UNIVERSITY OF CALIFORNIA

Los Angeles

A Correlation Thresholding Algorithm for Learning Factor Analysis Models

A thesis submitted in partial satisfaction
of the requirements for the degree Master of Science
in Statistics

by

Dale S. Kim

2020

ABSTRACT OF THE THESIS

A Correlation Thresholding Algorithm
for Learning Factor Analysis Models

by

Dale S. Kim
Master of Science in Statistics
University of California, Los Angeles, 2020
Professor Qing Zhou, Chair

We consider the problem of learning the structure of the factor analysis model. The traditional method of Exploratory Factor Analysis (EFA), despite its widespread application, is often criticized for its ad-hoc use of rotation criteria for learning solutions. Additionally, more recently developed penalized EFA methods partially address these issues, but remain computationally intense. We propose a fast correlation thresholding algorithm, that is theoretically motivated by graph theory, to simultaneously learn the structure of a factor analysis model for an unknown number of factors. We derive the conditions for structural identifiability and parameter uniqueness, as well as show asymptotic consistency for our algorithm. Finally, we present a simulation study and real data example to test and demonstrate its performance.

The thesis of Dale S. Kim is approved.

Steven Paul Reise

Peter M. Bentler

Qing Zhou, Committee Chair

University of California, Los Angeles

2020

Hallo o shet.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Factor analysis is a commonly used multivariate technique which conceptualizes a set of observed variables as a function of a set of unobserved latent factors. It is generally assumed that the number of latent factors is less than the number of observed variables, hence serving as a dimension simplification procedure. Many social sciences use factor analysis to relate observed variables to hypothetical constructs that cannot be directly observed. These may include personality, emotional states, social status, or political power [1].

1.1 Model and Notation

The factor analysis model is a causal model of the form:

$$X = \Lambda L + \epsilon, \tag{1.1}$$

where $X = \begin{bmatrix} X_1 & \dots & X_p \end{bmatrix}^T \in \mathbb{R}^{p \times 1}$ is a vector of observed variables, $L = \begin{bmatrix} L_1 & \dots & L_d \end{bmatrix}^T \sim \mathcal{N}_d(0, \Phi)$ is a vector of latent variables or factors, $\epsilon = \begin{bmatrix} \epsilon_1 & \dots & \epsilon_p \end{bmatrix}^T \sim \mathcal{N}_p(0, \Omega)$ is a vector of errors and Ω is diagonal, and $\Lambda = [\lambda_{ij}] \in \mathbb{R}^{p \times d}$ is a matrix of coefficients, or factor loadings. For convenience, an additive mean vector μ is omitted from the model without the loss of generality. The associated path model can be illustrated by Figure 1.1. We assume that $d < p$, reflecting the fact that factor analysis is generally used as a dimension simplification technique. Further, the only causal relations that are assumed exist are

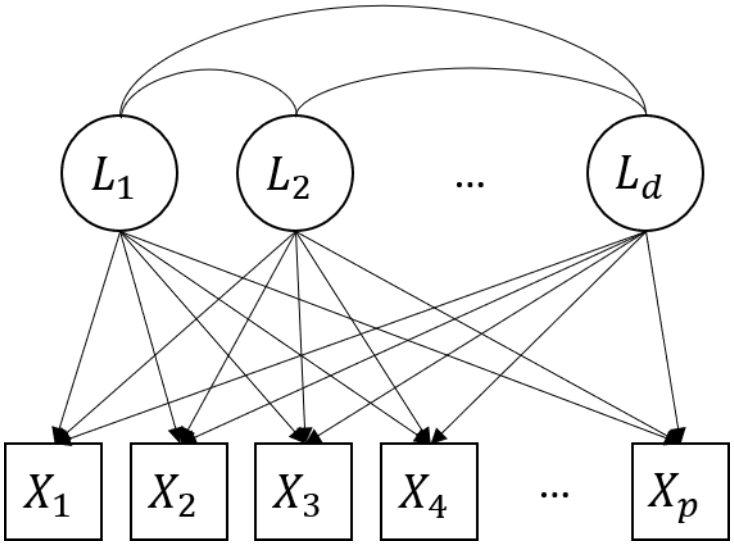

Figure 1.1: A path diagram for a general factor analysis model. Single arrow edges denote a causal relation. Non-arrow edges denote a correlation.

those from L to X. Thus we may say that L are the (causal) *parents* of X, and X are the *children* of L. No causal relations are assumed among the L variables, and they are only assumed to be correlated (oblique factor analysis) or uncorrelated (orthogonal factor analysis). We are considering the more general case of oblique factor analysis models in this study.

The model stated in Equation 1.1 implies a covariance structure Σ for X as follows:

$$\Sigma(\theta) \coloneqq \mathrm{Var}(X) = \mathrm{Var}(\Lambda L + \epsilon) = \Lambda\Phi\Lambda^T + \Omega, \tag{1.2}$$

letting $\theta = \{\Lambda, \Phi, \Omega\}$. We can write $\Sigma(\theta)$ to make explicit that we are referring to Σ as a function of the parameters $\Lambda, \Phi, and\Omega$.

At times, it will be easier to deal with observed variables which are unit variance scaled. Let D_σ be the Cholesky factor of the diagonal of Σ (i.e., the diagonal matrix of standard deviations). Then we define a unit variance scaled X as $\widetilde{X}$ in the following manner:

$$\widetilde{X} \coloneqq D_\sigma^{-1} X = D_\sigma^{-1}(\Lambda L + \epsilon) = \widetilde{\Lambda} L + \widetilde{\epsilon}, \tag{1.3}$$

where $D_\sigma^{-1}\Lambda = \widetilde{\Lambda}$ and $D_\sigma^{-1}\epsilon = \widetilde{\epsilon}$. Similarly, it follows that the population correlation

matrix $\widetilde{\Sigma}$ can be expressed as:

$$\begin{aligned}\widetilde{\Sigma}(\theta) &:= D_\sigma^{-1}\Sigma D_\sigma^{-1} \\ &= D_\sigma^{-1}(\Lambda\Phi\Lambda^T + \Omega)D_\sigma^{-1} \\ &= \widetilde{\Lambda}\Phi\widetilde{\Lambda}^T + \widetilde{\Omega},\end{aligned} \tag{1.4}$$

where $\widetilde{\Omega} = D_\sigma^{-1}\Omega D_\sigma^{-1}$. Note that the factor analysis model for Σ and $\widetilde{\Sigma}$ are often used interchangeably, and the elements of $\widetilde{\Sigma}(\theta)$ may be referred to as ρ_{ij}.

For estimation, maximum likelihood is the most widely used method. The Gaussian likelihood is particularly convenient since it can be directly parameterized in terms of the covariance:

$$\ell(\theta) = \frac{n}{2}\log|\Sigma(\theta)^{-1}| - \frac{n}{2}\text{tr}(\Sigma(\theta)^{-1}S), \tag{1.5}$$

where S is the sample covariance matrix of the observed variables X. From here, the maximum likelihood estimates are obtained by optimizing $\ell(\theta)$ with respect to θ. This function can also be augmented with penalty terms to promote sparsity in Λ, which we will briefly review in the next section.

The rest of this article is organized as follows. We first review traditional and recent methods of learning factor analysis structures in Section 1.2. We then delineate some common problems among current methods and provide the motivation of the current research in Section 1.3. Then we describe our Correlation Threshold Algorithm and develop theoretical justifications for its use in Section 2. We then test our Correlation Threshold Algorithm against other methods with a simulation study in Section 3 and a real data example in section 4. Finally, in Section 5, we provide some concluding remarks.

Notation throughout this article will be as follows. Let $A \subseteq \{1, \ldots, n\}$ and $B \subseteq \{1, \ldots, p\}$ be index sets. The complement of A will be denoted as A^c. For a matrix $M \in \mathbb{R}^{n\times p}$, we will define M_{AB} to be the submatrix of M consisting of the rows indexed by A and columns indexed by B. Similarly for a vector $V \in \mathbb{R}^{n\times 1}$, we will define V_A to be the subvector of V consisting of the entries indexed by A. We will use $\mathbf{0}$ or blank entries

to represent a rectangular matrix or vector of zeroes, whose dimension can be inferred from context and I_n will denote the $n \times n$ identity matrix.

1.2 Review of Structure Learning in Factor Analysis

Structure learning in the context of factor analysis typically refers to constraints imposed on Λ. We are interested in sparse structures, where many entries of Λ are zero. Sparse structures are favorable in that they allow a clean interpretation of the model so it is clear as to which latent variables relate to each observed variable.

For example, the most favored type of sparsity is row sparsity. If there is only one entry per row, then every observed variable has only one parent. We will call Λ a *simple* structure if it possesses this attribute, and may be called a "perfect simple structure" by other authors [2]. It is so called since mutually exclusive sets of the observed variables perfectly serve as the set of causal indications for any given latent factor.

1.2.1 Exploratory Factor Analysis

Currently, the main methods of learning a sparse structure on Λ fall under the umbrella of Exploratory Factor Analysis (EFA). In practice, it is an algorithm which works as follows:

1. Given d as an input, set $\Phi = I_d$ and estimate an unconstrained Λ and diagonal Ω.
2. Use a rotation criterion to find Φ.
3. (Optional) Set small elements of Λ to zero if less than some threshold τ.
4. (Optional) Use a model selection procedure to choose among several choices of d.

Arguably, the biggest criticism of structure learning with EFA is the lack of rotational uniqueness (the premise of Step 2 in EFA). This refers to the fact that if Λ is unconstrained, there are many pairs (Λ, Φ) for which $\Sigma(\theta) = \Lambda\Phi\Lambda^T + \Omega$ (this is further detailed in Section 2.5).

