Comparison of Smoothing Approaches to Polychoric Correlation Matrices in CFA

Yijun Cheng

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Education

University of Washington

2024

Committee:

Oscar Olvera Astivia

Chun Wang

Program Authorized to Offer Degree:

Education

University of Washington

Abstract

Comparison of Smoothing Approaches to Polychoric Correlation Matrices in CFA

Yijun Cheng

Chair of the Supervisory Committee:

Oscar Olvera Astivia

Measurement & Statistics College of Education

Confirmatory factor analysis (CFA) using polychoric correlations has become standard in psychometric and item analyses. Nevertheless, issues such as sparse data can lead to non-positive definite (NPD) polychoric correlation matrices, posing notable challenges. Smoothing algorithms to address this issue can play an important role in eliminating noise, enhancing signal quality, and regularizing data. In the present paper, a series of simulation studies were conducted to compare the eigenvalue substitution smoothing method with Higham's nearest correlation approach. Both aim to transform NPD matrices into positive definite ones but differ in technique. Eigenvalue substitution adjusts eigenvalues below a set threshold and rescales, while Higham's method employs iterative eigen decomposition, selectively choosing eigenvalues above a certain threshold and reconstructing the matrix until convergence. It was found that although Higham's correlation approach slightly outperforms the eigenvalue substitution method in terms of parameter bias, the converse was more efficient. Neither approach was particularly favorable at assessing fit. Recommendations for empirical data analysis and potential future avenues of research are discussed.

Introduction

In the context of probability theory and statistics, a covariance matrix, also known as an auto-covariance matrix, dispersion matrix, variance matrix, or variance–covariance matrix, is a square, symmetric matrix designed to articulate the linear relations between every pair of variables. A correlation matrix can be considered as a special type of a covariance matrix between standardized variables. In statistical analysis, correlation and covariance matrices are essential for understanding the relationships between variables within a dataset. A covariance matrix provides information about the degree to which variables change together, capturing the direction and magnitude of these changes (Rencher & Christensen, 2002).

Categorical Data in Factor Analysis

Confirmatory Factor Analysis (CFA) relies on the correlation or covariance matrix to model the relationships between latent constructs and observed indicators (Mueller & Hancock, 2015). Usually, the Pearson covariance (or correlation) matrix is employed when analyzing continuous variables. When working with ordered categorical variables, a notable challenge arises in the applicability of the Pearson correlation matrix. The Pearson correlation coefficient can exhibit undue issues when in the presence of categorical data. The primary issue stems from the fact that ordered categorical variables have distinct categories with meaningful order but lack equal intervals between them (Robitzsch, 2020). Since the Pearson correlation can be influenced by the scale (i.e., continuous VS discrete) and distribution of the variables being analyzed, it might not accurately capture the underlying association of the latent trait it is purporting to measure. Under the assumption that continuous variables underlie the observed ordinal responses, the matrix of Pearson correlations underestimates the correlation matrix among the

underlying continuous variables (Olsson, 1979). The polychoric correlation matrix is a more suitable alternative to address this challenge. Jöreskog and Sörbom (1996a) found that the polychoric correlations was a consistent and unbiased estimator of the true, latent correlation between underlying continuous traits. It assumes that a latent multivariate normal density underlies the observed ordinal responses, providing a more accurate representation of the relationships between variables with meaningful order (Robitzsch, 2020). The procedure involves estimating the correlation between two latent continuous variables based on the observed ordinal responses to implement the polychoric correlation matrix (Holgado-Tello et al., 2010). Let us suppose there are two ordinal items X_1 and X_2 with $m_1$and m_2 categories correspondingly. Further, suppose the underlying variables are jointly normally distributed. When $m_1 = m_2 = 2$, both items have only two categories and are dichotomous. In this instance, the correlation coefficient is referred to as the tetrachoric correlation with i categories for item 1 and j categories of item 2. It can be written as:

$$P\,[X = i, Y = j] = \; p_{ij} = \int_{a_{i-1}}^{a_i} \int_{b_{j-1}}^{b_j} \frac{1}{2\pi\sqrt{1-\rho^2}} exp^{-\frac{1}{2(1-\rho^2)}(x^2 - 2\rho xy + y^2)} dxdy \qquad (1)$$

where $P\,[X = i, Y = j] = \; p_{ij}$ defines the probability that the random variable X takes on value i, and the random variable Y takes on value j, denoted as p_{ij}, and ρ is the polychoric correlation coefficient between X and Y. The limits of integration, a_{i-1} to a_i for X, and b_{j-1} to b_j for Y, define the thresholds over the latent distribution which represents the bounds of the categories that i and j fall into.

The polychoric correlation can be estimated by maximizing the function of the maximum likelihood of the multinomial distribution:

$$\ln L = \sum_{i=1}^{m1}\sum_{i=1}^{m2} n_{ij} log p_{ij} \quad (2)$$

where n_{ij} means the observed frequency or count of occurrences for combining category i and category j. p_{ij} defines the probability that the random variable X takes on value i and the random variable Y takes on value j (Holgado-Tello et al., 2010).

The property of positive definiteness of covariance matrices

A positive definite matrix is a special symmetric matrix that plays a crucial role in linear algebra, regression analysis and various mathematical applications. A real, square symmetric matrix A is considered positive definite if and only if, for any non-zero column vector x of appropriate dimensions, the quadratic form $x^T Ax$ is always positive, except when x is the zero vector (Rencher & Christensen, 2002).

Mathematically, a real, square symmetric matrix A is positive definite if:

$$x^T Ax > 0 \quad (3)$$

for every non-zero column vector x.

The property of positive definiteness, as defined above, necessarily imply the following conditions to be true:

Eigenvalues: All eigenvalues of a positive definite matrix are positive.

Determinant: The determinant of a positive definite matrix is positive.

Cholesky Decomposition: A positive definite matrix can be decomposed into the product of a lower triangular matrix and its transpose using the Cholesky decomposition.

Negative-definite and negative semi-definite matrices are defined analogously. A matrix that is not positive semi-definite or negative semi-definite is sometimes called indefinite (e.g. Wothke, 1993). Positive definite matrices are used in various mathematical and statistical applications, including optimization, numerical analysis, and the definition of several multivariate distributions in statistics. They provide a foundation for stable and well-behaved mathematical operations (Bhatia, 2009).

Smoothing of non-positive definite covariance matrices

Due to the computational complexity of estimating the polychoric correlation (i.e., several multidimensional integrals need to be jointly estimated), each correlation coefficient is estimated bivariately, tackling one pair of items at a time (Ekström, 2021). In this situation, the risk of obtaining a non-positive definite matrix increases. To alleviate this problem, smoothing techniques can be used to address instability in covariance estimates and to reduce noise, especially by transforming these matrices into true positive-definite correlation matrices. These methodologies contribute significantly to refining data representations' stability, enhancing correlations' reliability, and fostering more robust analyses. Several smoothing methods are available in the literature. In this paper, we mainly focused on two popular smoothing methods designed to enhance the stability and reliability of correlation matrices.

Method 1: Eigenvalue substitution

The first method will be referred to as the "eigenvalue substitution" method, and it is implemented in the cor.smooth() function from the psych package (Revelle, version 2.3.12). The eigenvalue substitution method begins with the eigendecomposition of the correlation matrix.

Next, this method checks the size of the smallest eigenvalues against a predetermined threshold, $c = 10^{-12}$. Finally, a small constant value (100c in this case) is used to replace the smallest eigenvalue. Once the eigenvalues are adjusted, the correlation matrix is recalculated using the updated eigenvalues and the original eigenvectors. This thresholding aims to mitigate the impact of very small eigenvalues, which could easily make the matrix non-positive definite.

Let A be the $n \times n$ correlation matrix. This method begins with the eigen-decomposition of the matrix:

$$A = Q\Lambda Q^{-1} \tag{4}$$

where Q is the square $n \times n$ matrix whose ith column (for $i = 1, \dots, n$) is the eigenvector q_i of A, and Λ is the diagonal matrix whose diagonal elements are the corresponding eigenvalues, $\Lambda_{ii} = \lambda_i$.

