**JIRSS (2023)**
**Vol. 22, No. 02, pp 77-96**
**DOI: 10.22034/jirss.2024.2002863.1022**

# Practical Learning Directed Acyclic Graphs with General Noise Assumptions

**Russul Al Sarray** [1] **and Vahid Rezaeitabar** [1]

[1] Department of statistics, Faculty of Statistics, Mathematics & Computer, Allameh Tabataba'i University, Tehran, Iran.

Received: 21/05/2023, Accepted: 04/03/2024, Published online: 30/10/2024

**Abstract.** Directed Acyclic Graphs are investigated focusing on learning the coefficient matrix via continuous optimization. We have provided three learning strategies and their corresponding improvements in comparison with former algorithms using some numerical illustrations. Each method is widely introduced and its corresponding concepts are also studied. We have extended, the learning assumptions for each strategy. Lots of preliminary assumptions including normality of noises, and independent and identically distributed errors are removed and with these general considerations, the learning methods are even improved than some existing methods. Furthermore, the number of new criteria that can evaluate the learning processes are given and throughout simulation studies are analyzed. Their sensitivity analysis is also presented which can be useful due to the learning power of any presented strategy. Finally, some further discussions are introduced as guidance for future works.

**Keywords.** Directed Acyclic Graph, Learning, Linear, Not Independent and Not Identically, Structural Equation Model.

**MSC:** 60B99, 05C99, 62P99.

## 1 Introduction

Today we are witnessing an unprecedented growth of data in all fields. All companies and institutions, including big and small, seek to collect and analyze data related to their goals. For this purpose, various tools are being developed more and more. One

Corresponding Author: Russul Al Sarray (alsarray-rusul@atu.ac.ir )
Vahid Rezaeitabar (vhrezaei@atu.ac.ir).

of the most favorite and most important implementations is the graph theory discipline. The graphical models are related to the multivariate models and correspond to the joint densities of a family of variables confined by some conditional independence assumptions. In addition, it holds the conditional relation between corresponding random variables which are denoted by a graph. Furthermore, the graph theory can explain the structural equation models (SEM) using these variables. The coefficient is then shown in a direction between variables shown as a graph. In this regard, the more valuable graph utilized in many areas and SEM is called directed acyclic graphs (DAG). This term was first applied in Wright (1920a,1920b) and is known by the term DAG in statistical areas (Kratzer (2019); Spiegelhalter (1993); Kazempoor (2020)) and Bayesian network in artificial intelligence disciplines (Kratzer (2019); Zareifard (2021)). Data based learning for DAG models is investigated by several researchers (Cooper (1991); Spiegelhalter (1993)). It is also worth mentioning that all of the aforementioned investigations have been done following two concepts including score-based learning and search procedure. For more information about the definition, structure, performance, and usefulness of a DAG and graph theory see Giudici (2003) and Goudie (2016).

A famous representation model of a DAG that is also utilized in SEM is as follows:

$$X = BX + \epsilon, \tag{1.1}$$

where matrix $X$ presents the variables, matrix $B$ provides the graph structure or network representation, and matrix $\epsilon$ stands for the model residuals or noises. The favorite quantity in the modeling of DAGs is determining their structure. The problem is we can not calculate the structure-function of a DAG before we know its model. As it is clear, on one hand, the value of structure-function is so important and can play a useful role in determining the model of a DAG, and on the other hand, we must identify the model and afterwards calculate the mentioned function. There exist many papers that deal with the learning issue of matrix $B$. The methods involve theoretical ones like score-based learning (Aragam (2015)) and penalized likelihood (Aragam (2015)), practical learning such as bootstrap (Giudici (2003)), and empirical (Goudie (2016)) and software-based strategies similar to ICA-LiNGAM (Goudie (2016); Shimizu (2006)) and fast-ICA (Shimizu (2006)). Despite all studies, there is no study dealing with general assumptions such as non-normality, dependencies, and heterogeneous properties of residuals. The role of sample size has also been ignored which results in huge variation between the estimated coefficient matrix and the true one. These reasons cause lots of errors when we use these models in practical data sets. Here, we set to give innovative strategies to improve former learning. In this manner, we are willing to provide many practical instances considering various kinds of assumptions with their learning tests using some available methods and comparing them with each other and the given concepts.

The article introduces a groundbreaking learning methodology devoid of assumptions for a diverse set of DAGs. It systematically discards traditional assumptions, such as independence, homoscedasticity, and normality of errors, paving the way for a progressively evolving approach amalgamating insights from two distinct methods.

The novelty of the article lies in the presentation of an entirely new learning paradigm, hitherto unexplored. Both alternative methods showcased also outperform established approaches, and their collective performance is substantiated through illustrative examples. The initial contribution involves leveraging non-normal errors to estimate the coefficient matrix, facilitating the exploration of alternative models for learning. Subsequently, employing proper likelihood convergence in the second stage enables the accurate calculation of the coefficient matrix, adaptable to various error structures. The ultimate contribution unfolds a non-parametric process, enabling the execution of any learning type. The article's overarching contributions and innovations underscore the high quality, precision, and efficiency inherent in all three presented methods.

The structure of the rest of this study will be arranged as follows. In section 2, we illustrate what we have, what we can deal with or change in the preliminary assumptions, and what we want. A simple representation of an SEM will be given. We have also presented the main corresponding issues. For the aforementioned subjects, we provide the solution algorithms in section 3. According to former learning algorithms, we present several effective and fast algorithms that can find the coefficient matrix of an SEM. In the same manner, these algorithms can be utilized in learning a DAG. These topics will be ready in section 3. We have examined the algorithms in section 4, through extensive numerical analysis. The comparison of the provided algorithms with other existing alternatives is also given in this section. Section 5 deals with some further topics including time-consuming, the number of improvements of the algorithms, and convergence concepts of former and present learning algorithms. We also present some future study items that can be continued in the next investigations. Finally, the conclusion of our study will be available in section 6.

