad \frac{1}{n}\sum_{j=1}^{n} \text{vec}^+(\mathbf{x}_j \mathbf{x}_j^T).$$

Let us denote by τ_0 the true parameter vector. The $\sigma_2(\tau_0)$ can be estimated by the m_2 when n is enough large: $\sigma_2(\tau_0) \approx m_2$.

We now propose to evaluate the fit of the model by measuring the distance between the moments of the observed data m_2 and those based on the model $\sigma_2(\tau)$ in a weighted least-squares sense (see below for details). In the approach, a large residual can be considered as badness of fit of the model found to data, which would imply violation of the model assumptions. Thus, this approach gives information on validity of the assumptions.

6.2.2 SOME TEST STATISTICS TO EVALUATE A MODEL FIT

We provide some test statistics to examine an overall model fit. Here, the null and alternative hypotheses H_0 and H_1 are as follows:

$$H_0 : E(m_2) = \sigma_2(\tau) \quad \text{versus} \quad H_1 : E(m_2) \neq \sigma_2(\tau),$$

where $E(m_2)$ is the expectation of m_2. Assume that the fourth-order moments of $\mathbf{x}_i$ are finite. Let us denote by $\mathbf{V}$ the covariance matrix of m_2, which consists of fourth-order moments $\mathrm{cov}(x_i x_j, x_k x_l) = E(x_i x_j x_k x_l) - E(x_i x_j)E(x_k x_l)$. One can take a sample covariance matrix of m_2 as a nonparametric estimator $\widehat{\mathbf{V}}$ for $\mathbf{V}$.

Denote $\mathbf{J} = \partial\sigma_2(\tau)/\partial\tau^T$ and assume that $\mathbf{J}$ is of full column rank (see Appendix E for the exact form). Define

$$F(\widehat{\tau}) = \{m_2 - \sigma_2(\widehat{\tau})\}^T \widehat{\mathbf{M}} \{m_2 - \sigma_2(\widehat{\tau})\},$$

where

$$\begin{aligned} \widehat{\mathbf{M}} &= \widehat{\mathbf{V}}^{-1} - \widehat{\mathbf{V}}^{-1}\widehat{\mathbf{J}}(\widehat{\mathbf{J}}^T\widehat{\mathbf{V}}^{-1}\widehat{\mathbf{J}})^{-1}\widehat{\mathbf{J}}^T\widehat{\mathbf{V}}^{-1} \\ \widehat{\mathbf{J}} &= \left.\frac{\partial\sigma_2(\tau)}{\partial\tau^T}\right|_{\tau=\widehat{\tau}}. \end{aligned} \tag{4}$$

Then a test statistic $T_1 = n \times F(\widehat{\tau})$ could be used to test the null hypothesis H_0, that is, to examine a fit of the model considered to data. Under H_0, the statistic T_1 asymptotically approximates to a chi-square variate with degrees $u - v$ of freedom where u is the number of distinct moments employed and v is the number of parameters employed to represent the second-order moment structure $\sigma_2(\tau)$, that is, the number of elements of τ. The required assumption for this is that $\hat{\tau}$ is a $\sqrt{n}$-consistent estimator. No asymptotic normality is required (see Browne, 1984, for details). Acceptance of H_0 implies that the model assumptions fit data. Rejection of H_0 suggests that at least one model assumption is in error so that H_1 holds (Bollen, 1989). Thus, we can assess the overall fit of the estimated model to data.

However, it is often pointed out that this type of test statistics requires large sample sizes for T_1 to behave like a chi-square variate (e.g., Hu et al., 1992). Therefore, we would apply a proposal by Yuan and Bentler (1997) to T_1 to improve its chi-square approximation and employ the following test statistic T_2:

$$T_2 = \frac{T_1}{1 + F(\widehat{\tau})}.$$

6.2.3 A DIFFERENCE CHI-SQUARE TEST FOR MODEL COMPARISON OF NESTED MODELS

Let us consider the comparison of two models that are nested, that is, one is a simplified model of the other. Assume that Models 1 and 2 have q and $q-1$ edges, and Model 2 is a simplified version of

Model 1 by pruning one edge out. Denote by $T_2(q)$ and $T_2(q-1)$ the model fit statistics for Models 1 and 2, respectively. Then, the difference between $T_2(q) - T_2(q-1)$ asymptotically approximates to a chi-square variate with one degree of freedom (e.g., Bollen, 1989), by which we can test if the two models with q and $q-1$ edges have significantly different model fits. In principle, a more complex model fits better. If the two model fits are significantly different, the edge should not be pruned since the model fit becomes significantly worse. This means that we examine significance of the edge in terms of overall model fit.

6.3 A Method for Pruning Edges

Using the tests developed above, we now propose a sophisticated method for pruning the edges (connection strengths).

The Wald statistics above tell us how likely each edge is, which can be considered an evaluation of the individual fit of each edge to data. On the other hand, the chi-square test assesses the overall model fit by measuring the residual between the data covariance matrix and model-based covariance matrix. A straightforward approach would be to test the significance of remaining edges using Wald statistics only. That is, we prune all the non-significant edges with the p values higher than a significance level, for example, 0.05 (5%). However, it would be more effective (e.g., the test has more power) to use both the individual and overall fits for assessing significance of edges. Furthermore, it is also important that the pruned estimated model is accepted by the chi-square test of model fit. Thus, we propose a pruning method utilizing all the three tests above, Wald test, the chi-square test and the difference test (see Section 6.2.3 for the difference test). The algorithm is as follows:

Algorithm D: Pruning edges using Wald test, model fit test and difference test.

1. Set a significance level α (e.g., 0.05)
2. Find non-significant edges by applying Wald test to each edge
3. Set the least significant strictly lower triangular element of $\widetilde{\mathbf{B}}$ (in Step 5 of the LiNGAM discovery algorithm) among the non-significant edges accepted by Wald test to zero
4. Repeat until all the non-significant edges by Wald test are examined
   (a) Test if the overall model fits for the last model and current model with one less edge than the last model are significantly different by the difference test. Further test the model fit of the current model by the chi-square test. If both null hypotheses are accepted in the two tests, adopt the current model, that is, prune the edge out. Otherwise, adopt the last model, that is, do not prune the edge.
   (b) Additionally set the next least significant element of $\widetilde{\mathbf{B}}$ to zero
5. Return the pruned $\widetilde{\mathbf{B}}$

The pruned $\widehat{\mathbf{B}}$ can be obtained by the relation $\widehat{\mathbf{B}} = \mathbf{P}^T \widetilde{\mathbf{B}} \mathbf{P}$ (see step 5 in the LiNGAM algorithm). Thus, we would be able to find a pruned network that fits data. We conduct a simulation to study the empirical performance of this pruning method (Section 7.2).

A potential alternative to Wald statistics would be to use resampling techniques (e.g., Efron and Tibshirani, 1993). We provide a basic method using resamplings as an option in our Matlab code. In our implementation we take the causal ordering obtained from the LiNGAM algorithm, and then simply estimate the connection strengths using covariance information alone for different resamplings of the original data. In this way, it is possible to obtain measures of the variances of the estimates of the b_{ij}, and use these variances to prune those edges whose estimated means are low compared with their standard deviations. Future versions of our software packages should incorporate the more advanced methods including bootstrapping.

The issue of multiple comparisons also arises in this context. Usually, $\widetilde{\mathbf{W}}$ and $\mathbf{B}$ have more than one element. In many cases, we need to perform more than one test simultaneously to find out if all or a set of the coefficients are significantly large in an absolute value sense. Although a given significance level may be appropriate for each individual test, it is not for the set of all the tests. We could have a lot of spurious significance if we just repeat tests without any corrections. In such a case, it would be effective to employ multiple comparison procedures (see Hochberg and Tamhane, 1987, for details). A simple and basic method is the Bonferroni correction, where we simply divide a significance level by the number of tests to obtain the significance level for individual test. However, it is often pointed out that the Bonferroni method is too conservative when the number of tests is large. Some authors have improved the Bonferroni procedure or devised new techniques so that they have more power of test (e.g., Benjamini and Hochberg, 1995; Hochberg, 1988; Holm, 1979; Simes, 1986). We would like to study such multiple comparison techniques in future work and implement them in our software package.[5]

7. Simulations

To verify the validity of our method (and of our Matlab code), we performed extensive experiments with simulated data. All experimental code (including the precise code to produce Figures 2, 3, 4 and Table 7.2) is included in the LiNGAM code package.

