



THEORY AND METHODS

Algebraic Approach to Maximum Likelihood Factor Analysis

Ryoya Fukasaku¹, Kei Hirose², Yutaro Kabata³ and Keisuke Teramoto⁴

¹Faculty of Mathematics, Kyushu University, Fukuoka, Japan; ²Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan; ³Graduate School of Science and Engineering, Kagoshima University, Kagoshima, Japan; ⁴Department of Mathematics, Yamaguchi University, Yamaguchi, Japan

Corresponding author: Kei Hirose; Email: hirose@imi.kyushu-u.ac.jp

(Received 22 January 2025; revised 20 August 2025; accepted 20 August 2025)

Abstract

In maximum likelihood factor analysis, we need to solve a complicated system of algebraic equations, known as the normal equation, to get maximum likelihood estimates (MLEs). Since this equation is difficult to solve analytically, its solutions are typically computed with continuous optimization methods, such as the Newton–Raphson method. With this procedure, however, the MLEs are dependent on initial values since the log-likelihood function is highly non-concave. Particularly, the estimates of unique variances can result in zero or negative, referred to as improper solutions; in this case, the MLE can be severely unstable. To delve into the issue of the instability, we algebraically compute all candidates for the MLE. We provide an algorithm based on algebraic computations that is carefully designed for maximum likelihood factor analysis. To be specific, Gröbner bases are employed, powerful tools to get simplified sub-problems for given systems of algebraic equations. Our algebraic algorithm provides the MLE independent of the initial values. While computationally demanding, our algebraic approach is applicable to small-scale problems and provides valuable insights into the characterization of improper solutions. For larger-scale problems, we provide numerical methods as practical alternatives to the algebraic approach. We perform numerical experiments to investigate the characteristics of the MLE with our two approaches.

Keywords: computational algebra; Gröbner basis; improper solutions; maximum likelihood factor analysis

1. Introduction

Factor analysis is a practical tool for investigating the covariance structure of observed variables by constructing a small number of latent variables referred to as common factors. Factor analysis has been used in the social and behavioral sciences since its proposal more than 100 years ago (Spearman, 1904), and it has been recently applied to a wide variety of fields, including marketing, life sciences, materials sciences, and energy sciences (Kartechina et al., 2020; Lin et al., 2019; Shkeer & Awang, 2019; Shurrab et al., 2019; Vilkaite-Vaitone et al., 2022).

In exploratory factor analysis, the model parameter is often estimated by the maximum likelihood method under the assumption that the observed variables follow the multivariate-normal distribution. The maximum likelihood estimates (MLEs), solutions that maximize the likelihood function, are

© The Author(s), 2025. Published by Cambridge University Press on behalf of Psychometric Society.

This is an Open Access article, distributed under the terms of the Creative Commons Attribution-NonCommercial licence (<https://creativecommons.org/licenses/by-nc/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original article is properly cited. The written permission of Cambridge University Press must be obtained prior to any commercial use.

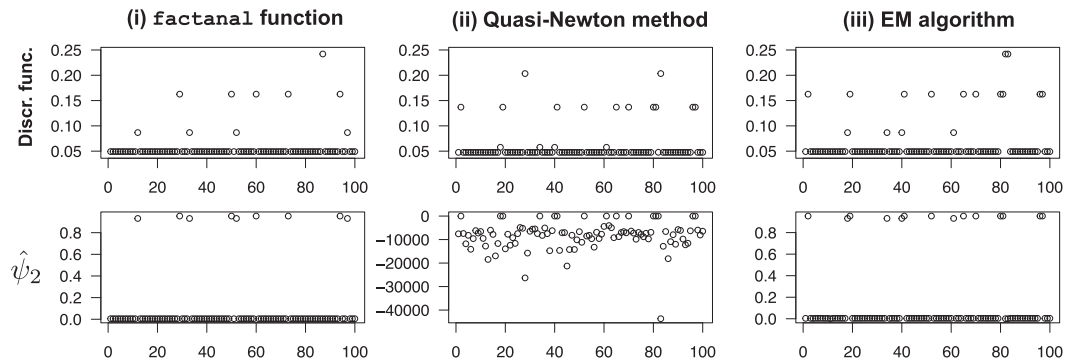


Figure 1. Convergence values of the discrepancy function and MLEs of unique variance for the second observed variable, $\hat{\psi}_2$.

Note: These convergence values are obtained by applying three estimation algorithms to an artificial dataset with 100 different initial values of unique variances, and the horizontal axis indicates the indices for these 100 initial values.

obtained by solving a complicated system of multivariate algebraic equations called the normal equation. The solutions to the system are usually intractable; that is, it is difficult to solve the system analytically from the standpoint of computation time/memory. Thus, these solutions are typically computed using continuous optimization techniques.

One of the most common approaches is to use the gradient of the log-likelihood function, such as the Newton–Raphson method or Quasi-Newton method (Jennrich & Robinson, 1969; Jöreskog, 1967; Lawley & Maxwell, 1971). For given unique variances, the estimation of the loading matrix turns out to be the eigenvalue/eigenvector problem for the standardized sample covariance matrix, which is similar to the problem of the principal component analysis (Tipping & Bishop, 1999). Substituting the loading matrix into the log-likelihood function, it is regarded as a function of unique variances, and these unique variances are optimized by the Newton–Raphson method or Quasi-Newton method. The above algorithm has been widely used for several decades; indeed, `factanal` function, one of the most popular functions that implements the maximum likelihood factor analysis in R, is based on Lawley and Maxwell’s (1971) algorithm. Another algorithm is the expectation–maximization (EM) algorithm (Rubin & Thayer, 1982), in which the unique factors are regarded as latent variables. The complete-data log-likelihood function corresponds to the log-likelihood function in multivariate regression; thus, the EM algorithm can be directly utilized in the penalized likelihood methods, such as the lasso and minimax concave penalty (Choi *et al.*, 2010; Hirose & Yamamoto, 2015), allowing to achieve high-dimensional sparse estimation.

Although the above-mentioned algorithms often converge, the convergence value is not always optimal due to the non-concavity of the log-likelihood function. To illustrate the potential issues with convergence in factor analysis, we present an example demonstrating how the converged values depend on the initial values and the estimation algorithm. Figure 1 shows the convergence values of the discrepancy function that corresponds to the negative log-likelihood function and the MLEs of unique variance for the second observed variable, say $\hat{\psi}_2$. These convergence values are obtained by applying estimation algorithms to an artificial dataset generated from “simulation model S2,” where the communality is relatively small, leading to unstable estimation. A detailed description of the simulation model S2 will be provided in Section 4. We employ three estimation algorithms: (i) the `factanal` function in R (i.e., Lawley and Maxwell’s (1971) algorithm), (ii) the Quasi-Newton method based on Jennrich & Robinson (1969), and (iii) the EM algorithm (Rubin & Thayer, 1982). One hundred different initial values for unique variances are generated from a uniform distribution $U(0,1)$ to investigate whether the convergence value depends on the initial value. It should be noted that we use the same sets of 100 initial values for these three algorithms.

The results indicate that the convergence values are dependent on the initial values and estimation algorithms. In particular, the estimate of unique variance, $\hat{\psi}_2$, is sensitive to the initial values with the

Quasi-Newton method based on Jennrich and Robinson's (1969) algorithm. We observe that improper solutions are obtained; that is, the estimates of unique variances turn out to be zero or negative. The Quasi-Newton method based on Jennrich and Robinson's (1969) algorithm allows the estimates of unique variances to take zero or negative values; thus, the $\hat{\psi}_2$ becomes exceptionally unstable. Meanwhile, the `factanal` function outputs parameters such that all of the unique variances are greater than some small threshold values; by default, this threshold is set to 0.005. The EM algorithm also does not produce negative estimates of unique variances because the update equation for unique variances in the M-step produces non-negative values (see Theorem 1 of Adachi, 2013). Although the `factanal` function and EM algorithm do not yield negative unique variances, their results are dependent on the initial values. Consequently, it is quite difficult to evaluate whether an optimal value of the parameter is obtained with these estimation algorithms.

As shown in the above example, the instability of the numerical estimates is often observed when we obtain an improper solution in our experience. To our knowledge, a theoretical approach to elucidate improper solutions has not yet been provided due to the complexity of the log-likelihood function; thus, the improper solutions remain to be a challenging problem. Instead of theoretical approaches, numerical approaches have been employed to investigate the characteristics of improper solutions for the last five decades (e.g., van Driel, 1978; Hayashi & Lu, 2014; Jöreskog, 1967; Kano, 1998; Krijnen et al., 1998; Sato, 1987). However, with the numerical approaches, the result is highly dependent on the estimation algorithm and its initial values, as shown in Figure 1.

To address this issue, it is essential to compute exact algebraic solutions to the system of multivariate algebraic equations (i.e., all candidates of the MLEs). Here, "exact" specifically refers to solutions computed without rounding errors. In this study, we employ an algebraic approach based on computational algebra to obtain the exact algebraic solutions. Computational algebra can solve complicated systems of multivariate algebraic equations by using the theory of Gröbner bases. Several software that implement computational algebra, such as Magma, Maple, and Mathematica, have been developed considerably in the last two decades. Thanks to the development of the algorithm and the rapid progress in computational technology, several real problems have been resolved by computational algebra (e.g., Hinkelmann et al., 2011; Laubenbacher & Sturmfels, 2009; Veliz-Cuba et al., 2014).

In maximum likelihood factor analysis, we observe that the original system of multivariate algebraic equations requires a lot of computer resources from an algebraic viewpoint, due to the existence of the inverse covariance matrix. To reduce the heavy computational loads, we introduce an algorithm specifically designed for maximum likelihood factor analysis. In our algebraic algorithm, we employ the theory of Gröbner bases to get simplified sub-problems for a given system of algebraic equations. After getting the simplified sub-problems, we compute all exact solutions to the system. The MLEs can be obtained by selecting a solution that maximizes the likelihood function. The estimates are independent of the initial value and estimation algorithm because the algebraic algorithm produces all exact solutions. Furthermore, based on the MLE obtained by this method, we classify the solutions into three patterns: no solution, positive solution, and negative solution. While computationally demanding, our algebraic approach is applicable to small-scale problems (e.g., five variables and two factors) and provides valuable insights into the characterization of improper solutions. For larger-scale problems, we provide numerical methods as practical alternatives to the algebraic approach.

Algebraic approaches to factor analysis have been explored by several earlier studies, as highlighted by Ardiyansyah & Sodomaco (2023), Drton & Xiao (2010), Drton et al. (2007), and Drton et al. (2025). In particular, Drton et al. (2007) employ computational algebra to study model invariants in factor analysis, and the final section spotlights algebraic study for maximum likelihood estimation and singularities as the next steps. These earlier works mostly focus on perspectives from mathematics. This study, however, aims to develop an algorithm for computing estimates, enabling us to delve into the characteristics of the MLEs. Specifically, an extensive Monte Carlo simulation is conducted to provide a detailed analysis of these characteristics. We observe that sometimes the dimension of the solution space can be more than zero, that is, the solution space can be a curve, a surface, etc. Furthermore, the MLEs do not exist in some cases, even if the parameter space for unique variances includes negative values. Such

a discussion is impossible with existing algorithms. Consequently, the proposed algorithm provides not only theoretical insights but also numerical analyses that significantly advance the understanding of the behavior of MLEs, including a characterization of improper solutions. The insights gained through our approach may potentially guide future methodological development for practical data analysis.

The remainder of this article is organized as follows: Section 2 briefly reviews the maximum likelihood factor analysis. A system of algebraic equations to get MLEs is also discussed. In Section 3, we introduce a novel algorithm for computing MLEs via computational algebra. Section 4 presents numerical results for artificial datasets. In addition, Section 5 reports numerical results for actual datasets. Conclusion and future perspectives are given in Section 6.

2. Maximum likelihood factor analysis

2.1. Maximum likelihood estimation

Let $\mathbf{X} = (X_1, \dots, X_p)^\top$ be a p -dimensional observed random vector. The factor analysis model is

$$\mathbf{X} = \boldsymbol{\mu} + \Lambda \mathbf{f} + \boldsymbol{\varepsilon},$$

where $\boldsymbol{\mu}$ is a mean vector, $\Lambda = (\lambda_{ij})$ is a $p \times k$ matrix of factor loadings, and $\mathbf{f} = (f_1, \dots, f_k)^\top$ and $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_p)^\top$ are unobservable random vectors. Here, A^\top denotes the transpose of a matrix A . The elements of \mathbf{f} and $\boldsymbol{\varepsilon}$ are referred to as common factors and unique factors, respectively. It is assumed that $\mathbf{f} \sim N_k(0, I_k)$ and $\boldsymbol{\varepsilon} \sim N_p(0, \Psi)$ and \mathbf{f} and $\boldsymbol{\varepsilon}$ are independent. Here, I_k is an identity matrix of order k and Ψ is a $p \times p$ diagonal matrix with the i -th diagonal element being ψ_i , referred to as a unique variance. Under these assumptions, the observed vector \mathbf{X} follows multivariate-normal distribution; $\mathbf{X} \sim N_p(\boldsymbol{\mu}, \Sigma)$ with $\Sigma = \Lambda \Lambda^\top + \Psi$. When the observed variables are standardized, the i -th diagonal element of $\Lambda \Lambda^\top$ is called communality, which stands for the proportion of variance in X_i explained by common factors.

It is well-known that factor loadings have a rotational indeterminacy since both Λ and ΛT generate the same covariance matrix Σ , where T is an arbitrary orthonormal matrix. To obtain an identifiable model, we often use a restriction that $\Lambda^\top \Psi^{-1} \Lambda$ is a diagonal matrix or the upper triangular elements of Λ are fixed by 0.

Suppose that we have a random sample of N observations $\mathbf{x}_1, \dots, \mathbf{x}_N$ from the p -dimensional normal population $N_p(\boldsymbol{\mu}, \Sigma)$. The MLEs of factor loadings and unique variances are obtained by maximizing the log-likelihood function

$$\log f(\mathbf{x}_1, \dots, \mathbf{x}_N | \Lambda, \Psi) = -\frac{N}{2} \left\{ p \log(2\pi) + \log |\Sigma| + \text{tr}(\Sigma^{-1} S) \right\}, \quad (1)$$

or equivalently, minimizing the discrepancy function:

$$q(\Lambda, \Psi) = \log |\Sigma| + \text{tr}(\Sigma^{-1} S) - \log |S| - p, \quad (2)$$

where $S = (s_{ij})$ is the sample variance-covariance matrix,

$$S = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^\top, \quad \bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n.$$

We remark that it is common to use the unbiased sample covariance matrix with $n = N - 1$, and the choice between N and $N - 1$ slightly affects the statistical properties of the estimator. In this study, however, the focus is on computing estimates rather than analyzing estimator properties. Since factor analysis typically uses the sample correlation matrix, the choice between N and $N - 1$ does not affect the discrepancy function.

We assume that S is non-singular to ensure the existence of S^{-1} , which plays an important role in constructing an algorithm to find MLEs. This assumption can be violated when the number of variables exceeds the sample size. In such a case, we may employ the ridge regularization on S ; that is, we use $S_\lambda := S + \lambda I_p$ with $\lambda > 0$ (Yuan & Chan, 2008).

2.2. Numerical algorithms

We briefly review the algorithms for computing the MLEs (Jöreskog, 1967; Lawley & Maxwell, 1971). The candidates of the MLEs, say $(\hat{\Lambda}, \hat{\Psi})$, are given as the solutions of

$$\frac{\partial q(\Lambda, \Psi)}{\partial \Lambda} = 0, \quad \frac{\partial q(\Lambda, \Psi)}{\partial \Psi_i} = 0. \quad (i = 1, \dots, p) \quad (3)$$

The solution to the above equation is given by

$$\begin{aligned} \Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Lambda &= 0, \\ \text{diag}(\Sigma^{-1}(\Sigma - S)\Sigma^{-1}) &= 0, \end{aligned} \quad (4)$$

where $\text{diag}(A)$ denotes the diagonal matrix consisting of the diagonal elements of a matrix A .

Since the solutions are not expressed in a closed form, some iterative algorithm is required. Typically, we use an algorithm based on the gradient of the log-likelihood function, such as the Newton–Raphson method or Quasi-Newton method. When Ψ is not singular, Eq. (3) results in the following expressions (Lawley & Maxwell, 1971, page 28, Eq. (4.10)):

$$(\Psi^{-\frac{1}{2}}S\Psi^{-\frac{1}{2}})\Psi^{-\frac{1}{2}}\Lambda = \Psi^{-\frac{1}{2}}\Lambda(I_k + \Lambda^T\Psi^{-1}\Lambda), \quad (5)$$

$$\Psi = \text{diag}(S - \Lambda\Lambda^T). \quad (6)$$

Let $\Lambda^T\Psi^{-1}\Lambda = \Delta$. The rotational indeterminacy is resolved when Δ is a diagonal matrix (e.g., Lawley & Maxwell, 1971); in this case, (5) is an eigenvalue/eigenvector problem. Let $(\theta_1, \mathbf{e}_1), \dots, (\theta_p, \mathbf{e}_p)$ be the eigenvalues and eigenvectors of $\Psi^{-\frac{1}{2}}S\Psi^{-\frac{1}{2}}$, respectively, where the eigenvalues are rearranged in a weakly decreasing order; that is, $\theta_1 \geq \theta_2 \geq \dots \geq \theta_p$. Let $P = (\mathbf{e}_1, \dots, \mathbf{e}_k)$ and $\Theta = \text{diag}(\theta_1, \theta_2, \dots, \theta_k)$. Eq. (5) implies

$$P(\Theta - I_k)^{\frac{1}{2}} = \Psi^{-\frac{1}{2}}\Lambda \quad (7)$$

or equivalently,

$$\Lambda = \Psi^{\frac{1}{2}}P(\Theta - I_k)^{\frac{1}{2}}. \quad (8)$$

Thus, for given Ψ , Λ can be calculated using (8). For given Λ , one can update Ψ with (6). However, the update with (6) can be slow when some of the diagonal elements of $\hat{\Psi}$ are small. Thus, the unique variances are often updated by using the Newton–Raphson method or Quasi-Newton method (Jöreskog, 1967; Lawley & Maxwell, 1971). The `factanal` function in R uses the `optim` function to update the unique variances.