## 2 An Illustration of Our Goal

In this section, the main problem is explained, and through an example, not only some related issues have been discussed, but also the original problems are introduced. In this regard, we present an example and the corresponding contexts are provided. Afterward, our challenging concerns were expressed and our solving strategies remain for the next section.

Here, assume that we are going to test the following SEM:

$$\begin{cases} x_1 &= 2x_3 + \epsilon_1, \\ x_2 &= 0.8x_1 + \epsilon_2, \\ x_3 &= \epsilon_3, \end{cases}$$

where the corresponding noises $\epsilon_i$ following the exponential ($\lambda$) model and noticed by $\epsilon_i \sim EXP(\lambda)$ or equivalently

$$P(\epsilon_i \leq x) = 1 - e^{-\lambda x}, \lambda > 0, x > 0.$$

The graphical representation of this SEM also can be expressed as:

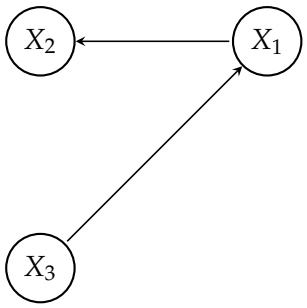


Figure 1: DAG of the above SEM

Adapting with the relation (1.1), it is immediately can be understood that:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 2 \\ 0.8 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix}.$$

It is obvious that for DAG, there exists a triangle representation of the coefficient $B$ and consequently, we have:

$$\begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0.8 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix} + \begin{bmatrix} \epsilon_3 \\ \epsilon_2 \\ \epsilon_1 \end{bmatrix}.$$

or with the same manner:

$$\begin{bmatrix} x_3 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 0.8 & 0 \end{bmatrix} \begin{bmatrix} x_3 \\ x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \epsilon_3 \\ \epsilon_1 \\ \epsilon_2 \end{bmatrix}.$$

As it is clear, the row number of the coefficient increases, and the dependencies of the corresponding variable increase. We look for a coefficient matrix that has the above property. For this example, the third variable does not impact any variable and should be placed in the first row.

Now, the practical and main problem arises. Considering the fixed number of generated samples $\underline{x}$, how can we learn about the coefficient matrix $B$ that can be represented as a DAG? However, our main contributions in the presented study are to develop and evaluate the learning of a DAG with the following concepts:

- How can we develop the independent assumption of noises? Many scholars consider that the residuals are independently distributed. It is still interesting what can we do for dependent consideration of the residuals.
- There exist some extensions on heterogeneous assumptions for the noises. This issue can be also fascinating to deal with and has two separate parts. The first one

is considering the non-normal variables for the residuals. This is a gap among related studies because there exist lots of papers dealing with normal assumptions of the noises. The second item is considering different models that the residuals follow. For instance, in the aforementioned example how can we learn DAG structure if $\epsilon_1 \sim EXP(1), \epsilon_2 \sim NOR(0,1), \epsilon_3 \sim LAP(0,1)$, where NOR$(\mu, \sigma)$ and LAP$(\alpha, \beta)$ respectively stand for normal and Laplace model with the form:

$$P(\epsilon_i \leq x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}, \mu \in \mathbb{R}, \sigma > 0, x \in \mathbb{R},$$

and

$$P(\epsilon_i \leq x) = \frac{1}{2\beta} \int_{-\infty}^{x} e^{-\frac{|t-\mu|}{\beta}}, \alpha \in \mathbb{R}, \beta > 0, x \in \mathbb{R}.$$

- It is also worth mentioning that the sample size effect on learning DAG should be evaluated. In accordance with different assumptions, the sample size has a significant role in estimating parameters. If the learning method has reasonable performance with a small sample size, it can be preferred due to its cost and time. This element will be discussed in detail and evaluated with numerous numerical results.

Further topics can be discussed around the features of $\hat{B}$ or estimated coefficient matrix. Two obvious properties include that $\hat{B}$ has lower triangular or upper triangular and its main diagonal should be zero. Without losing the generality, as mentioned earlier, we consider that $\hat{B}$ is lower triangular. Regarding relation (1.1), it is straightforward that $\hat{\epsilon} = X - \hat{B}X$. Hence $\hat{\epsilon}$ should follow the assumed residual model as well. The more fitting $\hat{\epsilon}$ with assumed residuals, the more quality of calculated $\hat{B}$. Consequently, the co-relation between $\hat{B}$ and $\hat{\epsilon}$ should be jointly considered.

Here, we are going to provide an outlook of the main and original existing learning strategies. The learning process of a DAG almost begins with the original work of Wright (1920a, 1920b) and is separately extended and developed in many areas by Darmois (1953) and Skitovich (1953). These frontline studies motivate other scholars dealing with theoretical assumptions and improvements in learning DAGs.

Among the research before 2000, Robinson (1977) introduced Bayesian network structure learning (BNSL), and the method was extended by Chickering (1996) under independent and identically distributed (IID) assumptions for residuals following normal models. Considering similar assumptions and for non-normal residuals, Spirtes (1991) gives a learning algorithm, but the given method was not efficient enough for non-normal assumptions in real-world applications. In this regard, a simple and rough example was discussed as an ALARM network in Beinlich (1989). For more information on these classical works see alsoChickering (1996), Robinson (1977), and Zareifard (2021) and reference therein.

After (2000), We are a perception of the large growth of scholars in this field dealing

with the practical algorithms in real data applications and their progression. The main contributions and instruments of these studies are constructed as follows:

- Neighborhood and penalized neighborhood regression are the methods that find $\hat{B}$ in relation (1.1) minimizing the existing errors. For more information see Spiegelhalter (1993), Aragam (2015), Gu (2019) and reference therein.

- Score-based and penalized score test learning are often utilized in continuation of the former item, but where the number of variables $p$ is greater than sample size $n$, it can be developed under a non-identifiable feature of a DAG. see Aragam (2015), Spirtes (1991), Robinson (1977) and reference therein for more detailed discussions.