To alleviate this problem, additional constraints called "rotation criteria" can be imposed to identify the parameters. One common example of such a rotation constraint is minimizing the following:

$$f(\Lambda) = (1-\kappa)\sum_{i=1}^{p}\sum_{j=1}^{d}\sum_{l\neq j}^{d}\lambda_{ij}^2\lambda_{il}^2 + \kappa\sum_{j=1}^{d}\sum_{i=1}^{p}\sum_{k\neq i}^{p}\lambda_{ij}^2\lambda_{kj}^2, \quad \kappa \in [0,1], \tag{1.6}$$

which is known as the Crawford-Ferguson family of rotation criteria [3]. We can see that the term $\sum_{j=1}^{d}\sum_{l\neq j}^{d}\lambda_{ij}^2\lambda_{il}^2 \geq 0$, where equality holds if and only if there is at most one non-zero element in the ith row of Λ. The term $\sum_{i=1}^{p}\sum_{k\neq i}^{p}\lambda_{ij}^2\lambda_{kj}^2$ behaves the same way except it acts upon the jth column of Λ. Thus, Equation 1.6 is a weighted penalty on the row and column sparsities of Λ, which is parameterized by κ. The most common parameterization choice is $\kappa = 1/p$, which is also known as varimax rotation [4].

Regardless of rotation criterion, the fact remains that different criteria may yield different solutions. Further, Step 3 of the EFA algorithm is another source of subjectivity. Even though a rotation criterion may minimize certain magnitudes of the entries of Λ, rotation alone is insufficient to produce entries that are exactly zero. Hence, one must choose an arbitrary threshold τ by which to set low magnitude entries of Λ to zero.

1.2.2 Penalized Exploratory Factor Analysis

As a potential solution to the subjectivity problems in EFA, penalized methods also have been developed. Instead of rotating factor coefficients, penalized EFA can achieve sparse solutions directly in estimation. While penalized estimation additionally requires tuning parameters, these can be selected in an objective manner, for example by using the Bayesian Information Criterion (BIC) or cross-validation (CV) [5].

These methods maximize a penalized likelihood (or optimize other loss functions) of the form:

$$\ell_p(\theta) = \ell(\theta) - p(\Lambda), \tag{1.7}$$

where $p(\cdot)$ is some penalty function. One example is the LASSO penalty [6], which has

been adapted to EFA [7, 8] as follows:

$$p_\kappa(\Lambda) = \kappa\|\Lambda\|_1 = \kappa \sum_{j=1}^{p}\sum_{k=1}^{d}|\lambda_{jk}|, \tag{1.8}$$

for a regularization parameter κ. Another common example is the minimax-concave penalty (MCP; [9]), which has been utilized in penalized EFA as well [10, 11]:

$$p_{\kappa,\gamma}(\Lambda) = \kappa \sum_{j=1}^{p}\sum_{k=1}^{d}\int_0^{|\lambda_{jk}|}\left(1 - \frac{x}{\kappa\gamma}\right)_+ dx, \tag{1.9}$$

where κ, γ are regularization parameters. In both cases, the regularization parameters are chosen by some model selection procedure (BIC, CV).

1.3 Motivation

The problems with current methods can be categorized into two main issues: identification of sparsities in Λ and the learning the number of latent variables, d. For identifying sparsities, EFA relies on methods of rotation and choosing thresholds for establishing structure. These have been criticized for their ad-hoc and subjective nature. Partially addressing this, penalized EFA methods utilize a penalty function to promote sparsity in a more principled manner. However, penalty functions generally also lack a theoretical basis from the model, and additionally requires a computationally intense search over the tuning parameters.

For learning d, neither EFA nor penalized EFA have intrinsic methods to estimate this parameter, and require it as an input. Many data based guidelines of proposing d have been suggested, but suffer from poor performance, lack of objectivity, or both (for a recent review see [12]). Addressing these issues, we propose a correlation thresholding algorithm to learn the structure of Λ and the number of latent variable simultaneously. Our method is fast and simple, utilizing graph theory to provide a motivating framework.

Chapter 2

The Correlation Thresholding Algorithm

2.1 Overview

To begin, we review several terms and definitions from graph theory. We define a graph $\mathcal{G}$ as an ordered pair (V, E), explicitly denoted as $\mathcal{G}(V, E)$. V is a set of vertices and $E \subseteq V \times V$ is a set of edges. For convenience, we will use $V = X$ to mean that the elements of the vertex set V represent the index set of the random vector X. We also restrict our attention to *undirected* graphs, where $(i, j) \in E$ if and only if $(j, i) \in E$. A *clique* of $\mathcal{G}(V, E)$ is a subset of vertices $C \subseteq V$ such that all pairs of distinct vertices in C are in the edge set E. Finally, a *maximal clique* is a clique that cannot be extended by including more vertices from V.

We now give a simple example to demonstrate how we will use graph theory to analyze factor analysis models. Consider the following parameters:

$$\widetilde{\Lambda} = \begin{bmatrix} \widetilde{\lambda}_{11} & \\ \widetilde{\lambda}_{21} & \\ \widetilde{\lambda}_{31} & \widetilde{\lambda}_{32} \\ & \widetilde{\lambda}_{42} \\ & \widetilde{\lambda}_{52} \end{bmatrix}, \Phi = \begin{bmatrix} 1 & \\ & 1 \end{bmatrix}, \widetilde{\Omega} = \begin{bmatrix} \widetilde{\omega}_1 & & & & \\ & \widetilde{\omega}_2 & & & \\ & & \widetilde{\omega}_3 & & \\ & & & \widetilde{\omega}_4 & \\ & & & & \widetilde{\omega}_5 \end{bmatrix}. \tag{2.1}$$

This model is illustrated in Figure 2.1 (left). Note that these matrices imply the following

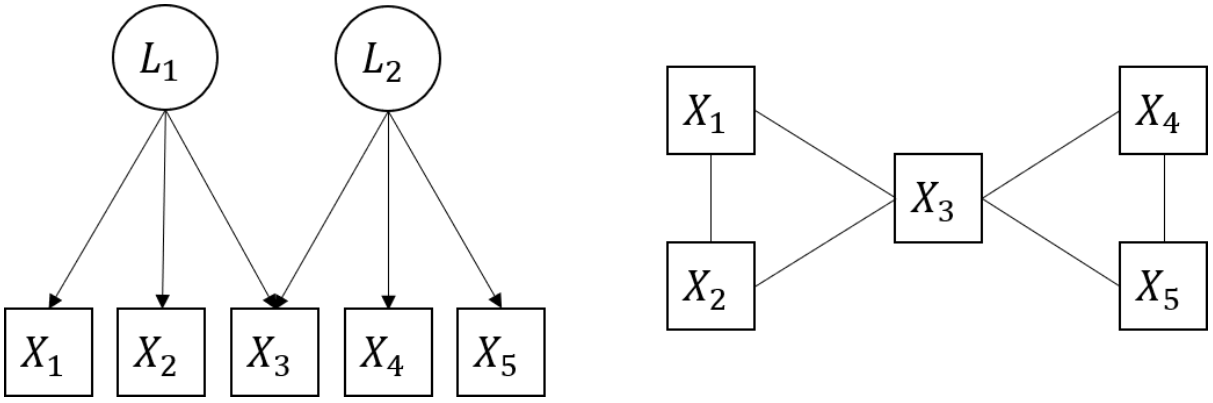

Figure 2.1: On the left we have a graphical representation of the model described in Equation 2.1. On the right we have the associated correlation graph.

correlation matrix:

$$\widetilde{\Sigma}(\theta) = \widetilde{\Lambda}\Phi\widetilde{\Lambda}^T + \widetilde{\Omega} = \begin{bmatrix} \widetilde{\lambda}_{11}^2 + \widetilde{\omega}_1 & \widetilde{\lambda}_{11}\widetilde{\lambda}_{21} & \widetilde{\lambda}_{11}\widetilde{\lambda}_{31} & & \\ \widetilde{\lambda}_{11}\widetilde{\lambda}_{21} & \widetilde{\lambda}_{21}^2 + \widetilde{\omega}_2 & \widetilde{\lambda}_{21}\widetilde{\lambda}_{31} & & \\ \widetilde{\lambda}_{11}\widetilde{\lambda}_{31} & \widetilde{\lambda}_{21}\widetilde{\lambda}_{31} & \widetilde{\lambda}_{31}^2 + \widetilde{\lambda}_{32}^2 + \widetilde{\omega}_3 & \widetilde{\lambda}_{32}\widetilde{\lambda}_{42} & \widetilde{\lambda}_{32}\widetilde{\lambda}_{52} \\ & & \widetilde{\lambda}_{32}\widetilde{\lambda}_{42} & \widetilde{\lambda}_{42}^2 + \widetilde{\omega}_4 & \widetilde{\lambda}_{42}\widetilde{\lambda}_{52} \\ & & \widetilde{\lambda}_{32}\widetilde{\lambda}_{52} & \widetilde{\lambda}_{42}\widetilde{\lambda}_{52} & \widetilde{\lambda}_{52}^2 + \widetilde{\omega}_5 \end{bmatrix}. \quad (2.2)$$

From here, let us convert $\widetilde{\Sigma}(\theta)$ to a graph in the following manner. Let the vertex set represent the observed variables X and its edge set be determined by the non-zero lower-triangular entries of $\widetilde{\Sigma}(\theta)$. That is:

$$\begin{aligned} V &= \{1, \ldots, 5\} \\ E &= \{(i,j) : |\rho_{ij}| > 0\}, \end{aligned} \quad (2.3)$$

for $(i,j) \in \{1, \ldots 5\}^2$. Then the graph $\mathcal{G}(V,E)$ is depicted in 2.1 (right). The key observation here is that the number of latent variables in the factor model correspond to the number of maximal cliques in the graph. Moreover, the children of each latent variable are correspondingly the members of these maximal cliques. In this way, we can gain insight to the unknown structure of Λ by converting a thresholded correlation matrix into a graph.