7.1 Estimation of B

We repeatedly performed the following experiment:

1. First, we randomly constructed a strictly lower-triangular matrix $\mathbf{B}$. Various dimensionalities (3, 5, 10, 20 and 100) were used. Both fully connected (no zeros in the strictly lower triangular part) and sparse networks (many zeros) were tested. We also randomly selected variances of the disturbance variables and values for the constants c_i.

2. Next, we generated data by independently drawing the disturbance variables e_i from Gaussian distributions and subsequently passing them through a power non-linearity (raising the absolute value to an exponent in the interval [0.5, 0.8] or [1.2, 2.0], but keeping the original sign) to make them non-Gaussian. Various data set sizes (200, 1000 and 5000) were tested. The e_i were then scaled to yield the desired variances, and the observed data $\mathbf{X}$ was generated according to the assumed recursive process.

5. It would also be possible to devise a Bayesian technique for scoring models as proposed by Geiger and Heckerman (1994) if we knew the distributions of non-Gaussian disturbances. However, in practice, it is quite difficult to model the exact functional form of the non-Gaussian distributions, and therefore it would be difficult to score the models requiring parametric models.

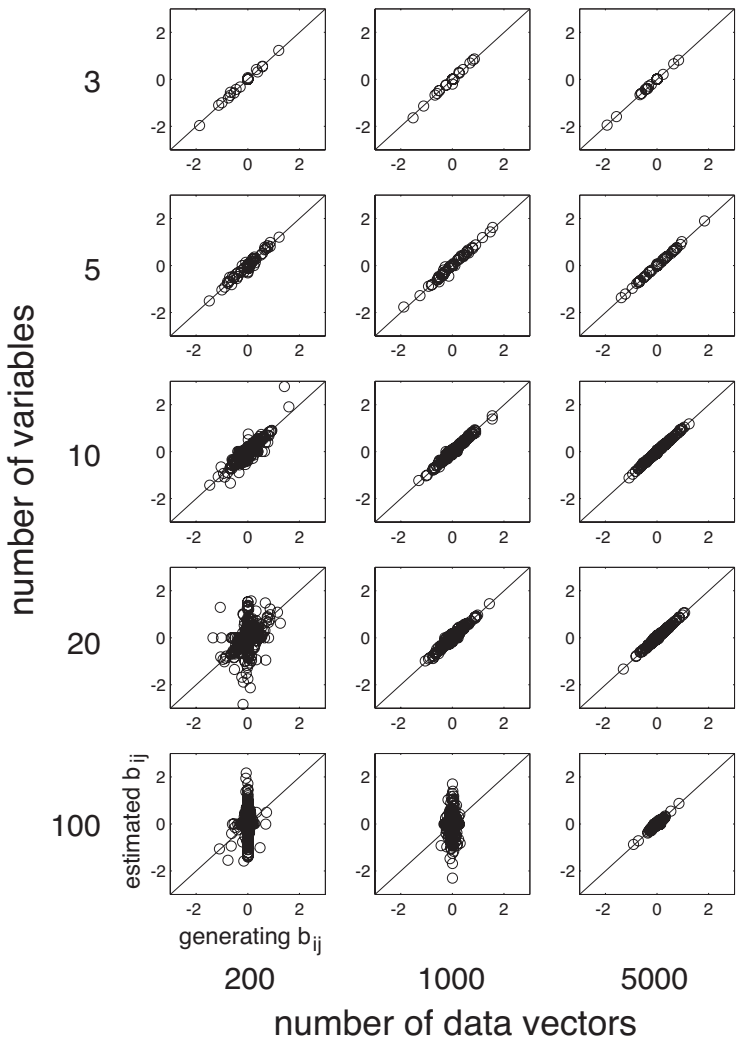

Figure 2: Scatterplots of the estimated b_{ij} versus the original (generating) values. The different plots correspond to different numbers of variables and different numbers of data vectors. Five data sets were generated for each scatterplot. For the last two rows, 1,000 plot points were randomly selected and plotted to improve the clarity of the figures.

3. Before feeding the data to the LiNGAM algorithm, we randomly permuted the rows of the data matrix $\mathbf{X}$ to hide the causal order with which the data was generated. At this point, we also permuted $\mathbf{B}$, the c_i, as well as the variances of the disturbance variables to match the new order in the data.

4. Finally, we fed the data to our discovery algorithm, and compared the estimated parameters to the generating parameters. In particular, we made a scatterplot of the entries in the estimated matrix $\widehat{\mathbf{B}}$ against the corresponding ones in $\mathbf{B}$.

Since the number of different possible parameter configurations is limitless, we feel that the reader is best convinced by personally running the simulations using various settings. Nevertheless, we here show some representative results.

Figure 2 gives combined scatterplots of the elements of **B** versus the generating coefficients. The different plots correspond to different dimensionalities (numbers of variables) and different data sizes (numbers of data vectors), where each plot combines the data for a number of different network sparseness levels and non-linearities. Although for very small data sizes the estimation often fails, when the data size grows the estimation works practically flawlessly, as evidenced by the grouping of the data points onto the main diagonal.

In summary, the experiments verify the correctness of the method and demonstrate that reliable estimation is possible even with fairly limited amounts of data. We note that for larger dimensions we clearly need more data, but the amounts of data required are still reasonable.

7.2 Pruning Edges

We examined the performance of the pruning method developed in Section 6.3 using artificial data. The simulation consisted of 1000 trials. In each trial, we generated five- and ten-dimensional data of sample size $n = 1000, 5000, 10000$ in the same manner as in Section 7.1 above.

The LiNGAM discovery algorithm was then applied to the data. We subsequently applied the pruning method to the estimated networks. The significance level was set at 5%. Then we computed the numbers of correctly identified edges (true positives) and the numbers of correctly identified absence of edges (true negatives) only in the strictly lower triangular part of the matrix **B** to see the performance of our pruning method. We also counted how many edges were falsely added (false positives) and how many were falsely missing (false negatives).

	True pos.	False neg.	True neg.	False pos.	Sums of false pos. and neg.
Dim.=5					
n =1000	8101 (90.5%)	849 (9.5%)	921 (87.7%)	129 (12.3%)	978 (9.8%)
5000	8556 (95.6%)	394 (4.4%)	943 (89.8%)	107 (10.2%)	501 (5.0%)
10000	8691 (97.1%)	259 (2.9%)	972 (92.6%)	78 (7.4%)	337 (3.4%)
Dim.=10					
n =1000	27825 (80.6%)	6698 (19.4%)	8171 (78.0%)	2306 (22.0%)	9004 (20.0%)
5000	31623 (91.6%)	2900 (8.4%)	9350 (89.2%)	1127 (10.8%)	4027 (8.9%)
10000	32477 (94.1%)	2046 (5.9%)	9466 (90.4%)	1011 (9.6%)	3057 (6.8%)

Table 1: Numbers of true positives, false negatives, true negatives, and false positives (1000 trials). n is sample size.

The results are shown in Table 7.2. Some representative pruned estimated networks are shown in Figures 3 and 4.[6] First, we examine the numbers of false positives for the edges that had non-zero values. The false positive rates were approximately 10% except the case with sample size 1000 for both 5 and 10 variables. Second, we see the statistical power of the test (numbers of true positives)

6. Graphs were plotted using the latest version of the LiNGAM package which connects seamlessly to the free Graphviz software, a sophisticated tool for plotting graphs.

for the other edges that had non-zero values. The power of 0.90 (8055 true positives for 5 variables and 31071 true positives for 10 variables) was achieved for all the conditions other than when the number of variables was 10 and the sample size was 1000. Finally, we would mention that the sums of the two errors (false negatives and false positives) were small enough (less than 10%) except for the case with the number of variables 10 and the sample size 1000. Thus, Table 7.2 implied that our pruning method worked well for reasonable sample sizes.