If $\lambda_i < c$, then 100c takes its place to obtain a new list of eigenvalues, λ' and correspondingly a new diagonal matrix of $\Lambda'_{ii} = \lambda'_i$. We defined the new matrix as A' as

$$A' = Q\Lambda' Q^{-1} \tag{5}$$

The new matrix A' will be positive definite after its eigenvalues have been substituted.

Method 2: Higham's nearest correlation

On the other hand, Higham's (2002) nearest correlation methodology delves into the issue of computing the nearest correlation matrix to a given symmetric matrix which is not positive definite. This approach is of paramount importance in finance to analyze stock correlations (Al-Homidan & AlQarni, 2012). The central idea of this approach is an iterative

algorithm based on the method of alternating projections, specifically designed to tackle this challenge by leveraging the spectral properties of square, symmetric matrices.

The development of this method is documented in a series of papers. Initial strategies focused on adjusting the diagonal elements to enforce unit diagonals, which proved insufficient (Dykstra, 1983). Early work in this area laid some of the foundational results but highlighted the need for methods to guarantee positive semidefiniteness (Vandenberghe & Boyd, 1996). Inspired by the principles outlined in works such as Dykstra's original paper on projection algorithms, the work by Vandenberghe and Boyd (1996) on semidefinite programming provides a foundational understanding of these techniques. The adaptation of semidefinite programming (SDP) represented a leap forward. Pioneering research in this domain showcased how SDP could address the dual constraints of unit diagonal and positive semidefiniteness, leveraging the computational power of interior-point methods (Dehghani et al., 2017). The introduction of the alternating projections algorithm represents a culmination of these insights, integrating optimization, spectral methods, and advanced projection techniques (Xu & Zikatanov, 2002). This algorithm further developed advancements in projection methods and provided a robust solution to the nearest correlation matrix problem.

The proposed algorithm is centered on the concept of iteratively projecting between two key sets: the set of symmetric positive semidefinite matrices (S), which is essential for maintaining the positive definiteness of the target matrix, and the set of matrices with unit diagonals (U), which ensures that the matrix exhibits the defining characteristics of a correlation matrix—symmetry with ones on the diagonal. The algorithm also relies on Dykstra's correction to address convergence issues that may arise from direct projection. The algorithm aims to

incrementally approximate the target matrix by ensuring that each projection step brings the current matrix closer to fulfilling the properties of a correlation matrix.

Projection onto Set *U*

The projection onto set *U* is defined as the set of symmetric matrices whose diagonal entries are all equal to 1. This set is particularly important in optimization problems and is used when adjusting matrices to have certain desirable properties, such as a correlation matrix, which by definition is a symmetric matrix with ones on the diagonal. Here, it is designed to adjust the diagonal elements of the matrix to unity while preserving the structure of the symmetric matrix, denoted as *A*. This is achieved by subtracting a diagonal matrix, where its entries are derived by solving the linear system:

$$(W^{-1} \circ W^{-1}) = diag(A - I) \tag{5}$$

where *W* is a symmetric, positive-definite matrix. This projection ensures the adjusted matrix has unit elements on its diagonal.

Projection onto Set *S*

The set *S* is defined as the set of all positive-definite, symmetric matrices denoted as $Y > 0$ for a symmetric matrix *Y*. The projection onto set *S* focuses on maintaining the positive definiteness of the matrix. The set is used in optimization problems, particularly those involving correlation matrices, which are a subset of the positive definite matrices with unit diagonal. This process is accomplished through the formula:

$$Ps(A) = W^{-1/2}((W^{1/2}AW^{1/2})_{+})W^{-1/2} \tag{6}$$

where $(\cdot)_+$ denotes the operation of retaining the positive semidefinite part of the matrix. This ensures that each projection step brings the matrix closer to A and satisfies the requirement of positive semidefiniteness.

The nearest correlation method iteratively executes the above two projection steps, applying Dykstra's correction after each step to ensure convergence. The essence of the process is to minimize the distance to the original matrix A while maintaining symmetry, positive semidefiniteness, and unit diagonal elements.

Use of the Frobenius Norm

Adopting the Frobenius norm as a distance metric to compare the smoothed VS non-smoothed correlation matrices became prevalent, due to its straightforward geometric interpretation and ease of computation (Havel, 1998). When comparing two matrices, the Frobenius norm of their difference (i.e., $||A - B||_F$) quantifies the distance between them in the space of matrices. Mathematically, the Frobenius norm of the difference between two matrices A and B is defined as:

$$||A - B||_F^2 = \sum_i \sum_j (a_{ij} - b_{ij})^2 \quad (7)$$

where a_{ij} and b_{ij} are the elements of matrices A and B respectively.

The Frobenius norm is utilized to quantify the "closeness" between the original matrix A and the resulting correlation matrix, i.e., the nearest correlation matrix. By minimizing the weighted Frobenius norm difference between A and the correlation matrix, the algorithm seeks the most accurate approximation of A that still retains the essential properties of a correlation

matrix. The choice of the Frobenius norm is pivotal because it simplifies the mathematical expression of the problem and, from a statistical standpoint, naturally reflects the sum of squared differences between matrix elements, offering an intuitive and effective method for measuring matrix discrepancies (Golub & Van Loan, 2013). In the minimization process, the algorithm considers a weighted Frobenius norm, allowing for a more flexible handling of uncertainties and variabilities present in the original data. Through this approach, the algorithm finds the correlation matrix closest to A and ensures that the resulting matrix has minimal deviation from the original data in a statistical sense.

In spite of the availability of the two approaches to alleviate the problem of non-positive definiteness, their use, and research regarding their use, is still somewhat lacking. Whereas Exploratory Factor Analysis (EFA) employs smoothing algorithms practically as a default, Confirmatory Factor Analysis (CFA) rarely sees its use. To address this gap, a Monte Carlo simulation study was conducted to evaluate the effectiveness of those two smoothing algorithms within the CFA framework. This study aims to assess the impact of such algorithms on parameter estimation, model fit, and efficiency. By systematically exploring the benefits and potential drawbacks of applying smoothing algorithms to CFA, the research anticipates uncovering valuable insights that could inform best practices of confirmatory factor models in various research scenarios.

METHODOLOGY

A Monte Carlo simulation was implemented to compare the two correlation smoothing approaches, namely cor.smooth() from the psych package (Revelle, version 2.3.12) and nearcor()

method (Rahman, 2018), by simulating sparse and small sample sizes (Lorenzo-Seva & Ferrando, 2021).

Data Generation

The simulations were conducted by employing a small sample size of $n = 100$ to induce sparseness in the contingency tables from which the tetrachoric correlations are calculated. The more sparseness in the data, the higher the chance that the estimation will be imprecise, generating a non-positive definite correlation matrix. Cohen's (1988) benchmark was used for the correlation metric for a "small effect size" (0.1) and a "large effect size" (0.5) across all the Factor model parameters (i.e., loadings λ and factor correlation ρ). This implied a 2x2 design with 1000 replications per condition ($R = 1000$).

Datasets were simulated using the lavaan package (Rosseel, 2012) following a correlated, two-factor model as shown in Figure 1. Each factor was measured by six binary items, resulting in a total of 12 items. To simulate variations in response patterns, these binary items were generated with specified thresholds of 1 and -1.5, applied iteratively.

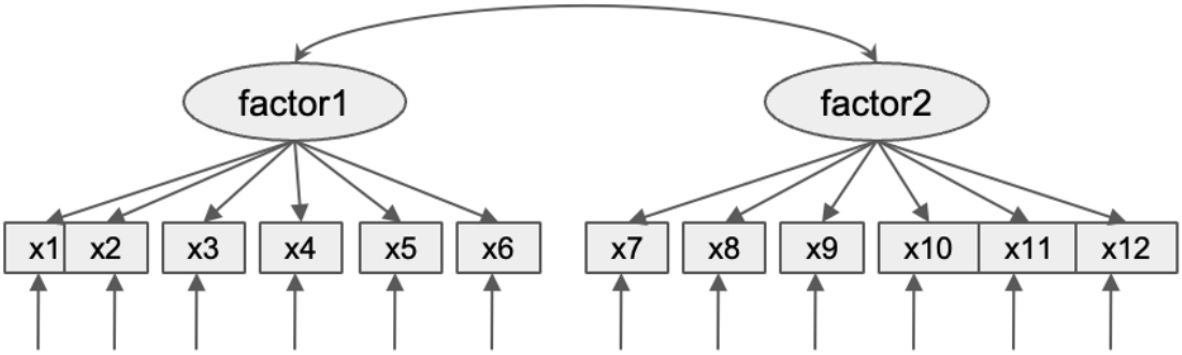

Figure 1: Two-factor model with 12 binary items. Factor 1 is associated with six items, x1 through x6, while factor 2 is linked to items, x7 through x12. Factor 1 and factor 2 are correlated..