Extending this logic to the oblique case ($\Phi \neq I_d$), it is clear that the edge detection procedure is not as simple as thresholding for non-zero correlations. Generally, we begin with a saturated correlation matrix (no sparsities), since variables that do not share parents will be correlated by virtue of their parents being correlated. However, in most practical settings, the correlation of pairs that do not share parents will have a lower magnitude than those pairs whose parents are shared. To see why this could be the case, consider Equation 2.2 again if $\Phi \neq I_d$:

$$\begin{bmatrix} \widetilde{\lambda}_{11}^2+\widetilde{\omega}_1^2 & \widetilde{\lambda}_{11}\widetilde{\lambda}_{21} & \widetilde{\lambda}_{11}\widetilde{\lambda}_{31}+\widetilde{\lambda}_{11}\widetilde{\lambda}_{32}\phi_{12} & \boldsymbol{\widetilde{\lambda}_{11}\widetilde{\lambda}_{42}\phi_{12}} & \boldsymbol{\widetilde{\lambda}_{11}\widetilde{\lambda}_{52}\phi_{12}} \\ \widetilde{\lambda}_{11}\widetilde{\lambda}_{21} & \widetilde{\lambda}_{21}^2+\widetilde{\omega}_2^2 & \widetilde{\lambda}_{21}\widetilde{\lambda}_{31}+\widetilde{\lambda}_{21}\widetilde{\lambda}_{32}\phi_{12} & \boldsymbol{\widetilde{\lambda}_{21}\widetilde{\lambda}_{42}\phi_{12}} & \boldsymbol{\widetilde{\lambda}_{21}\widetilde{\lambda}_{45}\phi_{12}} \\ \widetilde{\lambda}_{11}\widetilde{\lambda}_{31}+\widetilde{\lambda}_{11}\widetilde{\lambda}_{32}\phi_{12} & \widetilde{\lambda}_{21}\widetilde{\lambda}_{31}+\widetilde{\lambda}_{21}\widetilde{\lambda}_{32}\phi_{12} & \widetilde{\lambda}_{31}^2+\widetilde{\lambda}_{32}^2+\widetilde{\omega}_3^2 & \widetilde{\lambda}_{31}\widetilde{\lambda}_{42}\phi_{12}+\widetilde{\lambda}_{32}\widetilde{\lambda}_{42} & \widetilde{\lambda}_{31}\widetilde{\lambda}_{52}\phi_{12}+\widetilde{\lambda}_{32}\widetilde{\lambda}_{52} \\ \boldsymbol{\widetilde{\lambda}_{11}\widetilde{\lambda}_{42}\phi_{12}} & \boldsymbol{\widetilde{\lambda}_{21}\widetilde{\lambda}_{42}\phi_{12}} & \widetilde{\lambda}_{31}\widetilde{\lambda}_{42}\phi_{12}+\widetilde{\lambda}_{32}\widetilde{\lambda}_{42} & \widetilde{\lambda}_{42}^2+\widetilde{\omega}_4^2 & \widetilde{\lambda}_{42}\widetilde{\lambda}_{52} \\ \boldsymbol{\widetilde{\lambda}_{11}\widetilde{\lambda}_{52}\phi_{12}} & \boldsymbol{\widetilde{\lambda}_{21}\widetilde{\lambda}_{45}\phi_{12}} & \widetilde{\lambda}_{31}\widetilde{\lambda}_{52}\phi_{12}+\widetilde{\lambda}_{32}\widetilde{\lambda}_{52} & \widetilde{\lambda}_{42}\widetilde{\lambda}_{52} & \widetilde{\lambda}_{52}^2+\widetilde{\omega}_5^2 \end{bmatrix}. \tag{2.4}$$

Here we can see that Φ has a shrinking effect on the correlations between variables that do not share parents (bolded for emphasis). Suppose that there existed some threshold by which these correlations (bold) were below, and correlations among variables that shared parents (not bold) were above. Then this threshold could identify and eliminate pairs of variables that did not share parents, yielding a structurally informative graph as in Figure 2.1 (right). Finding such a threshold and using the thresholded correlation graph to learn the factor analysis structure is the premise of our algorithm.

We formalize the thresholded correlation graph as follows. Let the *parent set* of X_i be $\pi(X_i) \coloneqq \{j : \lambda_{ij} \neq 0, j \in \{1, \ldots, d\}\}$. Then, define the edge set of pairs who share parents as:

$$E \coloneqq \{(i,j) : \pi(X_i) \cap \pi(X_j) \neq \emptyset\}, \tag{2.5}$$

for all $(i,j) \in \{1, \ldots, p\}^2$. Subsequently, we will also work with the complement of E, which for clarity is:

$$E^c = \{(i,j) : \pi(X_i) \cap \pi(X_j) = \emptyset\}. \tag{2.6}$$

We would like to find some threshold that is able to separate the E and E^c sets by the magnitude of the correlations. We will define this notion as "thresholdable." Specifically, a set of parameters θ is called *thresholdable* if and only if there exists a threshold τ_0 such that:

$$\max\{|\rho_{kl}| : (k,l) \in E^c\} < \tau_0 < \min\{|\rho_{ij}| : (i,j) \in E\}. \tag{2.7}$$

That is, if θ is thresholdable, then we can correctly sort the index pairs of X into the E and E^c sets using τ_0. This allows us to move forward with the graphical logic under the graph $\mathcal{G}(X, E)$ as shown in the previous example with orthogonal factors (i.e., Figure 2.1). Further, we can also define an estimator of E for a candidate τ_k as:

$$\hat{E}(\tau_k) \coloneqq \{(i,j) : |r_{ij}| > \tau_k\}. \tag{2.8}$$

where r_{ij} denotes the sample correlation.

Putting these ideas together, the core task of the algorithm is to search for a suitable τ_0. This can be done by searching over a set of candidate set $\tau_k \in [0, 1]$ and analyzing their respective threholded correlation graphs $\mathcal{G}(X, \hat{E}(\tau_k))$. The aforementioned graphical concepts can then be leveraged to learn the number of latent variables and the structure of Λ. This essentially yields a set of candidate models for which we can utilize model selection procedures (e.g., BIC) to select a final model.

2.2 The Algorithm

We now apply the framework from the previous section to construct the Correlation Thresholding (CT) Algorithm. Given the sample correlation matrix $R = (r_{ij}) \in \mathbb{R}^{p \times p}$:

Algorithm 1: The Correlation Thresholding Algorithm

input : The sample correlation matrix R
output : Parameter estimates $\hat{\theta}$

1 Create a sequence of $k = 1, \ldots, m$ threshold levels $\tau_k \in [0, 1]$;
2 **for** $k = 1, \ldots, m$ **do**
3 Calculate $\hat{E}(\tau_k)$ and analyze $\mathcal{G}(X, \hat{E}(\tau_k))$ for a set of maximal cliques: $\mathcal{C}_k = \{C_1, \ldots C_{|\mathcal{C}_k|}\}$;
4 Set $d = |\mathcal{C}_k|$;
5 **forall** $(i, j) \in \{1, \ldots, p\} \times \{1, \ldots, d\}$ **do**
6 **if** $X_i \in C_j$ **then**
7 Set λ_{ij} as unconstrained;
8 **else**
9 Set $\lambda_{ij} = 0$;
10 **end**
11 **end**
12 Estimate the model constraints for Λ learned in Step 5 to obtain $\hat{\theta}_k$;
13 **end**
14 Select one of the k structures via a model selection procedure (e.g., BIC) ;

An overview of the procedure is displayed in Figure 2.2. The idea behind the CT Algorithm is as follows. Suppose we are dealing with the population correlation matrix $R = \widetilde{\Sigma}(\theta)$. Then if $\tau_0 \in \tau_k$, among other identifiability conditions (described below), the correct structure of Λ will be represented in one of the $\hat{E}(\tau_k)$ (Step 3). Then, given that the correct model is among the final set of candidate models, a consistent model selection criterion will be able to recover it (Step 14). In the following sections, we describe the precise conditions under which this can be achieved, as well as establishing statistical consistency for the algorithm.

2.3 On the Thresholdability of θ

One of the more fundamental assumptions of the CT Algorithm is the thresholdability of θ. In this section, we examine this assumption in more detail. Specifically, a necessary and sufficient condition for thresholdability is as follows:

Theorem 1 *Let (X_i, X_j, X_k, X_l) be a quadruplet of variables such that (X_i, X_j) share*