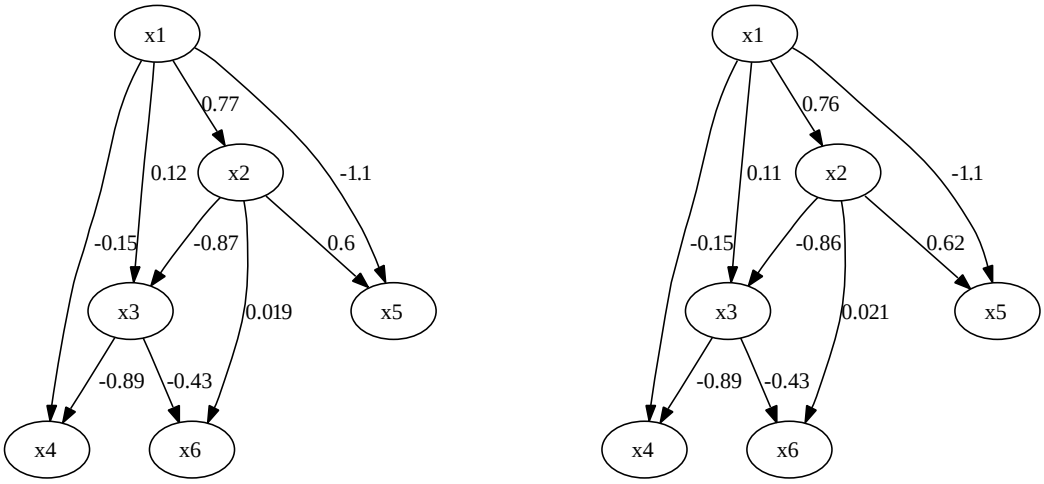

Figure 3: Left: example original network. Right: estimated network. The sample size was 10000. The structure of the original network was correctly estimated, and all the edge strengths were approximately correct.

8. Examples With Real-World Data

As a real-world example, we have applied the LiNGAM analysis to a set of time series. As a cause must precede its effect in time, we can expect the LiNGAM analysis to find the correct time ordering of the variables in any data generated from a LiNGAM model.

A time series can be approximated by a LiNGAM model if it is a stationary AR(p) process. An AR(p) process, or an *autoregressive process* of order p, is defined by the equation (Box and Jenkins, 1976)

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + a_t.$$

That is, the value of X_t is a weighted sum of p previous variables and white noise (a_t). The weights $\phi_1, \phi_2, \ldots, \phi_p$ are called the parameters of the process. A process is *stationary*, if the variance is finite, the mean remains the same over time, and the autocovariance function depends only on the time lag of two variables (Brockwell and Davis, 1987). The last condition also implies that the variance remains the same over time. If we want to approximate a stationary AR(p) process by a LiNGAM model, the white noise process must be non-Gaussian.

A time series must be presented to the LiNGAM analysis as a multivariate data set. To do this, the time series is divided into time windows with the same size as the number of variables in the

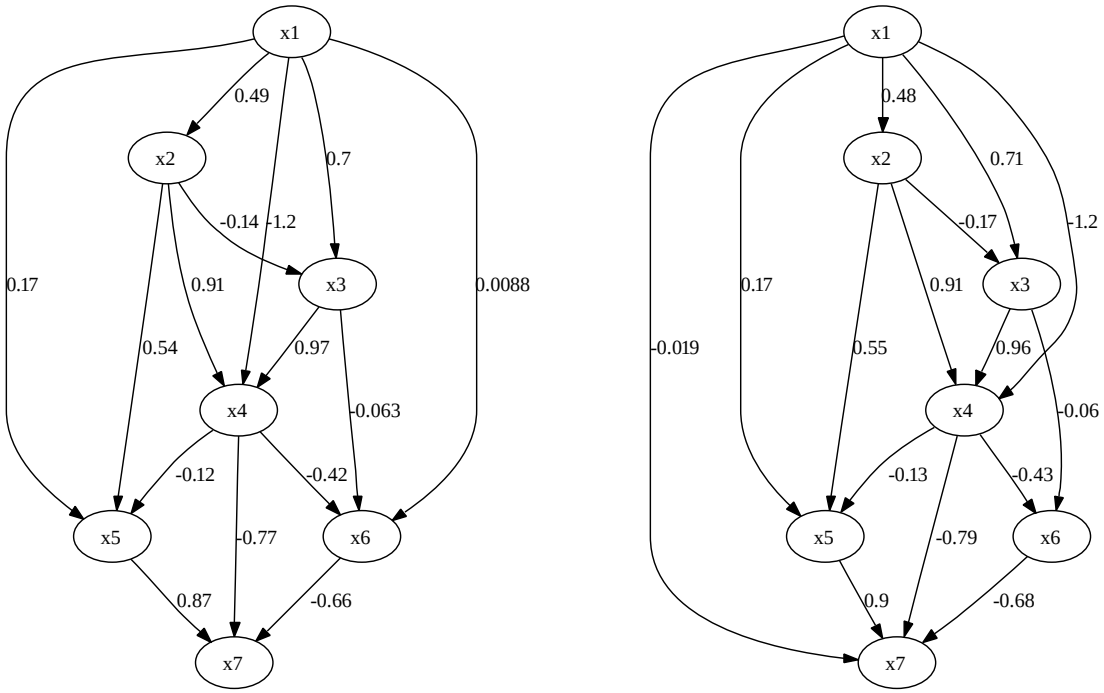

Figure 4: Left: example original network. Right: estimated network. The sample size was 10000. This shows what kind of mistakes the LiNGAM algorithm might make. The estimated network had one added edge ($x_1 \to x_7$) and one missing edge ($x_1 \to x_6$). However, both the added and missing edges had quite low strengths ($x_1 \to x_7$, -0.019 and $x_1 \to x_6$, 0.0088). Note that the other edges were correctly identified, and the connection strengths were approximately correct as well.

LiNGAM model. These time windows are then treated as samples of multivariate data. This introduces confounding variables to the model, against the assumptions of the LiNGAM analysis. To see this, consider an example of an AR(2) process, and a time window of three variables (Figure 5). Here, variables X_t and X_{t+1} are confounded by a variable outside the time window. In a general case, an AR(p) process introduces confounding variables to p first variables in a time window. The LiNGAM model holds strictly only for first order processes.

For the tests, a total of 22 data sets were selected from time series data repositories on the Internet (Hyndman, 2005; Statistical Software Information; National Statistics). We did not seek data sets that would fit the LiNGAM model, but a diverse set of data to see how well the LiNGAM analysis will perform with real-world data, when the assumptions of the model are violated at least to some extent. The data sets can be roughly categorized as economic time series and environmental time series. Economic time series included data sets like currency exchange rates and stock rates. Environmental time series included a more diverse set of data, ranging from monthly river-flows to daily temperatures. Before the tests, the sample autocorrelation and partial autocorrelation functions for the series were analyzed to gain insight into how well the series actually fit the AR(p) model.

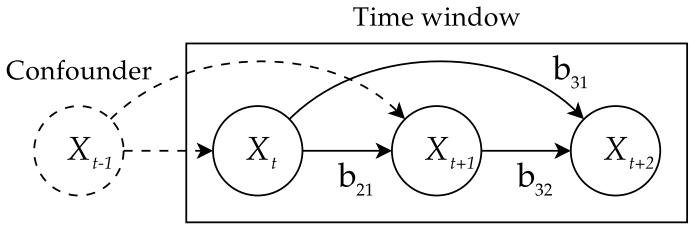

Figure 5: An example of confounding. Variables X_t and X_{t+1} are confounded by a variable outside the time window.

The LiNGAM analysis was run for each data set with varying parameters. The number of variables in the LiNGAM model was 3, 5, or 7, corresponding to AR(p) processes of orders 2, 4, and 6. Since the partial autocorrelations of economic time series indicated that the processes are at most second order processes, only models with 3 or 5 variables were tested for them. All possible time windows were used as multivariate data samples, including overlapping windows. For each number of variables, the LiNGAM analysis was run many times (100) to see if the analysis produced consistent results, where initial points of FastICA algorithm were randomized to assess the computational stability, that is, the effect of local minima.