To summarize, the estimation algorithm in Jöreskog (1967) is given as follows:

1. Set an initial value of Ψ .
2. For given Ψ , Eq. (8) implies that Λ is a function of Ψ ; that is, Λ can be written as $\Lambda = \Lambda(\Psi)$. Substituting $\Lambda(\Psi)$ into the discrepancy function (2), it is a function of Ψ . The elements of Ψ are then optimized by Newton–Raphson method or Quasi-Newton method.

2.3. A system of algebraic equations for the MLEs

In this study, we will employ a computational algebraic approach to get exact algebraic solutions to the system (4) (i.e., all candidates of the MLEs). However, Eq. (4) may not be directly used as it includes an inverse matrix Σ^{-1} that consists of significantly complicated rational functions. Consequently, a simpler system of algebraic equations is required. One may use (5) and (6) as they do not include Σ^{-1} , but Eq. (5) includes Ψ^{-1} , which is unavailable when Ψ is singular. As will be demonstrated in the numerical example in Section 4, there exist many estimates whose unique variances are exactly zeros. Thus, it is

essential to derive a system of algebraic equations that does not include complicated rational functions and also satisfies (4), even when Ψ is singular.

In the following theorem, we provide a necessary condition for (4) that does not have rational functions and holds even when Ψ is singular. Therefore, all solutions of (4) must satisfy that necessary condition.

Theorem 2.1. *Whether Ψ is singular or not, if (4) is satisfied, then we have*

$$\Lambda = (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda, \quad (9)$$

$$\Psi = \text{diag}(S - \Lambda\Lambda^\top). \quad (6)$$

Proof. It is shown that Eq. (9) is equivalent to the upper part of Eq. (4), since we have

$$\begin{aligned} \Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Lambda = 0 &\iff (\Sigma S^{-1}\Sigma)\Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Lambda = 0 \\ &\iff \Sigma S^{-1}\Lambda = \Lambda \iff (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda = \Lambda. \end{aligned}$$

Therefore, what we need to prove is that Eq. (4) implies Eq. (6). First, we have

$$\Sigma - S = (\Lambda\Lambda^\top + \Psi)\Sigma^{-1}(\Sigma - S)\Sigma^{-1}(\Lambda\Lambda^\top + \Psi)$$

by $\Sigma = \Lambda\Lambda^\top + \Psi$. Hence, the upper part of (4) implies that

$$\Sigma - S = \Psi\Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Psi.$$

To handle the case where Ψ is singular, suppose that ψ_1, \dots, ψ_q are not equal to zero and $\psi_{q+1}, \dots, \psi_p$ are equal to zero, where $q \leq p$. Let $J = \{1, \dots, q\}$, $K = \{q+1, \dots, p\}$, and let

$$T = \text{diag}\left(\frac{1}{\psi_1}, \dots, \frac{1}{\psi_q}, 1, \dots, 1\right), \quad I_{q \leq p} = \text{diag}\left(\overbrace{1, \dots, 1}^q, \overbrace{0, \dots, 0}^{p-q}\right).$$

Let M_{JK} denote a submatrix of a matrix M from which the rows corresponding to the indices in J and the columns corresponding to the indices in K are selected, respectively. Since T is not singular,

$$\begin{aligned} \Sigma - S = \Psi\Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Psi &\iff T(\Sigma - S)T = T\Psi\Sigma^{-1}(\Sigma - S)\Sigma^{-1}\Psi T \\ &= I_{q \leq p}\Sigma^{-1}(\Sigma - S)\Sigma^{-1}I_{q \leq p} \\ &= \begin{pmatrix} (\Sigma^{-1}(\Sigma - S)\Sigma^{-1})_{JJ} & 0 \\ 0 & 0 \end{pmatrix}. \end{aligned}$$

As $(T(\Sigma - S)T)_{JJ} = (\Sigma^{-1}(\Sigma - S)\Sigma^{-1})_{JJ}$ holds, the lower part of (4) implies that $\text{diag}((T(\Sigma - S)T)_{JJ}) = 0$. Hence, $\Psi_{JJ} = \text{diag}((S - \Lambda\Lambda^\top)_{JJ})$ holds because T is diagonal. Since $(\Sigma - S)_{KK} = I_{p-q}(\Sigma - S)_{KK}I_{p-q} = (T(\Sigma - S)T)_{KK} = 0$ holds, we also have $s_{ii} - \sum_{j=1}^k \lambda_{ij}^2 = 0 = \psi_i$ for each $i \in K$. Consequently, we obtain Eq. (6) without an assumption that Ψ is not singular. \square

Remark 1. As an alternative algorithm to obtain the MLEs given by Lawley & Maxwell (1971), Jennrich & Robinson (1969) used the eigenvalue/eigenvector problem of $S^{-\frac{1}{2}}\Psi S^{-\frac{1}{2}}$ instead of $\Psi^{-\frac{1}{2}}S\Psi^{-\frac{1}{2}}$ by using the following equation:

$$(S^{-\frac{1}{2}}\Psi S^{-\frac{1}{2}})S^{-\frac{1}{2}}\Lambda = S^{-\frac{1}{2}}\Lambda(I_k - \Lambda^\top S^{-1}\Lambda), \quad (10)$$

which is equivalent to Eq. (9). Hence, Jennrich and Robinson's (1969) algorithm can produce zero or negative unique variance estimates.

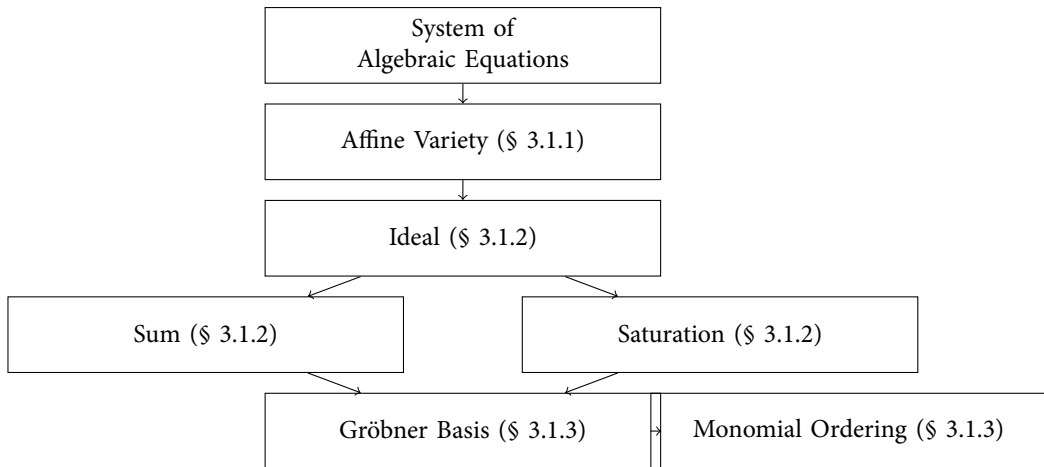


Figure 2. Algebraic concepts reviewed in Section 3.1.

3. MLEs via Gröbner bases

Theorem 2.1 implies that all solutions to (4) must satisfy (6) and (9); therefore, we first solve all exact solutions to (6) and (9) in order to find all the solutions of (4), and then obtain an optimal solution that maximizes the likelihood function. In this section, we propose an algebraic approach to finding exact solutions to (6) and (9). Section 3.1 reviews the theory of Gröbner bases for systems of algebraic equations (see Cox et al., 2015 for details). A simple illustrative example concluding Gröbner bases in maximum likelihood factor analysis is also presented. We provide an essential idea for constructing an algorithm to find the MLEs through the illustrative example. In particular, Section 3.1.1 explains the relationship between the concept of systems of algebraic equations and the concept of affine varieties; Section 3.1.2 describes properties of two ideals (sum and saturation) associated with affine varieties used in our algorithm; then, in Section 3.1.3, we review the concept of monomial orderings necessary to define Gröbner bases, and the theory of Gröbner bases necessary to compute the ideals reviewed in Section 3.1.2. For clarity, Figure 2 shows the relationship among these concepts reviewed in Section 3.1. With the algebraic concepts in this section, we can guarantee the correctness of our algorithm from properties of each algebraic concept by citing theorems and propositions from a textbook (e.g., Cox et al., 2015).

In Section 3.2, we introduce an algorithm for finding all candidates for the MLEs by using each concept reviewed in Section 3.1. Particularly, we decompose an original system of algebraic equations into several sub-problems, allowing us to get the MLE within a feasible computation time. Furthermore, Section 3.3 proposes an algorithm to obtain an optimal solution from the candidates for MLEs. Our proposed algorithm identifies a pattern of MLE among “Proper solution,” “Improper solution,” and “No solutions.”

3.1. Gröbner bases and systems of algebraic equations

3.1.1. Affine varieties and systems of algebraic equations

Let $\mathbb{R}[\mathbf{z}]$ denote the set of all polynomials in m variables $\mathbf{z} = (z_1, \dots, z_m)$ with coefficients in the real number field \mathbb{R} . Recall that $\mathbb{R}[\mathbf{z}]$ is a commutative ring, which is called the polynomial ring in variables \mathbf{z} with coefficients in \mathbb{R} . For details of fields and commutative rings, please refer to Appendix A. Let f_1, \dots, f_r be polynomials in the polynomial ring $\mathbb{R}[\mathbf{z}]$. We consider a system of algebraic equations which has the following general form:

$$f_1(\mathbf{z}) = \dots = f_r(\mathbf{z}) = 0. \quad (11)$$

Define

$$\mathbb{V}_W(f_1, \dots, f_r) = \{\mathbf{z} \in W^m : f_1(\mathbf{z}) = \dots = f_r(\mathbf{z}) = 0\} \subset W^m,$$

where $W = \mathbb{R}$ or \mathbb{C} . The set $\mathbb{V}_W(f_1, \dots, f_r)$ is referred to as an affine variety defined by f_1, \dots, f_r in W^m (Cox *et al.*, 2015, Section 1.2, Definition 1). Note that the solutions to (11) in W^m form the affine variety $\mathbb{V}_W(f_1, \dots, f_r)$, that is, the solution space to (11) in W^m coincides with the affine variety $\mathbb{V}_W(f_1, \dots, f_r)$.

Here, let us consider an illustrative example related to the system of algebraic equations (6) and (9).

Example 1. Let

$$S = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & 1 & \frac{2}{3} \\ \frac{1}{3} & \frac{2}{3} & 1 \end{pmatrix}, \quad \Lambda = \begin{pmatrix} \lambda_{11} \\ \lambda_{21} \\ \lambda_{31} \end{pmatrix}, \quad \Psi = \begin{pmatrix} \psi_1 & 0 & 0 \\ 0 & \psi_2 & 0 \\ 0 & 0 & \psi_3 \end{pmatrix}.$$

We consider the system of algebraic equations (6) and (9) for the above S, Λ, Ψ , that is,

$$\begin{cases} 0 = (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{11} = (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{22} = (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{33}, \\ 0 = (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{11} = (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{21} = (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{31}. \end{cases} \quad (12)$$

Because we have

$$S^{-1} = \begin{pmatrix} \frac{4}{3} & -\frac{2}{3} & 0 \\ -\frac{2}{3} & \frac{32}{15} & -\frac{6}{5} \\ 0 & -\frac{6}{5} & \frac{9}{5} \end{pmatrix},$$

we consider polynomials $f_1, \dots, f_6 \in \mathbb{R}[\boldsymbol{\psi}, \boldsymbol{\lambda}] = \mathbb{R}[\psi_1, \psi_2, \psi_3, \lambda_{11}, \lambda_{21}, \lambda_{31}]$ defined by

$$\begin{aligned} f_1 &= (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{11} = \psi_1 - (1 - \lambda_{11}^2), \\ f_2 &= (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{22} = \psi_2 - (1 - \lambda_{21}^2), \\ f_3 &= (\Psi - \text{diag}(S - \Lambda\Lambda^\top))_{33} = \psi_3 - (1 - \lambda_{31}^2), \\ f_4 &= (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{11} \\ &= \lambda_{11} - \left(\lambda_{11} \left(-\frac{2}{3}\lambda_{11}\lambda_{21} + \frac{4}{3}(\lambda_{11}^2 + \psi_1) \right) \right. \\ &\quad \left. + \lambda_{21} \left(\frac{32}{15}\lambda_{11}\lambda_{21} - \frac{6}{5}\lambda_{11}\lambda_{31} - \frac{2}{3}(\lambda_{11}^2 + \psi_1) \right) + \lambda_{31} \left(-\frac{6}{5}\lambda_{11}\lambda_{21} + \frac{9}{5}\lambda_{11}\lambda_{31} \right) \right), \\ f_5 &= (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{21} \\ &= \lambda_{21} - \left(\lambda_{11} \left(\frac{4}{3}\lambda_{11}\lambda_{21} - \frac{2}{3}(\lambda_{21}^2 + \psi_2) \right) \right. \\ &\quad \left. + \lambda_{21} \left(-\frac{2}{3}\lambda_{11}\lambda_{21} - \frac{6}{5}\lambda_{21}\lambda_{31} + \frac{32}{15}(\lambda_{21}^2 + \psi_2) \right) + \lambda_{31} \left(\frac{9}{5}\lambda_{21}\lambda_{31} - \frac{6}{5}(\lambda_{21}^2 + \psi_2) \right) \right), \\ f_6 &= (\Lambda - (\Lambda\Lambda^\top + \Psi)S^{-1}\Lambda)_{31} \\ &= \lambda_{31} - \left(\lambda_{11} \left(\frac{4}{3}\lambda_{11}\lambda_{31} - \frac{2}{3}\lambda_{21}\lambda_{31} \right) \right. \\ &\quad \left. + \lambda_{21} \left(-\frac{2}{3}\lambda_{11}\lambda_{31} + \frac{32}{15}\lambda_{21}\lambda_{31} - \frac{6}{5}(\lambda_{31}^2 + \psi_3) \right) + \lambda_{31} \left(-\frac{6}{5}\lambda_{21}\lambda_{31} + \frac{9}{5}(\lambda_{31}^2 + \psi_3) \right) \right). \end{aligned}$$

All solutions of the above equation form the affine variety defined by f_1, \dots, f_6 . Indeed, there are seven real solutions to (12), and these solutions, $(\boldsymbol{\psi}, \boldsymbol{\lambda}) = (\psi_1, \psi_2, \psi_3, \lambda_{11}, \lambda_{21}, \lambda_{31}) \in \mathbb{R}^6$, form the following affine variety:

$$\begin{aligned} &\mathbb{V}_{\mathbb{R}}(f_1, \dots, f_6) \\ &= \left\{ \left(0, \frac{3}{4}, \frac{8}{9}, \pm 1, \pm \frac{1}{2}, \pm \frac{1}{3} \right), \left(\frac{3}{4}, 0, \frac{5}{9}, \pm \frac{1}{2}, \pm 1, \pm \frac{2}{3} \right), \left(\frac{8}{9}, \frac{5}{9}, 0, \pm \frac{1}{3}, \pm \frac{2}{3}, \pm 1 \right), (1, 1, 1, 0, 0, 0) \right\}. \end{aligned}$$

3.1.2. Sums and saturations

In Example 1, we obtain only seven solutions to the system of algebraic equations (12). For a larger model, however, the number of solutions of (6) and (9) is considerably large and the solution space (i.e., the affine variety) is too complicated. In some cases, there are an infinite number of solutions.

Therefore, it is essential to enhance an efficient computation for obtaining all solutions. A key idea is a decomposition of (6) and (9) into several simple sub-problems. This decomposition is related to the theory of polynomial ideals and Gröbner bases.

We define the following set of polynomials:

$$\langle f_1, \dots, f_r \rangle = \left\{ \sum_{i=1}^r q_i f_i : q_i \in \mathbb{R}[\mathbf{z}] \right\},$$

where $f_1, \dots, f_r \in \mathbb{R}[\mathbf{z}]$. The above set is an ideal in the polynomial ring $\mathbb{R}[\mathbf{z}]$. So, it is called the ideal generated by f_1, \dots, f_r (Cox et al., 2015, Section 1.4, Definition 2). Let

$$\mathcal{J} = \langle f_1, \dots, f_r \rangle \subset \mathbb{R}[\mathbf{z}].$$

For details of the ideal in the commutative ring, please refer to Appendix A. The affine variety defined by \mathcal{J} in W^m is expressed as $\mathbb{V}_W(\mathcal{J}) = \{\mathbf{z} \in W^m : f(\mathbf{z}) = 0, \forall f \in \mathcal{J}\}$. Since $\mathbb{V}_W(f_1, \dots, f_r) = \mathbb{V}_W(\mathcal{J})$ holds, the solution space of (11) coincides with $\mathbb{V}_W(\mathcal{J})$.

Since ideals are algebraic objects, there exist natural algebraic operations. Essentially, such algebraic operations give us some simple sub-problems of the system of algebraic equations (6) and (9). Let \mathcal{K} be an ideal in $\mathbb{R}[\mathbf{z}]$. We note that any ideal in $\mathbb{R}[\mathbf{z}]$ is finitely generated by the Hilbert Basis Theorem (Cox et al., 2015, Theorem 4, Section 2.5), that is, there exist $h_1, \dots, h_s \in \mathcal{K}$ such that $\mathcal{K} = \langle h_1, \dots, h_s \rangle$.