Following data generation, confirmatory factor analysis (CFA) was performed using the true covariance matrix to get the weighted matrix. Subsequently, a polychoric correlation matrix was computed, and two smoothing methods were applied to it. Finally, CFA was conducted on the smoothed matrix from two different approaches for comparison.

Simulation outcomes

Convergence: Since small samples are known to induce issues of model non-convergence in SEM, the number of non-convergent models was recorded.

Bias: The raw bias of each simulation condition, $B(\theta_c)$ was calculated as:

$$B(\theta_c) = R^{-1}\Sigma_{r=1}^{R}(\hat{\theta}_{rc}) - \theta_c, \quad (9)$$

where $\hat{\theta}_{rc}$ is the parameter estimate for replication r in condition c, and R denotes the total number of replications. In this study, $R^{-1}\Sigma_{r=1}^{R}(\hat{\theta}_{rc})$ equals the mean of parameter estimates over 1000 replications, and θ_c is the true population parameter. The raw bias was compared in magnitude across simulation conditions and smoothing approaches regarding both loadings estimation and correlation between latent factors.

Residual: The Frobenius norm as a distance metric to measure the difference between the model-implied covariance matrices and the actual data covariance:

$$F_{Eigen} = ||\Sigma(\theta) - \Sigma_{Eigen}||_F \quad (10)$$

$$F_{Higham} = ||\Sigma(\theta) - \Sigma_{Higham}||_F \quad (11)$$

where $\Sigma(\theta)$ represents the true population covariance matrix, Σ_{Higham} represents the model-implied covariance matrix smoothed by Higham's nearest correlation method, and Σ_{Eigen} represents the model-implied covariance matrix from the 'eigenvalue substitution' method.

$$\overline{F_{Eigen}} = R^{-1}\Sigma_{r=1}^{R}F_{Eigen} \quad (12)$$

$$\overline{F_{Higham}} = R^{-1}\Sigma_{r=1}^{R}F_{Higham} \quad (13)$$

where $\overline{F_{Eigen}}$ and $\overline{F_{Higham}}$ are the average of F-norm. In this study, $R^{-1}\Sigma_{r=1}^{R}F_{Eigen}$ and $R^{-1}\Sigma_{r=1}^{R}F_{Higham}$ equal the mean of parameter estimates over 1000 replications.

Efficiency: Another criterion for a good estimator is efficiency which we define as:

$$V(\theta_c) = SD(\hat{\theta}_c) - R^{-1}\Sigma_{r=1}^{R}(SE(\hat{\theta}_{rc})) \quad (14)$$

where $SD(\hat{\theta})$ is the standard deviation of the empirical distribution for the parameter estimate, and $R^{-1}\Sigma_{r=1}^{R}(SE(\hat{\theta}_{rc}))$is the average standard error for parameter estimate $\hat{\theta}_c$ across R replications. $V(\theta_c)$ is the difference between $SD(\hat{\theta})$ and the average value of $SE(\hat{\theta})$.

Type I error rate: Since the correct model is being fitted to the data, the p-value associated with the χ^2 test of fit will be used to evaluate the empirical Type I error rate. This is measured as the proportion of p-values less than .05 across *R* replications.

Approximate model fit: Robust Comparative Fit Index (CFI) and Robust Root Mean Square Error of Approximation (RMSEA) are additional fit indices less sensitive to sample size. The CFI compares the specified model with a null model, assuming no relationships among the variables. Values closer to 1 indicate a better fit, with a threshold of 0.95 or above generally indicating a good model fit (Hu & Bentler, 1999). The RMSEA assesses how well the model, with unknown but optimally chosen parameter estimates, would fit the population's covariance matrix. Values of RMSEA less than 0.05 indicate a close fit (Browne & Cudeck, 1992). Since the correct model is being fitted to the data, we would expect CFI=1.0 and RMSEA=0.0. We will monitor the proportion of CFI and RMSEA values different from these values.

RESULTS

The results of this Monte Carlo study are presented as follows. First, raw biases of factor loadings for each item were presented across all simulation conditions, followed by the aggregated biases of factor loading across 12 items and simulation conditions. Second, the efficiency of the parameters was reported. Model fit, including Type I error and RMSEA and CFI, were summarized again according to the type of simulation conditions.

Convergence

No convergence rate issues were reported for either method. Approximately 15% of the cases were identified as Heywood cases (Kolenikov & Bollen, 2012), which may be attributed to the smoothing matrix used in the analysis differing from the original dataset.

Factor Loading Estimation

For the raw bias of the factor loadings, both methods exhibited comparable performance across all four simulation conditions. The aggregated results (Figure 6) revealed that raw biases remain consistently within the range of (-0.05, 0.05), indicating appropriate estimation even with a small sample and data sparseness. In the aggregated results (Figure 6), Higham's nearest correlation method exhibited less bias than the eigenvalue substitution method. However, it's noteworthy that the eigenvalue substitution function also performed well. A comparative analysis through unaggregated results (Figures 2-5) showed that each method exhibited its own merits under different conditions. The eigenvalue substitution function resulted in estimated values closer to their true values when compared to Higham's nearest correlation for specific items. Yet, for other items, Higham's nearest correlation showed less bias. This indicated that while one method may not universally outperform the other, both possess distinct advantages. The performance parity between the two methods may result from sampling variability. Secondly, another salient observation was the trend where Higham's nearest correlation method consistently yielded higher factor loading estimates than the eigenvalue substitution method across all examined conditions, both for the aggregated and unaggregated results. The range of differences between results from the two methods consistently manifested in the unaggregated results for all 12 items. Higham's nearest correlation produced a distribution around the population loading value in the unaggregated figures, exhibiting upward biases for some items

and downward biases for others. At the same time, the eigenvalue substitution predominantly showed downward biases. On the other hand, in the aggregated results, the biases of Higham's nearest correlation offset each other, resulting in precise and close-to-true-value performance.

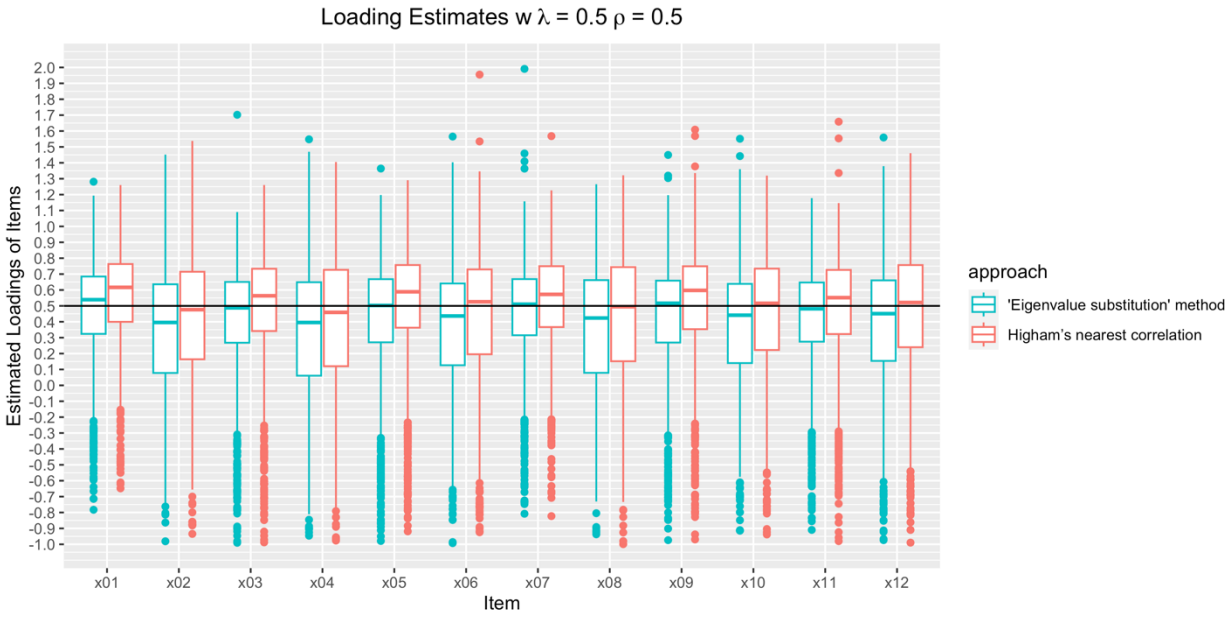

Figure 2 Loading Estimation w λ = 0.5 and ρ = 0.5. The image demonstrated loading estimations of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. The simulation condition here was λ = 0.5 and ρ = 0.5.