Keeping in mind that there are possibly deviations from the LiNGAM model in the data, there are different possible results for applying the LiNGAM analysis. If the data generating process is indeed a stationary AR(p) process with non-Gaussian noise, we can expect to find the correct time ordering of the variables. An important nonstationary process, commonly encountered in economic time series, is the random walk $X_t = X_{t-1} + a_t$. If the generating process is a random walk, we cannot determine the direction of time. Hence, both the correct time ordering and the reverse time ordering are possible results. It is also possible that the variables are not causally related, and the weight matrix **B** is a matrix, none of which elements are significantly different from zero. In this case, any ordering of the variables is plausible. Also, any of the assumptions may fail to hold in real-world data: the process might be non-linear, the error terms might have Gaussian distribution, or there might be confounding variables. In this case, the LiNGAM analysis will probably fail, producing unpredictable results.

Reflecting the possibilities explained above, four distinct categories of results were distinguished from the tests.

1. The correct causal order was found (5 cases).
2. The reverse causal order was found (9 cases).
3. The **B** matrix was estimated as a zero matrix (1 case).
4. No consistent estimate for causal order was found (7 cases).

For the first three categories, the estimates were consistent over all experiments. The last category included data sets for which no consistent estimate was found. It also included some data sets for which a consistent estimate was found for some test setting, but not for all settings. An example from each category is provided for further analysis. The examples are listed Table 2, together with

their sample sizes and descriptions. Figure 6 plots the example series. The sample autocorrelation function (acf) and the partial autocorrelation function (pacf) are plotted for each example in Figure 7. The horizontal lines in pacf plots are 99% (solid line) and 95% (dashed line) confidence intervals for the null hypothesis that the partial autocorrelation is zero.

Name	Size	Description
PEAS	768	Monthly precipitation in Eastport, USA [mm].
DAILYIBM	3333	Daily closing price of IBM stock.
PRECIP	792	Daily precipitation in Hveravellir [mm].
MLCO2	372	Monthly carbon dioxide above Mauna Loa, Hawaii [parts per million].

Table 2: Sample sizes and descriptions of the example data sets.

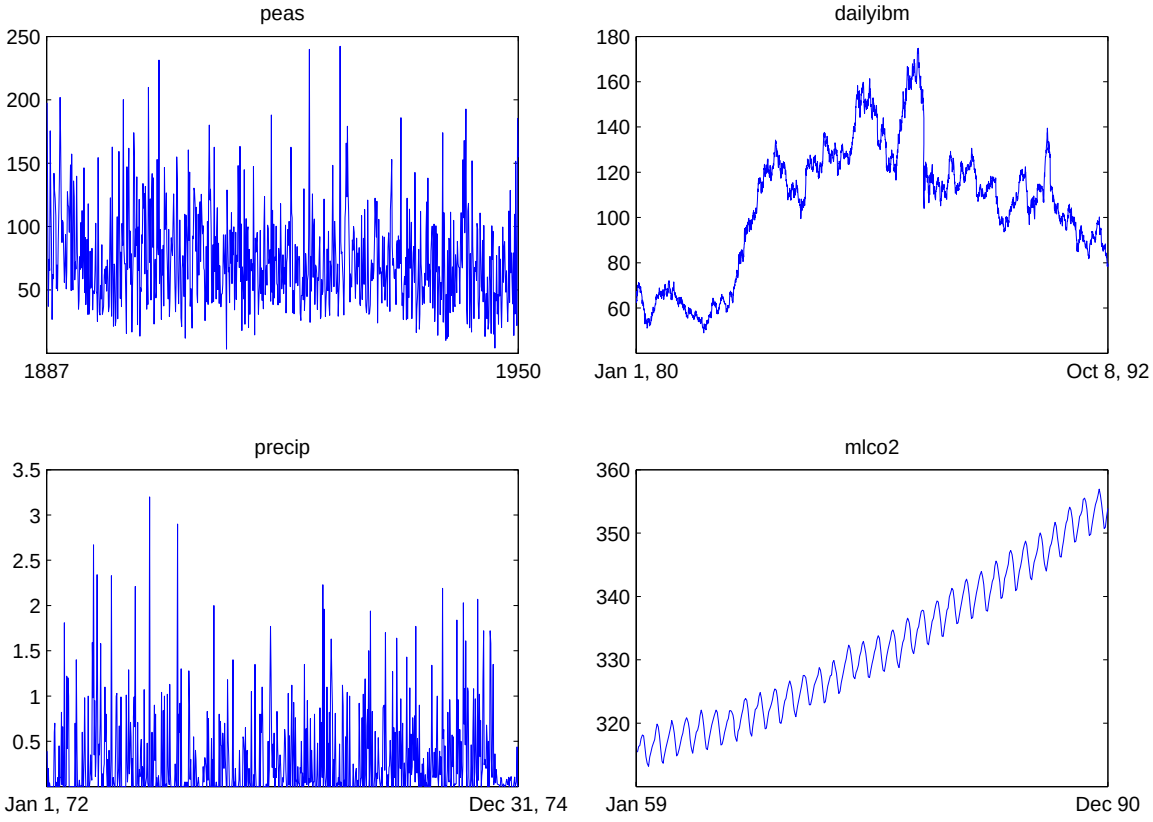

Figure 6: Plots for the example data sets.

The correct causal order was found from the PEAS data set. The statistical properties of the data indicate that the process could be modeled fairly well as an AR(2) process. There are no signs of a trend in the sample autocorrelation function, and the estimated partial correlations are significant for time lags 1 and 2. Still, there seems to be a small seasonal component, reflecting yearly seasonality

of precipitation. The most frequent model estimated by LiNGAM was an AR(2) process with small parameters, in accordance with the estimated partial correlations.

The reverse causal order was found from the DAILYIBM data set. The acf and the pacf of the data resemble those of a random walk, thus the result is expected. Also the estimated causal model is close to a random walk, values of nearly one are estimated consistently for the first parameter of the AR(p) process, other weight estimates being zero.

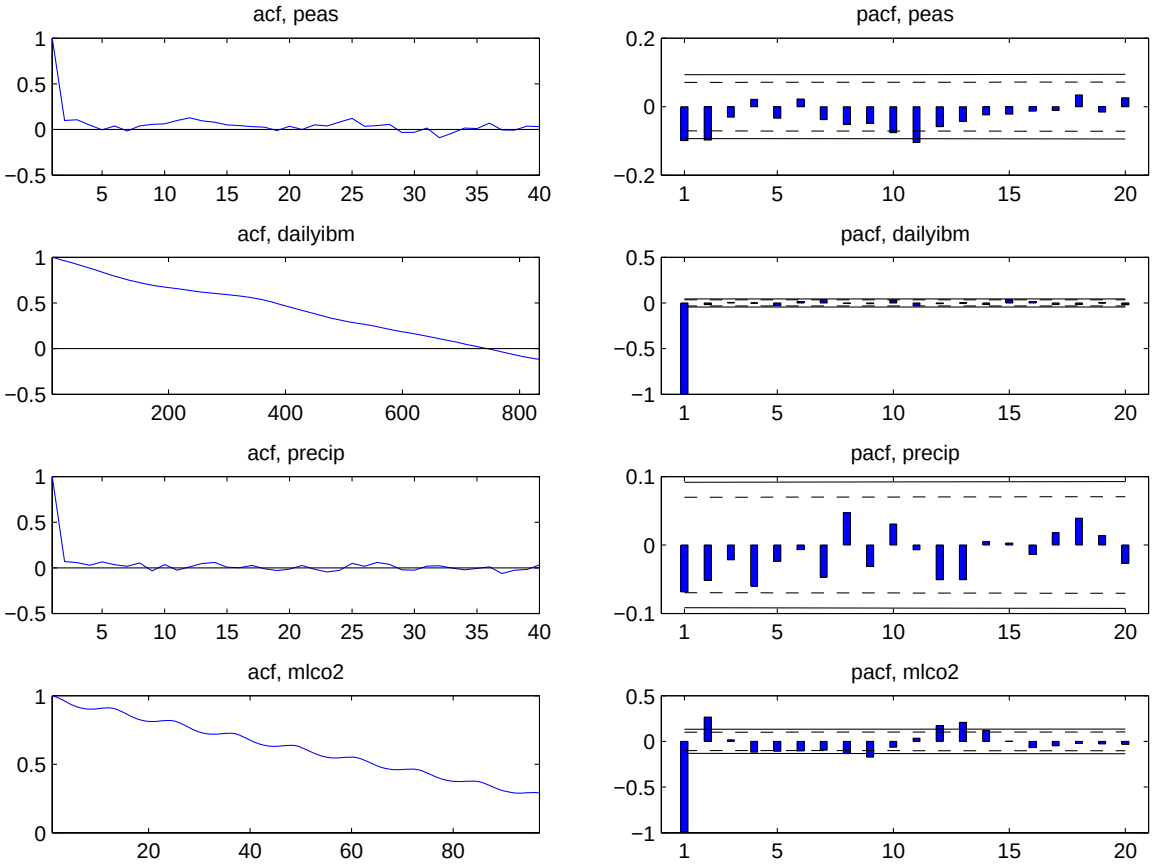

Figure 7: The sample autocorrelation and partial autocorrelation functions for the example data sets.