Now, we define two operations. As the first operation for ideals, we define the sum of \mathcal{J} and \mathcal{K} , say $\mathcal{J} + \mathcal{K}$, by

$$\mathcal{J} + \mathcal{K} = \{f + h : f \in \mathcal{J}, h \in \mathcal{K}\}.$$

For details of the definition of sums, see (Cox et al., 2015, Section 4.3, Definition 1). Note that any sum of ideals is an ideal in $\mathbb{R}[\mathbf{z}]$ (Cox et al., 2015, Proposition 2, Section 4.3). Moreover, the affine variety $\mathbb{V}_W(\mathcal{J} + \mathcal{K})$ coincides with $\mathbb{V}_W(\mathcal{J}) \cap \mathbb{V}_W(\mathcal{K})$ (Cox et al., 2015, Corollary 3, Section 4.3), that is, $\mathbb{V}_W(\mathcal{J} + \mathcal{K})$ is none other than the solution space of the equations

$$(11) \text{ and } h_1(\mathbf{z}) = \dots = h_s(\mathbf{z}) = 0.$$

As the second operation, we define the saturation of \mathcal{J} with respect to \mathcal{K} , say $\mathcal{J} : \mathcal{K}^\infty$, by

$$\mathcal{J} : \mathcal{K}^\infty = \{f \in \mathbb{R}[\mathbf{z}] : \text{for any } h \in \mathcal{K} \text{ there exists } t \in \mathbb{Z}_{\geq 0} \text{ such that } fh^t \in \mathcal{J}\},$$

where $\mathbb{Z}_{\geq 0} = \{t \in \mathbb{Z} : t \geq 0\}$, and fh^t means the product of the polynomial f and the t -th power of h denoted by h^t . For details of the definition of saturations, see (Cox et al., 2015, Section 4.4, Definition 8). Note that any saturation is an ideal in $\mathbb{R}[\mathbf{z}]$ (Cox et al., 2015, Proposition 9, Section 4.4). If $W = \mathbb{C}$, the affine variety $\mathbb{V}_{\mathbb{C}}(\mathcal{J} : \mathcal{K}^\infty)$ coincides with the Zariski closure of the subset $\mathbb{V}_{\mathbb{C}}(\mathcal{J}) \setminus \mathbb{V}_{\mathbb{C}}(\mathcal{K}) \subseteq \mathbb{C}^m$ (Cox et al., 2015, Theorem 10, Section 4.4), that is, the $\mathbb{V}_{\mathbb{C}}(\mathcal{J} : \mathcal{K}^\infty)$ is none other than the Zariski closure of the solution space over \mathbb{C} of

$$((11) \text{ and } h_1(\mathbf{z}) \neq 0) \text{ or } \dots \text{ or } ((11) \text{ and } h_s(\mathbf{z}) \neq 0),$$

where $\mathbb{V}_{\mathbb{C}}(\mathcal{J}) \setminus \mathbb{V}_{\mathbb{C}}(\mathcal{K}) \subseteq \mathbb{C}^m$ is defined by

$$\mathbb{V}_{\mathbb{C}}(\mathcal{J}) \setminus \mathbb{V}_{\mathbb{C}}(\mathcal{K}) = \{\mathbf{z} \in \mathbb{V}_{\mathbb{C}}(\mathcal{J}) : \mathbf{z} \notin \mathbb{V}_{\mathbb{C}}(\mathcal{K})\}.$$

Recall that the Zariski closure of a subset in $W^m = \mathbb{R}^m$ or \mathbb{C}^m is the smallest affine variety containing the subset (see Appendix A for details of Zariski closures).

Now, we provide a proposition that gives theoretical correctness to divide Eqs. (6) and (9) into some sub-problems (Cox et al., 2015, Theorem 10, Section 4.4).

Proposition 3.1. We have $\mathbb{V}_W(\mathcal{J}) = \mathbb{V}_W(\mathcal{J} + \mathcal{K}) \cup \mathbb{V}_W(\mathcal{J} : \mathcal{K}^\infty)$.

Proposition 3.1 tells that if we obtain generator sets of the sum $\mathcal{J} + \mathcal{K}$ and the saturation $\mathcal{J} : \mathcal{K}^\infty$, and both of their affine varieties are not empty, we get meaningful sub-problems of (11).

Now, let us turn our attention to the system of algebraic equations for the maximum likelihood factor analysis. The result of Example 1 shows that the affine variety includes many solutions whose Ψ is singular; 6 out of 7 solutions are singular. In fact, as we will see in the numerical example presented in Section 4, the solution space of (6) and (9) includes a surprisingly large number of solutions whose Ψ is singular. Therefore, we propose to decompose the original problem into the following sub-problems: $\psi_j = 0$ and $\psi_j \neq 0$ for $j = 1, \dots, p$. Now, we provide a simple example based on Example 1 to illustrate how the decomposition can be achieved. A rigorous procedure will be provided in Section 3.2.

Example 2. We illustrate the usefulness of Proposition 3.1 using the same example as in Example 1. Let us consider the ideal $\mathcal{J} = \langle f_1, \dots, f_6 \rangle$ in the polynomial ring $\mathbb{R}[\psi, \lambda]$, where f_1, \dots, f_6 are polynomials given in Example 1. We construct some ideals sequentially as follows: first, by using the ideal \mathcal{J} , we construct the following sum and saturation:

$$\mathcal{I}_0 = \mathcal{J} + \langle \psi_1 \rangle, \quad \mathcal{I}_1 = \mathcal{J} : \langle \psi_1 \rangle^\infty.$$

Next, by using the ideals \mathcal{I}_0 and \mathcal{I}_1 , we construct the following sums and saturations:

$$\mathcal{I}_{00} = \mathcal{I}_0 + \langle \psi_2 \rangle, \quad \mathcal{I}_{01} = \mathcal{I}_0 : \langle \psi_2 \rangle^\infty, \quad \mathcal{I}_{10} = \mathcal{I}_1 + \langle \psi_2 \rangle, \quad \mathcal{I}_{11} = \mathcal{I}_1 : \langle \psi_2 \rangle^\infty.$$

Last, by using the ideals $\mathcal{I}_{00}, \mathcal{I}_{01}, \mathcal{I}_{10}$, and \mathcal{I}_{11} , we construct the following sums and saturations:

$$\begin{aligned} \mathcal{I}_{000} &= \mathcal{I}_{00} + \langle \psi_3 \rangle, \quad \mathcal{I}_{001} = \mathcal{I}_{00} : \langle \psi_3 \rangle^\infty, \quad \mathcal{I}_{010} = \mathcal{I}_{01} + \langle \psi_3 \rangle, \quad \mathcal{I}_{011} = \mathcal{I}_{01} : \langle \psi_3 \rangle^\infty, \\ \mathcal{I}_{100} &= \mathcal{I}_{10} + \langle \psi_3 \rangle, \quad \mathcal{I}_{101} = \mathcal{I}_{10} : \langle \psi_3 \rangle^\infty, \quad \mathcal{I}_{110} = \mathcal{I}_{11} + \langle \psi_3 \rangle, \quad \mathcal{I}_{111} = \mathcal{I}_{11} : \langle \psi_3 \rangle^\infty. \end{aligned}$$

It follows from Proposition 3.1 that

$$\begin{aligned} \mathbb{V}_\mathbb{R}(\mathcal{J}) &= \mathbb{V}_\mathbb{R}(\mathcal{I}_0) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_1) \\ &= (\mathbb{V}_\mathbb{R}(\mathcal{I}_{00}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{01})) \cup (\mathbb{V}_\mathbb{R}(\mathcal{I}_{10}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{11})) \\ &= ((\mathbb{V}_\mathbb{R}(\mathcal{I}_{000}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{001})) \cup (\mathbb{V}_\mathbb{R}(\mathcal{I}_{010}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{011}))) \\ &\quad \cup ((\mathbb{V}_\mathbb{R}(\mathcal{I}_{100}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{101})) \cup (\mathbb{V}_\mathbb{R}(\mathcal{I}_{110}) \cup \mathbb{V}_\mathbb{R}(\mathcal{I}_{111}))) = \bigcup_{b \in \{0,1\}^3} \mathbb{V}_\mathbb{R}(\mathcal{I}_b). \end{aligned}$$

The affine varieties $\mathbb{V}_\mathbb{R}(\mathcal{I}_{000}), \mathbb{V}_\mathbb{R}(\mathcal{I}_{100}), \mathbb{V}_\mathbb{R}(\mathcal{I}_{010}), \mathbb{V}_\mathbb{R}(\mathcal{I}_{001})$ are empty sets. On the other hand,

$$\begin{aligned} \mathbb{V}_\mathbb{R}(\mathcal{I}_{011}) &= \left\{ \left(0, \frac{3}{4}, \frac{8}{9}, \pm 1, \pm \frac{1}{2}, \pm \frac{1}{3} \right) \right\}, & \mathbb{V}_\mathbb{R}(\mathcal{I}_{101}) &= \left\{ \left(\frac{3}{4}, 0, \frac{5}{9}, \pm \frac{1}{2}, \pm 1, \pm \frac{2}{3} \right) \right\}, \\ \mathbb{V}_\mathbb{R}(\mathcal{I}_{110}) &= \left\{ \left(\frac{8}{9}, \frac{5}{9}, 0, \pm \frac{1}{3}, \pm \frac{2}{3}, \pm 1 \right) \right\}, & \mathbb{V}_\mathbb{R}(\mathcal{I}_{111}) &= \{(1, 1, 1, 0, 0, 0)\} \end{aligned}$$

are non-empty proper subsets of $\mathbb{V}_\mathbb{R}(\mathcal{J})$. Thus, the solution space of (12) is divided into the four affine varieties: $\mathbb{V}_\mathbb{R}(\mathcal{I}_{011}), \mathbb{V}_\mathbb{R}(\mathcal{I}_{101}), \mathbb{V}_\mathbb{R}(\mathcal{I}_{110})$, and $\mathbb{V}_\mathbb{R}(\mathcal{I}_{111})$.

3.1.3. Gröbner bases

Example 2 showed that the solution space can be successfully decomposed by constructing sums and saturations. To obtain some simple sub-problems of the system of algebraic equations (6) and (9) for maximum likelihood factor analysis, in addition, we need to compute generator sets of the sum $\mathcal{J} + \mathcal{K}$ and the saturation $\mathcal{J} : \mathcal{K}^\infty$. The sum $\mathcal{J} + \mathcal{K}$ is generated by $f_1, \dots, f_r, h_1, \dots, h_s$, that is, $\mathcal{J} + \mathcal{K} = \langle f_1, \dots, f_r, h_1, \dots, h_s \rangle$. A generator set of the saturation $\mathcal{J} : \mathcal{K}^\infty$ is obtained by using a Gröbner basis. The definition of the Gröbner basis requires a monomial ordering. Therefore, we define monomial orderings before giving the definition of Gröbner bases.

Let us denote $\mathbf{z}^{\mathbf{a}} = z_1^{a_1} \dots z_m^{a_m}$ for $\mathbf{a} = (a_1, \dots, a_m) \in \mathbb{Z}_{\geq 0}^m$, which is called a monomial, where $\mathbb{Z}_{\geq 0}^m = \{(\alpha_1, \dots, \alpha_m) \in \mathbb{Z}^m : \alpha_1, \dots, \alpha_m \geq 0\}$. Let $M(\mathbf{z}) = \{\mathbf{z}^{\mathbf{a}} : \mathbf{a} \in \mathbb{Z}_{\geq 0}^m\}$. The definition of monomial orderings is the following (Cox et al., 2015, Section 2.2, Definition 1).

Definition 1. A monomial ordering $>$ on $\mathbb{R}[\mathbf{z}]$ is a relation on the set $M(\mathbf{z})$ satisfying that

1. $>$ is a total ordering on $M(\mathbf{z})$, that is, $>$ satisfies that
 - (a) $s \geq s$ for any $s \in M(\mathbf{z})$,
 - (b) $s \geq t$ and $t \geq s$ implies $s = t$ for any $s, t \in M(\mathbf{z})$,
 - (c) $s \geq t$ and $t \geq u$ implies $s \geq u$ for any $s, t, u \in M(\mathbf{z})$, and
 - (d) $s > t$ or $s = t$ or $s < t$ for any $s, t \in M(\mathbf{z})$,
2. $\mathbf{z}^{\mathbf{a}} > \mathbf{z}^{\mathbf{b}} \implies \mathbf{z}^{\mathbf{a}} \mathbf{z}^{\mathbf{c}} > \mathbf{z}^{\mathbf{b}} \mathbf{z}^{\mathbf{c}}$ for any $\mathbf{c} \in \mathbb{Z}_{\geq 0}^m$, and
3. any nonempty subset of $M(\mathbf{z})$ has a smallest element under $>$.

Definition 2. Fixing a monomial ordering on $\mathbb{R}[\mathbf{z}]$ and given a polynomial $f = \sum_{\mathbf{a} \in \mathbb{Z}_{\geq 0}^m} c_{\mathbf{a}} \mathbf{z}^{\mathbf{a}}$ for $c_{\mathbf{a}} \in \mathbb{R}$, we call the monomial $\text{LM}(f) = \max(\mathbf{z}^{\mathbf{a}} : c_{\mathbf{a}} \neq 0)$ the *leading monomial* of f .

We give two examples of monomial orderings. The first one is called a *lexicographic order* (or the *lex order* for short) (Cox et al., 2015, Section 2.2, Definition 3). The second one is called a *graded reverse lexicographic order* (or the *grevlex order* for short) (Cox et al., 2015, Section 2.2, Definition 6). These two monomial orderings satisfy the conditions of Definition 1 (Cox et al., 2015, Proposition 4 and Exercise 5, Section 2.2).

Definition 3 (Lexicographic Order; Lex). Let $\mathbf{a}, \mathbf{b} \in \mathbb{Z}_{\geq 0}^m$. We say $\mathbf{z}^{\mathbf{a}} >_{\text{lex}} \mathbf{z}^{\mathbf{b}}$ if the leftmost nonzero entry of the vector difference $\mathbf{a} - \mathbf{b}$ is positive. This ordering is called a *lexicographic order* (or the *lex order* for short).

Although a Gröbner basis with respect to the lex order defined above transforms a given system of algebraic equations into an equivalent system of algebraic equations that is easier to solve, it is widely known in computational algebra that such computations tend to be heavy in general. Therefore, we next define the graded lexicographic order, a monomial order under which Gröbner basis computation is generally known to be more efficient. Of course, a Gröbner basis with respect to the graded reverse lexicographic order also provides an equivalent transformation.

Definition 4 (Graded Reverse Lexicographic Order; Grevlex). Let $\mathbf{a} = (a_1, \dots, a_m) \in \mathbb{Z}_{\geq 0}^m$ and $\mathbf{b} = (b_1, \dots, b_m) \in \mathbb{Z}_{\geq 0}^m$. We say $\mathbf{z}^{\mathbf{a}} >_{\text{grevlex}} \mathbf{z}^{\mathbf{b}}$ if $|\mathbf{a}| = \sum_{i=1}^m a_i > |\mathbf{b}| = \sum_{i=1}^m b_i$ or the rightmost nonzero entry of the vector difference $\mathbf{a} - \mathbf{b}$ is negative. This ordering is called a *graded reverse lexicographic order* (or the *grevlex order* for short).

As shown in the following example, the leading monomial of a given polynomial is determined depending on a fixed monomial ordering.

Example 3 (Lexicographic Order; Lex). When we fix the lex order, $\text{LM}(3z_1^2 + 2z_1z_2^3) = z_1^2$. In addition, $\text{LM}(f_4) = \psi_1$, $\text{LM}(f_5) = \psi_2$, and $\text{LM}(f_6) = \psi_3$, where $f_4, f_5, f_6 \in \mathbb{R}[\boldsymbol{\psi}, \boldsymbol{\lambda}]$ are presented in Example 1.

Example 4 (Graded Reverse Lexicographic Order; Grevlex). When we fix the grevlex order, $\text{LM}(3z_1^2 + 2z_1z_2^3) = z_1z_2^3$. In addition, $\text{LM}(f_4) = \lambda_{11}^3$, $\text{LM}(f_5) = \lambda_{11}^2\lambda_{21}$, and $\text{LM}(f_6) = \lambda_{11}^2\lambda_{31}$, where $f_4, f_5, f_6 \in \mathbb{R}[\boldsymbol{\psi}, \boldsymbol{\lambda}]$ are presented in Example 1.

Now, we define Gröbner bases (Cox et al., 2015, Section 2.5, Definition 5).

Definition 5. Fix a monomial ordering on $\mathbb{R}[\mathbf{z}]$. A finite subset $G = \{g_1, \dots, g_t\}$ of an ideal $\mathcal{J} \subseteq \mathbb{R}[\mathbf{z}]$ is said to be a *Gröbner basis* of \mathcal{J} if we have

$$\langle \text{LM}(g_1), \dots, \text{LM}(g_t) \rangle = \langle \{\text{LM}(f) : f \in \mathcal{J}\} \rangle.$$

Note that every polynomial ideal has a Gröbner basis (Cox et al., 2015, Corollary 6, Section 2.5), and that any Gröbner basis of \mathcal{J} generates the ideal \mathcal{J} .

For the case $\mathcal{K} = \langle h \rangle \subseteq \mathbb{R}[\mathbf{z}]$, a generator set of the saturation $\mathcal{J} : \mathcal{K}^{\infty}$ is obtained by using a Gröbner basis as follows (Cox et al., 2015, Theorem 14(ii), Section 4.4):

1. let $\tilde{\mathcal{J}} = \langle f_1, \dots, f_r, 1 - yh \rangle \subseteq \mathbb{R}[y, z]$,
2. compute a Gröbner basis of $\tilde{\mathcal{J}}$, say \tilde{G} , with respect to the lex order,
3. $G = \tilde{G} \cap \mathbb{R}[z]$ is a Gröbner basis of the saturation $\mathcal{J} : \mathcal{K}^\infty$.