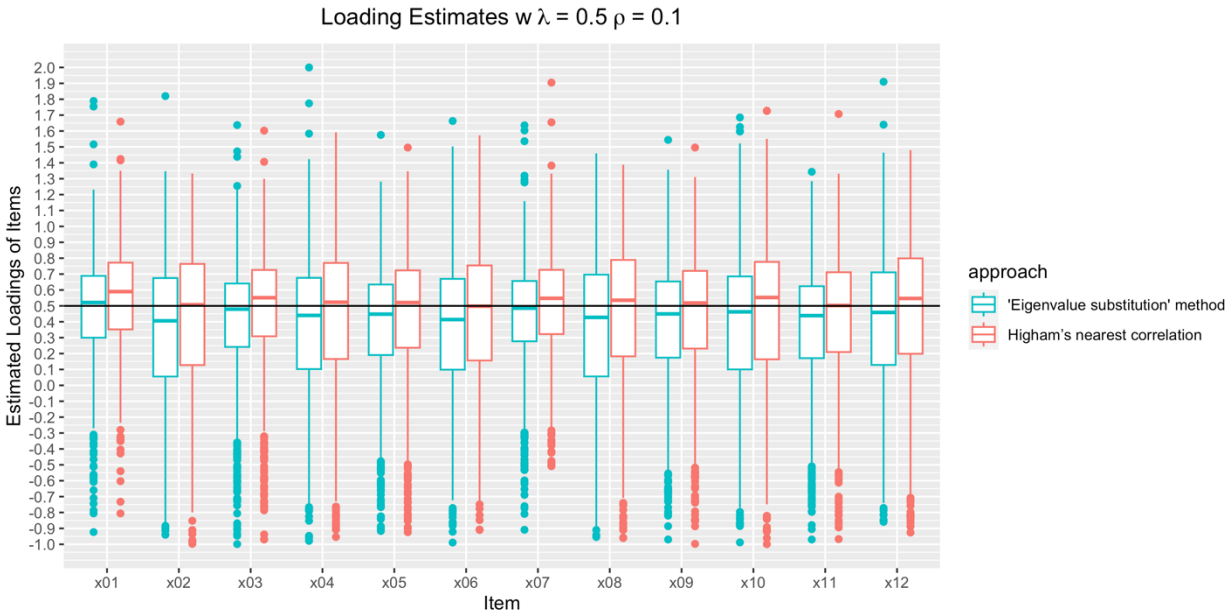

Figure 3 Loading Estimation w λ = 0.5 and ρ = 0.1. The image demonstrated loading estimations of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. The simulation condition here was λ = 0.5 and ρ = 0.1.

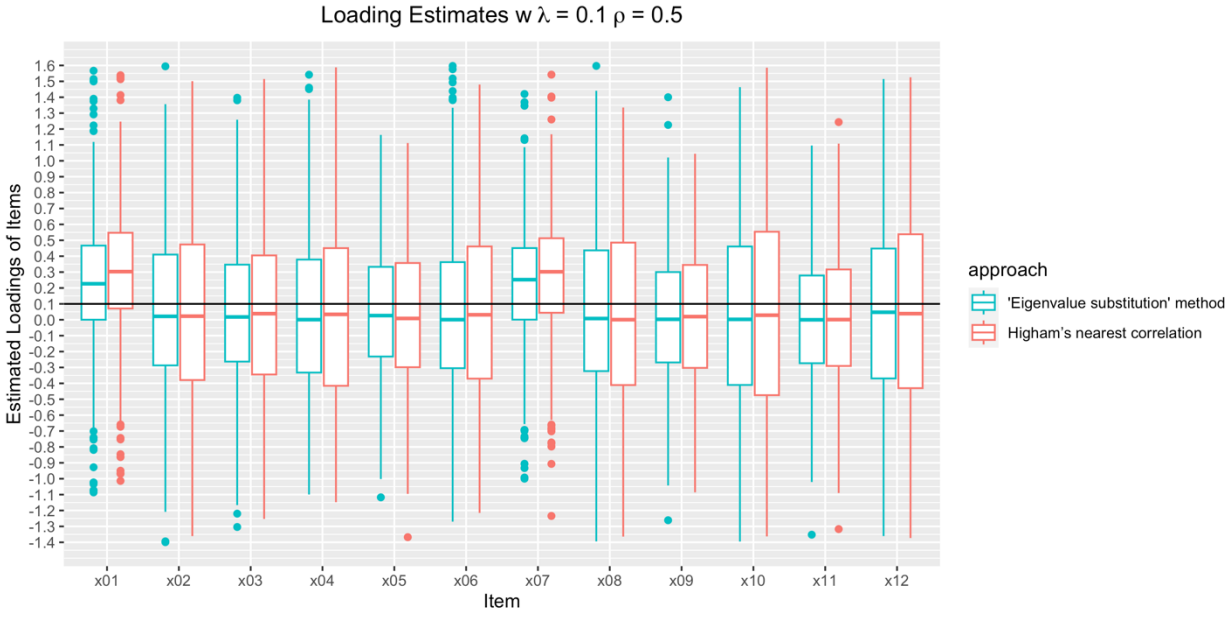

Figure 4 Loading Estimation w λ = 0.1 and ρ = 0.5. The image demonstrated loading estimations of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. The simulation condition here was λ = 0.1 and ρ = 0.5.

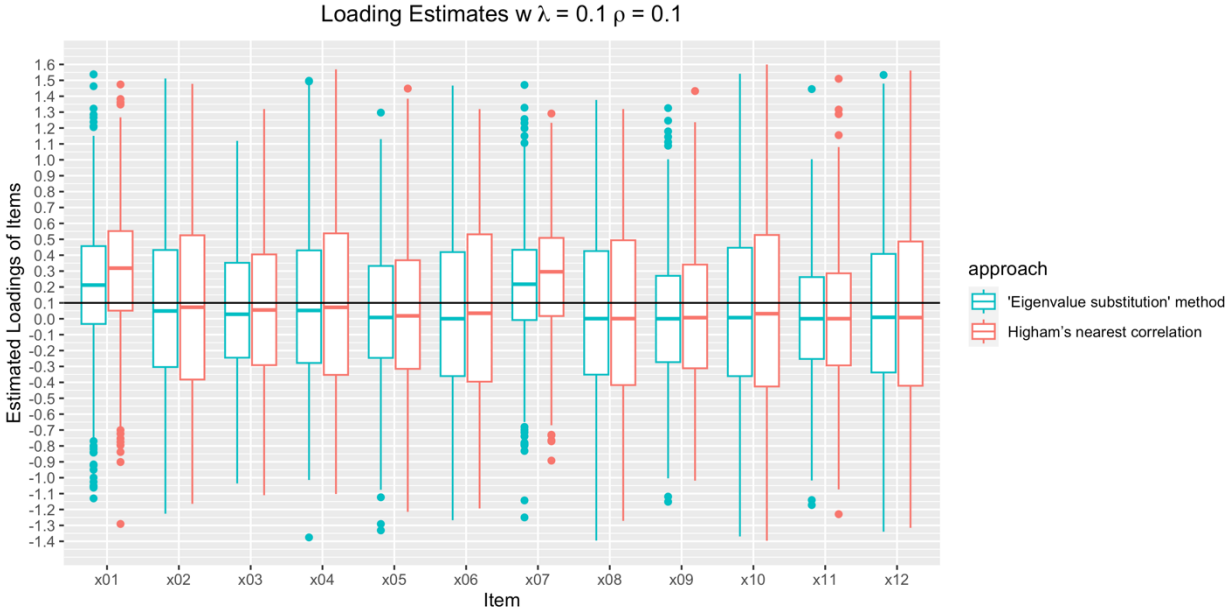

Figure 5 Loading Estimation w λ = 0.1 and ρ = 0.1. The image demonstrated loading estimations of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. The simulation condition here was λ = 0.1 and ρ = 0.1.