- Causal DAG learning arises in case of existing unique causal. Evaluating and understanding uniqueness property is the main question and should be asses by a learning algorithm. In this paper, we deal with extension problems in this category and the reader can also see Zheng (2020), Hyvarinen (1999), Stone (2004), and Shimizu (2006) for more explanations.

- Conditional independence learning is also rarely discussed in a number of papers and since it has no significant implication, we do not express more. See Aragam (2015) for a comprehensive and complementary explanation.

- Recently, estimating $\hat{B}$ according to the classical statistical method has also attracted lots of attention. Maximum likelihood estimation (MLE) is the main method in this category and is utilized by some scholars such as Beinlich (1989), and Cooper (1991).

- In the same manner, as the MLE method, the Bayesian strategy is also can be used to estimate the coefficient matrix. In continuation, using the Bayesian method has some manipulation problems should be fixed by Monte Carlo Markov Chain and Gibbs sampling. These ways are also explained in details by a number of scholars like Shimizu (2006), Chickering (1996), Goudie (2016), Giudici (2003), Cooper (1991), Zareifard (2021), Spiegelhalter (1993), and Kratzer (2019).

Despite the existing few kinds of literature on discrete data Gu (2019) , the main focus is on continuous data. It is because discrete data is a special case of continuous data and when we can deal with this type of data, the same procedures can be also done for discrete ones in a particular case. Hence, using continuous optimization can logically deal with the learning problems of a DAG. A good text on this issue is Zheng (2020) and includes many optimization processes and practical learning algorithms. Consequently, in the next section, we are going to provide some alternative algorithms comprising the existing algorithms Zheng (2020) in both theoretical and application aspects.

# 3 Learning Algorithm

In this section, we present some effective and practical learning methods for a DAG or SEM as its special case. In the following, we will present three different learning algorithms, each in its respective section. The three sections will progressively showcase learning algorithms that, in terms of learning quality, will improve successively. Assumptions made for the learning process will be successively reduced in each section, and in the final section, no assumptions will be made for the learning process. These assumptions can encompass the independence and dependence of errors, homoscedasticity, heteroscedasticity, and additionally, assumptions related to the sample size. The methods are enumerated respectively, considering that we have matrix-formed sample sizes and then, we should determine the coefficient matrix $B$ and evaluate our learning results. These processes are deeply introduced.

## 3.1 A general Data-based Learning

As mentioned earlier, in this sub-section, we are willing to discuss learning issues from a fixed number of available data. Consider [1] $x^i, i = 1, 2, \ldots, n$ represent for the $i$-th sample that is available as:

$$X^i = \begin{bmatrix} x_1^i \\ x_2^i \\ \vdots \\ x_n^i \end{bmatrix}.$$

Applying the relation (1.1), consequence that:

$$X^i = BX^i + \epsilon^i, i = 1, 2, \ldots, n, \tag{3.1}$$

where $\epsilon^i$ is the noise matrix of the $i$-th sample and can be defined as:

$$\epsilon^i = \begin{bmatrix} \epsilon_1^i \\ \epsilon_2^i \\ \vdots \\ \epsilon_n^i \end{bmatrix}.$$

It is worth mentioning that $B$ is fixed during the sampling processes, i.e,

$$\begin{cases} x^1 & = Bx^1 + \epsilon^1, \\ x^2 & = Bx^2 + \epsilon^2, \\ \vdots & \\ x^m & = Bx^m + \epsilon^m, \end{cases}$$

[1] Notation $X^i$ stands for all data of $i$-th individual including $n$ rows. $n$ shows the number of all collected features of all individuals. $x_j^i$ represent the collected data of $i$-th individual and the $j$-th property. The same idea is right for the other notations like as $\epsilon^i, \epsilon_j^i$, and so on.

Aggregating the above equations results in:

$$x^1 + x^2 + \cdots + x^m = B(x^1 + x^2 + \cdots + x^m) + (\epsilon^1 + \epsilon^2 + \cdots + \epsilon^m),$$

or equivalently as a simple representation

$$\frac{x^1 + x^2 + \cdots + x^m}{m} = B\frac{x^1 + x^2 + \cdots + x^m}{m} + \frac{\epsilon^1 + \epsilon^2 + \cdots + \epsilon^m}{m},$$

which can be reformed as

$$\bar{x}_m = B\bar{x}_m + \bar{\epsilon}_m. \tag{3.2}$$

This equation can guide us in finding a learning strategy. There are two unknown matrices in relation (3.2) including $B$ and $\epsilon$, respectively with $n^2 - n$ and $n$ unknown parameters. Therefore there need $n^2$ necessary equations and at least $m = n^2$. However, this intuition has some problems and in fact, the sample size is often independent of $n$. It is also obvious that the learning process can be continued if we suppose some random noises. Here, we are going to assess this argument that we consider noises and then the validating system of estimated $B$ can be performed. This method until now, does not need any noise assumptions and the detailed process can be mentioned as follows:

I: According to the relation (3.2), if the matrix $\epsilon$ is known, then

$$\hat{B} = (\bar{x}_m - \hat{\epsilon_m})\bar{x}_m^{-1},$$

where $\bar{x}_m^{-1}$ stands for the inverse of $\bar{x}_m$, $\hat{\epsilon_m}$ is the assumed noise matrix and $\hat{B}$ is estimated coefficient matrix.

II: The assumed noises have an important role in the learning procedures and even can be transformed between some of the known models. The free choices and assumptions can play a significant effect on the output results and should be reviewed via numerical results.

III: Since we are willing to investigate the causal of a random variable for other ones, and using section 2, it is easy to check that the diagonal of $B$ is zero and the same point will be noticed for $\hat{B}$. The outlined principle implies a broader concept where dependencies among variables are such that each variable is exclusively determined by other variables in the system. Consequently, when expressing any variable within the system, the coefficient corresponding to that variable is consistently zero.