Example 5. Let us reconsider Example 1, which is related to the system of algebraic equations (6) and (9). Gröbner bases of the sums and saturations described above give us simplified sub-problems. For $b \in \{0, 1\}^3$, a Gröbner basis G_b of \mathcal{I}_b with respect to the lex order is given by

$$\begin{aligned} G_{000} &= \{1\}, \quad G_{001} = \{1\}, \quad G_{010} = \{1\}, \quad G_{100} = \{1\}, \\ G_{011} &= \left\{ \psi_1, \psi_2 - \frac{3}{4}, \psi_3 - \frac{8}{9}, \lambda_{11} - 3\lambda_{31}, \lambda_{21} - \frac{3}{2}\lambda_{31}, \lambda_{31}^2 - \frac{1}{9} \right\}, \\ G_{101} &= \left\{ \psi_1 - \frac{3}{4}, \psi_2, \psi_3 - \frac{5}{9}, \lambda_{11} - \frac{3}{4}\lambda_{31}, \lambda_{21} - \frac{3}{2}\lambda_{31}, \lambda_{31}^2 - \frac{4}{9} \right\}, \\ G_{110} &= \left\{ \psi_1 - \frac{8}{9}, \psi_2 - \frac{5}{9}, \psi_3, \lambda_{11} - \frac{1}{3}\lambda_{31}, \lambda_{21} - \frac{2}{3}\lambda_{31}, \lambda_{31}^2 - 1 \right\}, \\ G_{111} &= \left\{ \psi_1 + \frac{81}{25}\lambda_{31}^2 - 1, \psi_2 - 1, \psi_3 + \lambda_{31}^2 - 1, \lambda_{11} + \frac{9}{5}\lambda_{31}, \lambda_{21} + \lambda_{31}^3 + \frac{5}{27}\lambda_{31} \right\}. \end{aligned}$$

So the Gröbner bases $G_{011}, G_{101}, G_{110}, G_{111}$ divide (12) into the following simple sub-problems:

$$\begin{aligned} \begin{cases} \psi_1 = 0, \\ \psi_2 = \frac{3}{4}, \\ \psi_3 = \frac{8}{9}, \\ \lambda_{11} = 3\lambda_{31}, \\ \lambda_{21} = \frac{3}{2}\lambda_{31}, \\ 0 = \lambda_{31}^2 - \frac{1}{9}, \end{cases} & \begin{cases} \psi_1 = \frac{3}{4}, \\ \psi_2 = 0, \\ \psi_3 = \frac{5}{9}, \\ \lambda_{11} = \frac{3}{4}\lambda_{31}, \\ \lambda_{21} = \frac{3}{2}\lambda_{31}, \\ 0 = \lambda_{31}^2 - \frac{4}{9}, \end{cases} & \begin{cases} \psi_1 = \frac{8}{9}, \\ \psi_2 = \frac{5}{9}, \\ \psi_3 = 0, \\ \lambda_{11} = \frac{1}{3}\lambda_{31}, \\ \lambda_{21} = \frac{2}{3}\lambda_{31}, \\ 0 = \lambda_{31}^2 - 1, \end{cases} & \begin{cases} \psi_1 = 1 - \frac{81}{25}\lambda_{31}^2, \\ \psi_2 = 1, \\ \psi_3 = 1 - \lambda_{31}^2, \\ \lambda_{11} = -\frac{9}{5}\lambda_{31}, \\ \lambda_{21} = 0, \\ 0 = \lambda_{31}^3 + \frac{5}{27}\lambda_{31}. \end{cases} \end{aligned}$$

Gröbner bases have a good property that plays an important role in solving a system of algebraic equations of the form (11). The following proposition shows that Gröbner bases transform (11) into easily solvable problems, without being subject to chance. The following proposition is called the finiteness theorem (Cox et al., 2015, Theorem 6, Section 5.3).

Proposition 3.2. Fix a monomial ordering on $\mathbb{R}[z]$ and let G be a Gröbner basis of an ideal $\mathcal{J} \subseteq \mathbb{R}[z]$. Consider the following four statements:

1. for each $i = 1, \dots, m$, there exists $t_i \geq 0$ such that $z_i^{t_i} \in \langle \text{LM}(f) : f \in \mathcal{J} \rangle$,
2. for each $i = 1, \dots, m$, there exists $t_i \geq 0$ and $g \in G$ such that $\text{LM}(g) = z_i^{t_i}$,
3. for each $i = 1, \dots, m$, there exists $t_i \geq 0$ and $g \in G$ such that $\text{LM}(g) = z_i^{t_i}$ and $g \in \mathbb{R}[z_i, \dots, z_m]$, if we fix the lex order,
4. the affine variety $\mathbb{V}_W(\mathcal{J})$ is a finite set.

Then, statements 1–3 are equivalent and they all imply statement 4. An ideal \mathcal{J} satisfying statement 1, 2, or 3 is called a zero-dimensional ideal. Otherwise, it is called a non-zero-dimensional ideal. Furthermore, if $W = \mathbb{C}$, then statements 1–4 are all equivalent.

The above proposition shows that if we can obtain a Gröbner basis, we can determine whether the affine variety is a finite or infinite set, and if the affine variety is a finite set, the Gröbner basis with respect to the lex order gives a triangulated representation (statement 3 of Proposition 3.2). This representation produces a much simpler system of algebraic equations than the original one; for example, when we need to solve linear equations, its Gröbner basis provides a triangular matrix, corresponding to the triangularization of linear equations. In general, the computation of a Gröbner basis with respect to the grevlex order is more efficient than that with respect to the lex order. Also, if \mathcal{J} is zero-dimensional, we have an efficient algorithm which converts a Gröbner basis from one monomial ordering to another, which is called the FGLM algorithm (Faugère et al., 1993). Hence, in our algebraic approach to maximum likelihood factor analysis, we first compute Gröbner bases with respect to the grevlex order, and then convert them to Gröbner bases with respect to the lex order.

Remark 2. Primary ideal decompositions can *more completely* give us simplified sub-problems of a system of algebraic equations. However, unfortunately, the primary ideal decompositions require *heavy* computational loads in general. Thus, we employ Proposition 3.1 instead of primary ideal decompositions.

3.2. An algorithm to find all solutions

We provide an algorithm to find all solutions to Eq. (4). To get the solutions, we solve the following equation:

$$\{S - \Lambda\Lambda^\top - \text{diag}(S - \Lambda\Lambda^\top)\}S^{-1}\Lambda = 0, \quad (13)$$

which is derived by substituting (6) into (9). Note that, according to Theorem 2.1, the system of algebraic equations (13) is a necessary condition for the likelihood equation (4). Since the solution space of (13) contains that of (4), we can find all solutions to (4) by solving (13). Eq. (13) is much easier to solve than Eq. (4); this is because Eq. (4) includes the inverse matrix Σ^{-1} that consists of rational functions, while Eq. (13) does not include any inverse matrix.

As described in Example 2 in Section 3.1, (13) can have a large number of, and sometimes infinite, solutions when Ψ is singular. Therefore, as in Example 2, we use Proposition 3.1 to get some sub-problems of Eq. (13).

Considering a rotational indeterminacy, we suppose that the upper triangular elements of Λ are fixed by zero. Let us consider the following ideal in $\mathbb{R}[\lambda] = \mathbb{R}[\lambda_{ij} : 1 \leq i \leq p, 1 \leq j \leq \min(i, k)]$:

$$\mathcal{J} = \left\langle \{S - \Lambda\Lambda^\top - \text{diag}(S - \Lambda\Lambda^\top)\}S^{-1}\Lambda\right\rangle_{ij} : 1 \leq i \leq p, 1 \leq j \leq k \Big\rangle.$$

We have to choose the ideals of the form $\mathcal{K} = \langle h \rangle \subseteq \mathbb{R}[\lambda]$ well to get sub-problems of (13).

Recall that (6) is equivalent to the lower part of (4) when Ψ is not singular, and is a necessary condition of (4) when Ψ is singular, as shown in Theorem 2.1. Focusing on (6), we consider ψ_1, \dots, ψ_p as polynomials defined by

$$\psi_i = s_{ii} - \sum_{j=1}^{\min(i, k)} \lambda_{ij}^2 \in \mathbb{R}[\lambda] \quad (i = 1, \dots, p).$$

For each $\mathcal{K}_i = \langle \psi_i \rangle \subseteq \mathbb{R}[\lambda]$, we construct sums and saturations as in our algorithm.

Now, we propose our algorithm in Algorithm 1. In Step 3 of Algorithm 1, the sum $\mathcal{I}_0 = \mathcal{J} + \mathcal{K}_1$ and the saturation $\mathcal{I}_1 = \mathcal{J} : \mathcal{K}_1^\infty$ are constructed, where $\mathcal{J}, \mathcal{K}_1, \dots, \mathcal{K}_p$ are given in Steps 1–3 of Algorithm 1. Proposition 3.1 implies that

$$\mathbb{V}_W(\mathcal{J}) = \mathbb{V}_W(\mathcal{I}_0) \cup \mathbb{V}_W(\mathcal{I}_1).$$

Second, we construct the following sums and saturations as in $i = 1$ of Steps 4–8 of Algorithm 1:

$$\mathcal{I}_{00} = \mathcal{I}_0 + \mathcal{K}_2, \quad \mathcal{I}_{01} = \mathcal{I}_0 : \mathcal{K}_2^\infty, \quad \mathcal{I}_{10} = \mathcal{I}_1 + \mathcal{K}_2, \quad \mathcal{I}_{11} = \mathcal{I}_1 : \mathcal{K}_2^\infty.$$

Since Proposition 3.1 implies

$$\mathbb{V}_W(\mathcal{I}_0) = \mathbb{V}_W(\mathcal{I}_{00}) \cup \mathbb{V}_W(\mathcal{I}_{01}), \quad \mathbb{V}_W(\mathcal{I}_1) = \mathbb{V}_W(\mathcal{I}_{10}) \cup \mathbb{V}_W(\mathcal{I}_{11}),$$

we have

$$\mathbb{V}_W(\mathcal{J}) = \mathbb{V}_W(\mathcal{I}_{00}) \cup \mathbb{V}_W(\mathcal{I}_{01}) \cup \mathbb{V}_W(\mathcal{I}_{10}) \cup \mathbb{V}_W(\mathcal{I}_{11}).$$

In the same way, for each $b \in \{0, 1\}^i, 2 \leq i \leq p-1$, we sequentially construct

$$\mathcal{I}_{b0} = \mathcal{I}_b + \mathcal{K}_{i+1}, \quad \mathcal{I}_{b1} = \mathcal{I}_b : \mathcal{K}_{i+1}^\infty$$

as in each $2 \leq i \leq p-1$ of Steps 4–8 of Algorithm 1. By Proposition 3.1, we have

$$\mathbb{V}_W(\mathcal{J}) = \bigcup_{b \in \{0,1\}^p} \mathbb{V}_W(\mathcal{I}_b).$$

Finally, we compute all solutions to (13) by using the following steps for each $b \in \{0,1\}^p$ as in Steps 9–17 of Algorithm 1:

1. compute a Gröbner basis $\tilde{G}_b = \{\tilde{g}_1, \dots, \tilde{g}_s\}$ of \mathcal{I}_b with respect to the grevlex order as in Step 10 of Algorithm 1,
2. if \mathcal{I}_b is zero-dimensional, compute a Gröbner basis $G_b = \{g_1, \dots, g_r\}$ of \mathcal{I}_b with respect to the lex order by converting \tilde{G}_b to a Gröbner basis with respect to the lex order as in Step 12 of Algorithm 1, and then solve the equation $g_1 = \dots = g_r = 0$ as in Step 13 of Algorithm 1,
3. else, compute a sample point from each connected component defined by $\tilde{g}_1 = \dots = \tilde{g}_s = 0$ as in Step 15 of Algorithm 1.

Remark 3. In practice, the system of algebraic equations (13) is divided into subproblems by also using polynomials other than $\psi_1, \dots, \psi_p \in \mathbb{R}[\lambda]$ because the system of algebraic equations (13) has a large number of solutions in general. In addition, whether \mathcal{I}_b is zero-dimensional or not, it can be determined by statement 2 of Proposition 3.2. Since we divide (13) into a large number of sub-problems, even if \mathcal{I}_b is non-zero-dimensional, \tilde{G}_b has a simple representation. If \mathcal{I}_b is zero-dimensional, the Gröbner basis G_b gives us an easily solvable sub-problem, as described in Example 5 in Section 3.1.3 and statement 3 of Proposition 3.2. For details of the decomposition other than $\psi_1, \dots, \psi_p \in \mathbb{R}[\lambda]$, please refer to Appendix B.

Algorithm 1 An algorithm to find all solutions.

Require: Sample variance-covariance matrix $S = (s_{ij})$

Ensure: All solutions to the system of algebraic equations (6) and (9) (S01.)

```

1:  $\mathcal{J} \leftarrow$  the ideal  $\langle \{S - \Lambda\Lambda^\top - \text{diag}(S - \Lambda\Lambda^\top)\} S^{-1} \Lambda \rangle_{ij} : 1 \leq i \leq p, 1 \leq j \leq k \rangle$ 
2:  $h_i \leftarrow$  the polynomial  $s_{ii} - \sum_{j=1}^k \lambda_{ij}^2$  for  $1 \leq i \leq p$ ,  $\mathcal{K}_i \leftarrow$  the ideal  $\langle h_i \rangle$  for  $1 \leq i \leq p$ 
3:  $\mathcal{I}_0 \leftarrow$  the sum  $\mathcal{J} + \mathcal{K}_1$ ,  $\mathcal{I}_1 \leftarrow$  the saturation  $\mathcal{J} : \mathcal{K}_1^\infty$ 
4: for  $1 \leq i \leq p-1$  do
5:   for  $b \in \{0,1\}^{i-1}$  do
6:      $\mathcal{I}_{b0} \leftarrow$  the sum  $\mathcal{I}_b + \mathcal{K}_{i+1}$ ,  $\mathcal{I}_{b1} \leftarrow$  the saturation  $\mathcal{I}_b : \mathcal{K}_{i+1}^\infty$ 
7:   end for
8: end for
9: for  $b \in \{0,1\}^p$  do
10:   $\tilde{G}_b = \{\tilde{g}_1, \dots, \tilde{g}_s\} \leftarrow$  a Gröbner basis of  $\mathcal{I}_b$  with respect to the grevlex order
11:  if  $\mathcal{I}_b$  is zero-dimensional then
12:     $\{g_1, \dots, g_r\} \leftarrow$  a Gröbner basis of  $\mathcal{I}_b$  with respect to the lex order by converting  $\tilde{G}_b$ 
13:    S01.  $\leftarrow$  S01.  $\cup \{(\Lambda, \text{diag}(S - \Lambda\Lambda^\top)) \in \mathbb{R}^{p \times k} \times \mathbb{R}^{p \times p} : g_1 = \dots = g_r = 0\}$ 
14:  else
15:    S01.  $\leftarrow$  S01.  $\cup \left\{ \begin{array}{l} (\Lambda, \text{diag}(S - \Lambda\Lambda^\top)) \in \mathbb{R}^{p \times k} \times \mathbb{R}^{p \times p} : \\ \lambda \text{ is a sample point from each connected component} \\ \text{defined by } \tilde{g}_1 = \dots = \tilde{g}_s = 0 \end{array} \right\}$ 
16:  end if
17: end for
```

Remark 4. The ideals \mathcal{I}_b may have multiple roots. Therefore, in practice, Step 10 computes a Gröbner basis of the radical ideals for \mathcal{I}_b . The radical ideal $\sqrt{\mathcal{I}_b}$ is defined by

$$\sqrt{\mathcal{I}_b} = \{p : \text{there exists } t \in \mathbb{Z}_{\geq 0} \text{ such that } p^t \in \mathcal{I}_b\}.$$

In general, radical ideals have only solutions with multiplicity 1 (i.e., simple roots). So our algebraic approach can treat very simple sub-problems, and does not generate multiple solutions.

3.3. An algorithm to identify a solution pattern

We provide an algorithm to categorize the MLEs into three patterns: “Proper solution,” “Improper solution,” and “No solutions.” The algorithm is detailed in Algorithm 2. First, we find all solutions to Eq. (4) with Algorithm 1. Then, we get a solution $(\hat{\Lambda}_t, \hat{\Psi}_t)$ that minimizes the discrepancy function in (2) as in Step 1 of Algorithm 2. When the Hessian matrix H_q at the solution $(\hat{\Lambda}_t, \hat{\Psi}_t)$ is positive definite, there exists an MLE that minimizes (2), where the Hessian matrix H_q is defined by the following:

$$H_q = \begin{pmatrix} \frac{\partial^2 q}{\partial \lambda_{11} \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{11} \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{11} \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{11} \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \lambda_{11} \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \lambda_{11} \partial \psi_p} \\ \vdots & \cdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 q}{\partial \lambda_{1m} \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{1m} \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{1m} \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{1m} \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \lambda_{1m} \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \lambda_{1m} \partial \psi_p} \\ \vdots & \cdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 q}{\partial \lambda_{p1} \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{p1} \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{p1} \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{p1} \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \lambda_{p1} \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \lambda_{p1} \partial \psi_p} \\ \vdots & \cdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 q}{\partial \lambda_{pm} \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{pm} \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{pm} \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \lambda_{pm} \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \lambda_{pm} \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \lambda_{pm} \partial \psi_p} \\ \frac{\partial^2 q}{\partial \psi_1 \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \psi_1 \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \psi_1 \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \psi_1 \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \psi_1 \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \psi_1 \partial \psi_p} \\ \vdots & \cdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 q}{\partial \psi_p \partial \lambda_{11}} & \cdots & \frac{\partial^2 q}{\partial \psi_p \partial \lambda_{1m}} & \cdots & \frac{\partial^2 q}{\partial \psi_p \partial \lambda_{p1}} & \cdots & \frac{\partial^2 q}{\partial \psi_p \partial \lambda_{pm}} & \frac{\partial^2 q}{\partial \psi_p \partial \psi_1} & \cdots & \frac{\partial^2 q}{\partial \psi_p \partial \psi_p} \end{pmatrix}.$$

We are able to test whether H_q at the solution $(\hat{\Lambda}_t, \hat{\Psi}_t)$ is positive definite or not by using its observed Fisher information matrix that is defined as $H_q/2$. So Step 2 of Algorithm 2 uses its observed Fisher information matrix for testing whether H_q at the solution $(\hat{\Lambda}_t, \hat{\Psi}_t)$ is positive definite or not. If it is positive definite, the estimate is categorized as either proper or improper according to the sign of the unique variances, as in Step 4 or 6 of Algorithm 2. When all unique variances are positive, the estimate is referred to as a “proper solution,” as in Step 4 of Algorithm 2. Meanwhile, when at least one of the unique variances is not positive, the estimate is referred to as an “improper solution,” as in Step 6 of Algorithm 2. When the Hessian matrix at the solution is not positive definite, there are no optimal MLEs. In this case, the estimate is referred to as a “no solutions,” as in Step 9 of Algorithm 2. With Algorithm 2, we can investigate the tendency of the solution pattern through Monte Carlo simulations, which will be described in the following section.