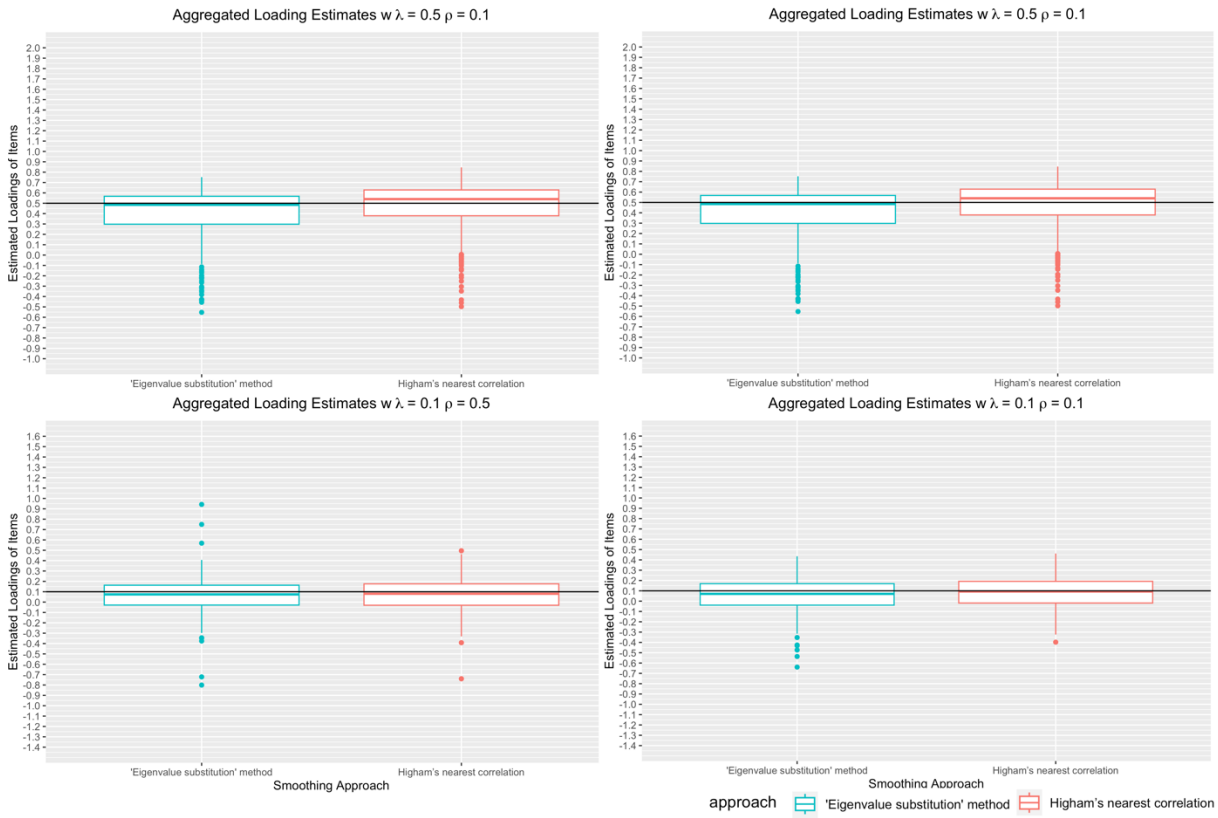

Figure 6: Aggregated Loading Estimation. The image demonstrated aggregated loading estimations for four scenarios across 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. Panel 1 presented a simulation scenario wherein the $\lambda = 0.5$ and $\rho = 0.5$. In Panel 2, the simulation maintains $\lambda = 0.5$, but adjusted the correlation $\rho = 0.1$. Panel 3 depicted a condition with the $\lambda = 0.1$ and $\rho = 0.1$. Finally, Panel 4 illustrated a simulation with the loading $\lambda = 0.1$ and $\rho = 0.5$.

Residual Correlation Matrix

In the current study, the Frobenius norm was used as a metric to quantify the divergence between model-implied covariance matrices smoothed by the algorithms and the actual covariance matrix from where the data was simulated. The method that produced a model-implied covariance matrix with the smallest Frobenius norm distance from the true covariance matrix was deemed to offer more accurate parameter estimates. The analysis conducted across four distinct simulation scenarios revealed that Higham's nearest correlation method demonstrated slightly superior performance, as detailed in Table 1.

Table 1. The table showed the Frobenius norm between the model-implied covariance matrices and the actual data covariance from the 'eigenvalue substitution' method and Higham's nearest correlation method over 1000 replications. When the simulation condition was $\lambda = 0.5$ and $\rho = 0.5$. The average f-norm was 5.99 for Higham's nearest correlation method and 6.14 for the 'eigenvalue substitution' method. When the simulation condition was $\lambda = 0.5$ and $\rho = 0.1$. The average f-norm was 6.00 for Higham's nearest correlation method and 6.13 for the 'eigenvalue substitution' method. When the simulation condition was $\lambda = 0.1$ and $\rho = 0.5$. The average f-norm was 5.82 for Higham's nearest correlation method and 5.92 for the 'eigenvalue substitution' method. When the simulation condition was $\lambda = 0.1$ and $\rho = 0.1$. The average f-norm was 5.72 for Higham's nearest correlation method and 5.88 for the 'eigenvalue substitution' method.

Loading λ	**Correlation of Factors ρ**	**Higham's nearest correlation method**	**Eigenvalue substitution method**
0.5	0.5	5.99	6.14
	0.1	6.00	6.13
0.1	0.5	5.82	5.92
	0.1	5.72	5.88

Factor Correlation Estimation

Figure 7 showcases the factor correlation estimates across the four simulation conditions. Both the eigenvalue substitution and Higham's nearest correlation methods yielded correlation estimates close to the simulation conditions, as indicated by the central black lines. This demonstrated a general accuracy in the estimation process across both methods. However, the distribution of these estimates, particularly the spread and presence of outliers, offered a deeper insight into each method's sensitivity to the underlying conditions. When comparing the scenarios with the simulation condition of a loading of 0.5 ($\lambda = 0.5$) against those with a simulation condition of a loading of 0.1 ($\lambda = 0.1$), a downward shift in the estimated values was observed for both methods. The average correlation estimates exhibit a downward bias as the loading decreases from $\lambda = 0.5$ to $\lambda = 0.1$. Notably, the variance of the estimated values increases with a decrease in loading size ($\lambda = 0.5$ to $\lambda = 0.1$). This was apparent in the elongation of the boxplots and the growing number of outliers for both 'eigenvalue substitution' and Higham's nearest correlation, regardless of whether the simulation condition for the correlation was low or high. This trend points to a decrease in the precision of the factor

correlation estimates as the simulation condition of loading decreases from $\rho = 0.5$ to $\rho = 0.1$, which could imply that the estimator's efficiency decreases under these conditions. Upon closer inspection, the 'eigenvalue substitution' method showed a tighter distribution around the population values. In fact, when the correlation and loadings were low ($\lambda = 0.1$ *and* $\rho = 0.1$), the Higham's nearest correlation method appeared to produce a wider spread of estimates.