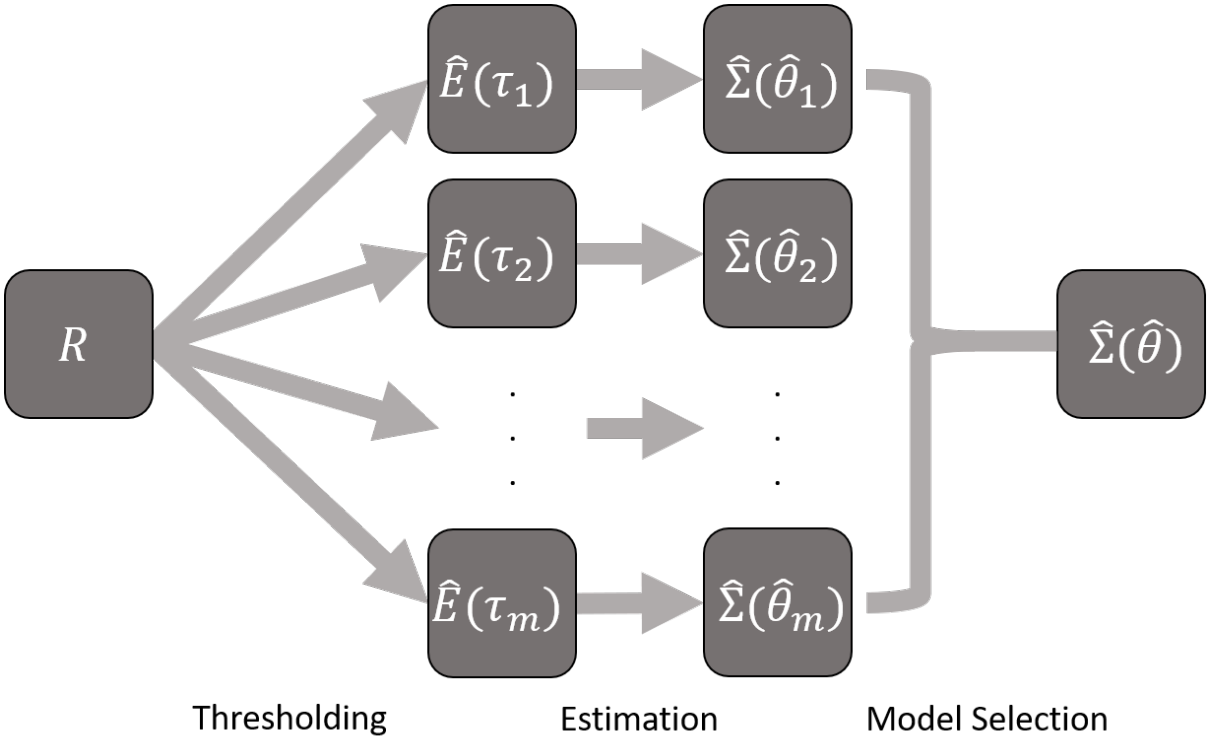

Figure 2.2: Overview of the CT Algorithm.

parents and (X_k, X_l) *do not. Then, a set of parameters* θ *is thresholdable if and only if:*

$$\max_{(k,l)} |\widetilde{\Lambda}_{kE}\Phi_{EF}\widetilde{\Lambda}_{lF}^T| < \min_{(i,j)} |\widetilde{\Lambda}_{iA}\Phi_{AB}\widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iC}\Phi_{CB}\widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iA}\Phi_{AC}\widetilde{\Lambda}_{jC}^T + \widetilde{\Lambda}_{iC}\Phi_{CC}\widetilde{\Lambda}_{jC}^T|, \quad (2.9)$$

where $A = \pi(X_i)\backslash\pi(X_j)$, $B = \pi(X_j)\backslash\pi(X_i)$, $C = \pi(X_i) \cap \pi(X_j)$, $E = \pi(X_k)$, *and* $F = \pi(X_l)$, *and* $i \neq j$ *and* $k \neq l$.

Proof. First it will be convenient to partition the parent variables of any pair (X_i, X_j) as $\pi(X_i) \cup \pi(X_j) = \{L_A, L_B, L_C\}$, where:

$$\begin{aligned} A &= \pi(X_i)\backslash\pi(X_j) \\ B &= \pi(X_j)\backslash\pi(X_i) \\ C &= \pi(X_i) \cap \pi(X_j). \end{aligned} \quad (2.10)$$

Then we may re-cast Equation 1.1 for any pair $(\widetilde{X}_i, \widetilde{X}_j)$ as follows:

$$\begin{bmatrix} \widetilde{X}_i \\ \widetilde{X}_j \end{bmatrix} = \begin{bmatrix} \widetilde{\Lambda}_{iA} & \mathbf{0} & \widetilde{\Lambda}_{iC} \\ \mathbf{0} & \widetilde{\Lambda}_{jB} & \widetilde{\Lambda}_{jC} \end{bmatrix} \begin{bmatrix} L_A \\ L_B \\ L_C \end{bmatrix} + \begin{bmatrix} \widetilde{\epsilon}_i \\ \widetilde{\epsilon}_j \end{bmatrix}. \tag{2.11}$$

We then obtain the correlation of between X_i and X_j from this form as follows:

$$\text{Var}\left(\begin{bmatrix} \widetilde{X}_i \\ \widetilde{X}_j \end{bmatrix}\right) = \begin{bmatrix} \widetilde{\Lambda}_{iA} & \mathbf{0} & \widetilde{\Lambda}_{iC} \\ \mathbf{0} & \widetilde{\Lambda}_{jB} & \widetilde{\Lambda}_{jC} \end{bmatrix} \begin{bmatrix} \Phi_{AA} & \Phi_{AB} & \Phi_{AC} \\ \Phi_{BA} & \Phi_{BB} & \Phi_{BC} \\ \Phi_{CA} & \Phi_{CB} & \Phi_{CC} \end{bmatrix} \begin{bmatrix} \widetilde{\Lambda}_{iA}^T & \mathbf{0} \\ \mathbf{0} & \widetilde{\Lambda}_{jB}^T \\ \widetilde{\Lambda}_{iC}^T & \widetilde{\Lambda}_{jC}^T \end{bmatrix} + \begin{bmatrix} \widetilde{\omega}_i & 0 \\ 0 & \widetilde{\omega}_j \end{bmatrix}, \tag{2.12}$$

for which we multiply through and take the off-diagonal to be:

$$\rho_{ij} = \widetilde{\Lambda}_{iA}\Phi_{AB}\widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iC}\Phi_{CB}\widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iA}\Phi_{AC}\widetilde{\Lambda}_{jC}^T + \widetilde{\Lambda}_{iC}\Phi_{CC}\widetilde{\Lambda}_{jC}^T. \tag{2.13}$$

Writing ρ_{ij} in this way yields a useful decomposition with respect to the structure of the factor analysis model. Specifically, this can be thought of as the correlation between X_i and X_j due to their non-shared parents being correlated (Φ_{AB}), their non-shared parents being correlated with their shared parents (Φ_{AC}, Φ_{CB}) and simply having shared parents (Φ_{CC}). Thus, if X_i and X_j have no shared parents, then the index set C is empty. This reduces Equation 2.13 to:

$$\rho_{ij} = \widetilde{\Lambda}_{iA}\Phi_{AB}\widetilde{\Lambda}_{jB}^T. \tag{2.14}$$

The result of Theorem 1 follows by characterizing the definition of thresholdability (Equation 2.7) directly in terms of θ. That is, if for all (X_i, X_j) that share parents and for

all (X_k, X_l) that do not share parents, θ is thresholdable if and only if:

$$\max_{(k,l)} |\widetilde{\Lambda}_{kE} \Phi_{EF} \widetilde{\Lambda}_{lF}^T| < \min_{(i,j)} |\widetilde{\Lambda}_{iA} \Phi_{AB} \widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iC} \Phi_{CB} \widetilde{\Lambda}_{jB}^T + \widetilde{\Lambda}_{iA} \Phi_{AC} \widetilde{\Lambda}_{jC}^T + \widetilde{\Lambda}_{iC} \Phi_{CC} \widetilde{\Lambda}_{jC}^T|. \quad (2.15)$$

□

The application of Theorem 1 can be illustrated by inspecting a specific example. Consider the commonly used simple structure factor analysis model. Recall that simple structures have only one non-zero entry per row of Λ. This implies that each observed variable has only one latent variable parent, and therefore the right-hand side of Equation 2.9 reduces to the Φ_{CC} term, since Φ_{AB}, Φ_{CB}, and Φ_{AC} do not exist and $\Phi_{CC} = 1$. Hence, in the case of simple structure models, the thresholdability condition is met if and only if:

$$\max_{(k,l)} |\widetilde{\lambda}_{ke} \widetilde{\lambda}_{lf} \phi_{ef}| < \min_{(i,j)} |\widetilde{\lambda}_{ic} \widetilde{\lambda}_{jc}^T|, \quad (2.16)$$

where $\pi(X_i) = \pi(X_j) = C$, $\pi(X_k) = E$, and $\pi(X_l) = F$. From here, we can ascertain that if the non-zero entries of $\widetilde{\Lambda}$ and the off-diagonal entries of Φ are equal or relatively homogenous, then the model is thresholdable.

More generally speaking, it can be seen that thresholdability holds as Φ tends toward I_d, and/or as the non-zero entries of $\widetilde{\Lambda}$ tends toward 1. Both of these conditions are desirable properties of factor analytic designs. First, it has been suggested that latent variable models should be designed such that the latent factors be distinguishable from one another, or that they are not too highly correlated [13]. If the latent factors are too highly correlated, then a factor solution with less dimensions may be better suited. Second, higher magnitudes of the non-zero entries of $\widetilde{\Lambda}$ reflect better measurement of the latent variable. That is, observed variables serve as proxies for the latent variables, hence stronger regression coefficients provide more information [14].

2.4 Structural Identifiability via Thresholded Correlation Graphs

In this section we study the conditions under which the structure for Λ can be recovered from the thresholded correlation graph. To demonstrate the problem of structural identifiability, consider Figure 2.3. Assuming each displayed models are thresholdable, all these structures will yield the same thresholded correlation graph. Specifically, the maximal cliques that are yielded by them are $\{1, 2, 3\}$ and $\{3, 4, 5\}$, despite all having different structures. This can be seen by noting that some latent variables do not yield maximal cliques in $\mathcal{G}(X, E)$, or yield the same maximal clique as another latent variable. For example, in Figure 2.3b, both L_2 and L_3 yield the clique $\{3, 4, 5\}$. Thus, L_2 and L_3 cannot be distinguished from each other through maximal cliques alone. Similarly, in Figure 2.3c, L_3 yields the clique $\{4, 5\}$, but it is not maximal since L_2 yields $\{3, 4, 5\}$. In this case, L_2 cannot be identified as latent variable, since its clique is subsumed by the one yielded by L_3. Hence, we must consider the problem of multiple structures corresponding to the same thresholded correlation graph.