Algorithm 2 An algorithm to identify the solution pattern.

Require: All solutions to (13), say $(\hat{\Lambda}_t, \hat{\Psi}_t)_{t=1, \dots, T}$.

Ensure: Solution pattern (Sol . Pat .)

- 1: Select a solution that minimizes the discrepancy function in Eq. (2); that is, obtain $(\hat{\Lambda}_t, \hat{\Psi}_t)$, where $t^* = \arg \min_{t \in \{1, \dots, T\}} \lambda(\hat{\Lambda}_t, \hat{\Psi}_t)$.
 - 2: **if** the solution $(\hat{\Lambda}_t, \hat{\Psi}_t)$ satisfies the condition (4) and H_q is positive-definite **then**
 - 3: **if** the minimum value of diagonal element of $\hat{\Psi}_t$ is positive **then**
 - 4: Sol . Pat . \leftarrow “Proper solution.”
 - 5: **else**
 - 6: Sol . Pat . \leftarrow “Improper solution.”
 - 7: **end if**
 - 8: **else**
 - 9: Sol . Pat . \leftarrow “No solutions.”
 - 10: **end if**
-

4. Monte Carlo simulation

We conduct Monte Carlo simulations to investigate the characteristics of solution patterns of MLEs in various situations. First, we consider the following three simulation models:

$$S1: \Lambda = \begin{pmatrix} 0.9 & 0 \\ 0.8 & 0 \\ 0.7 & 0 \\ 0 & 0.8 \\ 0 & 0.7 \end{pmatrix}, \quad S2: \Lambda = \begin{pmatrix} 0.5 & 0 \\ 0.5 & 0 \\ 0.5 & 0.5 \\ 0 & 0.5 \\ 0 & 0.5 \end{pmatrix}, \quad S3: \Lambda = \begin{pmatrix} 0.9 & 0 \\ 0.8 & 0 \\ 0.6 & 0.7 \\ 0 & 0.8 \\ 0 & 0.9 \end{pmatrix},$$

and $\Psi = \text{diag}(I - \Lambda\Lambda^T)$. On S1, the second column of Λ has only two nonzero elements, suggesting that a necessary condition for identification is not satisfied (Anderson & Rubin, 1956, Theorem 5.6). In this case, the estimate can be unstable due to the identifiability problem (van Driel, 1978), and improper solutions are often obtained. S2 has small communalities and large unique variances, and then improper solutions sometimes occur due to the sampling fluctuation (van Driel, 1978). In S3, communalities are large and a necessary condition on identification is satisfied; thus, it is expected that the numerical estimate is stable and improper solutions are unlikely to be obtained.

For all simulation models, we generate $N = 100$ observations from $N(0, \Lambda\Lambda^T + \Psi)$, and compute the sample covariance matrix. The elements of the sample covariance matrix are rounded to one decimal place. The most fundamental reason for this is that numbers with many decimal places are generally expressed as complicated rational numbers. Many implementations for computations of Gröbner bases deal with polynomials with rational coefficients, rather than with coefficients with decimal places. Such computations with complicated rational coefficients in computational algebra can require large amounts of computer memory as the coefficients become big due to the “four arithmetic operations of coefficient” performed internally. For that reason, the elements of the sample covariance matrix are rounded to one decimal place. As described in Appendix C, we have also confirmed that rounding the values of the sample elements from two digits to one digit does not significantly affect the accuracy in most cases. Furthermore, we have confirmed that the results with two-digit values and those with more than two-digit values are almost identical.

For each simulation model, we run ten times and investigate the characteristics of estimates obtained by computational algebra. We note that such a small number of replications of the Monte Carlo simulation is due to the heavy computational loads of computational algebra. For implementation, we used the highest-performance computer that we had (Intel Xeon Gold 6134 processors, 3.20 GHz, 32 CPUs, 256 GB memory, and Ubuntu 18.04.2 LTS), and made efforts to optimize the algorithm of computational algebra as much as possible, as detailed in Appendix B. However, the computational timing varied from 2 weeks to 4 weeks to get the result for a single dataset.

For comparison, we also perform `factanal` function in R. We note that one may use the algorithm of Jennrich and Robinson (1969), which allows for negative values. However, in our experience, the algorithm often diverges and is highly unstable when improper solutions are observed. In contrast, the `factanal` function provides stable estimates and is also easy to implement. Therefore, we decided to use the `factanal` function.

4.1. Investigation of MLEs

We investigate the MLEs obtained by computational algebra and `factanal`. Table 1 shows the solution pattern of the MLE computed using Algorithm 2. For S3, we obtain proper solutions in all ten simulations, implying that the results for S3 are stabler than S1 and S2. Meanwhile, several improper solutions are obtained, or no solutions are found for S1 and S2. In particular, S1 tends to produce “no solutions” more frequently than S2.

With `factanal` function, however, it would be difficult to identify the solution pattern as in Table 1 because the `factanal` provides estimates under a restriction that all unique variances are

Table 1. Solution pattern obtained by Algorithm 2: proper solution (P), improper solution (I), and no solution (NA)

Sim no.	1	2	3	4	5	6	7	8	9	10
S1	P	NA	NA	NA	NA	I	I	P	P	P
S2	P	P	I	NA	P	I	I	P	P	P
S3	P	P	P	P	P	P	P	P	P	P

larger than some threshold, such as 0.005. Moreover, the observed Fisher information matrices of the MLEs computed with `factanal` function are positive definite for all simulated datasets; therefore, it is remarkably challenging to distinguish “improper solution” and “no solutions” using `factanal` function.

To study the solution pattern in more detail, we investigate the number of solutions obtained by computational algebra, as shown in Table 2. When counting the number of solutions, the identifiability with respect to the sign of column vectors in a loading matrix is addressed. We categorize the algebraic solutions based on the signs of the unique variances:

- For (a)–(b), all unique variances are positive.
- For (c)–(d), all unique variances are non-zero but some are negative.
- For (e)–(h), some unique variances are zero.

When some unique variances are zero, the solution does not always satisfy the original normal equation in (4). Therefore, we check whether the solution satisfies (4) in such cases numerically. Meanwhile, when all unique variances are non-zero, the solutions to (6) and (9) are equivalent to those of (4). For this reason, we do not verify (4) for cases where all unique variances are non-zero.

Moreover, interestingly, in our simulation, a very complicated solution space is computed when all unique variances are zero. In such cases, it is not possible to evaluate the value of the log-likelihood function. Therefore, we do not analyze these cases in this simulation.

For all cases, the number of solutions that satisfy the conditions (6) and (9) is relatively large, but most of them do not have a positive-definite observed Fisher information matrix. Thus, it would be essential to check if the observed Fisher information matrix is positive definite or not. Another noteworthy point is that although we get a number of solutions with zero unique variances using (6) and (9), about 98.5 % of them are not the solution of (4) (see (e) and (f)), implying that most of the solutions to (6) and (9) with zero unique variances are inappropriate. Furthermore, in some simulations, such solution spaces consist of a finite number of points and subspaces containing an infinite number of points. We note that, for subspaces consisting of an infinite number of points, (e) and (f) in Table 2 show the number of sample points from the connected components computed by our algorithm.

Here, we denote that the j th dataset on simulation model S_i is S_{i-j} ($i = 1, 2, 3; j = 1, \dots, 10$). Further empirical comparison for each case in Table 2 is presented below.

4.1.1. Optimal solution does not exist: S_{1-2} , S_{1-3} , S_{1-4} , S_{1-5} , and S_{2-4}

We first investigate the detailed results in which there are no local maxima. The result (e) in Table 2 suggests that there exist a number of improper solutions with zero unique variances obtained by (6) and (9), and some of them provide a positive-definite observed Fisher information matrix (see the result (f)). However, all of these solutions do not satisfy (4) from the result (g). Interestingly, the estimate computed using `factanal` function is close to one of the solutions of (6) and (9) (i.e., the result (f)), suggesting that the `factanal` can converge to an improper solution when the optimal solution does not exist. With the `factanal` function, the convergence value of the loading matrix constructed by (5) does not always satisfy (4) when Ψ is singular; thus, the mixture of update equation (5) and Quasi-Newton

Table 2. The number of solutions obtained by computational algebra

Sim no.	1	2	3	4	5	6	7	8	9	10
(a) All solutions to (6) and (9) whose unique variances are all positive.										
S1	3	1	2	3	3	3	0	3	1	1
S2	8	2	3	5	3	0	2	2	3	4
S3	1	1	3	2	1	1	2	1	2	1
(b) Solutions of (a) whose observed Fisher information matrix is positive definite.										
S1	1	0	0	0	0	0	0	1	1	1
S2	1	1	0	0	1	0	0	1	1	1
S3	1	1	1	1	1	1	1	1	1	1
(c) All solutions to (6) and (9) where all unique variances are non-zero and some of them are negative.										
S1	2	7	4	2	3	0	10	0	3	0
S2	4	3	2	5	3	4	2	2	3	2
S3	0	6	1	1	2	4	3	7	8	5
(d) Solutions of (c) whose observed Fisher information matrix is positive definite.										
S1	0	0	0	0	0	0	1	0	0	0
S2	0	0	2	0	0	1	1	0	0	0
S3	0	0	0	0	0	0	0	0	0	0
(e) All solutions to (6) and (9) where some of the unique variances are zero.										
S1	31	28	32	30	30	22	32	28	24	26
S2	31	22	26	28	27	23	24	27	26	25
S3	22	22	23	24	23	24	27	23	28	25
(f) Solutions of (e) whose observed Fisher information matrix is positive definite.										
S1	1	5	4	1	3	5	1	2	2	0
S2	2	3	6	2	0	3	1	1	1	3
S3	1	4	3	1	1	1	2	2	2	3
(g) Solutions of (e) that satisfy (4).										
S1	0	0	0	0	0	1	1	0	0	4
S2	0	0	0	0	0	0	1	0	0	0
S3	4	0	0	0	0	0	1	0	0	0
(h) Solutions of (f) that satisfy (4).										
S1	0	0	0	0	0	1	0	0	0	0
S2	0	0	0	0	0	0	0	0	0	0
S3	0	0	0	0	0	0	0	0	0	0

Note: When counting the number of solutions, the identifiability with respect to the sign of column vectors in a loading matrix is addressed.

method with respect to unique variances can lead to the improper solution when the optimal solution does not exist.

The results also give insight into the conventional algorithm of Jennrich & Robinson (1969). For all simulated datasets, Jennrich and Robinson’s (1969) algorithm tends to highly depend on initial values,

as demonstrated in Figure 1. Furthermore, one of the unique variances results in an exceptionally large negative value, even if the initial value is close to the estimate computed with the `factanal` function. We numerically observe that the estimate obtained by Jennrich and Robinson's algorithm (1969) satisfies (4) but does not satisfy (5). Recall that `factanal` function uses (5); therefore, `factanal` function and Jennrich and Robinson's (1969) algorithm result in different estimates. Furthermore, the observed Fisher information matrix computed with Jennrich and Robinson's algorithm (1969) is not positive definite; one of the eigenvalues of the observed Fisher information matrix is nearly zero, suggesting that the confidence interval is unavailable. These results are often observed due to the identification problem (van Driel, 1978; Kano, 1998). Indeed, the true loading matrix of S1-2, S1-3, S1-4, and S1-5 is not unique even if rotational indeterminacy is taken out, because theorem 5.6 in Anderson & Rubin (1956) is not satisfied.

4.1.2. Improper solutions: S1-6, S1-7, S2-3, S2-6, and S2-7

We get five improper solutions whose unique variances are negative or zero. These negative estimates are obtained when the proper solutions do not exist. The number of negative estimates is relatively large for each model from the result (c) of Table 2, but most of them do not have a positive-definite observed Fisher information matrix (see result (d)). As a result, the negative estimates are almost uniquely determined. An exception is S2-3; there are two solutions that are completely different but have similar likelihood values.

We also find that when the optimal values of unique variances are strictly negative, there often exists a solution to (6) and (9) whose corresponding unique variances are exactly zero and other parameters have similar values to the optimal value. Furthermore, that solution is quite similar to that obtained by the `factanal` function. The corresponding Fisher information matrix is positive-definite.

4.1.3. Proper solutions: S1-1, S1-8, S1-9, S1-10, S2-1, S2-2, S2-8, S2-9, S2-10, and S3

We briefly investigate the results where all unique variances are positive. For all models, we obtain several proper solutions (result (a) in Table 2) and only one proper solution has a positive-definite observed Fisher information matrix (result (b)). Furthermore, we obtain no improper solutions whose observed Fisher information matrix is positive (results (d) and (h)), suggesting that the likelihood function has only one local maximum that is proper.

Remark 5. As shown in Section 3.2, our approach computes all candidates for the MLE. If a candidate is a root of a non-zero-dimensional ideal, the candidate may not provide an identifiable model. On the other hand, if the candidate is the root of a zero-dimensional ideal, that candidate does not suffer from identification problems. In our experiments, we often obtain candidates that are roots of a non-zero-dimensional ideal (see Appendix B for details). Fortunately, however, those candidates do not reach the maximum value of the likelihood.

4.2. Investigation of the best estimate obtained by computational algebra and `factanal`

We further compare the characteristics of the loading matrix obtained by computational algebra and `factanal`. Figure 3 shows the heatmaps of the estimated factor loadings along with the true ones. The loading matrices of computational algebra are unavailable when the optimal solution does not exist (i.e., S1-2, S1-3, S1-4, S1-5, and S2-4).

We first investigate whether the estimates can approximate the true loading matrix. For S1, the results for the heatmap of the loading matrix in Figure 3 suggest that both computational algebra and `factanal` function can well approximate the true loading matrix. In particular, even if no solution is found with the computational algebra, the `factanal` function can approximate the true loadings. This result is surprising to us because the true loading matrix has an identifiability issue that has been considered as a severe problem (Kano, 1998). In contrast to our expectation, the estimates of S2 are

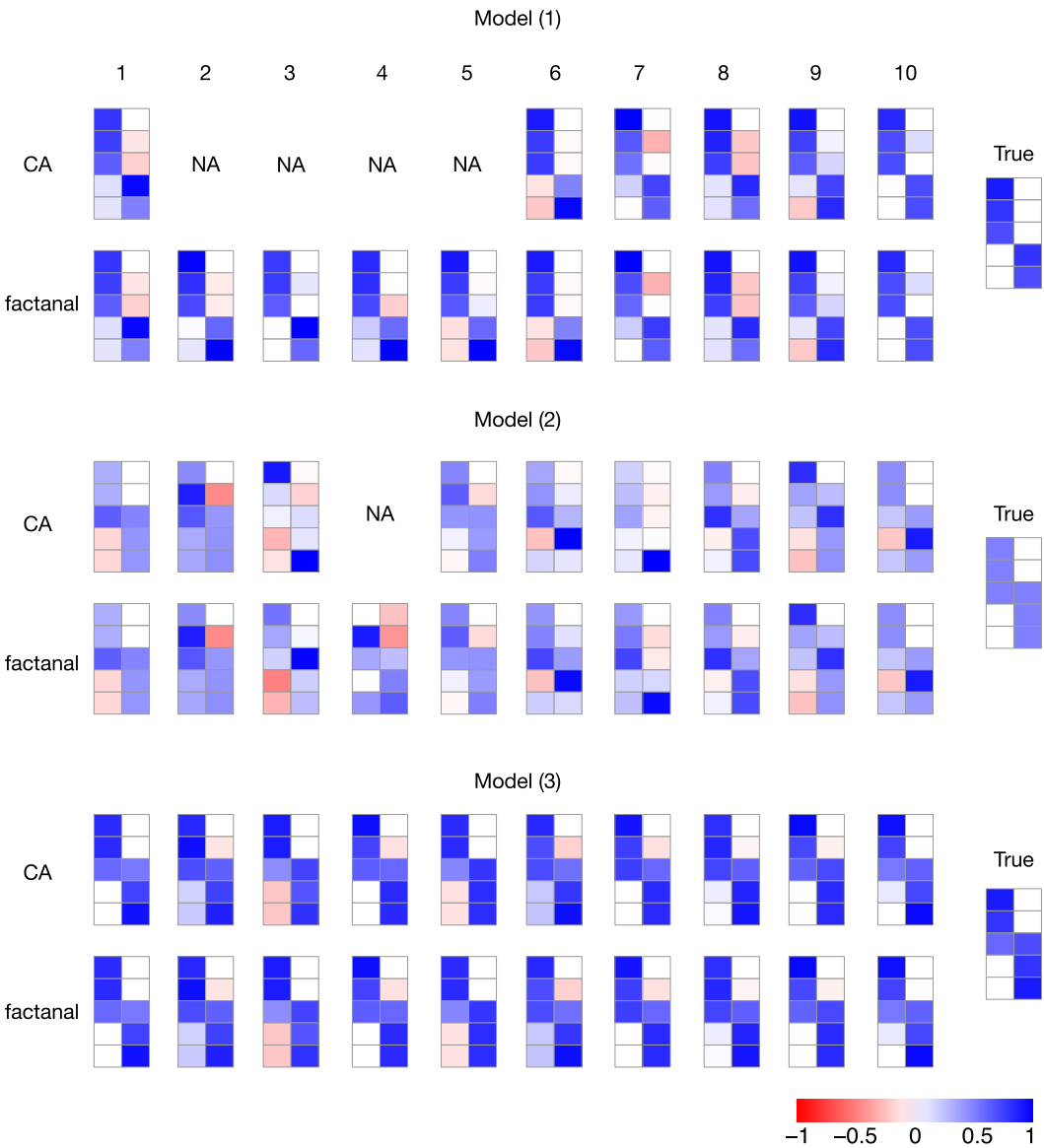


Figure 3. Heatmap of the factor loadings obtained by computational algebra and `factanal` function.

much more unstable than those of S1. Thus, the small communalities rather than the identification problem would make the estimation difficult in our simulation settings. For S3, we obtain proper and stable estimates for all settings.