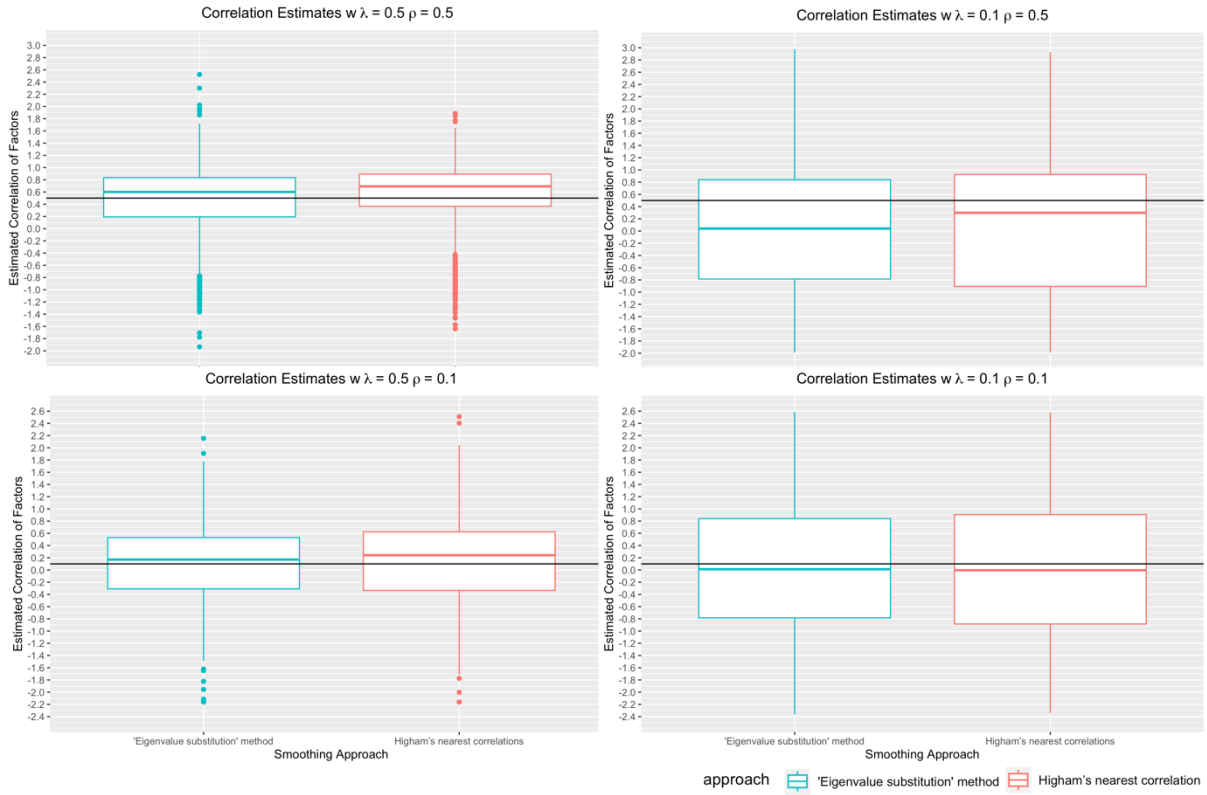

Figure 7 Correlation Estimation. The image demonstrated correlation estimations for four scenarios for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. Panel 1 presented a simulation scenario wherein $\lambda = 0.5$ and $\rho = 0.5$. *In Panel 2, the simulation maintains the correlation between the latent factors* $\rho = 0.5$ but adjusted *loading* $\lambda = 0.1$. *Panel 3 depicted a condition with* $\lambda = 0.1$ and $\rho = 0.1$. *Finally, Panel 4 illustrated a simulation with* $\lambda = 0.5$ and $\rho = 0.1$.

To better measure the difference between Higham's nearest correlation and the Eigenvalue substitution method, the mean values of the estimated correlations in the range of [-1, 1] were reported in Table 2. In this table, as the loading condition decreased from $\lambda = 0.5$ to

$\lambda = 0.1$, the average factor correlation estimate by Higham's nearest correlation method showed a significant decrease, moving from a positive correlation to no correlation or even negative. Conversely, the Eigenvalue substitution method maintained more consistent estimates. Under the condition of high inter-factor correlation and high loadings ($\lambda = 0.5$ and $\rho = 0.5$), Higham's nearest correlation method performed better than in the low loading condition ($\lambda = 0.1$) . For the low correlation condition ($\rho = 0.1$), Higham's nearest correlation performance was notably worse, providing less accurate estimates. The Eigenvalue substitution method, while not as accurate as when the loading and factor correlation conditions were high, showed less variability and bias under the low loading condition. Both methods struggled with accuracy at lower loadings ($\lambda = 0.1$), especially when the simulation condition of correlation between latent factors was high ($\rho = 0.5$).

Table 2 Correlation Estimation. The table showed the average correlation estimation across replications from the 'eigenvalue substitution' method and Higham's nearest correlation method over 1000 replications. The loading parameters in the simulation were iteratively set at $\lambda = 0.5$, and $\lambda = 0.1$, respectively. Furthermore, the simulations assumed a correlation of latent factors was iteratively set $\rho = 0.5$ and $\rho = 0.1$, respectively.

Correlation of Factors ρ	**Loading λ**	**Higham's nearest correlation method**	**Eigenvalue substitution method**
0.5	0.5	0.47	0.40
	0.1	-0.01	0.09
0.1	0.5	0.12	0.10
	0.1	-0.04	0.01

Efficiency

Tables 3 to 6 showcase the results for the efficiency of the two methods. First, when the simulation condition of loading decreased from $\lambda = 0.5$ to $\lambda = 0.1$, there was a general increase in the standard error (SE) for both methods, which indicated a decrease in the estimation

efficiency. This pattern is consistent across all levels of correlation in simulation conditions ($\rho = 0.5$ and $\rho = 0.1$). Second, Higham's nearest correlation method consistently showed a lower standard error than the Eigenvalue substitution method. However, when the simulation condition of loading was low ($\lambda = 0.1$), the performance gap between the two methods grew, with Higham's nearest correlation still performing better; that said, the Average SE and SD were very close. Third, there were noticeable outliers, especially in Tables 5 and 6, where the standard error for the Eigenvalue substitution method significantly increased for certain variables (e.g., X1, X6 in Table 5, and X3, X6, X8, and X10 in Table 6). These numbers were marked in yellow, highlighting the instances where the Eigenvalue substitution method's efficiency was substantially lower. In addition, across all tables, the difference between the Average SE and SD (standard deviation) appeared relatively consistent within each method. All four simulated conditions exhibited larger SEs when compared to the SD of the empirical distribution of the parameter estimates.

Table 3 Efficiency w $\lambda = 0.5$ and $\rho = 0.5$. *The table demonstrated the efficiency analysis of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included average SE across all replications, true SD, and their differences. The simulation condition here was* $\lambda = 0.5$ and $\rho = 0.5$.

	Higham's nearest correlation			**Eigenvalue substitution**		
	Average SE	SD	Difference	Average SE	SD	Difference
X1	0.15	0.06	0.09	0.17	0.08	0.09
X2	0.16	0.01	0.14	0.17	0.01	0.15
X3	0.13	0.04	0.09	0.15	0.08	0.08
X4	0.13	0.01	0.12	0.15	0.01	0.14
X5	0.12	0.01	0.11	0.14	0.04	0.10
X6	0.13	0.01	0.11	0.15	0.05	0.09
X7	0.81	0.06	0.74	0.62	0.09	0.52
X8	0.12	0.01	0.10	0.13	0.01	0.12
X9	0.13	0.01	0.11	0.14	0.01	0.13
X10	0.12	0.01	0.11	0.15	0.08	0.07
X11	0.13	0.01	0.12	0.15	0.01	0.13
X12	0.13	0.08	0.05	0.16	0.08	0.08

Table 4 Efficiency w $\lambda = 0.5$ and $\rho = 0.1$. The table demonstrated the efficiency analysis of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included average SE across all replications, true SD, and their differences. The simulation condition here was $\lambda = 0.5$ and $\rho = 0.1$.