For the PRECIP data set, the estimates for causal order were consistent, but not consistently either correct or reverse. Most of the time, the estimated **B** matrix was a zero matrix. As a zero **B** matrix is consistent with any time ordering of the variables, this largely explains the results. For the cases when the **B** matrix was not a zero matrix, the estimated weights were small (less than 0.1), and did not correspond to an AR(p) model of any order, but rather random estimation errors. The pacf of this series supports the results of the LiNGAM analysis: none of the partial autocorrelations is statistically significant.

For the MLCO2 data set, the LiNGAM analysis produced inconsistent estimates of the causal order. There are several possible reasons for this. First of all, the series has a trend and a seasonal

component. More probably, the basic assumptions of the LiNGAM analysis are violated. The process might be non-linear, or the level of carbon dioxide might be caused by other environmental variables, leading to a confounded model. It is also possible, although unlikely, that the data is Gaussian.

9. Conclusions

Developing methods for causal inference from non-experimental data is a fundamental problem with a very large number of potential applications. Although one can never fully prove the validity of a causal model from observational data alone, such methods are nevertheless crucial in cases where it is impossible or very costly to perform experiments.

Previous methods developed for linear causal models (Bollen, 1989; Spirtes et al., 2000; Pearl, 2000) have been based on an explicit or implicit assumption of Gaussianity, and have hence been based solely on the covariance structure of the data. Because of this, additional information (such as the time-order of the variables) is usually required to obtain a full causal model of the variables. Without such information, algorithms based on the Gaussian assumption cannot in most cases distinguish between multiple equally possible causal models.

In this paper, we have shown that an assumption of non-Gaussianity of the disturbance variables, together with the assumption of linearity and causal sufficiency, allows the causal model to be completely identified. Furthermore, we have proposed a practical algorithm which estimates the causal structure under these assumptions and provided a number of tests to prune the graph and to see whether the estimated model fits the data.

The practical value of the LiNGAM analysis needs to be determined by applying it to real-world data sets and comparing it to other methods for causal inference from non-experimental data. The real data examples reported here are rather limited. Also, in many cases involving real-world data, practitioners in the field already have a fairly good understanding of the causal processes underlying the data. An interesting question is how well methods such as ours do on such data sets. These are important topics for future work.

Acknowledgments

This work was partially carried out at Division of Mathematical Science, Graduate School of Engineering Science, Osaka University and Transdisciplinary Research Integration Center, Research Organization of Information and Systems. The authors would like to thank Aristides Gionis, Heikki Mannila, and Alex Pothen for discussions relating to algorithms for solving the permutation problems, Niclas Börlin for contributing the Matlab code for solving the assignment problem, Yutaka Kano and Michiwo Kanekiyo for comments on the manuscript and Michael Jordan and three anonymous reviewers for useful comments to improve this paper. S.S. was supported by Grant-in-Aid for Scientific Research from Japan Society for the Promotion of Science. P.O.H. was supported by the Academy of Finland project #204826. A.H. was supported by the Academy of Finland through an Academy Research Fellow Position and project #203344.

Appendix A. Proof of Uniqueness of Row Permutation

Here, we show that, were the estimates of ICA exact, there is only a single permutation of the rows of $\mathbf{W}$ which results in a diagonal with no zero entries.

It is well-known (Bollen, 1989) that the DAG structure of the network guarantees that for some permutation of the variables, the matrix $\mathbf{B}$ is strictly lower-triangular. This implies that the correct $\widetilde{\mathbf{W}}$ (where the disturbance variables are aligned with the observed variables) can be permuted to lower-triangular form (with no zero entries on the diagonal) by equal row and column permutations, that is,

$$\widetilde{\mathbf{W}} = \mathbf{P}_d \mathbf{M} \mathbf{P}_d^T,$$

where $\mathbf{M}$ is lower-triangular and has no zero entries on the diagonal, and $\mathbf{P}_d$ is a permutation matrix representing a causal ordering of the variables. Now, ICA returns a matrix with randomly permuted rows,

$$\mathbf{W} = \mathbf{P}_{\text{ica}} \widetilde{\mathbf{W}} = \mathbf{P}_{\text{ica}} \mathbf{P}_d \mathbf{M} \mathbf{P}_d^T = \mathbf{P}_1 \mathbf{M} \mathbf{P}_2^T,$$

where $\mathbf{P}_{\text{ica}}$ is the random ICA row permutation, and on the right we have denoted by $\mathbf{P}_1 = \mathbf{P}_{\text{ica}} \mathbf{P}_d$ and $\mathbf{P}_2 = \mathbf{P}_d$, respectively, the row and column permutations from the lower triangular matrix $\mathbf{M}$.

We now prove that $\mathbf{W}$ has no zero entries on the diagonal if and only if the row and column permutations are equal, that is, $\mathbf{P}_1 = \mathbf{P}_2$. Hence, there is only one row permutation of $\mathbf{W}$ which yields no zero entries on the diagonal, and it is the one which finds the correspondence between the disturbance variables and the observed variables.

Lemma 1 *Assume* $\mathbf{M}$ *is lower triangular and all diagonal elements are non-zero. A permutation of rows and columns of* $\mathbf{M}$ *has only non-zero entries in the diagonal if and only if the row and column permutations are equal.*

Proof First, we prove that if the row and columns permutations are not equal, there will be zero elements in the diagonal.

Denote by $\mathbf{K}$ a lower triangular matrix of all ones in the lower triangular part. Denote by $\mathbf{P}_1$ and $\mathbf{P}_2$ two permutation matrices. The number of non-zero diagonal entries in a permuted version of $\mathbf{K}$ is $\text{tr}(\mathbf{P}_1 \mathbf{K} \mathbf{P}_2^T)$. This is the maximum number of non-zero diagonal entries when an arbitrary lower triangular matrix is permuted.

We have $\text{tr}(\mathbf{P}_1 \mathbf{K} \mathbf{P}_2^T) = \text{tr}(\mathbf{K} \mathbf{P}_2^T \mathbf{P}_1)$. Thus, we can consider permutations of columns only, given by $\mathbf{P}_2^T \mathbf{P}_1$. Assume the columns of $\mathbf{K}$ are permuted so that the permutation is not equal to identity. Then, there exists an index i so that the column of index i has been moved to column index j where $j < i$ (If there were no such columns, all the columns would be moved to the right, which is impossible.) Obviously, the diagonal entry in the j-th column in the permuted matrix is zero. Thus, any column permutation not equal to the identity creates at least one zero entry in the diagonal.

Thus, to have non-zero diagonal, we must have $\mathbf{P}_2^T \mathbf{P}_1 = \mathbf{I}$. This means that the column and row permutations must be equal.

Next, assume that the row and column permutations are equal. Consider $\mathbf{M} = \mathbf{I}$ as a worst-case scenario. Then the permuted matrix equals $\mathbf{P}_1 \mathbf{I} \mathbf{P}_2^T$ which equals identity, and all the diagonal elements are non-zero. Adding more non-zero elements in the matrix only increases the number of non-zero elements in the permuted version.

Thus, the lemma is proven.