Clearly, to identify distinct structures from maximal cliques, there must be a bijective correspondence between the structures of each latent variable and the set of maximal cliques. If such a correspondence holds for a given Λ, we will call Λ (or θ) *maximal clique identifiable*. We propose one such mapping as follows. Let the *child set* of a latent variable be denoted $\text{ch}(L_k) = \{i : \lambda_{ik} \neq 0, i \in \{1, \ldots, p\}\}$. Then, a sufficient condition maximal clique identifiability is as follows:

$$\text{ch}(L_k) / \bigcup_{k \neq l} \text{ch}(L_l) = U_k, \tag{2.17}$$

where $U_k \neq \emptyset$ and indexes the unique children variables for L_k for all $k \in \{1, \ldots, d\}$. If this condition holds for Λ (or θ), we will say that the *unique child condition* holds for Λ. It essentially means that all latent parents have at least one unique child variable.

Theorem 2 *If θ is thresholdable and the unique child condition holds in Λ, then the set of latent variable children $\{\text{ch}(L_k) : k \in \{1, \ldots, d\}\}$ has a bijective correspondence to the set of maximal cliques in $\mathcal{G}(X, E)$.*

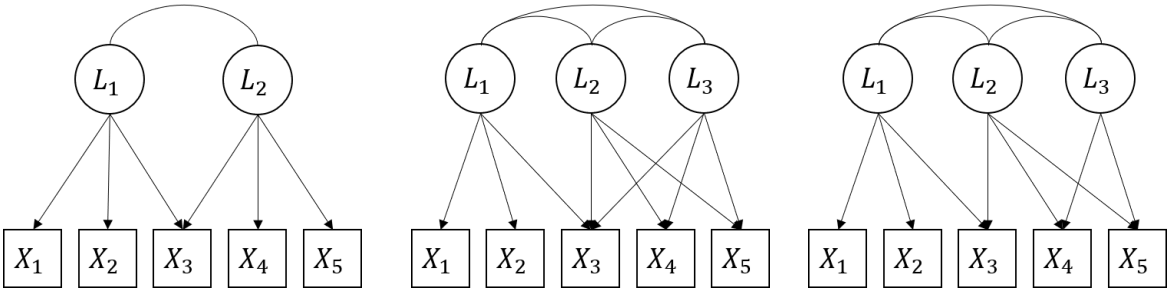

Figure 2.3: Three structures that yield the same thresholded correlation graph.

Proof. First, let us consider an alternative definition of E:

$$E = \{(i, j) : \lambda_{ik} \neq 0, \lambda_{jk} \neq 0, k \in \{1, \ldots, d\}\}. \tag{2.18}$$

This simply re-writes Equation 2.5 in terms of the structure of Λ. Then by the definition of $\text{ch}(\cdot)$, each $\text{ch}(L_k)$ forms a clique in $\mathcal{G}(X, E)$. We can denote such a clique formed this way as C_k. Thus, we can consider E as a mapping $E : \{\text{ch}(L_k)\} \to \{C_k\}$.

Under the unique child condition, there is a unique $U_k \subseteq \text{ch}(L_k)$, implying $U_k \subseteq C_k$ by definition of E. Thus, the correspondence of each U_k to each $\text{ch}(L_k)$ and C_k makes E a one-to-one mapping. Trivially, E is also an onto mapping, as $\{C_k\}$ consists only of maximal cliques generated by E, which are the only maximal cliques considered by the CT Algorithm. Taken together, we have a bijective correspondence between $\{\text{ch}(L_k)\}$ and $\{C_k\}$. □

2.5 Rotational Uniqueness

An important consideration with factor analysis models is the uniqueness of θ. For the CT Algorithm, we show that the unique child condition guarantees a type of uniqueness for θ. To demonstrate, when Λ is unconstrained (e.g., EFA), there may be many (Λ, Φ) pairs that exist such that $\Sigma(\theta) = \Lambda\Phi\Lambda^T + \Omega$. Let $M \in \{\text{Invertible } \mathbb{R}^{d \times d}\}$ be a so-called

rotation matrix. Then:

$$\begin{aligned}\Sigma(\theta) &= \Lambda\Phi\Lambda^T + \Omega \\ &= \Lambda M M^{-1}\Phi M^{-T} M^T \Lambda^T + \Omega \\ &= \Lambda_M \Phi_M \Lambda_M^T + \Omega,\end{aligned} \tag{2.19}$$

letting $\Lambda_M = \Lambda M$ and $\Phi_M = M^{-1}\Phi M^{-T}$. Hence there are multiple pairs (Λ, Φ) that can construct the same $\Sigma(\theta)$ matrix. Following this, we formally define the rotational uniqueness as follows. Let valid rotation matrices be denoted as $M \in \mathcal{M} = \{\text{Invertible } \mathbb{R}^{d\times d} : \Sigma(\theta) = \Lambda_M \Phi_M \Lambda_M^T + \Omega\}$:

1. If $\mathcal{M} = I_d$, then (Λ, Φ) is said to be *globally rotationally unique.*
2. If $\mathcal{M} \subseteq \{\text{Signature Matrix} \in \mathbb{R}^{d\times d}\}$, then (Λ, Φ) is said to be *locally rotationally unique*,

where signature matrices are diagonal matrices whose diagonal elements are ± 1.

Corollary 1 *If the unique child condition holds in* Λ*, then* (Λ, Φ) *is locally rotationally unique.*

Proof. To begin, we list two sufficient conditions for Λ that yield local rotational uniqueness for our model. Adapting these conditions from [15], we have:

Condition 1: Λ has at least $d-1$ fixed zeroes in each column.

Condition 2: $\text{rank}(\Lambda^{[j]}) = d-1$ for all $j \in \{1, \dots d\}$,

where $\Lambda^{[j]}$ is defined as the submatrix of Λ, which consists of the rows of Λ which have fixed zeroes in the jth column, and consists of the columns of Λ except for the jth. An

example of $\Lambda^{[j]}$ is as follows:

$$\Lambda = \begin{bmatrix} \lambda_{11} & 0 & 0 \\ \lambda_{21} & \lambda_{22} & 0 \\ \lambda_{31} & 0 & 0 \\ 0 & \lambda_{42} & 0 \\ 0 & \lambda_{52} & \lambda_{53} \\ 0 & \lambda_{62} & 0 \\ 0 & 0 & \lambda_{73} \\ 0 & 0 & \lambda_{83} \\ \lambda_{91} & 0 & \lambda_{93} \end{bmatrix}, \quad \Lambda^{[1]} = \begin{bmatrix} \lambda_{42} & 0 \\ \lambda_{52} & \lambda_{53} \\ \lambda_{62} & 0 \\ 0 & \lambda_{73} \\ 0 & \lambda_{83} \end{bmatrix}, \quad \Lambda^{[2]} = \begin{bmatrix} \lambda_{11} & 0 \\ \lambda_{31} & 0 \\ 0 & \lambda_{73} \\ 0 & \lambda_{83} \\ \lambda_{91} & \lambda_{93} \end{bmatrix}, \quad \Lambda^{[3]} = \begin{bmatrix} \lambda_{11} & 0 \\ \lambda_{21} & \lambda_{22} \\ \lambda_{31} & 0 \\ 0 & \lambda_{42} \\ 0 & \lambda_{62} \end{bmatrix}. \tag{2.20}$$

These conditions can be seen to be satisfied by the unique child condition as follows. For all $j, k \in \{1, \ldots, d\}$, and $i \in \{1, \ldots, p\}$ we can re-cast U_j as:

$$U_j = \{i : \lambda_{ij} \neq 0, \lambda_{ik} = 0, k \neq j\}, \tag{2.21}$$

and let the index of non-unique variables be:

$$\overline{U} = \{i : i \notin \cup_{j=1}^{d} U_j\}. \tag{2.22}$$

Let us permute the rows of Λ according to an order that satisfies $(U_1, \ldots, U_d, \overline{U})$. Denoting a permutation matrix that yields such a row ordering as P, we have:

$$P\Lambda = \begin{bmatrix} \Lambda_{U_1 1} & & \\ & \ddots & \\ & & \Lambda_{U_d d} \\ \Lambda_{\overline{U}1} & \cdots & \Lambda_{\overline{U}d} \end{bmatrix}. \tag{2.23}$$

That is, we can permute the rows of Λ such that its upper part is block-diagonal with d blocks. Then, by definition of a block-diagonal matrix, there must be at least $d-1$ zeroes

in each column, satisfying Condition 1. It is easily seen that $P\Lambda$ also satisfies Condition 2, as any $(P\Lambda)^{[j]}$ will also have its upper part be block-diagonal, and thus full rank $(d-1)$. □

2.6 Consistency

In this section, we establish the consistency of the CT Algorithm. The crucial part of the argument depends on the consistency of the algorithm's structural learning aspect. Since the structure of the model is determined by the graph $\mathcal{G}(X, E)$, structural consistency will follow if $\hat{E}(\tau_0) \xrightarrow{\mathcal{P}} E$. To determine this, we study the finite sample error bounds for the event $\hat{E}(\tau_0) = E$.