We compare the computational algebra's estimates and `factanal`'s ones. In many cases, these estimates are essentially identical. In some cases where the computational algebra provides negative unique variances (i.e., S1-7, S2-3, S2-6, and S2-7), we find more or less difference between the two methods. This is because the computational algebra provides negative unique variances while the `factanal` provides 0.005 due to the restriction of the parameter space. However, the overall interpretation of these two estimates is essentially identical for S1-7, S2-6, and S2-7. An exception is S2-3; in this case, the results are entirely different from each other. Such a difference may be caused by the fact that the `factanal` function uses only one initial value. We prepare 100 initial values from

Table 3. Solution pattern obtained by Algorithm 3 over 100 simulation runs

	Proper solution	Improper solution	No solutions
S1	30	36	34
S2	66	16	18
S3	100	0	0

$U(0, 1)$, perform the `factanal` function, and select a solution that maximizes the likelihood function; as a result, the `factanal`'s estimate is essentially identical to computational algebra's one. The result suggests that performing `factanal` function with multiple initial values would be essential when we obtain the improper solutions.

4.3. A numerical algorithm to identify the solution pattern

Because the computational algebra remains a challenge in heavy computation, it would be difficult to apply it to a large model. Still, our empirical result is helpful for constructing a method to determine whether the estimate exists or not with a numerical approach. Specifically, we obtain the estimate by Jennrich and Robinson's (1969) algorithm with multiple initial values. Subsequently, we employ the same procedure as Algorithm 2 based on the numerical approach. The algorithm for identifying the solution pattern with the numerical algorithm is detailed in Algorithm 3.

Algorithm 3 A numerical algorithm to identify the cause of improper solutions.

Require: Dataset $\{x_1, \dots, x_n\}$ or its sample covariance matrix S , and the number of iterations T to get the maximum likelihood estimate.

Ensure: Solution pattern.

- 1: **for** $t = 1$ to T **do**
- 2: Obtain a maximum likelihood estimate with Jennrich and Robinson's (1969) algorithm in (10) using t -th initial value.
- 3: **end for**
- 4: Same as lines 1–10 in Algorithm 2.

With Algorithm 3, it is possible to further investigate the tendency of solution pattern with a number of simulation iterations. Table 3 shows the distribution of solution pattern with Algorithm 3 over 100 simulation runs. The results indicate that only 30% of S1 provides proper solutions, while S2 has more than double the percentage of proper solutions compared to S1. Thus, S1 tends to provide more improper solutions than S2. Interestingly, we get several "improper solutions" and "no solutions" for both S1 and S2, suggesting that various solution patterns are obtained, whether they arise from identifiability issue (S1) or sampling fluctuation (S2). In other words, it would be difficult to identify the cause of improper solutions by using the solution pattern. However, the distribution of the solution pattern appears to differ among S1–S3. Therefore, it would be essential to find the distribution of solution pattern with a number of simulations (e.g., bootstrap method) for identifying the cause of improper solutions.

5. Actual data analysis

This section consists of two sections. In Section 5.1, we apply our algebraic approach to one of the famous datasets (Davis, 1944) that produces improper solutions. However, as the dataset contains many observed variables, it is almost impossible to conduct algebraic computations due to computational limitations. Therefore, we first select a subset of variables and fit a two-factor model. In Section 5.2, we apply a numerical approach based on Algorithm 3, which can be used for medium- to large-scale

Table 4. Solution pattern obtained by Algorithm 2: proper solution (P), improper solution (I), and no solution (NA)

Sim	$S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$	$S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$	$S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$
	P	I	NA

data, to five well-known datasets that yield improper solutions. We remark that the aim of the real data analysis in this research is not to find the best-fitting model for each dataset, but rather to gain a deeper understanding of the characteristics of solution patterns in maximum likelihood estimation.

5.1. Algebraic approach

In this section, we analyze the actual data studied in Davis (1944). The data have the following sample correlation matrix:

$$S = \begin{pmatrix} 1 & 0.72 & 0.41 & 0.28 & 0.52 & 0.71 & 0.68 & 0.51 & 0.68 \\ & 1 & 0.34 & 0.36 & 0.53 & 0.71 & 0.68 & 0.52 & 0.68 \\ & & 1 & 0.16 & 0.34 & 0.43 & 0.42 & 0.28 & 0.41 \\ & & & 1 & 0.30 & 0.36 & 0.35 & 0.29 & 0.36 \\ & & & & 1 & 0.64 & 0.55 & 0.45 & 0.55 \\ & & & & & 1 & 0.76 & 0.57 & 0.76 \\ & & & & & & 1 & 0.59 & 0.68 \\ & & & & & & & 1 & 0.58 \\ & & & & & & & & 1 \end{pmatrix}.$$

As reported in Jöreskog (1967), by applying the maximum likelihood factor analysis model with two common factors to this data, we obtain an improper solution. It is not easy for Algorithm 1 to handle data with nine observed variables and two common factors from a realistic computation time/memory point of view. Therefore, we select five observed variables out of nine, and perform analysis based on these selected variables. In other words, we select five rows and five columns from the sample correlation matrix above. Note that such a variable selection procedure would not typically be performed by an analyst in practice. We reduce the number of variables only to enable the algebraic approach within a realistic computation time and to investigate whether the resulting solution patterns are consistent with those observed in the simulation study in Section 4.

By applying a variable selection algorithm with group lasso (Hirose & Konishi, 2012) to the sample correlation matrix above, rows/columns 1, 2, 6, 7, 9 are selected. For this input, that is, for the sample correlation sub-matrix, say $S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$, the `factanal` function in R outputs a proper solution. Changing the third row and column from 6 to 3, we obtain the sample correlation sub-matrix $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$. For $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$, the `factanal` function produces an improper solution. Moreover, we select rows/columns 1, 2, 3, 4, 5. For $S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$, the `factanal` function provides an improper solution as well. In this section, we analyze the sample correlation submatrixes $S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$, $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$ and $S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$ by using Algorithm 2.

The MLEs are computed based on Algorithm 1 from the sub-matrices expressed as real numbers to the second decimal place. The computation time for the case of the second decimal place was about several months. Table 4 shows the solution patterns obtained by Algorithm 2 for $S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$, $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$, and $S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$.

For the selected subsets of rows and columns, a proper solution (P) is obtained for $\{1,2,6,7,9\}$, an improper solution (I) for $\{1,2,3,7,9\}$, and no solution (NA) for $\{1,2,3,4,5\}$. As in the investigation in Section 4.1, proper solutions, improper solutions, and no solutions are obtained from algebraic computations for the actual data.

Table 5. The number of solutions obtained by computational algebra

Sim	$S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$	$S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$	$S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$
(a) All solutions to (6) and (9) whose unique variances are all positive.	4	2	3
(b) Solutions of (a) whose observed Fisher information matrix is positive definite.	1	0	0
(c) All solutions to (6) and (9) where all unique variances are non-zero and some of them are negative.	0	1	0
(d) Solutions of (c) whose observed Fisher information matrix is positive definite.	0	1	0
(e) All solutions to (6) and (9) where some of the unique variances are zero.	15	15	16
(f) Solutions of (e) whose observed Fisher information matrix is positive definite.	2	2	2
(g) Solutions of (e) satisfying (4).	0	0	1
(h) Solutions of (f) satisfying (4).	0	0	0

Note: When counting the number of solutions, the identifiability with respect to the sign of column vectors in a loading matrix is addressed.

We also investigate the performance of the numerical algorithms. Specifically, we apply `factanal` and the method of Jennrich & Robinson (1969) to $S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$, $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$, and $S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$. Several important findings are summarized as follows:

- For $S_{\{1,2,6,7,9\}\{1,2,6,7,9\}}$, `factanal` and computational algebra provide almost identical results with proper solutions.
- For $S_{\{1,2,3,7,9\}\{1,2,3,7,9\}}$, `factanal` and computational algebra yield similar estimates for most elements of the loading matrix. An exception is the estimate of λ_{22} due to the negative unique variance ($\hat{\psi}_2 = -0.212$) with computational algebra, compared to $\hat{\psi}_2 = 0.005$ with `factanal`.
- For $S_{\{1,2,3,4,5\}\{1,2,3,4,5\}}$, the algorithm based on Jennrich & Robinson (1969) produces a very large negative value of unique variance ($\hat{\psi}_1 = -11642.5$).

These results indicate that the numerical findings are largely consistent with those obtained with Monte Carlo simulation.

We perform Algorithm 3 to numerically investigate the solution pattern for such findings. The results are identical to those obtained by algebraic computation shown in Table 4. As Algorithm 3 completes the analysis in just seconds with a standard laptop, the algorithm would be useful when sufficient computational resources are unavailable. Furthermore, Algorithm 3 can be applied to larger models. We apply Algorithm 3 to the original covariance matrix, S ; in this case, no optimal solution is obtained.

Table 5 shows the number of solutions to (6) and (9), which closely resembles the results in Table 2 from the Monte Carlo simulation in Section 4.1. When a proper solution is obtained, we get a unique stationary point of the log-likelihood function, with a positive definite observed Fisher information matrix and positive unique variances. When an improper solution is obtained, a stationary point of the log-likelihood function is obtained, with a positive definite observed Fisher information matrix and some negative unique variances. In addition, it can be seen that “NA” does not have these stationary points. Interestingly, most of the solutions to (6) and (9) with zero unique variances do not satisfy (4), as is the case with Monte Carlo simulations in Section 4. Thus, it is confirmed that the results of the

Table 6. Distribution of solution patterns for five datasets

Dataset	(i)	(ii)			(iii)		
	Solution pattern	P	I	NA	P	I	NA
Davis	NA	55	15	30	24	50	26
Harman	NA	42	20	38	19	17	64
Maxwell	NA	38	46	16	36	21	43
Thurstone	NA	39	49	12	28	43	29
Kendall	I	33	51	16	11	28	61

Note: (i) Original data; (ii) bootstrap (estimated parameters); (iii) bootstrap (modified loading matrices). P: proper solution, I: improper solution, and NA: no solution.

actual data analysis in Table 5 show patterns similar to those of the Monte Carlo simulation presented in Table 2.

The overall analysis of the actual data provides results similar to those from the Monte Carlo simulations. Further investigation would involve resampling methods, such as bootstrap, but this is beyond the scope of our research. We would like to take this as a future research topic.

5.2. Numerical computation

The previous section uses a low-dimensional data to apply algebraic computation. In this section, we apply Algorithm 3, a numerical algorithm to investigate the behavior of improper solutions, to five commonly-used benchmark datasets. It is known that each dataset exhibits improper solutions under certain conditions: the Davis data in the previous example ($n = 421$ and $p = 9$, improper at $m = 2$) (Davis, 1944); Harman’s 24 mental tests ($n = 145$ and $p = 24$, improper at $m = 6$) (Harman, 1976); Maxwell’s normal children dataset ($n = 810$ and $p = 10$, improper at $m = 4$) (Maxwell, 1961); the Bechtoldt dataset based on the first half-sample of Thurstone and Thurstone’s (1941) 17-variable test battery ($n = 212$ and $p = 17$, improper at $m = 6$) (Bechtoldt, 1961); and Kendall’s job applicant ratings dataset ($n = 48$ and $p = 15$, improper at $m = 4$) (Kendall, 1975). For each dataset, we conduct the following three analyses:

- (i) We apply Algorithm 3 to the original data or its correlation matrix to investigate the solution patterns of improper solutions.
- (ii) We investigate the distribution of solution patterns by simply applying the parametric bootstrap with $B = 100$ resamples. The MLEs with varimax rotation of the loading matrix and unique variances are treated as the true values, and datasets with the same sample sizes are generated from a multivariate normal distribution using these parameters.
- (iii) Same as (ii), but with a different loading matrix. We select the column vector with the smallest sum of squares and modify it so that the two elements with the largest absolute values are retained and all others are set to zero. As a result, the column vector with the smallest sum of squares has only two nonzero entries. Therefore, the loading matrix is not identifiable because the condition of theorem 5.6 in Anderson & Rubin (1956) is not satisfied. With this modified loading matrix, the unique variances are computed such that the diagonal elements of the covariance matrix of the observed variables are equal to one; that is, $\Psi = \text{diag}(I - LL^T)$. The bootstrap is then performed using this loading matrix and the corresponding unique variances.

Table 6 shows the results for the above-mentioned three analyses. When we apply Algorithm 3 to the original data, the first four datasets yield “no solutions,” while Kendall’s dataset exhibits an improper solution with a negative unique variance.

Next, in (ii), we employ the parametric bootstrap. Based on the results from the original datasets, we expect a large number of cases with no solutions for the first four datasets, while improper solutions are expected for Kendall's dataset. However, we actually observe that the first two datasets (Davis and Harman) yield mostly proper solutions, and the next two (Maxwell and Thurstone) produce mostly improper solutions. For Kendall's dataset, we observe many improper solutions, but the proportion is still only 51%. Therefore, the distribution is quite dispersed rather than concentrated, and does not always coincide with the solution pattern of the original dataset.

In (iii), the bootstrap is conducted using loading matrices that are not identifiable. Therefore, we expect most of the results for the solution patterns to be no solutions. However, even when the true loading matrix is not identifiable, we often observe negative unique variances (especially with the Davis data), and sometimes observe positive unique variances as well.

These bootstrap results are consistent with Table 1 from the simulation study. The results suggest that it would be difficult to infer the true loading matrix based solely on the observed solution patterns. One possible reason is that the loss (discrepancy) function is highly complex. Because of this complexity, even when the true loading matrix suffers from identifiability issues, it is still possible to obtain either a proper or improper solution. As a result, the boundaries between proper solutions, improper solutions, and no solutions are ambiguous, and the outcome can easily change depending on sampling variability. While this kind of consideration may have been discussed at an intuitive level, our algebraic approach (Algorithm 2) and its numerical mimic (Algorithm 3) provide further formal support.

6. Conclusion and future perspectives

We have proposed an algorithm to get the exact MLEs via computational algebra. The method is based on the system of algebraic equations in Jennrich and Robinson's (1969) algorithm. We obtain all candidates based on Eq. (13), and select a solution that minimizes the discrepancy function (i.e., exact estimate). If the estimate's observed Fisher information matrix is positive-definite, that estimate is optimal; otherwise, no estimate is obtained. As a result, we can identify the solution pattern of the MLE with computational algebra. The conventional analysis may not identify the solution pattern; thus, our proposed approach represents a significant advancement in studying improper solutions in maximum likelihood factor analysis.

This study compared the performance of the computational algebra and `factanal` function. Unlike the `factanal` function, the computational algebraic method itself offers practical advantages. First, it rigorously detects divergence or improper solutions, ensuring reliability in challenging scenarios where `factanal` may produce improper solutions. Second, it does not rely on random initial values, providing a unique result without requiring multiple runs. While the empirical effectiveness of `factanal` is notable, the computational algebraic method can offer greater precision and reliability, particularly in applications requiring rigorous validation.

This study also highlights the strengths and limitations of the `factanal` function. While the `factanal` function performs well in practice, it imposes a constraint that unique variances are greater than a threshold, such as 0.005. This constraint ensures stability but may limit the investigation of improper solutions. By comparing the `factanal` results with solutions obtained via computational algebra, we rigorously confirm its empirical effectiveness. These findings provide strong evidence that `factanal` can be used with confidence in many practical scenarios.

As shown in Table 2, there are multiple local minima, making the `factanal` function highly dependent on initial values. As `factanal` uses a single initial value by default, it can fail to converge to the global solution. Therefore, we suggest preparing a sufficiently large set of initial values and selecting one that minimizes the discrepancy function. This suggestion is supported by the findings from our extensive analysis using computational algebra.

The current approach is computationally demanding due to the complexity of solving systems of multivariate algebraic equations. To enhance the computational speed, we proposed an algorithm that

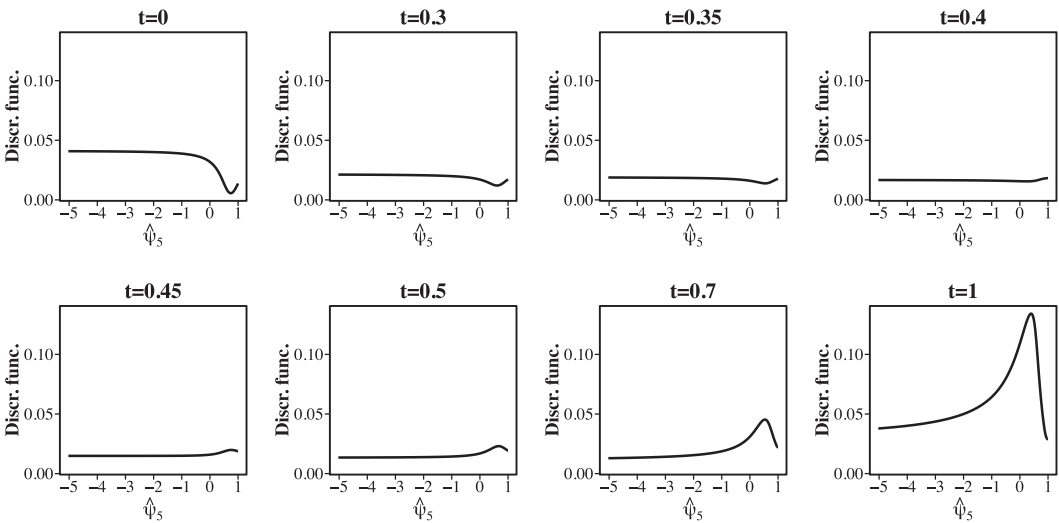


Figure 4. Discrepancy function in (2) as a function of $\hat{\psi}_5$.

incorporated advanced computational algebra techniques, such as sum and saturation, as detailed in Appendix B. We also used a fast computer algebra system “Magma.” After these extensive efforts, we successfully conducted the Monte Carlo simulations and actual data analysis presented in Sections 4 and 5.

While these efforts allowed us to achieve our current results, further improvements in computational speed are essential for making this approach practically applicable. We believe that advancements in computer technologies, such as parallel processing, GPU acceleration, and quantum computing, will play a crucial role in the next 10–20 years. Furthermore, we expect computer algebra systems will continue to evolve, significantly reducing computation time from a perspective of system advancements.

With advancements in computational power, more extensive Monte Carlo simulations under various scenarios will enable a deeper investigation for identifying the causes of unusual parameter estimation results. Understanding these causes when encountering improper solutions will make further investigation in parameter estimation procedure. Such an extensive simulation is definitely important and remains a future research topic.