This overarching observation leads to a more extensive conclusion: in any representation or formulation involving these variables, the principal diagonal of the associated coefficient matrix is universally zero. This stems from the fact that the main diagonal signifies the coefficients of each variable about itself. As a corollary of the previously established result, it follows that all entries along this diagonal must be zero, reflecting the intrinsic nature of the variable relationships in the system.

Before we start to present an example of what we search for, it should be noticed that matrix $B$ is lower triangular considered with zero diagonal elements. This is held, just after we solve the problem or after determining causals. Hence, many scholars provide matrix $W$ instead of $B$, discovering the causal relationship between random variables. In this regard, the relation (3.2) is replaced with $\bar{x}_m = W\bar{x}_m + \bar{\epsilon}_m$, or in practical situations, it can be available as $\hat{W} = (\bar{x}_m - \hat{\epsilon_m})\bar{x}_m^{-1}$. The remained discussion is postponed until the future example.

*Example* 1. Beginning with a rough example that is presented in Zheng (2020) , assume that

$$\begin{cases} x_1 &= e_1, \\ x_2 &= 1.2x_1 + e_2, \\ x_3 &= -0.8x_2 + e_3, \end{cases}$$

where $e_i$'s are independent and uniformly distributed. To compare the results with (Zheng 2020) , in the first step consider $n = 1000$ and use the typical FastICA algorithm (Hyvarinen 1999) , we can get:

$$\hat{W} = \begin{bmatrix} -0.0548 & -0.0001 & -0.0001 \\ 0.0031 & -0.046 & -0.0564 \\ -0.0663 & 0.0532 & -0.0009 \end{bmatrix}.$$

As a remarkable point, the matrix $\hat{W}$ contains lots of small values that are very close to zero. The point is not only due to numerical miscellaneous in optimization procedures but also returns to the inevitable random statistical error in our learning processes. To overcome this problem in practical situations, ICA-LiNGAM steps are suggested. Here, we regret the details processes and the reader can find the practical steps Zheng (2020). After pruning derived estimated matrix, we have:

$$\hat{B} = \begin{bmatrix} 0 & 0 & 0 \\ 1.23964 & 0 & 0 \\ 0 & -0.8006 & 0 \end{bmatrix}.$$

In this manner, the $\hat{B}$ is respectively estimated according to small sample sizes as:

$$\hat{B}_{n=100} = \begin{bmatrix} 0.1 & 0 & 0 \\ 1.05 & 0 & 0 \\ 0.07 & -0.6 & 0 \end{bmatrix}.$$

and

$$\hat{B}_{n=10} = \begin{bmatrix} 0.7 & 0 & 0 \\ 1.1 & 0 & 0 \\ 0.14 & -0.96 & 0 \end{bmatrix}.$$

In this regard, if we generate the noises from $N(0, \frac{1}{3})$, that has the same mean and

variance with $U(-1,1)$, we can determine these matrix as:

$$\hat{B}_{n=1000} = \begin{bmatrix} 0.004 & 0 & 0 \\ 1.2003 & 0 & 0 \\ 0.001 & -0.809 & 0 \end{bmatrix},$$

$$\hat{B}_{n=100} = \begin{bmatrix} 0.0467 & 0 & 0 \\ 1.0987 & 0 & 0 \\ -0.1 & -0.567 & 0 \end{bmatrix},$$

and

$$\hat{B}_{n=10} = \begin{bmatrix} 0.34 & 0 & 0 \\ 00.98 & 0 & 0 \\ -0.14 & -0.643 & 0 \end{bmatrix}.$$

An estimated matrix in the case of $n = 1000$ is so close to the real considered issue in both items of considering uniform and normal models for the noises. A similar argument can be stated for any model with mean 0 and variance $N(0, \frac{1}{3})$. For instance, considering the Laplace model with the aforementioned parameters, we can estimate the coefficient matrix $B$ as:

$$\hat{B}_{n=1000} = \begin{bmatrix} 0.007 & 0 & 0 \\ 1.2809 & 0 & 0 \\ 0.076 & -0.8052 & 0 \end{bmatrix}.$$

This is a powerful showcase of the presented algorithms that can find the coefficients matrix as well as the real one. In comparison with the estimated matrix in Shimizu (2006) , the advantages of this strategy are obvious, especially for small sample sizes $n = 100$ and $n = 10$. The sample sizes decrease, the quality of estimation also decreases and the estimation error increases. This strategy can estimate the coefficient matrix for small sample sizes as well as large ones. Despite this fact, this algorithm is unable for non-normal and heterogeneous assumptions. The independence assumption also can be removed in this area and it shows, the power and quality of the presented algorithm. It can again be mentioned that under different model assumptions for the noises, the learning performances do not change much.

### 3.2 The MLE General Data-based Learning

Despite the good performances of the previous algorithm, we should assume an exact parametric distribution for the noises at the first step. In practical situations, we do not have such information about the residuals and we can not specify their exact distributions. However, we know that the mean residuals should be zero and for the mentioned noise distributions like Normal, Laplace, and Uniform, if we know the mean, there is a need to know just about variance to determine their precise forms. Table (1), provides the exact form of these models according to their variances with the assumptions of zero mean. Now, we can present the related algorithm as:

Table 1: Variance-based representation of the Uniform, Normal, and Laplace distributions with zero mean.

| Distribution name | Variance | Model representation |
|---|---|---|
| Uniform | $\sigma_u^2$ | $f_U(u) = \frac{1}{\sqrt{12\sigma_u^2}}$ |
| Normal | $\sigma_n^2$ | $f_N(n) = \frac{1}{\sigma_n \sqrt{2\pi}} e^{-\frac{n^2}{2\sigma_n^2}}$ |
| Laplace | $\sigma_l^2$ | $f_L(l) = \frac{1}{\sigma_l \sqrt{2}} e^{-\frac{\lvert l \rvert \sqrt{2}}{\sigma_l}}$ |

I: Consider one distribution provided in Table (1) with a specific initial variance $\sigma_0^2$.

II: Perform the previous algorithm and estimate $\hat{B}$.