Appendix B. An Example of the Permutation Problem

Here we show with an example that if the permutation is not correctly determined, the parameters b_{ij} can have very different values, yet give the same data distribution. This example considers the general case where the system is not DAG. For simplicity, let us consider the two variables case. Assume we parameterize the mixing model in (2) as

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{1}{1-b_{12}b_{21}} \begin{bmatrix} 1 & b_{12} \\ b_{21} & 1 \end{bmatrix} \mathrm{diag}(\sigma_1,\sigma_2) \begin{bmatrix} s_1 \\ s_2 \end{bmatrix},$$

where σ_1 and σ_2 are standard deviations of e_1 and e_2, and s_1 and s_2 are normalized versions of e_1 and e_2, that is, e_1/σ_1 and e_2/σ_2.

Then, take the following new set of parameters:

$$\begin{aligned} b'_{12} &= 1/b_{21} \\ b'_{21} &= 1/b_{12} \\ \sigma'_1 &= \sigma_2/b_{21} \\ \sigma'_2 &= \sigma_1/b_{12}, \end{aligned}$$

and do the permutation and sign change:

$$\begin{aligned} s'_1 &= -s_2 \\ s'_2 &= -s_1. \end{aligned}$$

Then, the two parameterizations give the same data, that is, the same model fit. This is because

$$\begin{aligned} &\frac{1}{1-b'_{12}b'_{21}} \begin{bmatrix} 1 & b'_{12} \\ b'_{21} & 1 \end{bmatrix} \mathrm{diag}(\sigma'_1,\sigma'_2) \begin{bmatrix} s'_1 \\ s'_2 \end{bmatrix} \\ &= \frac{1}{1-1/(b_{12}b_{21})} \begin{bmatrix} 1 & 1/b_{21} \\ 1/b_{12} & 1 \end{bmatrix} \mathrm{diag}(\sigma_2/b_{21},\sigma_1/b_{12}) \begin{bmatrix} -s_2 \\ -s_1 \end{bmatrix} \\ &= \frac{-b_{12}b_{21}}{1-b_{12}b_{21}} \begin{bmatrix} 1/b_{21} & 1/(b_{21}b_{12}) \\ 1/(b_{12}b_{21}) & 1/b_{12} \end{bmatrix} \mathrm{diag}(\sigma_2,\sigma_1) \begin{bmatrix} -s_2 \\ -s_1 \end{bmatrix} \\ &= \frac{b_{12}b_{21}}{1-b_{12}b_{21}} \begin{bmatrix} 1/b_{21} & 1/(b_{21}b_{12}) \\ 1/(b_{12}b_{21}) & 1/b_{12} \end{bmatrix} \mathrm{diag}(\sigma_2,\sigma_1) \begin{bmatrix} s_2 \\ s_1 \end{bmatrix} \\ &= \frac{1}{1-b_{12}b_{21}} \begin{bmatrix} b_{12} & 1 \\ 1 & b_{21} \end{bmatrix} \mathrm{diag}(\sigma_2,\sigma_1) \begin{bmatrix} s_2 \\ s_1 \end{bmatrix} \\ &= \frac{1}{1-b_{12}b_{21}} \begin{bmatrix} 1 & b_{12} \\ b_{21} & 1 \end{bmatrix} \mathrm{diag}(\sigma_1,\sigma_2) \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}. \end{aligned}$$

Therefore, the parameter sets with or without "prime" are equivalent. The model fit is the same with two different sets of parameters. Estimation of the model can equally well give any of these two sets. However, the numerical values of b and b' are quite different. In this example, the system was not constrained to be a DAG. In fact, if the original system is a DAG, the system with primes has infinite coefficients. Thus, we see how the constraint of DAG is helpful.

Appendix C. ML Derivation of Objective Function for Finding the Correct Row Permutation

Since the ICA estimates are never exact, all elements of $\mathbf{W}$ will be non-zero, and one cannot base the permutation on exact zeros. Here we show that the objective function for step 2 of the LiNGAM algorithm can be derived from a maximum likelihood framework.

Let us denote by e_{it} the value of disturbance variable i for the t-th data vector of the data set. Assume that we model the disturbance variables e_{it} by a generalized Gaussian density:

$$\log p(e_{it}) = -|e_{it}|^{\alpha}/\beta + Z,$$

where the α, β are parameters and Z is a normalization constant. Then, the log-likelihood of the model equals

$$\sum_t \sum_i -\frac{1}{\beta}\left|\frac{e_{it}}{w_{ii}}\right|^{\alpha} = -\sum_i \frac{1}{\beta |w_{ii}|^{\alpha}} \sum_t |e_{it}|^{\alpha},$$

because each row of $\mathbf{W}$ is subsequently divided by its diagonal element. To maximize the likelihood, we find the permutation of rows for which the diagonal elements maximize this term. For simplicity, assuming that the pdf's of all independent components are the same, this means we solve

$$\min_{\text{all row perms}} \sum_i \frac{1}{|w_{ii}|^{\alpha}}.$$

In principle, we could estimate α from the data using ML estimation as well, but for simplicity we fix it to unity because it does not really change the qualitative behavior of the objective function. Regardless of its value, this objective function heavily penalizes small values on the diagonal, as we intuitively (based on the argumentation in Section 4) require.

Appendix D. Asymptotic Variance of ICA

Several authors studied asymptotic variance of ICA (Pham and Garrat, 1997; Hyvärinen, 1997; Cardoso and Laheld, 1996; Tichavský et al., 2006), where the theory of estimating functions (Godambe, 1991) was often used. Let us consider a semiparametric model $p(\mathbf{x}|\theta)$, where θ is a r-dimensional parameter vector of interest. Note that the density function $p(\mathbf{x}|\theta)$ is unknown. Let us denote by θ_0 the true parameter vector of interest. A r-dimensional vector-valued function $f(\mathbf{x}, \theta)$ is called an estimating function when it satisfies the following conditions for any $p(\mathbf{x}|\theta_0)$:

$$\begin{aligned} E[f(\mathbf{x}, \theta_0)] &= \mathbf{0} \\ |\det \mathbf{J}| \neq 0, &\quad \text{where } \mathbf{J} = E\left[\frac{\partial}{\partial \theta^T} f(\mathbf{x}, \theta)\Big|_{\theta=\theta_0}\right] \\ E[\|f(\mathbf{x}, \theta_0)\|^2] &< \infty, \end{aligned}$$

where the expectation E is taken over $\mathbf{x}$ with respect to $p(\mathbf{x}|\theta_0)$.

Let $\mathbf{x}(1), \cdots, \mathbf{x}(n)$ be a random sample from $p(\mathbf{x}|\theta_0)$. Then an estimator $\widehat{\theta}$ is obtained by solving the estimating equation:

$$\sum_{i=1}^{n} f(\mathbf{x}(i), \theta) = \mathbf{0}.$$

Under some regularity conditions including identification conditions for θ, the estimator $\widehat{\theta}$ is consistent when n goes to infinity and asymptotically distributes according to the Gaussian distribution $N(\theta_0, \mathbf{G})$, and

$$\mathbf{G} = \frac{1}{n}\mathbf{J}^{-1}E[f(\mathbf{x},\theta_0)f^T(\mathbf{x},\theta_0)]\mathbf{J}^{-T}. \tag{5}$$

Pham and Garrat (1997) derived an estimating function for (quasi-) maximum likelihood estimation. Kawanabe and Müller (2005) provided estimating functions for JADE (Cardoso and Souloumiac, 1993) and for ICA based on non-Gaussianity maximization with orthogonality (uncorrelatedness) constraints including FastICA (Hyvärinen, 1999).

In this paper, we restrict ourselves to testing mixing and demixing coefficients estimated by FastICA. In the FastICA, we first center the data to make its mean zero and whiten the data by computing a matrix $\mathbf{V}$ such that the covariance matrix of $z = \mathbf{V}\mathbf{x}$ is the identity matrix. After that, we find an orthogonal matrix $\mathbf{Q}$ so that components of $\mathbf{Q}^T z = \mathbf{Q}^T\mathbf{V}\mathbf{x}$ have maximum non-Gaussianity. Then we obtain estimates of $\mathbf{A}$ and $\mathbf{W}$ by $\mathbf{A} = \mathbf{V}^{-1}\mathbf{Q}$ and $\mathbf{W} = \mathbf{Q}^T\mathbf{V}$.