For further investigation, we may study transition points among various solution patterns. For example, we use two sample covariance matrices, S1-1 and S1-2, described in Section 4; S1-1 and S1-2 correspond to a positive solution and no solutions, respectively. We linearly interpolate these two sample covariance matrices; that is, we construct $S(t) := (1 - t)S_{1-1} + tS_{1-2}$ with $t \in [0, 1]$. We note that the maximum likelihood estimation can be performed for any $t \in [0, 1]$ because $S(t)$ is positive-definite. The transition point of positive and negative estimates can be observed by investigating the change in the solution pattern with varying t .

Figure 4 shows the minimum value of the discrepancy function as a function of $\hat{\psi}_5$ with varying t (note that $\hat{\psi}_5 \rightarrow -\infty$ with Jennrich and Robinson’s (1969) algorithm when $t = 1$). The results indicate that as t increases, the local minimizer decreases and becomes negative around $t = 0.41$. As t increases further, the discrepancy function becomes flat, and the local minimum disappears at the transition point around $t = 0.45$. Beyond this point, there are no global minima. The discrepancy function varies with a small change in t around 0.4–0.5, and the change is smooth but drastic. Such a drastic change would make investigating the improper solutions difficult.

For further study, an analysis of geometrical structure based on singularity theory and bifurcation theory would be helpful. Moreover, in addition to investigating the change in solution pattern between S1-1 and S1-2, it would be interesting to study various change patterns near the transition points with

numerous simulations. These considerations will definitely be significant for further investigating the improper solutions, but they are beyond the scope of this research. We will take them as future research topics.

Applying algebraic approaches to the least squares method, which is more commonly used in factor analysis than the maximum likelihood method, would also be a promising direction. Preliminary experiments under similar settings suggest that the algebraic approach for the least squares method requires significantly less computational effort compared to that for the maximum likelihood method. This difference appears to stem from the fact that the solution space of the normal equation for the maximum likelihood method includes non-stationary points of the likelihood function, whereas the solution space for the least squares method consists solely of stationary points of the mean squared error. A detailed comparison between these two methods, along with further theoretical exploration, will also be an important subject for future work.

Data availability statement. This manuscript does not contain any data that requires a data availability statement.

Acknowledgements. We are grateful to an associate editor and three reviewers for their detailed and insightful comments and suggestions, which have significantly improved the quality of the article.

Funding statement. This work was supported by JSPS KAKENHI Grant Number JP23K10988 (R.F.), JP25H01482 (R.F.), JP23H04474 (K.H.), JP20K14312 (Y.K.), JP21K18312 (Y.K.), Japan, and the Institute of Mathematics for Industry, Joint Usage/Research Center in Kyushu University (FY2020 Short-term Visiting Researcher (Reference No. 20200025)).

Competing interests. The authors declare none.

References

- Adachi, K. (2013). Factor analysis with EM algorithm never gives improper solutions when sample covariance and initial parameter matrices are proper. *Psychometrika*, 78(2), 380–394. <https://doi.org/10.1007/s11336-012-9299-8>
- Anderson, T. W., & Rubin, H. (1956). Statistical inference in factor analysis. In J. Neyman (Ed.), *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 5, pp. 111–150). University of California Press.
- Ardiyansyah, M., & Sodomo, L. (2023). Dimensions of higher order factor analysis models. *Algebraic Statistics*, 14(1), 91–108. <https://doi.org/10.2140/astat.2023.14.91>
- Bechtoldt, H. P. (1961). Some latent structures in the primary mental abilities of college freshmen. *Psychological Monographs: General and Applied*, 75(12), 1–31. <https://doi.org/10.1037/h0093829>
- Choi, J., Zou, H., & Oehlert, G. (2010). A penalized maximum likelihood approach to sparse factor analysis. *Statistics and its Interface*, 3(4), 429–436. <https://doi.org/10.4310/sii.2010.v3.n4.a1>
- Cox, D. A., Little, J., & O’Shea, D. (2015). *Ideals, varieties, and algorithms: An introduction to computational algebraic geometry and commutative algebra* (4th ed.) Springer Publishing Company.
- Davis, F. B. (1944). Fundamental factors of comprehension in reading. *Psychometrika*, 9, 185–197. <https://doi.org/10.1007/BF02288722>
- van Driel, O. P. (1978). On various causes of improper solutions in maximum likelihood factor analysis. *Psychometrika*, 43(2), 225–243. <https://doi.org/10.1007/bf02293865>
- Drton, M., & Xiao, H. (2010). Finiteness of small factor analysis models. *Annals of the Institute of Statistical Mathematics*, 62(4), 775–783. <https://doi.org/10.1007/s10463-010-0293-6>
- Drton, M., Sturmfels, B., & Sullivant, S. (2007). Algebraic factor analysis: Tetrads, pentads and beyond. *Probability Theory and Related Fields*, 138(3), 463–493. <https://doi.org/10.1007/s00440-006-0033-2>
- Drton, M., Grosdos, A., Portakal, I., & Sturmfels, B. (2025). Algebraic sparse factor analysis. <https://doi.org/10.48550/arxiv.2312.14762>
- Faugère, J.-C., Gianni, P., Lazard, D., & Mora, T. (1993). Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4), 329–344. <https://doi.org/10.1006/jsc.1993.1051>
- Harman, H. H. (1976). *Modern factor analysis* (3rd ed.) University of Chicago Press.
- Hayashi, K., & Lu, L. (2014). On detection of improper solutions in factor analysis by the em algorithm. *Behaviormetrika*, 41(2), 245–268. <https://doi.org/10.2333/bhmk.41.245>
- Hinkelmann, F., Brandon, M., Guang, B., McNeill, R., Blekherman, G., Veliz-Cuba, A., & Laubenbacher, R. (2011). ADAM: Analysis of discrete models of biological systems using computer algebra. *BMC Bioinformatics*, 12(1), 295. <https://doi.org/10.1186/1471-2105-12-295>

- Hirose, K., & Konishi, S. (2012). Variable selection via the weighted group lasso for factor analysis models. *The Canadian Journal of Statistics*, 40(2), 345–361. <https://doi.org/10.1002/cjs.11129>
- Hirose, K., & Yamamoto, M. (2015). Sparse estimation via nonconcave penalized likelihood in factor analysis model. *Statistics and Computing*, 25(5), 863–875. <https://doi.org/10.1007/s11222-014-9458-0>
- Jennrich, R. I., & Robinson, S. M. (1969). A Newton-Raphson algorithm for maximum likelihood factor analysis. *Psychometrika*, 34(1), 111–123. <https://doi.org/10.1007/bf02290176>
- Jöreskog, K. G. (1967). Some contributions to maximum likelihood factor analysis. *Psychometrika*, 32(4):443–482. <https://doi.org/10.1007/bf02289658>
- Kano, Y. (1998). Improper solutions in exploratory factor analysis: Causes and treatments. In A. Rizzi, M. Vichi, & H.-H. Bock (Eds.), *Advances in data science and classification* (pp. 375–382). Springer.
- Kartechina, N. V., Bobrovich, L. V., Nikonorova, L. I., Pchelinceva, N. V., & Abaluev, R. N. (2020). Practical application of variance analysis of four-factor experience data as a technology of scientific research. *IOP Conference Series: Materials Science and Engineering*, 919(5), 52030. <https://doi.org/10.1088/1757-899x/919/5/052030>
- Kendall, M. G. (1975). *Multivariate analysis*. Griffin.
- Krijnen, W. P., Dijkstra, T. K., & Gill, R. D. (1998). Conditions for factor (in)determinacy in factor analysis. *Psychometrika*, 63(4), 359–367. <https://doi.org/10.1007/bf02294860>
- Laubenbacher, R., & Sturmfels, B. (2009). Computer algebra in systems biology. *The American Mathematical Monthly*, 116(10), 882–891. <https://doi.org/10.4169/000298909x477005>
- Lawley, D. N., & Maxwell, A. E. (1971). *Factor analysis as a statistical method* / D.N. Lawley and A. E. Maxwell. (2nd ed.) Butterworths.
- Lin, Y., Ghazanfar, S., Wang, K. Y. X., Gagnon-Bartsch, J. A., Lo, K. K., Su, X., Han, Z.-G., Ormerod, J. T., Speed, T. P., Yang, P., & Yang, J. Y. H. (2019). scMerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell RNA-seq datasets. *Proceedings of the National Academy of Sciences*, 116(20), 9775–9784. <https://doi.org/10.1073/pnas.1820006116>
- Maxwell, A. E. (1961). Recent trends in factor analysis. *Journal of the Royal Statistical Society. Series A (General)*, 124(1), 49–83. <https://doi.org/10.2307/2982623>
- Rubin, D. B., & Thayer, D. T. (1982). EM algorithms for ML factor-analysis. *Psychometrika*, 47(1), 69–76. <https://doi.org/10.1007/bf02293851>
- Sato, M. Pragmatic treatment of improper solutions in factor analysis. *Annals of the Institute of Statistical Mathematics*, 39(2), 443–455, 1987. <https://doi.org/10.1007/bf02491481>
- Shkeer, A. S., & Awang, Z. (2019). Exploring the items for measuring the marketing information system construct: An exploratory factor analysis. *International Review of Management and Marketing*, 9(6), 87–97. <https://doi.org/10.32479/irmm.8622>
- Shurrab, J., Hussain, M., & Khan, M. (2019). Green and sustainable practices in the construction industry. *Engineering, Construction and Architectural Management*, 26(6), 1063–1086. <https://doi.org/10.1108/ecam-02-2018-0056>
- Spearman, C. (1904). General intelligence,” objectively determined and measured. *The American Journal of Psychology*, 15(2), 201. <https://doi.org/10.2307/1412107>
- Thurstone, L. L., & Thurstone, T. G. (1941). *Factorial studies of intelligence*. University of Chicago Press.
- Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3), 611. <https://doi.org/10.1111/1467-9868.00196>
- Veliz-Cuba, A., Aguilar, B., Hinkelmann, F., & Laubenbacher, R. (2014). Steady state analysis of Boolean molecular network models via model reduction and computational algebra. *BMC Bioinformatics*, 15(1), 221. <https://doi.org/10.1186/1471-2105-15-221>
- Vilkaite-Vaitone, N., Skackauskiene, I., & Díaz-Meneses, G. (2022). Measuring green marketing: Scale development and validation. *Energies*, 15(3), 718. <https://doi.org/10.3390/en15030718>
- Yuan, K.-H., & Chan, W. (2008). Structural equation modeling with near singular covariance matrices. *Computational Statistics & Data Analysis*, 52(10), 4842–4858. <https://doi.org/10.1016/j.csda.2008.03.030>

Appendix

A. Basic concepts

In this section, we review basic concepts related to computational algebra. In particular, we deal with basic contents related to fields, rings, and affine varieties along with some concrete examples. First, we give the definition of fields (Cox *et al.*, 2015, section A.1, definition 1).

Definition A.1. A field consists of a set F and two binary operations “+” and “ \cdot ” defined on F satisfying the following conditions:

1. for any $a, b, c \in F$, $(a + b) + c = a + (b + c)$ and $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ (associativity),
2. for any $a, b, c \in F$, $a \cdot (b + c) = a \cdot b + a \cdot c$ (distributivity),

3. for any $a, b \in F$, $a + b = b + a$ and $a \cdot b = b \cdot a$ (commutativity),
4. for any $a \in F$, there exists $0, 1 \in F$ such that $a + 0 = a \cdot 1 = a$ (identities),
5. given $a \in F$, there exists $b \in F$ such that $a + b = 0$ (additive inverses),
6. given $a \in F$, $a \neq 0$, there exists $b \in F$ such that $a \cdot b = 1$ (multiplicative inverses).

For example, \mathbb{Q} , \mathbb{R} , and \mathbb{C} are fields, since they satisfy the above conditions with the sum “+” and product “.”. On the other hand, \mathbb{Z} is not a field, since it does not satisfy the last condition (multiplicative inverses). Indeed, the element $2 \in \mathbb{Z}$ does not have $b \in \mathbb{Z}$ such that $2 \cdot b = 1$.

Next, we define a commutative ring (Cox et al., 2015, section A.1, definition 2).

Definition A.2. A commutative ring consists of a set R and two binary operations “+” and “.” defined on R satisfying conditions 1–5 of Definition A.1.

As mentioned above, \mathbb{Z} is not a field; however, it is a commutative ring. Moreover $\mathbb{R}[z] = \mathbb{R}[z_1, \dots, z_m]$ is a commutative ring. In particular, $\mathbb{R}[z]$ is known as a polynomial ring.

Then, we introduce ideals. In general, ideals can be defined for any ring. For simplicity, we give the definition of ideals for commutative rings (Cox et al., 2015, section 1.4, definition 1).

Definition A.3. Let R be a commutative ring. A subset $\mathcal{I} \subset R$ is an ideal if it satisfies that

$$0 \in \mathcal{I}, \quad a, b \in \mathcal{I} \Rightarrow a + b \in \mathcal{I}, \quad a \in \mathcal{I}, b \in R \Rightarrow b \cdot a \in \mathcal{I}.$$

The definition of ideals is similar to that of linear sub-spaces; both have to be closed under addition and multiplication. However, a difference lies in multiplication; for a linear sub-space, we multiply an element in the linear sub-space by an element in the field, whereas for an ideal, we multiply an element in the ideal by an element in the ring.

In Section 3.1, we defined the subset in $\mathbb{R}[z]$ as follows:

$$\langle f_1, \dots, f_r \rangle = \left\{ \sum_{i=1}^r q_i f_i : q_i \in \mathbb{R}[z] \right\} \subset \mathbb{R}[z],$$

where $f_1, \dots, f_r \in \mathbb{R}[z]$. $\langle f_1, \dots, f_r \rangle$ is referred to as the ideal generated by f_1, \dots, f_r . The subset $\langle f_1, \dots, f_r \rangle$ is similar to a span of a finite number of vectors. In each case, one takes linear combinations, using field coefficients for a span and polynomial coefficients for an ideal. We show that $\langle f_1, \dots, f_r \rangle$ is an ideal in $\mathbb{R}[z]$.

Lemma A.4. $\langle f_1, \dots, f_r \rangle \subseteq \mathbb{R}[z]$ is an ideal.

Proof. Let $\mathcal{J} = \langle f_1, \dots, f_r \rangle$. If we put $h_1 = \dots = h_r = 0$, then $0 = \sum_{i=1}^r h_i f_i$. So we have $0 \in \mathcal{J}$. Let $h, g \in \mathcal{J}$. The definition of \mathcal{J} implies that $h = \sum_{i=1}^r h_i f_i, g = \sum_{i=1}^r g_i f_i$ for some $h_1, \dots, h_r, g_1, \dots, g_r \in \mathbb{R}[z]$. Since $h + g = \sum_{i=1}^r h_i f_i + \sum_{i=1}^r g_i f_i = \sum_{i=1}^r (h_i + g_i) f_i$ holds, we have $h + g \in \mathcal{J}$ by $h_1 + g_1, \dots, h_r + g_r \in \mathbb{R}[z]$ and the definition of \mathcal{J} . Let $h \in \mathcal{J}$ and $c \in \mathbb{R}[z]$. Then, $h = \sum_{i=1}^r h_i f_i$ holds for some $h_1, \dots, h_r \in \mathbb{R}[z]$. Since $ch = c \sum_{i=1}^r h_i f_i = \sum_{i=1}^r (ch_i) f_i$ holds, we have $ch \in \mathcal{J}$ by $ch_1, \dots, ch_r \in \mathbb{R}[z]$ and the definition of \mathcal{J} . Thus, $\mathcal{J} \subseteq \mathbb{R}[z]$ is an ideal. \square

Let $U \subseteq W^m$, where $W = \mathbb{R}$ or \mathbb{C} . Recall that any affine variety is the solution space in W^m of a system of algebraic equations which has the form (11). Zariski closures are defined by the following (Cox et al., 2015, section 4.4, definition 2).

Definition A.5. The Zariski closure of U , denoted by \overline{U} , is the smallest affine variety containing U (in the sense that if $V \subseteq W^m$ is any affine variety containing U , then $\overline{U} \subseteq V$).

Note that the Zariski closure \overline{U} is the smallest solution space in W containing U among solution spaces of systems of algebraic equations which have the form (11). If the subset U is an affine variety, then U is the Zariski closure of U , that is, the smallest solution space containing U . We conclude this section with some examples for Zariski closures.

Example A.6. Let $P = \mathbb{V}_W(z_1(z_3^2 + 1), z_2(z_3^2 + 1))$, $Q = \mathbb{V}_W(z_3^2 + 1)$, and $U = P \setminus Q$. Since the subset U is an affine variety, we have $U = \overline{U}$. If $W = \mathbb{R}$, we have $\mathbb{V}_{\mathbb{R}}(z_1, z_2) = \overline{U} = U = P$. If $W = \mathbb{C}$, then we have $\mathbb{V}_{\mathbb{C}}(z_1, z_2) = \overline{U} = U \subsetneq P$.

Example A.7. If $P = \mathbb{V}_W(z_1 z_2)$, $Q = \mathbb{V}_W(z_1, z_2)$, and $U = P \setminus Q$, then $P = \overline{U} \supsetneq U$.

B. Implementation details

As a matter of fact, the simulations conducted in Section 4 have a very large number of candidates for MLEs. However, since these candidates tend to fall into several types, it is possible to construct more simplified subproblems that further subdivide the subproblems constructed in Algorithm 1. In this section, we will show details of these candidates, and present details of the implementation for obtaining the candidates.