III: Utilizing the equation (1.1)[2], and calculate the MLE of noise variance.

IV: Repeat the previous items until the differences between the estimated variance are close to your desire[3].

V: Do the previous steps for other distributions in Table (1).

VI: Choose the distribution that the coefficient estimates are also convergent and if this does not exist this property, choose the corresponding estimates of the Normal model[4].

VII: End the learning strategy.

To clarify these steps, we provide the following Example.

*Example* 2. Considering standard exponential noises, we are going to construct the DAG according to the below relations:

$$\begin{cases} x_1 &= e_1, \\ x_2 &= 0.6x_1 + e_2, \\ x_3 &= e_3, \\ x_4 &= 1.5x_3 + e_4, \\ x_5 &= \frac{\pi}{2}x_4 + e_5, \end{cases}$$

[2]All of these distributions can keep linear transformations and their families are not changed under any linear combinations (Rohatgi 2015; Casella 2021).

[3]The convergence of the MLEs is one of the important features that is held in this situation and can be used even in the whole exponential family of distributions (Casella 2021).

[4]The Normal distribution has lots of interesting properties and is also used by many scholars in all areas like learning methods and it is reasonable that if we have not seen any advantages by other models, we are going to prefer this classical density (Rohatgi 2015).

where $e_i$'s are independent and distributed as standard exponential.

Now, we aim to perform the second algorithm. It is worth mentioning that in this case, we assume that we do not have any knowledge of the generated noises. As same of the previous example, at first assume that $n = 1000$, and also assume that all three variances of the algorithm distribution should equal 1. The $\hat{W}$ of the first estimating procedures, respectively are:

$$\hat{W}_u = \begin{bmatrix} 0 & 0.003 & -0.00456 & -0.00702 & 0 \\ 0.65 & 0 & -0.0094 & 0 & -0.0098 \\ 0 & 0 & 0 & 0 & 0.006504 \\ -0.0043 & 0 & 1.87 & 0 & 0.09806 \\ -0.00932 & -0.00346 & 0 & 1.9854 & 0 \end{bmatrix},$$

$$\hat{W}_n = \begin{bmatrix} 0.008 & 0.00433 & -0.00656 & -0.009802 & 0.00605 \\ 0.635 & -0.006103 & -0.00654 & 0 & -0.00918 \\ 0.0073 & 0.089 & 0.075 & 0.94 & 0.003504 \\ -0.0053 & 0.0076 & 1.97 & 0.086045 & 0.05606 \\ -0.00952 & -0.00646 & 0.0093 & 1.6854 & -0.00354 \end{bmatrix},$$

and

$$\hat{W}_l = \begin{bmatrix} 0.035 & 0.01573 & -0.08756 & -0.068802 & 0.006785 \\ 0.455 & -0.000103 & -0.0467 & -0.0078 & -0.04788 \\ 0.0053 & 0.067 & 0.096 & 0.43 & 0.002467 \\ -0.0087 & 0.0038 & 1.956 & 0.067003 & 0.04706 \\ -0.00752 & -0.00387 & 0.0065 & 1.2384 & -0.00589 \end{bmatrix}.$$

Hence the first learning step is:

$$\hat{B}_u = \begin{bmatrix} 0.02 & 0 & 0 & 0 & 0 \\ 0.65 & 0.01 & 0 & 0 & 0 \\ 0.001 & 0.01 & 0.02 & 0 & 0 \\ 0 & -0.0091 & 1.634 & -0.0023 & 0 \\ 0.002 & -0.065 & 0.098 & 1.643 & 0 \end{bmatrix},$$

$$\hat{B}_n = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.645 & 0.046 & 0 & 0 & 0 \\ -0.0041 & 0.0134 & 0.0296 & 0 & 0 \\ 0 & -0.0041 & 1.434 & -0.0043 & 0 \\ -0.0052 & -0.0745 & 0.0108 & 1.479 & 0 \end{bmatrix},$$

and

$$\hat{B}_l = \begin{bmatrix} -0.003 & 0 & 0 & 0 & 0 \\ 0.701 & -0.046 & 0 & 0 & 0 \\ -0.0031 & 0.0574 & 0.0862 & 0 & 0 \\ 0.0076 & 0.0067 & 1.5907 & 0.0093 & 0 \\ 0.0064 & 0.0876 & 0.0108 & 1.57806 & 0 \end{bmatrix}.$$

In this regard, if we consider 0.01 as the maximum threshold of our derived variances, we need 35, 25, and 41 times performing the algorithm respectively for the Uniform, Normal, and Laplace model, and the calculated MLEs are 0.23, 0.27, and 0.24. The final learning matrix then can be available as:

$$\hat{B}_u = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.56 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0.097 & 0 & 0 \\ 0 & 0 & 1.89 & 0.002 & 0 \\ 0.001 & 0 & -0.023 & 1.73 & 0 \end{bmatrix},$$

$$\hat{B}_n = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 \\ 0.76 & 0.09 & 0 & 0 & 0 \\ 0 & 0.08 & -0.086 & 0 & 0 \\ 0 & 0 & 1.64 & 0.002 & 0 \\ -0.001 & 0 & -0.043 & 1.63 & 0 \end{bmatrix},$$

and

$$\hat{B}_l = \begin{bmatrix} -0.007 & 0 & 0 & 0 & 0 \\ 0.654 & -0.19 & 0 & 0 & 0 \\ 0.002 & -0.09 & -0.043 & 0 & 0 \\ 0.0043 & -0.004 & 1.606 & -0.0022 & 0 \\ 0.003 & -0.0075 & 0.079 & 1.578 & 0 \end{bmatrix}.$$

It is clear that the number of repeated steps in all three models is low or the convergence rate is high. The fast convergence property is useful in case of time-consuming software computations. However, this feature does not decrease the estimation precision and it can be understood that the $\hat{B}$ are so close for all three models. Even though we can not see a significant difference between the models, it is worth considering that this note may not be guaranteed, and maybe none of the models or some of them are convergent in variance estimation. Meanwhile, the considered noise models have suitable performances in many studies and can be utilized.