	Higham's nearest correlation			**Eigenvalue substitution**		
	Average SE	SD	Difference	Average SE	SD	Difference
X1	0.16	0.05	0.10	0.19	0.11	0.08
X2	0.17	0.02	0.16	0.18	0.09	0.08
X3	0.15	0.09	0.06	0.16	0.05	0.11
X4	0.14	0.05	0.09	0.16	0.01	0.15
X5	0.13	0.01	0.12	0.15	0.04	0.12
X6	0.14	0.01	0.12	0.17	0.01	0.15
X7	0.16	0.05	0.10	0.19	0.05	0.14
X8	0.13	0.02	0.12	0.15	0.01	0.14
X9	0.14	0.01	0.13	0.16	0.04	0.12
X10	0.13	0.02	0.12	0.17	0.07	0.10
X11	0.14	0.01	0.13	0.16	0.01	0.15
X12	0.13	0.01	0.12	0.16	0.09	0.08

Table 5 Efficiency w $\lambda = 0.1$ and $\rho = 0.5$. The table demonstrated the efficiency analysis of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included average SE across all replications, true SD, and their differences. The simulation condition here was $\lambda = 0.1$ and $\rho = 0.5$.

	Higham's nearest correlation			**Eigenvalue substitution**		
	Average SE	SD	Difference	Average SE	SD	Difference
X1	0.16	0.07	0.09	3.04	0.12	2.92
X2	0.17	0.02	0.15	0.46	0.08	0.37
X3	0.15	0.07	0.08	0.66	0.10	0.56
X4	0.15	0.09	0.06	0.19	0.13	0.07
X5	0.14	0.01	0.13	0.18	0.01	0.16
X6	0.14	0.06	0.09	5.16	0.08	5.08
X7	0.14	0.01	0.13	0.37	0.05	0.32
X8	0.36	0.08	0.28	0.17	0.15	0.03
X9	0.14	0.08	0.05	0.15	0.01	0.14
X10	0.15	0.09	0.05	0.18	0.10	0.08
X11	0.13	0.01	0.12	0.18	0.01	0.16
X12	0.16	0.06	0.10	0.26	0.14	0.12

Table 6 Efficiency w $\lambda = 0.1$ and $\rho = 0.1$. *The table demonstrated the efficiency analysis of 12 binary items for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included average SE across all replications, true SD, and their differences. The simulation condition here was* $\lambda = 0.1$ and $\rho = 0.1$.

	Higham's nearest correlation			**Eigenvalue substitution**		
	Average SE	SD	Difference	Average SE	SD	Difference
X1	0.15	0.01	0.14	0.22	0.10	0.12
X2	0.18	0.05	0.12	0.24	0.08	0.16
X3	0.81	0.12	0.69	0.20	0.14	0.05
X4	0.16	0.02	0.14	0.18	0.06	0.12
X5	0.14	0.04	0.10	0.16	0.01	0.15
X6	0.15	0.02	0.13	13.70	0.08	13.6
X7	0.15	0.10	0.04	0.24	0.13	0.11
X8	0.15	0.05	0.11	5.60	0.08	5.52
X9	0.14	0.08	0.06	0.16	0.07	0.09
X10	2.20	0.09	2.10	4.72	0.13	4.58
X11	0.13	0.01	0.12	0.16	0.07	0.09
X12	0.16	0.10	0.06	0.20	0.13	0.07

Fit Assessment

Tables 7 to 10 show the two methods did not fit the model well. However, the eigenvalue substitution method provided better-fit statistics than the Higham's nearest correlation method across different simulation conditions. A Type I error rate inflation of about 10% to 20% was reported when using the eigenvalue substitution method, while the inflation went over 50% for the case of Higham's nearest correlation method. The worst Type I error inflation happened for this method, when the factor correlation was low ($\rho = 0.1$), and the loadings were large ($\lambda = 0.5$). When the loadings were small ($\lambda = 0.1$), the correlation condition did not impact the Type I error rate. Those scenarios tended to have the least Type I error inflation among the four simulation conditions.

Robust CFI and RMSEA behaved poorly for those two methods but the 'eigenvalue substitution method' outperformed Higham's method slightly. Each smoothing method's

performance changed as a function of the simulation condition of loading and correlation values. Lower simulation conditions of loadings ($\lambda = 0.1$) tended to have detrimental effects on model fit. However, when the simulation condition of loading was small ($\lambda = 0.1$), the decrease ($\rho = 0.5$ to $\rho = 0.1$) in the correlation condition did not change much.

Using Table 7 as an example, the simulation condition for loading was set at 0.5 ($\lambda = 0.5$), with a factor correlation of 0.5 ($\rho = 0.5$). For Higham's nearest correlation method, the average Robust CFI was 0.77, with only 7.69% of 1000 replications exceeding the threshold of 0.95. In contrast, the eigenvalue substitution method yielded a mean Robust CFI of 0.78, with only 11.85% of replications surpassing the recommended threshold of 0.95. Regarding the Robust RMSEA values for Higham's method, the mean was 0.09, with only 8.81% of replications falling below 0.05. Similarly, the eigenvalue substitution method had an average Robust RMSEA of 0.068, with only 22.27% replications under 0.05. When considering P-values, 56.6% were less than 0.05 for Higham's method, as opposed to 21.8% for the eigenvalue substitution method, indicating a noticeable inflation in the Type I error rates.

Table 7 Fit w $\lambda = 0.5$ and $\rho = 0.5$ *with Criteria: Robust CFI > 0.95, Robust RMSEA < 0.05, and P-value < 0.05. The table demonstrated the fit assessment for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included the expected value and proportions of robust CFA, robust RMSEA, and P-value. The simulation condition here was* $\lambda = 0.5$ and $\rho = 0.5$.

	Higham's nearest correlation		**Eigenvalue substitution method**	
	Mean	Cut Off	Mean	Cut Off
Robust CFI	0.77	7.69%	0.78	11.85%
Robust RMSEA	0.09	8.81%	0.068	22.27%
Type I error	56.6%		21.8%	

Table 8 Fit w $\lambda = 0.5$ and $\rho = 0.1$ with Criteria: Robust CFI > 0.95, Robust RMSEA < 0.05, and P-value < 0.05. The table demonstrated the fit assessment for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included the expected value and proportions of robust CFA, robust RMSEA, and P-value. The simulation condition here was $\lambda = 0.5$ and $\rho = 0.1$.

	Higham's nearest correlation		**Eigenvalue substitution method**	
	Mean	Cut Off	Mean	Cut Off
Robust CFI	0.72	2.85 %	0.74	6.55%
Robust RMSEA	0.10	4.78%	0.068	18.73%
Type I error	61.1%		23.8%	

Table 9 Fit w $\lambda = 0.1$ and $\rho = 0.5$ with Criteria: Robust CFI > 0.95, Robust RMSEA < 0.05, and P-value < 0.05. The table demonstrated the fit assessment for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included the expected value and proportions of robust CFA, robust RMSEA, and P-value. The simulation condition here was $\lambda = 0.1$ and $\rho = 0.5$.

	Higham's nearest correlation		**Eigenvalue substitution method**	
	Mean	Cut Off	Mean	Cut Off
Robust CFI	0.67	1.65%	0.70	6.74%
Robust RMSEA	0.09	7.12%	0.06	22.63%
Type I error	53.8%		14.1%	

Table 10 Fit w $\lambda = 0.1$ and $\rho = 0.1$ with Criteria: Robust CFI > 0.95, Robust RMSEA < 0.05, and P-value < 0.05. The table demonstrated the fit assessment for the 'eigenvalue substitution' method and Higham's nearest correlation over 1000 replications. It included the expected value and proportions of robust CFA, robust RMSEA, and P-value. The simulation condition here was $\lambda = 0.1$ and $\rho = 0.1$.

	Higham's nearest correlation		**Eigenvalue substitution method**	
	Mean	Cut Off	Mean	Cut Off
Robust CFI	0.67	1.65 %	0.70	7.17%
Robust RMSEA	0.09	6.5 %	0.06	22.89%
Type I error	53.6%		14.2%	

Conclusion

In this Monte Carlo simulation, a thorough investigation was conducted to compare the eigenvalue substitution method and Higham's nearest correlation method for smoothing non-positive definite correlation matrices within the context of confirmatory factor analysis (CFA) for ordinal data. This study provided a detailed assessment of the conditions under which each method demonstrated superior performance.