Theorem 3 *Let $X \sim \mathcal{N}_p(0, \Sigma(\theta))$ be a random vector. Assume all correlations between all pairs (X_i, X_j) are bounded such that $|\rho_{ij}| \leq M < 1$, for all $(i,j) \in \{1, \ldots, p\}^2$. Then, the following inequality holds:*

$$\mathbb{P}(\hat{E}(\tau_0) \neq E) \leq C_2 p(p-1)(n-2)\left(\frac{4-\gamma^2}{4+\gamma^2}\right)^{n-4}, \tag{2.24}$$

where $0 < C_2 < \infty$ only depends on M and γ is defined as:

$$\gamma := \frac{\min(|\rho_{ij}| \in E) - \max(|\rho_{ij}| \in E^c)}{2}. \tag{2.25}$$

Proof. Our goal is to give a bound for the event $\hat{E}(\tau_0) \neq E$. For clarity, let us first consider the event $\hat{E}(\tau_0) = E$, which by definition, holds if and only if:

$$\Big(\bigcap_{(i,j)\in E} |r_{ij}| > \tau_0\Big) \cap \Big(\bigcap_{(i,j)\notin E} |r_{ij}| < \tau_0\Big). \tag{2.26}$$

Then by De Morgan's laws, we can say $\hat{E}(\tau_0) \neq E$ if and only if:

$$\Big(\bigcup_{(i,j)\in E} |r_{ij}| \leq \tau_0\Big) \cup \Big(\bigcup_{(i,j)\notin E} |r_{ij}| \geq \tau_0\Big), \tag{2.27}$$

which is to say that $\hat{E}(\tau_0) \neq E$ holds if and only if any r_{ij} is on the opposite side of τ_0 as

their population analog ρ_{ij}. From here, the strategy is to derive bounds for $\mathbb{P}(|r_{ij}| \leq \tau_0)$ if $|\rho_{ij}| > \tau_0$, and $\mathbb{P}(|r_{ij}| \geq \tau_0)$ if $|\rho_{ij}| < \tau_0$, for all (i, j). To determine these bounds, we make use of a concentration inequality for $\mathbb{P}(|r_{ij} - \rho_{ij}| \geq \epsilon)$ from Lemma 1 of [16]. We re-state this as follows:

Lemma 1 *Assuming X_i and X_j are Gaussian random variables and $|\rho_{ij}| \leq M < 1$ for all (i, j), then for any $0 < \epsilon \leq 2$, the quantity $|r_{ij} - \rho_{ij}| \geq \epsilon$ is bounded as follows:*

$$\mathbb{P}(|r_{ij} - \rho_{ij}| \geq \epsilon) \leq C_1(n-2)\left(\frac{4-\epsilon^2}{4+\epsilon^2}\right)^{n-4}, \tag{2.28}$$

where $0 < C_1 < \infty$ only depends on M.

For our purposes, we set $\epsilon = \gamma$, which will be the best choice of ϵ to uniformly bound all $\mathbb{P}(|r_{ij}| \leq \tau_0)$ if $|\rho_{ij}| > \tau_0$ and $\mathbb{P}(|r_{ij}| \geq \tau_0)$ if $|\rho_{ij}| < \tau_0$. The uniformity of the bound follows by seeing that $\gamma \leq \big||\rho_{ij}| - \tau_0\big|$ for all (i, j). That is, there is no ρ_{ij} that is closer to τ_0 than the length of γ. It is the best choice in that we would like ϵ to be as large as possible for fastest decay. This is achieved by selecting the mid-point of $\min(|\rho_{ij}| \in E)$ and $\max(|\rho_{ij}| \in E^c)$.

We begin with the scenario where $|\rho_{ij}| < \tau$. Given the left-hand side of Equation 2.28 and setting $\epsilon = \gamma$, we have:

$$\begin{aligned}
\mathbb{P}(|r_{ij} - \rho_{ij}| \geq \gamma) &\geq \mathbb{P}(|r_{ij}| - |\rho_{ij}| \geq \gamma) \\
&\geq \mathbb{P}(|r_{ij}| - |\rho_{ij}| \geq \tau_0 - |\rho_{ij}|) \\
&= \mathbb{P}(|r_{ij}| \geq \tau_0).
\end{aligned} \tag{2.29}$$

Hence, $\mathbb{P}(|r_{ij}| \leq \tau_0)$ is bounded from above by the right-hand side of Equation 2.28 if $|\rho_{ij}| < \tau$. We can use the same strategy if $|\rho_{ij}| > \tau$:

$$\begin{aligned}
\mathbb{P}(|r_{ij} - \rho_{ij}| \geq \gamma) &\geq \mathbb{P}(|\rho_{ij}| - |r_{ij}| \geq \gamma) \\
&\geq \mathbb{P}(|\rho_{ij}| - |r_{ij}| \geq |\rho_{ij}| - \tau_0) \\
&= \mathbb{P}(-|r_{ij}| \geq -\tau_0) \\
&= \mathbb{P}(|r_{ij}| \leq \tau_0).
\end{aligned} \tag{2.30}$$

Since these two events have the same upper bound, let us combine them by defining:

$$B_{ij} = B(r_{ij}, \tau_0) := \begin{cases} |r_{ij}| \geq \tau_0 & \text{if } |\rho_{ij}| < \tau \\ |r_{ij}| \leq \tau_0 & \text{if } |\rho_{ij}| > \tau, \end{cases} \tag{2.31}$$

Noting that $\hat{E}(\tau_0) \neq E(\tau_0)$ holds if and only if $\bigcup_{(i,j)} B_{ij}$ holds, what remains is to find a bound of the latter event. This can be done as follows:

$$\begin{aligned} \mathbb{P}\Big(\bigcup_{(i,j)} B_{ij} \Big) &\leq \sum_{(i,j)} \mathbb{P}\left(B_{ij}\right) \\ &\leq \frac{p(p-1)}{2} \max_{(i,j)} \mathbb{P}(B_{ij}) \\ \Rightarrow \mathbb{P}(\hat{E}(\tau_0) \neq E(\tau_0)) &\leq C_2 p(p-1)(n-2) \left(\frac{4-\gamma^2}{4+\gamma^2} \right)^{n-4}, \end{aligned} \tag{2.32}$$

where $0 < C_2 < \infty$ only depends on M. The final result follows by recognizing that all B_{ij} are uniformly bounded as in Lemma 1. □

The immediate corollary of Theorem 3, is that $\lim_{n\to\infty} \mathbb{P}(\hat{E}(\tau_0) \neq E(\tau_0)) = 0$, from which structural consistency can easily be seen. Overall parameter consistency of the CT Algorithm follows under two additional considerations. First, we must have $\tau_0 \in \tau_k$ which can be obtained if τ_k constructed to be large and dispersed enough. Second, a consistent parameter estimation and/or model selection procedure need to be used in the algorithm. A straightforward choice would be to use maximum likelihood estimation in conjunction with BIC model selection. Then, asymptotically, the CT Algorithm will produce the correct model structure with consistent parameter estimates.

Some additional observations regarding the bound on $\mathbb{P}(\hat{E}(\tau_0) \neq E(\tau_0))$ are as follows. The bound is independent of d, thus consistency holds regardless of the number of latent variables. Second, the term $(4-\gamma^2)/(4+\gamma^2)$ decays at an exponential rate with n. This allows the number of variables p to grow at up to a polynomial rate with n while maintaining consistency.

Chapter 3

Simulation Study

3.1 Method

We generated data sets from a Gaussian distribution. The mean vector was set to $\mu = \mathbf{0}$ for all conditions, and the covariance matrix Σ was parameterized by θ which varied by condition. Λ followed a simple structure (one non-zero entry per row). The number of latent variables (d) was set to 2, 3, 4 and 5, with the number of children per latent variable set to 5. The non-zero entries of Λ were drawn from a uniform distribution, $\lambda_{ij} \sim \text{Uniform}(0.6, 0.8)$. Additionally, the off-diagonal entries of Φ were drawn from a uniform distribution, with $\phi_{ij} \sim \alpha\text{Uniform}(0.6, 0.8)$. The scaling parameter α controlled the frequency of which θ was thresholdable, and was set to 1, 0.75, 0.5, 0.25, and 0. As we empirically show later, $\alpha = 0.5, 0.25, 0$ corresponded to thresholdable conditions, while $\alpha = 1, 0.75$ corresponded to non-thresholdable conditions, generally. For the cutoffs τ_k, 40 equidistant points from 0 to 1 were used, and the sample size was set to $n = 1000$. Overall, this design resulted in $4 \times 5 = 20$ conditions, for which we conducted 100 replications per condition. Finally, we generated two data sets per replication, one for training purposes and one for testing purposes.

We tested the performance of the CT Algorithm with the other methods of factor analysis structure learning. These methods were the MLE with known structure (baseline), EFA, EFA-LASSO, and EFA-MCP. Note that the three EFA methods all require d as an input. To make a comparison as fair as possible, we implemented a modified version of the CT Algorithm for these methods. That is, the CT Algorithm was replicated except for

Step 4. For this step, instead of placing constraints on the entries of Λ, an EFA procedure was carried out instead, with $d = |\mathcal{C}|$. The algorithm continues as written thereafter, including the model selection procedure. Essentially, the CT Algorithm was used to give the EFA methods a set of d to work with and select from, while using their own procedures for structure learning.

The simulations were written in the `R` language (3.6.1) [17]. The `lavaan` package [18] was used in the estimation phases of the CT Algorithm, and was used to estimate the baseline MLE solution. For EFA, the `psych` package [19] was used to obtain MLE solutions for unconstrained Λ solutions. And finally, the LASSO and MCP variants of EFA were estimated with the `fanc` package [10, 11].

3.2 Outcomes

We collected five outcomes pertaining to the performance of true model recovery. In terms of model fit, we calculated BIC and testing log-likelihood differences from the baseline estimation procedure. For outcomes related to model structure, we collected Structural Hamming Distance (SHD) and the learned dimension ($\hat{d}$) of the latent variable vector.

In addition to model recovery performance, note that we varied the extent of which θ was thresholdable with a scaling parameter α. Therefore we calculated whether it was possible for $\widetilde{\Sigma}(\theta)$ and R to be fully and correctly sorted by τ_0, which we termed $\widetilde{\Sigma}(\theta)$ sortable and R sortable, respectively. That is, for every generated θ we determined if there existed a τ_0 to correctly sort the entries of $\widetilde{\Sigma}(\theta)$, and if that persisted after sampling variation for sorting the entries of R. In addition, we also calculated the maximum proportion of ρ_{ij} and r_{ij} that could be correctly sorted. These measures were termed ρ sortable and r sortable, respectively. We used a direct application of Theorem 1 to make these calculations.