### 3.3 An Effective Test-error Learning

In this sub-section, using the concepts of previous sub-sections, we develop the learning processes without considering any assumptions on residuals. The specific points are expressed below.

I: Utilizing ICA-LiNGAM algorithm Shimizu (2006) , directly estimate $\hat{B}$ from collected data.

II: Using equation (1.1), and determine noises as $\hat{\epsilon} = X - \hat{B}X$.

III: Choose a random sub-sample of calculated $\hat{\epsilon}_j$, $j = 1, 2, \ldots, N$.

IV: According to new determined residuals $\hat{\epsilon}_j$, calculate new coefficient matrix $\hat{B}_j$.

V: Do steps III and IV for $N$ times.

VI: Estimate the original coefficient matrix as $\hat{B}_1 = \dfrac{\sum_{j=1}^{N} \hat{B}_j}{N}$.

VII: Repeat steps III to VI and calculate again $\hat{B}_2$ in the same way.

VIII: If $\hat{B}_1$ and $\hat{B}_2$ are enough close, end the algorithm, and if else repeat this algorithm for greater $N$ until achieving the suitable and close estimations.

Returning to our attention, we present the following example clarifying our algorithm procedures.

*Example* 3. Considering an arbitrary model for the noises, we are going to construct the DAG according to the following relations:

$$\begin{cases} x_1 &= e_1, \\ x_2 &= 2.5x_1 + e_2, \\ x_3 &= 0.7x_2 + e_3, \\ x_4 &= e_4, \end{cases}$$

where $e_i$'s are generated in accordance with your choices.

## 4 Performances and Sensitivity Analysis

In this section, we are going to examine our algorithms through a number of figures and numerical results. To do this, we present an illustrative showcase of what we generate and then we aim to find the corresponding coefficient matrix. Assume that the following construction of an SEM,

$$\begin{cases} x_1 = 2x_2 + \epsilon_1, & x_2 = 0.9x_3 + \epsilon_2, \\ x_3 = 3.1x_4 + \epsilon_3, & x_4 = \epsilon_4, \\ x_5 = 1.7x_6 + \epsilon_5, & x_6 = \epsilon_6, \\ x_7 = 0.4x_6 + \epsilon_7, & x_8 = 3.5x_7 + \epsilon_8, \\ x_9 = 2.6x_8 + \epsilon_9, & x_{10} = 1.6x_9 + \epsilon_{10}. \end{cases}$$

Now consider these models generated such that

I: The residuals $\epsilon$s follow the standard normal distribution.

II: The residuals $\epsilon$s follow the standard Laplace distribution.

III: The residuals $\epsilon$s follow $U(-1, 1)$ distribution.

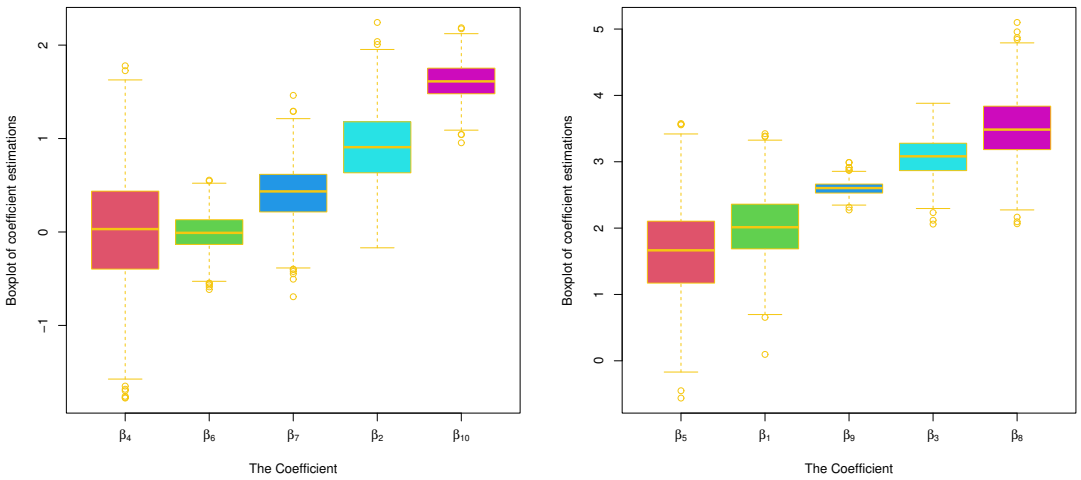


Figure 2: Boxplot representations of the estimated coefficients under the first strategy for considered SEM (Section 4)

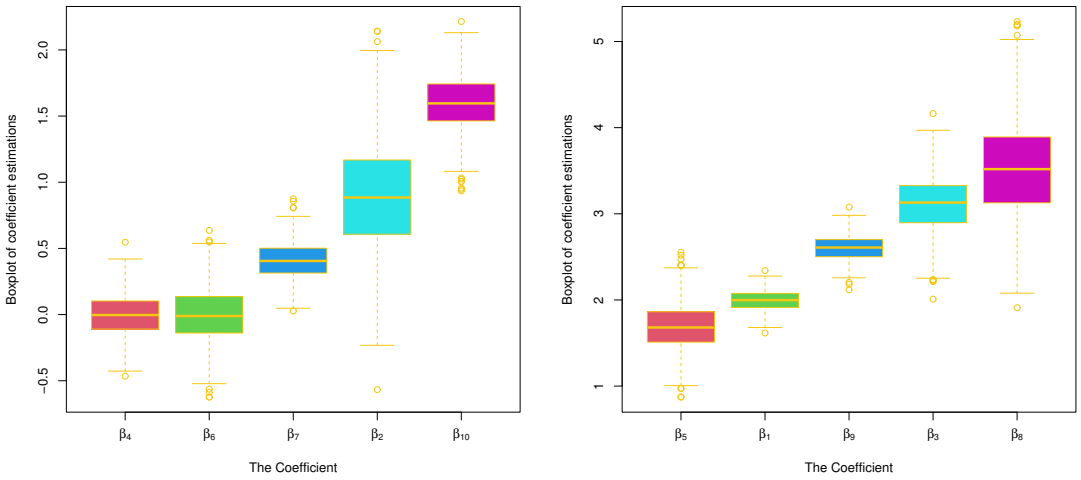


Figure 3: Boxplot representations of the estimated coefficients under the second strategy for considered SEM (Section 4)