Let us consider the following function:

$$\mathbf{F}(\mathbf{x},\mathbf{Q}) = yy^T - \mathbf{I} + yg^T(y) - g(y)y^T,$$

where $y = \mathbf{W}\mathbf{x} = \mathbf{Q}^T\mathbf{V}\mathbf{x} = \mathbf{Q}^T z$ and $g(u)$ is the non-linearity. The estimating function for FastICA is obtained as $f = \mathrm{vec}(\mathbf{F})$ taking $\theta = \mathrm{vec}(\mathbf{Q})$ (Kawanabe and Müller, 2005). Here, $\mathrm{vec}(\cdot)$ denotes the vectorization operator which creates a column vector from a matrix by stacking its columns.

According to the estimating function theory, we obtain the asymptotic covariance matrix of $\mathrm{vec}(\mathbf{Q})$ by (5). Here we assume that the variance in the estimate of $\mathbf{V}$ is negligible with respect to the variance in $\mathbf{Q}$. Then we obtain the asymptotic covariance matrix of $\mathrm{vec}(\mathbf{A})$ and $\mathrm{vec}(\mathbf{W})$ as follows:

$$\begin{aligned}
\mathrm{acov}\{\mathrm{vec}(\mathbf{A})\} &= \mathrm{acov}\{\mathrm{vec}(\mathbf{V}^{-1}\mathbf{Q})\} \\
&= (\mathbf{I}\otimes\mathbf{V}^{-1})\mathrm{acov}\{\mathrm{vec}(\mathbf{Q})\}(\mathbf{I}\otimes\mathbf{V}^{-1})^T \\
\mathrm{acov}\{\mathrm{vec}(\mathbf{W})\} &= \mathrm{acov}\{\mathrm{vec}(\mathbf{Q}^T\mathbf{V})\} \\
&= (\mathbf{V}^T\otimes\mathbf{I})\mathrm{acov}\{\mathrm{vec}(\mathbf{Q}^T)\}(\mathbf{V}^T\otimes\mathbf{I})^T,
\end{aligned}$$

where $\otimes$ denotes the Kronecker product.[7]

The formula of $\mathrm{acov}\{\mathrm{vec}(\mathbf{Q})\}$ for FastICA is written as

$$\mathrm{acov}\{\mathrm{vec}(\mathbf{Q})\} = \frac{1}{n}\mathbf{J}^{-1}E[\mathrm{vec}\{\mathbf{F}(\mathbf{x},\mathbf{Q})\}\mathrm{vec}\{\mathbf{F}(\mathbf{x},\mathbf{Q})\}^T]\mathbf{J}^{-T}.$$

Let us denote by F_{pq} and $\mathbf{F}_q$ the (p,q)-element and the q-th column of $\mathbf{F}$, respectively. We shall provide $E(F_{pq}F_{rs})$ to compute $E\{\mathrm{vec}(\mathbf{F})\mathrm{vec}(\mathbf{F})^T\}$. Denote by i,j,k,l four different subscripts. Then

7. The Kronecker product $\mathbf{Y}\otimes\mathbf{Z}$ of matrices $\mathbf{Y}$ and $\mathbf{Z}$ is defined as a partitioned matrix with (i,j)-th block equal to $y_{ij}\mathbf{Z}$.

we have

$$\begin{aligned}
E(F_{ii}F_{ii}) &= E(s_i^4)+1,\ E(F_{ii}F_{jj})=2,\ E(F_{ki}F_{ij})=-E\{g(s_k)\}E\{g(s_j)\}\\
E(F_{ki}F_{li}) &= E\{g(s_k)\}E\{g(s_l)\},\ E(F_{ki}F_{kj})=E\{g(s_i)\}E\{g(s_j)\}\\
E(F_{ii}F_{li}) &= -E(s_i^3)E\{g(s_l)\},\ E(F_{ki}F_{ii})=-E(s_i^3)E\{g(s_k)\}\\
E(F_{ii}F_{ij}) &= E(s_i^3)E\{g(s_j)\},\ E(F_{ji}F_{jj})=E(s_j^3)E\{g(s_i)\}\\
E(F_{ii}F_{lj}) &= 0,\ E(F_{ki}F_{jj})=0,\ E(F_{ki}F_{lj})=0\\
E(F_{ji}F_{ij}) &= 1+2E\{s_ig(s_i)\}E\{s_jg(s_j)\}-E\{g(s_i)^2\}-E\{g(s_j)^2\}\\
E(F_{ji}F_{lj}) &= 1+E\{s_ig(s_i)\}+E\{s_jg(s_j)\}+E\{s_ig(s_i)\}E\{s_jg(s_j)\}-E\{g(s_i)\}E\{g(s_l)\}\\
E(F_{ki}F_{ki}) &= 1+2E\{s_ig(s_i)\}-2E\{s_kg(s_k)\}+E\{g(s_i)^2\}+E\{g(s_k)^2\}\\
&\quad -2E\{s_ig(s_i)\}E\{s_kg(s_k)\}.
\end{aligned}$$

Further we shall give $E\left\{(\partial\mathbf{F}_i)/(\partial q_j^T)\right\}$ to compute $\mathbf{J}=E\left[\{\partial\mathrm{vec}(\mathbf{F})\}/\{\partial\mathrm{vec}(\mathbf{Q})^T\}\right]$:

$$\begin{aligned}
E\left[\frac{\partial\mathbf{F}_i}{\partial q_i^T}\right] &= \begin{cases} 2E(q_i^Tzz^T) \\ E\{q_k^Tzz^T-z^Tg(q_k^Tz)+q_k^Tzg'(q_i^Tz)z^T\}\end{cases}\\
&= \begin{cases} 2q_i^T & (i\text{−th row})\\ [1-E\{s_kg(s_k)\}+E\{g'(s_i)\}]\,q_k^T & (k\text{−th row},\ k\neq i)\end{cases}\\
E\left[\frac{\partial\mathbf{F}_i}{\partial q_j^T}\right] &= \begin{cases} E[\{1-g'(q_j^Tz)\}q_i^Tzz^T+z^Tg(q_i^Tz)] \\ \mathbf{0}^T\end{cases}\\
&= \begin{cases} [1-E\{g'(s_j)\}+E\{s_ig(s_i)]q_i^T & (j\text{−th row},\ j\neq i)\\ \mathbf{0}^T & (k\text{−th row},\ k\neq j)\end{cases}.
\end{aligned}$$

Appendix E. Exact Form of $\mathbf{J}=\partial\sigma_2(\tau)/\partial\tau^T$

We here derive the exact form of $\mathbf{J}=\partial\sigma_2(\tau)/\partial\tau^T$ in (4). Let us denote by Σ_2 the covariance matrix based on the model or $E(\mathbf{x}\mathbf{x}^T)$. The $\sigma_2(\tau)$ in (3) is obtained by $\mathrm{vec}^+(\Sigma_2)$. Let us rewrite the LiNGAM model as:

$$\begin{aligned}
\mathbf{x} &= (\mathbf{I}-\mathbf{B})^{-1}\mathbf{e}\\
&= (\mathbf{I}-\mathbf{B})^{-1}\mathbf{D}^{\frac{1}{2}}\widetilde{\mathbf{e}},
\end{aligned}$$

where $\mathbf{D}=\mathrm{cov}(\mathbf{e})$ and $\widetilde{\mathbf{e}}=\mathbf{D}^{-\frac{1}{2}}\mathbf{e}$. Note that $E(\mathbf{e})=0$ is assumed. Then the model-based covariance matrix Σ is:

$$\begin{aligned}
\Sigma &= (\mathbf{I}-\mathbf{B})^{-1}\mathbf{D}^{\frac{1}{2}}\mathrm{cov}(\widetilde{\mathbf{e}})\mathbf{D}^{\frac{1}{2}}(\mathbf{I}-\mathbf{B})^{-T}\\
&= \left\{\mathbf{D}^{-\frac{1}{2}}(\mathbf{I}-\mathbf{B})\right\}^{-1}\left\{\mathbf{D}^{-\frac{1}{2}}(\mathbf{I}-\mathbf{B})\right\}^{-T}\\
&= \mathbf{Y}\mathbf{Y}^T,
\end{aligned}$$

where

$$\mathbf{Y} = \left\{\mathbf{D}^{-\frac{1}{2}}(\mathbf{I}-\mathbf{B})\right\}^{-1}.$$