Table B.1. The number of all real solutions to (3) satisfying that Ψ is not singular

Sim no.	1	2	3	4	5	6	7	8	9	10
S1	29	47	37	33	37	17	53	21	25	13
S2	61	25	27	53	33	23	21	27	33	∞
S3	17	41	29	25	∞	33	39	45	49	39

Table B.2. The number of all complex solutions to (3) satisfying that Ψ is not singular

Sim no.	1	2	3	4	5	6	7	8	9	10
S1	1301	1301	1301	1301	1301	∞	1299	1301	1301	∞
S2	1301	1301	1301	1301	1301	1301	1301	1301	1301	∞
S3	∞	1301	∞	∞	∞	1301	∞	1301	1301	1301

Considering a rotational indeterminacy, we suppose that the upper triangular elements of Λ are fixed by zero in Section 3.2. In addition, we consider three models with $p = 5, k = 2$ in Section 4. Therefore, we suppose the following form of factor loading matrices in our simulation:

$$\Lambda = \begin{pmatrix} \lambda_{11} & 0 \\ \lambda_{21} & \lambda_{22} \\ \vdots & \vdots \\ \lambda_{51} & \lambda_{52} \end{pmatrix}.$$

Before discussing details for our implementation, we show details of solutions to Eqs. (6) and (9). Table B.1 shows the number of all real solutions to (6) and (9) satisfying that Ψ is not singular, or equivalently

$$s_{11} - \lambda_{11}^2 \neq 0, \quad s_{22} - (\lambda_{21}^2 + \lambda_{22}^2) \neq 0, \quad s_{33} - (\lambda_{31}^2 + \lambda_{32}^2) \neq 0, \\ s_{44} - (\lambda_{41}^2 + \lambda_{42}^2) \neq 0, \quad s_{55} - (\lambda_{51}^2 + \lambda_{52}^2) \neq 0.$$

Note that if Ψ is not singular, the likelihood equation (4) is equivalent to Eqs. (6) and (9). Therefore, Table B.1 shows the number of all real solutions to (3) satisfying that Ψ is not singular. In addition, Table B.2 shows the number of all complex solutions to (3) satisfying that Ψ is not singular. Note that the simulations with infinite complex solutions, namely, S1-6, S1-10, S2-10, S3-1, S3-3, S3-4, S3-5, and S3-7, form non-zero dimensional ideals. Since we would like to discuss the difficulty of computing all solutions from an algebraic computational aspect, Tables B.1 and B.2 do not address the identifiability with respect to the sign of column vectors in a loading matrix, unlike Table 2. Furthermore, Table 2 excludes candidates that form a one-factor model or the zero loading matrix, while Tables B.1 and B.2 do not exclude these candidates. As shown in Tables B.1 and B.2, Eq. (3) has a surprisingly large number of solutions satisfying that Ψ is not singular. In particular, the number of complex solutions is significantly large. Real radical ideals can remove unwanted complex solutions, but they are heavier than primary ideal decompositions. Therefore, due to such a practical problem, we must skillfully compute real solutions while unfortunately also having unnecessary complex solutions at hand.

Now, let us show the number of all complex solutions to (3) satisfying that Ψ is not singular, for each of four spaces satisfying the following conditions: (a) $\lambda_{11} = 0$, (b) $\lambda_{11} \neq 0$ and $\lambda_{i2} = 0$ for some $i = 2, \dots, 5$, (c) $\lambda_{11} \neq 0$, $\lambda_{i2} \neq 0$ for any $i = 2, \dots, 5$ and $\det \Lambda_{J\{1,2\}} = 0$ for some $J \subset \{1, \dots, 5\}$ satisfying $\#J = 2$, and (d) $\lambda_{11} \neq 0$, $\lambda_{i2} \neq 0$ for any $i = 2, \dots, 5$ and $\det \Lambda_{J\{1,2\}} \neq 0$ for any $J \subset \{1, \dots, 5\}$ satisfying $\#J = 2$. Table B.3 provides details for Table B.2 based on these classifications.

As can be seen from Table B.3, the spaces with an infinite number of solutions are limited to the spaces satisfying condition (a), and we have only a finite number of solutions when condition (a) is not satisfied. If we use sum ideals and saturation ideals related to the polynomial λ_{11} , we can divide the solution spaces into the spaces which satisfy $\lambda_{11} = 0$ or not.

Since the number of solutions satisfying condition (d) is greater than that satisfying conditions (b) and (c), the computation of real solutions satisfying condition (d) seems to require heavy computational loads. In reality, however, there are several cases where computing real solutions satisfying condition (b) requires more heavy computational loads than (c) and (d). The solution space for (c) is more simple than that for (d). For ease of comprehension, we will compare outlines of the systems of algebraic equations satisfying (b) and (d) for a specific dataset, S2-10, which has a large number of solutions satisfying condition (b). After making the comparison, we present the strategy in our implementation.

Table B.3. Detailed results for Table B.2: the number of all complex solutions to (3) satisfying that Ψ is not singular for four different parameter spaces

Sim no.	1	2	3	4	5	6	7	8	9	10
(a) All complex solutions to (3) satisfying that $\det \Psi \neq 0$ and $\lambda_{11} = 0$.										
S1	1	1	1	1	1	∞	1	1	1	∞
S2	1	1	1	1	1	1	1	1	1	∞
S3	∞	1	∞	∞	∞	1	∞	1	1	1
(b) All complex solutions to (3) satisfying that $\det \Psi \neq 0$, $\lambda_{11} \neq 0$, and $\lambda_{i2} = 0$ for some $i = 2, \dots, 5$.										
S1	212	212	212	212	212	316	210	212	212	346
S2	220	212	212	212	212	212	212	212	212	398
S3	370	212	288	346	398	212	346	212	212	212
(c) All complex solutions to (3) satisfying that $\det \Psi \neq 0$, $\lambda_{11} \neq 0$, and $\lambda_{i2} \neq 0$ for any $i = 2, \dots, 5$, and $\det \Lambda_{J\{1,2\}} = 0$ for some $J \subset \{1, \dots, 5\}$ satisfying $\#J = 2$.										
S1	0	0	0	0	0	48	0	0	0	128
S2	0	0	0	0	0	0	0	0	0	128
S3	104	0	112	52	128	0	52	0	0	0
(d) All complex solutions to (3) satisfying that $\det \Psi \neq 0$, $\lambda_{11} \neq 0$, and $\lambda_{i2} \neq 0$ for any $i = 2, \dots, 5$, and $\det \Lambda_{J\{1,2\}} \neq 0$ for any $J \subset \{1, \dots, 5\}$ satisfying $\#J = 2$.										
S1	1088	1088	1088	1088	1088	816	1088	1088	1088	800
S2	1080	1088	1088	1088	1088	1088	1088	1088	1088	768
S3	784	1088	896	896	768	1088	896	1088	1088	1088

Now, let us consider the case of S2-10. A Gröbner basis transforms the system of algebraic equations satisfying $\lambda_{22} = 0$ into a form consisting of the 15 polynomials $g_1^{(b)}, \dots, g_{15}^{(b)}$, which have the following leading monomials:

$$\begin{array}{ll}
 \text{LM}(g_1^{(b)}) = \lambda_{11}, & g_1^{(b)} \in \mathbb{R}[\lambda_{11}, \lambda_{21}, \lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_2^{(b)}) = \lambda_{21}, & g_2^{(b)} \in \mathbb{R}[\lambda_{21}, \lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_3^{(b)}) = \lambda_{22}, & g_3^{(b)} \in \mathbb{R}[\lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_4^{(b)}) = \lambda_{31}^2, & g_4^{(b)} \in \mathbb{R}[\lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_5^{(b)}) = \lambda_{31}\lambda_{42}, & g_5^{(b)} \in \mathbb{R}[\lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_6^{(b)}) = \lambda_{31}\lambda_{51}^7, & g_6^{(b)} \in \mathbb{R}[\lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_7^{(b)}) = \lambda_{31}\lambda_{52}, & g_7^{(b)} \in \mathbb{R}[\lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_8^{(b)}) = \lambda_{32}, & g_8^{(b)} \in \mathbb{R}[\lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_9^{(b)}) = \lambda_{41}, & g_9^{(b)} \in \mathbb{R}[\lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_{10}^{(b)}) = \lambda_{42}^2, & g_{10}^{(b)} \in \mathbb{R}[\lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_{11}^{(b)}) = \lambda_{42}\lambda_{51}^{12}, & g_{11}^{(b)} \in \mathbb{R}[\lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_{12}^{(b)}) = \lambda_{42}\lambda_{52}, & g_{12}^{(b)} \in \mathbb{R}[\lambda_{42}, \lambda_{51}, \lambda_{52}], \\
 \text{LM}(g_{13}^{(b)}) = \lambda_{51}^{148}, & g_{13}^{(b)} \in \mathbb{R}[\lambda_{51}, \lambda_{52}],
 \end{array}$$

$$\begin{aligned}\mathrm{LM}\left(g_{14}^{(b)}\right) &= \lambda_{51}^2 \lambda_{52}, & g_{14}^{(b)} &\in \mathbb{R}[\lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_{15}^{(b)}\right) &= \lambda_{51}^{15}, & g_{15}^{(b)} &\in \mathbb{R}[\lambda_{52}].\end{aligned}$$

On the other hand, the system of algebraic equations of S2-10 satisfying (d), transformed into a form that is easy to solve in the Gröbner basis, consists of the nine polynomials $g_1^{(d)}, \dots, g_9^{(d)}$ which have the following leading monomials:

$$\begin{aligned}\mathrm{LM}\left(g_1^{(d)}\right) &= \lambda_{11}, & g_1^{(d)} &\in \mathbb{R}[\lambda_{11}, \lambda_{21}, \lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_2^{(d)}\right) &= \lambda_{21}, & g_2^{(d)} &\in \mathbb{R}[\lambda_{21}, \lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_3^{(d)}\right) &= \lambda_{22}, & g_3^{(d)} &\in \mathbb{R}[\lambda_{22}, \lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_4^{(d)}\right) &= \lambda_{31}, & g_4^{(d)} &\in \mathbb{R}[\lambda_{31}, \lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_5^{(d)}\right) &= \lambda_{32}^2, & g_5^{(d)} &\in \mathbb{R}[\lambda_{32}, \lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_6^{(d)}\right) &= \lambda_{41}, & g_6^{(d)} &\in \mathbb{R}[\lambda_{41}, \lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_7^{(d)}\right) &= \lambda_{42}, & g_7^{(d)} &\in \mathbb{R}[\lambda_{42}, \lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_8^{(d)}\right) &= \lambda_{51}^2, & g_8^{(d)} &\in \mathbb{R}[\lambda_{51}, \lambda_{52}], \\ \mathrm{LM}\left(g_9^{(d)}\right) &= \lambda_{52}^{192}, & g_9^{(d)} &\in \mathbb{R}[\lambda_{52}].\end{aligned}$$

It is noteworthy that the leading monomials for (b), namely, $\mathrm{LM}(g_i^{(b)})$, are relatively more complicated than those for (d), namely, $\mathrm{LM}(g_i^{(d)})$, except for the last polynomial $g_{15}^{(b)}, g_9^{(d)}$ in each case. All real solutions satisfying condition (d) can be obtained by simple substitutions and simple real number tests once $g_9^{(d)}$ is factorized, since it has only simple leading monomials except for $g_9^{(d)}$. On the other hand, it requires heavy computational loads for obtaining real solutions satisfying condition (b), since there are some leading monomials of high degree, even if $g_{15}^{(b)}$ is factorized.

Therefore, the following strategy, which decomposes the solution space satisfying condition (b) as delicately as possible, is actually employed. Let

$$\mathcal{J} = \{ \{S - \Lambda\Lambda^\top - \mathrm{diag}(S - \Lambda\Lambda^\top)\} S^{-1} \Lambda \}_{ij} : 1 \leq i \leq 5, 1 \leq j \leq 2 \}.$$

Put

$$h_1 = s_{11} - \lambda_{11}^2, \quad h_2 = s_{22} - (\lambda_{21}^2 + \lambda_{22}^2), \quad \dots, \quad h_5 = s_{55} - (\lambda_{51}^2 + \lambda_{52}^2).$$

The above polynomials are used in Algorithm 1. Moreover, we give the polynomials

$$\begin{aligned}h_6 &= \lambda_{11}, \quad h_7 = \lambda_{22}, \quad h_8 = \lambda_{32}, \quad h_9 = \lambda_{42}, \quad h_{10} = \lambda_{52}, \\ h_{11} &= \det \Lambda_{\{2,3\}\{1,2\}}, \quad h_{12} = \det \Lambda_{\{2,4\}\{1,2\}}, \\ h_{13} &= \det \Lambda_{\{2,5\}\{1,2\}}, \quad h_{14} = \det \Lambda_{\{3,4\}\{1,2\}}, \\ h_{15} &= \det \Lambda_{\{3,5\}\{1,2\}}, \quad h_{16} = \det \Lambda_{\{4,5\}\{1,2\}}.\end{aligned}$$

First, we construct the sum $\mathcal{I}_0 = \mathcal{J} + \mathcal{K}_1$ and the saturation $\mathcal{I}_1 = \mathcal{J} : \mathcal{K}_1^\infty$. Second, we construct the sums $\mathcal{I}_{00} = \mathcal{I}_0 + \mathcal{K}_2, \mathcal{I}_{10} = \mathcal{I}_1 + \mathcal{K}_2$ and saturations $\mathcal{I}_{01} = \mathcal{I}_0 : \mathcal{K}_2^\infty, \mathcal{I}_{11} = \mathcal{I}_1 : \mathcal{K}_2^\infty$. In the same way, for each $b \in \{0, 1\}^i, 2 \leq i \leq 13$, we sequentially construct $\mathcal{I}_{b0} = \mathcal{I}_b + \mathcal{K}_{i+1}, \mathcal{I}_{b1} = \mathcal{I}_b : \mathcal{K}_{i+1}^\infty$. Note that, by Proposition 3.1, we have $\mathbb{V}_W(\mathcal{J}) = \bigcup_{b \in \{0,1\}^{16}} \mathbb{V}_W(\mathcal{I}_b)$. Finally, we compute all real solutions in $\mathbb{V}_{\mathbb{R}}(\mathcal{I}_b)$ for each $b \in \{0, 1\}^{16}$. Although there are cases where such a decomposition yields an empty set of the affine variety for a sum or saturation, the above strategy has proven to be highly effective, enabling us to obtain all real solutions.

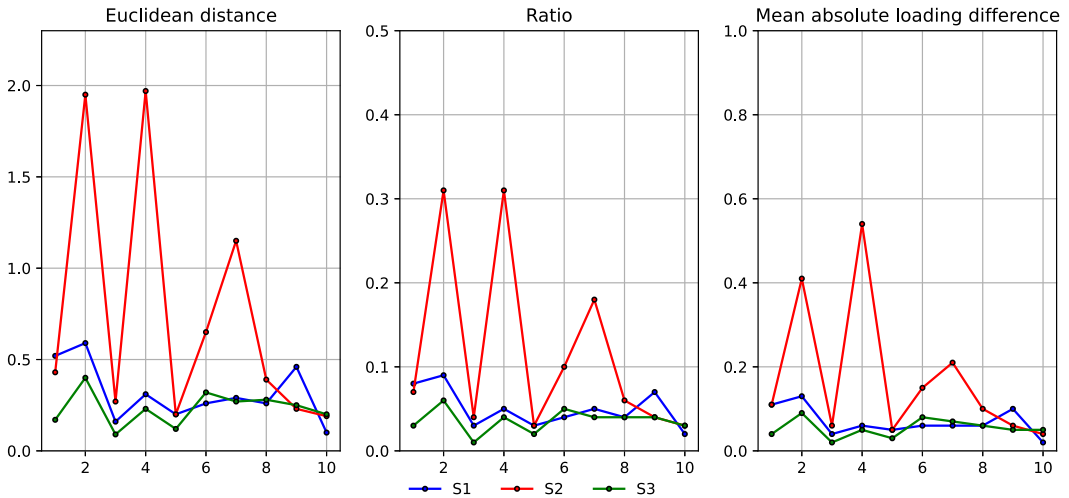
C. Analysis with different decimal precision in sample covariance matrices

In Section 4, the sample covariance matrices were rounded to one decimal place. However, the precision of the results can be improved by rounding to more decimal places. Since computer algebra can be time-consuming when using more than one decimal place (e.g., several months), we further investigate the results of factanal between one and two decimal places.

Table C.4 and Figure C.1 show differences between loading matrices obtained from sample covariance matrices, with the elements rounded to one and two decimal places. The column “Euclidean distance” in Table C.4 shows the average of the Euclidean distances, the column “Ratio” indicates the ratio of “Euclidean distance” (the average distance) to the maximum

Table C.4. Difference in estimates between sample covariance matrices with elements rounded to one and two decimal places

Average	Euclidean distance	Ratio	Mean absolute loading difference
S1	0.32	0.05	0.07
S2	0.74	0.12	0.17
S3	0.23	0.04	0.06

**Figure C.1.** Difference in estimates between sample covariance matrices with elements rounded to one and two decimal places.

distance $\sqrt{10 \times 2^2} \approx 6.32$, and the column “Mean absolute loading difference” shows the average of factor loading differences. Note that, for two loading matrices $A, B \in \mathbb{R}^{5 \times 2}$, they are given by

$$\begin{aligned}
 \text{“Euclidean distance”} &= \sqrt{\sum_{i=1}^5 \sum_{j=1}^2 (A_{ij} - B_{ij})^2}, \\
 \text{“Ratio”} &= \frac{1}{\sqrt{10 \times 2^2}} \times \text{“Euclidean distance,”} \\
 \text{“Mean absolute loading difference”} &= \frac{1}{5 \times 2} \sum_{i=1}^5 \sum_{j=1}^2 |A_{ij} - B_{ij}|.
 \end{aligned}$$

Moreover, each graph in Figure C.1 shows “Euclidean distance,” “Ratio,” and “Mean absolute loading difference” for each simulation in types S1 (blue), S2 (red), and S3 (green). The horizontal axis denotes the simulation number for each type, and the vertical axes correspond to “Euclidean distance” (left graph), “Ratio” (middle graph), and “Mean absolute loading difference” (right graph).

From Table C.4, S1 and S3 seem to have small differences, while S2 appears to have a large difference. However, if we look at Figure C.1, the large difference in S2 appears only in some specific cases (S2-2, S2-4, and S2-7), and the other differences are as small as those in S1 and S3. This indicates that the apparent large difference in S2 in Table C.4 is caused by these specific cases (S2-2, S2-4, and S2-7). Therefore, the differences between results with one-decimal and two-decimal precision are not critical in most cases.