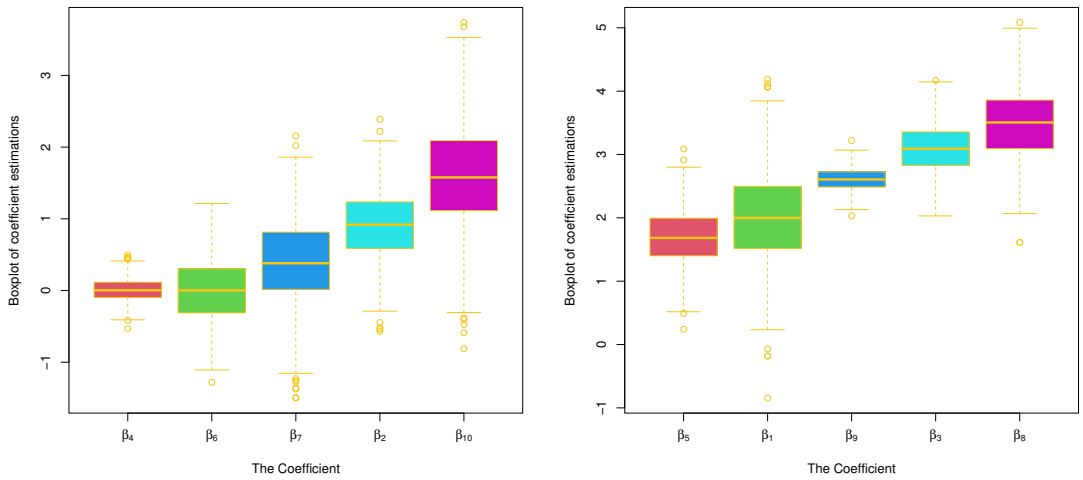


Figure 4: Boxplot representations of the estimated coefficients under the third strategy for considered SEM (Section 4)

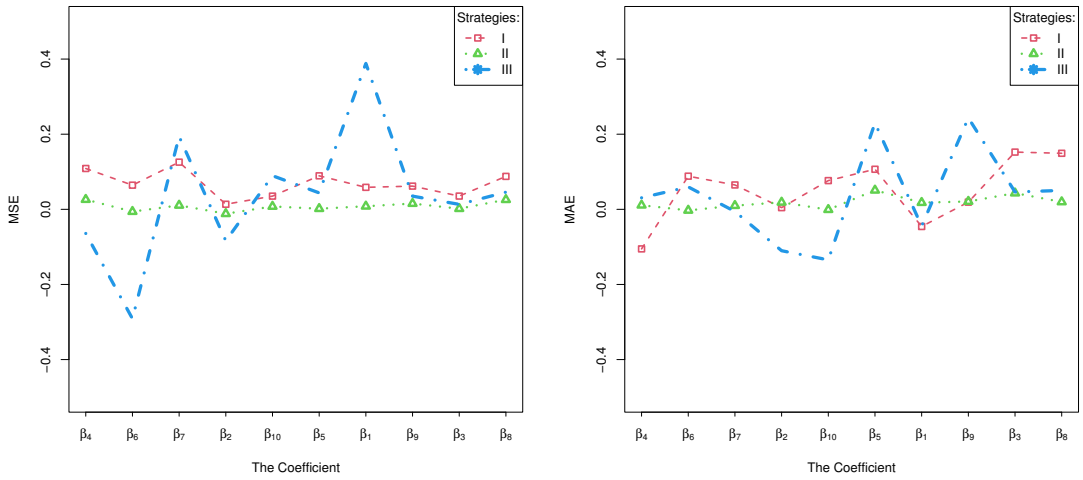


Figure 5: Errors-based evaluation of three considered methods for the mentioned example (Section 4)

Under the above circumstances, there have been lots of related concepts. Figures (2), (3), and (4) provide the dispersion representations of all estimated parameters one by one, respectively under strategies I, II, and III. Despite different performances of the estimated coefficients under the three strategies, it can be easily seen that the estimations are so close to the considered parameters, in the case of mean and median. Overall, the figures not only give a reasonable insist on utilizing these strategies but also show that the first and second strategies can stand with more acceptance in comparison to the third one. However, we can not make any precise or numerical differentiation among the aforementioned strategies in accordance with these figures. Hereby, we are going to present a comparative showcase in figure (5) that can enable us to this end. Figure (5), presents respectively the MSE and MAE of the estimated coefficient for all ten considered parameters in the Example given in the first of this section. Accordingly, the advantages of the first and second methods to the third one can be clearly understood. The same proposition still held for both MSE and MAE criteria.

We have incorporated two evaluation metrics, MSE and MAE, to assess and compare the efficacy of the proposed methodologies. In addition to these metrics, a penalty factor of 1/3 has been introduced, taking into consideration its inverse proportionality to the square root of the number of parameters. This penalty term has been subtracted from both MSE and MAE, ensuring a uniform penalty application across all parameters based on the specific conditions of the problem.

This nuanced approach accounts for the inherent trade-off between model complexity and the penalty applied, as the penalty scales with the inverse square root of the number of parameters. By incorporating this penalty factor into the evaluation metrics, we aim to provide a more comprehensive and fair comparison of the proposed methods, considering not only predictive accuracy but also the model's complexity and generalization capabilities concerning the given problem constraints.