Now we need to compute the following derivatives:

$$\begin{aligned}\frac{\partial \Sigma_{ij}}{\partial b_{kl}} &= \frac{\partial (\mathbf{Y}\mathbf{Y}^T)_{ij}}{\partial b_{kl}} = \sum_p \sum_q \frac{\partial (\mathbf{Y}\mathbf{Y}^T)_{ij}}{\partial \mathbf{Y}_{pq}} \frac{\partial \mathbf{Y}_{pq}}{\partial b_{kl}} \\ \frac{\partial \Sigma_{ij}}{\partial d_{kk}} &= \frac{\partial (\mathbf{Y}\mathbf{Y}^T)_{ij}}{\partial d_{kk}} = \sum_p \sum_q \frac{\partial (\mathbf{Y}\mathbf{Y}^T)_{ij}}{\partial \mathbf{Y}_{pq}} \frac{\partial \mathbf{Y}_{pq}}{\partial d_{kk}}.\end{aligned}$$

We provide $\partial(\mathbf{Y}\mathbf{Y}^T)_{ij}/\partial \mathbf{Y}_{pq}$, $\partial \mathbf{Y}_{pq}/\partial b_{kl}$, and $\partial \mathbf{Y}_{pq}/\partial d_{kk}$ to compute the derivatives above:

$$\begin{aligned}\frac{\partial (\mathbf{Y}\mathbf{Y}^T)_{ij}}{\partial \mathbf{Y}_{pq}} &= \begin{cases} 2\mathbf{Y}_{pq} & (i=p,\ j=p) \\ \mathbf{Y}_{jq} & (i=p,\ j\neq p) \\ \mathbf{Y}_{iq} & (i\neq p,\ j=p) \\ 0 & (i\neq p,\ j\neq p) \end{cases} \\ \frac{\partial \mathbf{Y}}{\partial b_{kl}} &= -\mathbf{Y}\frac{\partial \mathbf{Y}^{-1}}{\partial b_{kl}}\mathbf{Y} \\ &= \mathbf{Y}\mathbf{D}^{-\frac{1}{2}}\mathbf{J}^{kl}\mathbf{Y} \\ \frac{\partial \mathbf{Y}}{\partial d_{kk}} &= -\mathbf{Y}\frac{\partial \mathbf{Y}^{-1}}{\partial d_{kk}}\mathbf{Y} \\ &= \mathbf{Y}\frac{d_{kk}^{-3/2}}{2}\mathbf{J}^{kk}(\mathbf{I}-\mathbf{B})\mathbf{Y},\end{aligned}$$

where $\mathbf{J}^{kl}$ is the single-entry matrix with 1 at (k,l) and zero elsewhere. Thus, we can compute $\mathbf{J} = \partial \sigma_2/\partial \tau^T = \partial \mathrm{vec}^+(\Sigma_2)/\partial \tau^T$.

References

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 57:289–300, 1995.

K. A. Bollen. *Structural Equations with Latent Variables*. John Wiley & Sons, 1989.

G. E. P. Box and G. M. Jenkins. *Time Series Analysis: forecasting and control*. Holden-Day, Oakland, California, USA, revised edition, 1976.

P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods*. Springer-Verlag, New York, USA, 1987.

M. W. Browne. Asymptotically distribution-free methods for the analysis of covariance structures. *British Journal of Mathematical and Statistical Psychology*, 9:665–672, 1984.

R. E. Burkard and E. Cela. Linear assignment problems and extensions. In P. M. Pardalos and D. Z. Du, editors, *Handbook of Combinatorial Optimization - Supplement Volume A*, pages 75–149. Kluwer, 1999.

J.-F. Cardoso and B. H. Laheld. Equivariant adaptive source separation. *IEEE Trans. on Signal Processing*, 44:3017–3030, 1996.

J.-F. Cardoso and A. Souloumiac. Blind beamforming for non Gaussian signals. *IEE Proceedings-F*, 140(6):362–370, 1993.

P. Comon. Independent component analysis – a new concept? *Signal Processing*, 36:287–314, 1994.

Y. Dodge and V. Rousson. On asymptotic properties of the correlation coefficient in the regression setting. *The American Statistician*, 55(1):51–54, 2001.

B. Efron and R. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

D. Geiger and D. Heckerman. Learning gaussian networks. In *Proceedings of the 10th Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, pages 235–243, 1994.

V. P. Godambe. *Estimating functions*. Oxford University Press, New York, 1991.

J. Himberg, A. Hyvärinen, and F. Esposito. Validating the independent components of neuroimaging time-series via clustering and visualization. *Neuroimage*, 22:1214–1222, 2004.

Y. Hochberg. A sharper Bonferroni procedure for multiple tests of significance. *Biometrika*, 4: 800–802, 1988.

Y. Hochberg and A. C. Tamhane. *Multiple comparison procedures*. John Wiley & Sons, New York, 1987.

S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70, 1979.

P. O. Hoyer, S. Shimizu, A. Hyvärinen, Y. Kano, and A. J. Kerminen. New permutation algorithms for causal discovery using ICA. In *Proceedings of International Conference on Independent Component Analysis and Blind Signal Separation, Charleston, SC, USA*, pages 115–122, 2006a.

P. O. Hoyer, S. Shimizu, and A. J. Kerminen. Estimation of linear, non-gaussian causal models in the presence of confounding latent variables. In *Proc. the third European Workshop on Probabilistic Graphical Models (PGM2006)*, 2006b. In press.

L. Hu, P. M. Bentler, and Y. Kano. Can test statistics in covariance structure analysis be trusted? *Psychological Bulletin*, 112:351–362, 1992.

R. J. Hyndman. Time series data library, 2005. URL `http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/`. [June 2005].

A. Hyvärinen. One-unit contrast functions for independent component analysis: A statistical analysis. In *Neural Networks for Signal Processing VII (Proceedings of IEEE Workshop on Neural Networks for Signal Processing)*, pages 388–397, 1997.

A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley Interscience, 2001.

M. Kawanabe and K. R. Müller. Estimating functions for blind separation when sources have variance dependencies. *Journal of Machine Learning Research*, 6:453–482, 2005.

National Statistics, 2005. URL `http://www.statistics.gov.uk/`. [June 2005].

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

D. T. Pham and P. Garrat. Blind separation of mixture of independent sources through a quasi-maximum likelihood approach. *Signal Processing*, 45:1457–1482, 1997.

S. Shimizu, A. Hyvärinen, P. O. Hoyer, and Y. Kano. Finding a causal ordering via independent component analysis. *Computational Statistics & Data Analysis*, 50(11):3278–3293, 2006a.

S. Shimizu, A. Hyvärinen, Y. Kano, and P. O. Hoyer. Discovery of non-gaussian linear causal models using ICA. In *Proc. the 21st Conference on Uncertainty in Artificial Intelligence (UAI-2005)*, pages 526–533, 2005.

S. Shimizu, A. Hyvärinen, Y. Kano, P. O. Hoyer, and A. J. Kerminen. Testing significance of mixing and demixing coefficients in ICA. In *Proceedings of International Conference on Independent Component Analysis and Blind Signal Separation, Charleston, SC, USA*, pages 901–908, 2006b.

S. Shimizu and Y. Kano. Use of non-normality in structural equation modeling: Application to direction of causation. *Journal of Statistical Planning and Inference*, 2006. In press.

R. J. Simes. An improved Bonferroni procedure for multiple tests of significance. *Biometrika*, 73: 751–754, 1986.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search, 2nd ed.* MIT Press, 2000.

Statistical Software Information, 2005. URL `http://www-unix.oit.umass.edu/~statdata/`. [June 2005].

P. Tichavský, Z. Koldovský, and E. Oja. Performance analysis of the FastICA algorithm and Cramèr-Rao bounds for linear independent component analysis. *IEEE Trans. on Signal Processing*, 54(4):1189–1203, 2006.

K-H. Yuan and P. M. Bentler. Mean and covariance structure analysis: Theoretical and practical improvements. *Journal of the American Statistical Association*, 92(438):767–774, 1997.