Finally, we were interested in comparing the general computational efficiency of each method. To be agnostic towards the numerical idiosyncrasies between software packages, we simply counted the number of solutions each method estimated. For the CT Algorithm, this is simply the number of unique structures obtained by the sequence of τ_k. For EFA,

this translates to the number of unique d obtained by the sequence of τ_k. For EFA-LASSO and EFA-MCP, this is the number of tuning parameter combinations to search over, per unique d in the sequence of τ_k. The number of tuning parameters were left at the package defaults, which was 30 values of κ for EFA-LASSO (Equation 1.8) and 270 combinations of (κ, γ) for EFA-MCP (Equation 1.9).

3.3 Results

The results of the model fit outcomes are displayed in Figure 3.1 and the results of the structural outcomes are displayed in Figure 3.2. For BIC, the performance of the EFA methods ranked from best to worst as EFA-MCP, EFA-LASSO, and EFA, and was generally stable across α and the number of latent variables. However, the performance of the CT Algorithm varied across levels of α. When $\alpha = 0.5, 0.25, 0$, the CT Algorithm performed just as well as EFA-MCP, which was on par with the known structure MLE performance. For $\alpha = 0.75$, performance dropped slightly behind EFA-MCP but remained better than EFA-LASSO. When $\alpha = 1$ it began to perform worse than EFA-LASSO, but never worse than EFA.

For the testing log-likelihood, the performance of the EFA methods ranked from EFA-MCP, EFA, and EFA-LASSO, from best to worst. This rank order was generally stable across α and number of latent variables. Notable, there was a general drop in performance among all EFA methods at $\alpha = 1$. For the CT Algorithm, it performed the best along with EFA-MCP for $\alpha = 0.5, 0.25, 0$, across all number of latent variables. Once again, it displayed a slight performance drop at $\alpha = 0.75$, and performed the worst at $\alpha = 0$, along with EFA-LASSO.

For SHD, the results were similar to the pattern exhibited by BIC. The performance of EFA methods ordered from best to worst was EFA-MCP, EFA-LASSO, and EFA. This was stable across α and the number of latent variables. Once again, the performance of the CT Algorithm varied across levels of α. When $\alpha = 0.5, 0.25, 0$, the CT Algorithm performed on par with the known structure MLE performance, similar to EFA-MCP. For $\alpha = 0.75$, performance dropped slightly behind EFA-MCP, but better than EFA-LASSO.

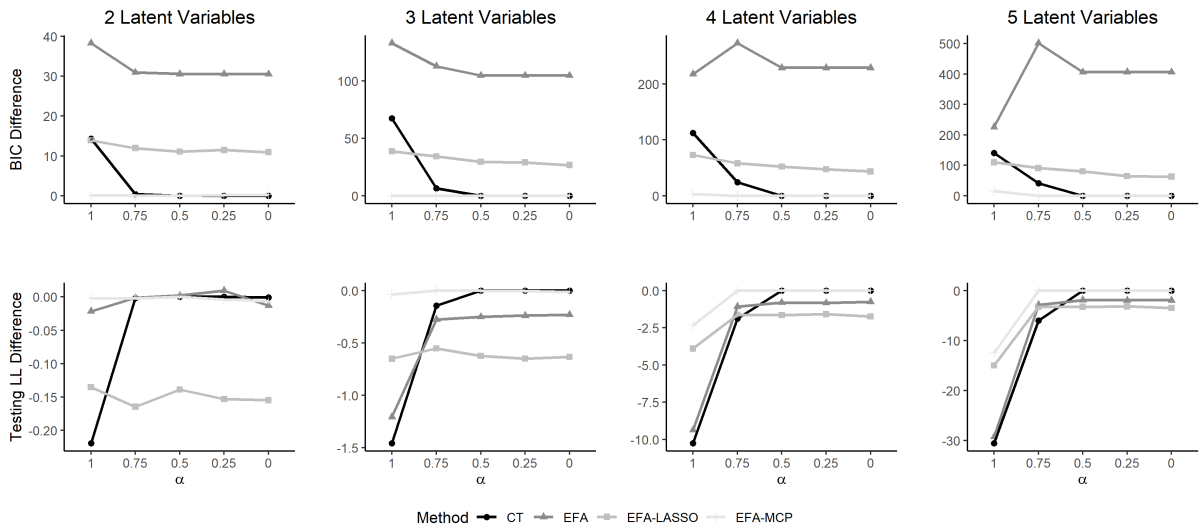

Figure 3.1: Averages for the model fit outcomes of the simulation study are displayed. All values had the known structure MLE subtracted for standardization. Hence, a value of zero corresponds to no difference vs. the known structure MLE method.

When $\alpha = 1$ it began to perform worse than EFA-LASSO, but not worse than EFA.

For the learned dimension, all methods performed nearly perfectly for $\alpha = 0.5, 0.25, 0$, across all numbers of latent variables. When $\alpha = 0.75$, both EFA and the CT Algorithm begin to perform slightly worse than EFA-MCP and EFA-LASSO, both of which maintain nearly perfect recovery rates. When $\alpha = 1$, generally all methods begin to drop in performance, with the CT Algorithm and EFA performing the worst.

The results of the sortability statistics are displayed in Table 3.1. In general, sortability decreases as both α and d increase. Specifically, when $\alpha = 1, 0.75$, the quantities $\widetilde{\Sigma}(\theta)$, ρ, R, and r all were almost never fully sortable. Conversely, they were almost always sortable when $\alpha = 0.5, 0.25, 0$. These results directly corroborate the results of the model fit and structure outcomes. That is, as the more sortable the correlations are, the better the CT Algorithm performs. As might be expected, the CT Algorithm performs perfectly if R is sortable.

For computational efficiency, the average number estimated solutions used is displayed in Table 3.2. In all practicality, the number of solutions depended on the method used. EFA used the lowest amount of solutions, with an overall average of 5.74. The CT algorithm used 9.98 on average, while the EFA-LASSO and EFA-MCP were orders of magnitude higher, using 172.31 and 1550.75 average solutions respectively.

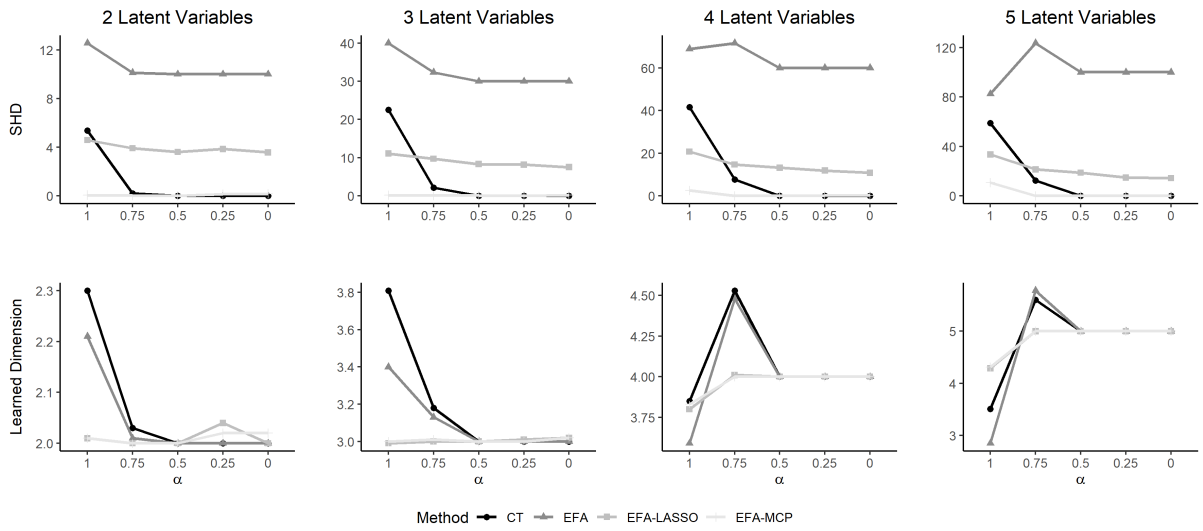

Figure 3.2: Averages for all structural outcomes of the simulation study are displayed.

3.4 Discussion

Unsurprisingly, much of the performance of the CT Algorithm depends on the thresholdability of θ. When thresholdability is met, the performance of the CT Algorithm performs just as well with the known structure MLE. Moreover, in terms of the number of solutions estimated, it is orders of magnitude more efficient than the competing method of EFA-MCP. While the CT Algorithm is robust to small violations of thresholdability (i.e., $\alpha = 0.75$), its performance suffers against large violations of thresholdability (i.e., $\alpha = 1$). However, α need not be too small in order to make θ thresholdable. As we have empirically shown, at $\alpha = 0.75$ thresholdability is not violated very often.

d	Sortable	$\alpha = 1.00$	$\alpha = 0.75$	$\alpha = 0.50$	$\alpha = 0.25$	$\alpha = 0$
2	$\widetilde{\Sigma}(\theta)$	0.450	1.000	1.000	1.000	1.000
	ρ	0.884	1.000	1.000	1.000	1.000
	R	0.230	0.920	1.000	1.000	1.000
	r	0.792	0.996	1.000	1.000	1.000
3	$\widetilde{\Sigma}(\theta)$	0.080	0.980	1.000	1.000	1.000
	ρ	0.841	1.000	1.000	1.000	1.000
	R	0.040	0.720	1.000	1.000	1.000
	r	0.708	0.986	1.000	1.000	1.000
4	$\widetilde{\Sigma}(\theta)$	0	1.000	1.000	1.000	1.000
	ρ	0.817	1.000	1.000	1.000	1.000
	R	0	0.400	0.990	1.000	1.000
	r	0.647	0.976	1.000	1.000	1.000
5	$\widetilde{\Sigma}(\theta)$	0.010	0.980	1.000	1.000	1.000
	ρ	0.802	1.000	1.000	1.000	1.000
	R	0	0.180	1.000	1.000	1.000
	r	0.631	0.975	1.000	1.000	1.000