The analysis focused on the accuracy of parameter estimation, computational efficiency, and model fit assessment. It was found that both methods effectively estimated factor loadings across a range of simulated conditions, maintaining biases within acceptable limits. This indicates their robustness in accurately estimating factor loadings, even at small sample sizes or sparse data. Results using the Frobenius norm showed that Higham's nearest correlation method more accurately reflected the true data-generating covariance structure on average, as evidenced by lower average Frobenius norm values compared to the eigenvalue substitution method.

The analysis also explored both methods' accuracy in estimating factor correlations, finding that both methods generally produced accurate estimates under various conditions. Nevertheless, Higham’s nearest correlation method exhibited a smaller bias in correlation estimation, suggesting better parameter estimation despite minor differences. Notably, the eigenvalue substitution method demonstrated greater stability in scenarios of low loadings and low factor correlations, indicating its robustness in more challenging contexts. Conversely, Higham’s method showed potential limitations in stability, as evidenced by wider spreads of estimates under similar conditions.

Although Higham’s method yielded less bias in parameter estimation, the eigenvalue substitution method consistently showcased better fit statistics across various conditions,

highlighting its efficacy in maintaining slightly lower Type I error rates and achieving superior fit indices under different loading and factor correlation scenarios. Both the robust Comparative Fit Index (CFI) and the robust Root Mean Square Error of Approximation (RMSEA) metrics indicated challenges in achieving optimal model fit, suggesting further research is needed to refine these methods or explore alternative strategies. Generally speaking, though, neither method could be considered optimal as far as model fit assessment is concerned.

The findings revealed that neither method offers a universal superior solution for correcting non-positive definite matrices in confirmatory factor analysis for ordinal variables. Higham's nearest correlation method provides more precise parameter estimation, crucial for accurate theoretical interpretations. Conversely, the eigenvalue substitution method achieved better-fit indices, demonstrating adaptability and effectiveness across varied conditions. These differences in performance emphasized the importance of selecting a method based on the specific objectives and conditions of the confirmatory factor analysis, balancing the need for parameter estimation precision against the fit of the statistical model to the data.

Limitation and Future Direction

Robust corrections within SEM rely on the weight matrix of asymptotic covariances of the model. Throughout this simulation study we noticed that, since the smoothing algorithm alters the original polychoric correlation matrix from which this weight matrix is derived, it no longer corresponds to the model-implied covariance matrix, which may explain the poor performance in terms of fit. Future research should aim to understand how the weight matrix is influenced by the smoothing algorithm. Also, it is known that the chi-square test of fit only converges to its

distribution with large samples. Exploring how to induce data sparseness while maintaining some semblance of a "large sample" will help disentangle the effect that smoothing algorithms and small samples have in the asymptotic distribution of the fit indices. Last but not least, other important models such as growth mixture models or bifactor models which are more complex should be studied to further our understanding of how smoothing algorithms can aid in the estimation and interpretation of SEM.

References

Al-Homidan, S., & AlQarni, M. (2012). Structure methods for solving the nearest correlation matrix problem. *Positivity, 16,* 497-508. doi: 10.1007/s11117-012-0180-x

Bhatia, R. (2009). Positive definite matrices. In *Positive Definite Matrices.* Princeton university press. doi: 10.1177/0049124192021002005

Bollen, K. A., & Long, J. S. (Eds.). (1993). *Testing structural equation models* (Vol. 154). Sage.

Browne, M. W., & Cudeck, R. (1992). Alternative ways of assessing model fit. *Sociological methods & research, 21*(2), 230-258. doi: 10.1177/0049124192021002

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences (2nd ed.).* Routledge. doi: 10.4324/9780203771587

Dehghani, A., Goffin, J. L., & Orban, D. (2017). A primal–dual regularized interior-point method for semidefinite programming. *Optimization Methods and Software, 32*(1), 193-219. doi: 10.1080/10556788.2016.1235708

Dykstra, R. L. (1983). An algorithm for restricted least squares regression. *Journal of the American Statistical Association, 78*(384), 837-842. doi: 10.1080/01621459.1983.10477029

Ekström, J. (2011). A generalized definition of the polychoric correlation coefficient. *UCLA: Department of Statistics, UCLA*. Retrieved from https://escholarship.org/uc/item/583610fv

Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations.* JHU press.

Havel, T. F. (2002). Distance geometry: Theory, algorithms, and chemical applications. *Encyclopedia of Computational Chemistry, 120,* 723-742. doi: 10.1002/0470845015.cda018

Higham, N. J. (2002). Computing the nearest correlation matrix—a problem from finance. *IMA journal of Numerical Analysis, 22*(3), 329-343. doi: 10.1093/imanum/22.3.329

Holgado–Tello, F. P., Chacón–Moscoso, S., Barbero–García, I., & Vila–Abad, E. (2010). Polychoric versus Pearson correlations in exploratory and confirmatory factor analysis of ordinal variables. *Quality & Quantity, 44,* 153-166. doi: 10.1007/s11135-008-9190-y

Hu, L. T., & Bentler, P. M. (1999). Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural equation modeling: a multidisciplinary journal, 6*(1), 1-55. doi: 10.1080/10705519909540118

Jöreskog, K. G., & Sörbom, D. (1996). *LISREL 8: User's reference guide.* Scientific Software International

Kolenikov, S., & Bollen, K. A. (2012). Testing negative error variances: Is a Heywood case a symptom of misspecification?. *Sociological Methods & Research, 41*(1), 124-167. doi: 10.1177/0049124112442138

Lorenzo-Seva, U., & Ferrando, P. J. (2021). Not positive definite correlation matrices in exploratory item factor analysis: Causes, consequences and a proposed solution. *Structural Equation Modeling: A Multidisciplinary Journal, 28*(1), 138-147. doi: 10.1080/10705511.2020.1735393

Mueller, R. O., & Hancock, G. R. (2015). Factor analysis and latent structure analysis: confirmatory factor analysis. doi: 10.1016/B978-0-08-097086-8.25009-5

Maechler, M., Stahel, W., Ruckstuhl, A., Keller, C., Halvorsen, K., Hauser, A., & Buser, C. (2024). Package ‘sfsmisc’. https://github.com/mmaechler/sfsmisc

Olsson, U. (1979). On the robustness of factor analysis against crude classification of the observations. *Multivariate behavioral research, 14*(4), 485-500. doi: 10.1207/s15327906mbr1404_7

Rahman, A. (2018). sdpt3r: Semidefinite Quadratic Linear Programming in R. *R Journal, 10*(2). doi: 10.32614/CRAN.package.sdpt3r

Rencher, A. C., & Christensen, W. F. (2002). Méthods of multivariate analysis. a john wiley & sons. *Inc. Publication, 727,* 2218-0230.

Revelle, W., & Revelle, M. W. (2015). Package 'psych'. *The comprehensive R archive network, 337*(338).

Robitzsch, A. (2020, October). Why ordinal variables can (almost) always be treated as continuous variables: Clarifying assumptions of robust continuous and ordinal factor analysis estimation methods. In *Frontiers in education* (Vol. 5, p. 589965). Frontiers Media SA. doi: 10.3389/feduc.2020.589965

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of statistical software, 48,* 1-36. doi: 10.18637/jss.v048.i02

Vandenberghe, L., & Boyd, S. (1996). Semidefinite programming. *SIAM review, 38*(1), 49-95. doi: 10.1137/103800

Wackerly, D. D., Mendenhall, W., & Scheaffer, R. L. (2008). *Mathematical statistics with applications* (Vol. 7). Belmont, CA: Thomson Brooks/Cole.

Wothke, W. (1993). Nonpositive definite matrices in structural modeling. *Sage Focus Editions, 154,* 256-256.

Xu, J., & Zikatanov, L. (2002). The method of alternating projections and the method of subspace corrections in Hilbert space. *Journal of the American Mathematical Society, 15*(3), 573-597. doi: 10.1090/S0894-0347-02-00398-3