We start this section with a large SEM model consisting of ten coefficients that can be generated using three random models and with completely different situations. The considered extension can guarantee that whether the model is good, it can be involved also in other examples. In continuation, there have been four figures that can help the performances of the studied strategies in general cases. Finally, the first and second methods are preferred to learn the structure of a linear model.

## 5 Further Discussion

As it is clear, learning concepts can include numerous methods under many analytical pieces of literature. In addition to the mentioned procedures in the present text and referred studies in the introduction, one may interested in performing the learning ways via the Bayesian algorithms or even completely non-parametric and semi-parametric processes. The other processes can also be presented and investigated as well as the theoretical ones like machine-based methods. Here we provide some pioneer topics for future studies.

I: Combining the classical estimating methods such as Bayesian, non-parametric, and semi-parametric strategies with the existing ones and providing some suitable learning explanations.

II: Providing machine-based learning algorithms in all different aspects of data involving parametric, non-parametric, and semi-parametric ones.

III: Developing the existence of parametric algorithms in case of different assumptions of the residuals models, and their dependencies structure.

IV: Evaluating the learning algorithms can also be extended. The results not only can be tested with their preciseness but also can be examined with their time-consuming, the memory needed for performing, and so on.

V: Extension around direct weights play also an important role in such areas. Discussion on this concept including precise estimating of the weights can be expressed in the next papers.

It is also worth mentioning that, every learning discussion should have a clear idea of what measure can be utilized. A practical criterion is a sum of differences between the real and estimated values, but the problem is related to when we do not know any information about the real or exact values. Marginal issues also can be investigated in this case. We regret the further topics and we will be prepared some concepts for our future studies.

## 6 Conclusion

Three learning algorithms aiming to discover relationships between random nodes in a continuous DAG have been developed. Their basic constructions are introduced and their corresponding details are also described. Their benefits rather than previous methods are not only explained but also their practical advantages including preciseness, and time-consuming have been expressed. To maintain the features of introduced algorithms, a section on sensitivity analysis and their performances in practical situations are highlighted. The famously considered assumptions are removed and under general assumptions, these algorithms are provided. The role of these assumptions is investigated under some criteria that were also developed in this study. For future works, we give some guidance and items that can be extended in the next studies.

## References

Aragam, B., Amini, A. A., and Zhou, Q. (2015), Learning directed acyclic graphs with penalized neighbourhood regression. arXiv preprint arXiv:1511.08963.

Beinlich, I. A., Suermondt, H. J., Chavez, R. M., and Cooper, G.F. (1989), The ALARM monitoring system: A case study with two probabilistic inference techniques for

belief networks. *Lecture Notes in Medical Informatics*, **89**, 247–256. doi:10.1007/978-3-642-93437-7-28.

Casella, G., and Berger, R. L. (2021), *Statistical inference*. Cengage Learning.

Chickering, D.M. (1996), Learning Bayesian networks is NP-complete. In *Learning from data: Artificial intelligence and statistics V* (Vol. 12, pp. 121-130). doi:10.1007/978-1-4612-2404-4-12.

Cooper, G. F., and Herskovits, E. (1991), A Bayesian method for constructing Bayesian belief networks from databases. *Uncertainty Proceedings*, **1**, 86-94. doi:10.1016/B978-1-55860-203-8.50015-2.

Darmois, G. (1953), Analyse générale des liaisons stochastiques: etude particulière de l'analyse factorielle linéaire. *Revue de l'Institut international de statistique*, **2**(1), 2-8. doi:10.2307/1401511.

Giudici, P., and Castelo, R. (2003), Improving Markov chain Monte Carlo model search for data mining. *Machine learning*, **50**(1), 127-158. doi:10.1023/A:1020202028934.

Goudie, R., and Mukherjee, S. (2016), *A Gibbs Sampler for Learning DAGs*. Microtome Publishing.

Gu, J., Fu, F., and Zhou, Q. (2019), Penalized estimation of directed acyclic graphs from discrete data. *Statistics and Computing*, **29**(1), 161-176. doi:10.1007/s11222-018-9801-y.

Hyvarinen, A. (2000), Independent component analysis: algorithms and applications. *Neural networks*, **13**(5), 411-430. doi:10.1016/S0893-6080(00)00026-5.

Kratzer, G., and Furrer, R. (2019), mcmcabn: a structural MCMC sampler for DAGs learned from observed systemic datasets. R package version 3(1): https://cran.microsoft.com/snapshot/2020-04-08/web/packages/mcmcabn/vignettes/mcmcabn.html.

Kazempoor, J., Habibirad, A., and Okhli, K. (2020), Bounds for CDFs of order statistics arising from INID random variables. *Journal of the Iranian Statistical Society*, **19**(1), 39-57.

Robinson, R. W. (1977), Counting unlabeled acyclic digraphs. Springer Berlin Heidelberg.

Rohatgi, V. K., and Saleh, A.K.M.E.S. (2015), An introduction to probability and statistics. John Wiley & Sons.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M.I. (2006), A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, **7**, 2003-2030.

Skitovitch, V. P. (1953), On a property of the normal distribution. *DAN SSSR*, **89**(1), 217-219.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., and Cowell, R. G. (1993), Bayesian analysis in expert systems. *Statistical science*, **8**(3), 219-247.

Spirtes, P., and Glymour, C. (1991), An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, **9**(1), 62-72.

Stone, J. V. (2004), Independent component analysis: a tutorial introduction. MIT press.

Wright, S. (1920a), Principles of livestock breeding. US Department of Agriculture.

Wright, S. (1920b), The relative importance of heredity and environment in determining the piebald pattern of guinea-pigs. US Department of Agriculture.

Zareifard, H., Rezaei Tabar, V., and Plewczynski, D. (2021), A Gibbs sampler for learning DAG: a unification for discrete and Gaussian domains. *Journal of Statistical Computation and Simulation*, **91**(14), 2833-2853.

Zheng, X. (2020), Learning DAGs with Continuous Optimization [PhD diss., University of Pittsburgh Medical Center].