Table 3.1: The average sortable statistics for $\widetilde{\Sigma}(\theta), \rho, R$, and r. Note that some of these numbers may not reflect exactly 1 or 0 due to rounding.

d	α	CT Algorithm	EFA	EFA-LA	EFA-MCP
2	1.00	10.40	4.58	137.4	1236.6
	0.75	10.54	4.34	130.2	1171.8
	0.50	9.87	4.18	125.4	1128.6
	0.25	9.24	4.20	126.0	1134.0
	0	7.61	4.10	123.0	1107.0
3	1.00	9.91	6.05	181.5	1633.5
	0.75	11.12	5.96	178.8	1609.2
	0.50	10.57	5.70	171.0	1539.0
	0.25	9.88	5.42	162.6	1463.4
	0	8.89	5.09	152.7	1374.3
4	1.00	8.40	5.95	178.5	1606.5
	0.75	11.07	6.67	200.1	1800.9
	0.50	11.17	6.56	196.8	1771.2
	0.25	10.27	6.34	190.2	1711.8
	0	9.70	6.04	181.2	1630.8
5	1.00	7.79	5.80	174.0	1566.0
	0.75	11.03	7.05	211.5	1903.5
	0.50	11.47	7.20	216.0	1944.0
	0.25	10.41	6.98	209.4	1884.6
	0	10.35	6.66	199.8	1798.2

Table 3.2: Average number of solutions estimated by each method.

Chapter 4

Real Data Example

4.1 Method

We examined a widely used factor analysis dataset comprised of intelligence test scores of middle school students [20]. The data consist of 9 variables designed to measure 3 factors of intelligence. These were a spatial factor (visual perception tasks), a verbal factor (paragraph comprehension, sentence completion, and word meaning), and a speed factor (speed tests of addition, counting groups of dots, and discrimination of straight and curved capitals). As in the simulation study, we applied the MLE (of the hypothesized structure), the CT Algorithm, EFA, EFA-LASSO, and EFA-MCP methods. Again, for a fair comparison, we implemented modified CT algorithms for each of the EFA methods as we did in the simulation study. We compared the learned dimension of the latent variable vector, the number of parmaters, BIC, and the CV log-likelihood (10 fold).

4.2 Results

The results are displayed in Table 4.1 and the structures are displayed in Figure 4.1. In terms of both BIC and CV log-likelihood, the results are similar across the CT Algorithm, EFA-LASSO, and EFA-MCP methods, all three being the best methods. The hypothesized MLE structure performed slightly behind these methods and EFA performed the worst.

All methods slightly differed in the structure learned. The CT Algorithm and EFA-MCP learned four latent variables, while EFA-LASSO learned three and EFA learned

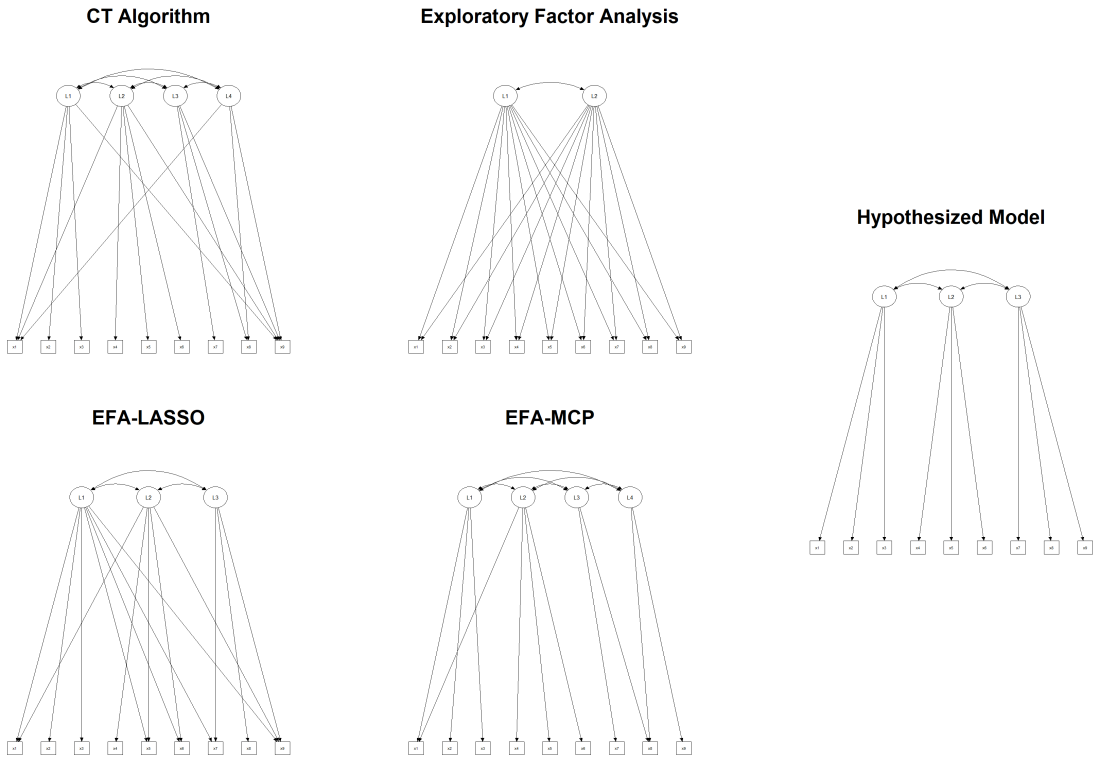

Figure 4.1: Path models for each method in real data example.

Method	$\hat{d}$	Parameters	BIC	CV Likelihood (10 Fold)
Hypothesis	3	21	7604.37	-3765.41
CT Algorithm	4	30	7602.52	-3749.60
EFA	2	28	7818.52	-3823.14
EFA-LASSO	4	30	7600.58	-3751.82
EFA-MCP	4	26	7581.61	-3751.37

Table 4.1: Results of real data example.

two. Notably, aside from EFA, the methods all suggested some additional structure for the 9th item (speed test of capital discrimination). EFA-MCP suggested an extra latent variable toward this item and the 8th item (speed of counting dots), EFA-LASSO suggested extra paths from the spatial and verbal factors toward this item, and the CT Algorithm suggested both.

Chapter 5

Concluding Remarks

The CT Algorithm is graphical method for learning factor analysis structures. In this article, we motivated the algorithm using thresholded correlation graphs, and established the conditions for structural identifiability, parameter uniqueness, and asymptotic consistency. In our simulation study, the CT Algorithm performs very well when the assumption of thresholdability is met, and also showed that this assumption may be quite plausible in practice. Further, the computational efficiency of the CT Algorithm is unrivaled relative to the EFA-LASSO and EFA-MCP methods, as it checks 10 and 100 times less models, respectively. Overall, the CT Algorithm may be a promising method of learning factor analysis models.

Bibliography

[1] Kenneth A. Bollen. *Structural equations with latent variables*. John Wiley & Sons, 1989.

[2] Robert I. Jennrich. Rotation to simple loadings using component loss functions: The oblique Case. *Psychometrika*, 71(1):173–191, 2006.

[3] Charles B. Crawford and George A. Ferguson. A general rotation criterion and its use in orthogonal rotation. *Psychometrika*, 35(3):321–332, 1970.

[4] Michael W. Browne. An overview of analytic rotation in exploratory factor analysis. *Multivariate Behavioral Research*, 36(1):111–150, 2001.

[5] Florian Scharf and Steffen Nestler. Should regularization replace simple structure rotation in exploratory factor analysis? *Structural Equation Modeling*, 26(4):576–590, 2019.

[6] Robert Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B*, 58(1):267–288, 1996.

[7] Jang Choi, Hui Zou, and Gary Oehlert. A penalized maximum likelihood approach to sparse factor analysis. *Statistics and Its Interface*, 3(4):429–436, 2010.

[8] Lipeng Ning and Tryphon T. Georgiou. Sparse factor analysis via likelihood and ℓ_1-regularization. In *Proceedings of the 50th IEEE Conference on Decision and Control*, pages 5188–5192, 2011.

[9] Cun Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, 38(2):894–942, 2010.

[10] Kei Hirose and Michio Yamamoto. Sparse estimation via nonconcave penalized likelihood in factor analysis model. *Statistics and Computing*, 25(5):863–875, 2014.

[11] Kei Hirose and Michio Yamamoto. Estimation of an oblique structure via penalized likelihood factor analysis. *Computational Statistics and Data Analysis*, 79:120–132, 2014.

[12] Max Auerswald and Morten Moshagen. How to determine the number of factors to retain in exploratory factor analysis: A comparison of extraction methods under realistic conditions. *Psychological Methods*, 24(4):468–491, 2019.

[13] Susan E. Whitely. Construct validity: Construct representation versus nomothetic span. *Psychological Bulletin*, 93(1):179–197, 1983.

[14] Ledyard R. Tucker. The objective definition of simple structure in linear factor analysis. *Psychometrika*, 20(3):209–225, 1955.

[15] Carel F. W. Peeters. Rotational uniqueness conditions under oblique factor correlation metric. *Psychometrika*, 77(2):288–292, 2012.

[16] Markus Kalisch and Peter Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007.

[17] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019.

[18] Yves Rosseel. lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2):1–36, 2012.

[19] William Revelle. *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois, 2019.

[20] K. J. Holzinger and F. Swineford. A study in factor analysis: The stability of a bi-factor solution. *Supplementary Educational Monographs*, 1939.