Computational Statistics and Data Analysis

Sparse logistic tensor decomposition for binary data

--Manuscript Draft--

Manuscript Number:	CSDA-D-22-00657
Article Type:	Research Paper
Section/Category:	II. Statistical Methodology for Data Analysis
Keywords:	Binary tensor; Majorization-Minimization; Sparsity; Tensor decomposition
Abstract:	Tensor data are increasingly available in many application domains. We develop several tensor decomposition methods for binary tensor data. Different from classical tensor decompositions for continuous-valued data with squared error loss, we formulate logistic tensor decompositions for binary data with a Bernoulli likelihood. To enhance the interpretability of estimated factors and improve their stability further, we propose sparse formulations of logistic tensor decomposition by considering l 1 -norm and l 0 -norm regularized likelihood. To handle the resulting optimization problems, we develop computational algorithms which combine the strengths of tensor power method and majorization-minimization (MM) algorithm. Through simulation studies, we demonstrate the utility of our methods in analysis of binary tensor data. To illustrate the effectiveness of the proposed methods, we analyze a dataset concerning nations and their political relations and perform co-clustering of estimated factors to find associations between the nations and political relations.

Sparse logistic tensor decomposition for binary data

Jianhao Zhang, Yoonkyung Lee
The Ohio State University, Columbus

Abstract

Tensor data are increasingly available in many application domains. We develop several tensor decomposition methods for binary tensor data. Different from classical tensor decompositions for continuous-valued data with squared error loss, we formulate logistic tensor decompositions for binary data with a Bernoulli likelihood. To enhance the interpretability of estimated factors and improve their stability further, we propose sparse formulations of logistic tensor decomposition by considering ℓ_1-norm and ℓ_0-norm regularized likelihood. To handle the resulting optimization problems, we develop computational algorithms which combine the strengths of tensor power method and majorization-minimization (MM) algorithm. Through simulation studies, we demonstrate the utility of our methods in analysis of binary tensor data. To illustrate the effectiveness of the proposed methods, we analyze a dataset concerning nations and their political relations and perform co-clustering of estimated factors to find associations between the nations and political relations.

Keywords: Binary tensor, Majorization-Minimization, Sparsity, Tensor decomposition.
2020 MSC: Primary 62H25, Secondary 62J12

1. Introduction

As a natural generalization of vectors and matrices, tensors have appeared frequently as a data form in many fields including social networks [1], recommender systems [2] and genomics [3]. As a result, tensor decomposition has attracted interests from machine learning and statistics with applications in

chemometrics [4], computer vision and signal processing. See [5] for a comprehensive review. In general, there are two approaches to decomposition of tensor data: CP decomposition and Tucker decomposition. CP decomposition is the abbreviation of canonical decomposition (CANDECOMP) and parallel factor (PARAFAC) analysis, which were proposed independently in psychometrics by [6] and [7]. Tucker decomposition [8] is a generalization of singular value decomposition for higher-order data, which includes CP decomposition as a special case.

There has been a growing body of literature that extends tensor decomposition methods for real valued data to other types such as tensors with binary outcomes or counts for dimensionality reduction and latent factor modeling. As a closely related problem, various types of principal component analysis (PCA) and matrix factorization methods have been developed for discrete non-Gaussian matrix data [9, 10, 11, 12, 13]. Extending the matrix factorization approach in [9] to binary tensor data, [14] considered a Tucker decomposition of the logit parameter tensor, and [15] considered a CP decomposition of the logit parameter tensor with max-norm constraint and investigated its statistical optimality. More generally, [16] proposed a CP decomposition of the natural parameter tensor for exponential family data.

In this paper, we focus on binary tensor data and consider settings where sparse latent factors are desired for modeling the underlying logit parameter tensor. Taking collaborative filtering as an example, how users interact with items in different contexts can be organized in the form of a tensor with user, item and context as three modes. Presence or absence of a user's interaction with an item in each context then makes up a binary tensor. Relations between the users and items could be context-specific, and they may involve a small subset of users, items, or contexts, rendering such relational factors sparse. Benefits of sparse factors and principal components for high dimensional matrix data have been well understood. Similar to sparse PCA [17, 18, 19], sparsity or regularization of factors is often desired in tensor decompositions. Sparsity in estimated factor matrices can provide a concise description of the latent

structure and improved understanding of the latent factors in relation to the observable features. For real valued tensors, [20] and [21] proposed sparse tensor decomposition methods based on the CP decomposition with an ℓ_1-norm penalty and ℓ_0-norm constraint on factor matrices, respectively. Besides, [22] considered sparse tensor decomposition with generalized lasso penalties on factor matrices to obtain smoothly varying factors. [23] proposed a sparse tensor singular value decomposition based on the Tucker decomposition and studied its statistical optimality.

To handle binary tensor data efficiently, we combine dimensionality reduction with regularization and selection of features and consider sparse decomposition of a logit parameter tensor. We propose a formulation of sparse logistic tensor decomposition by imposing an ℓ_1-norm penalty or ℓ_0-norm constraint on the factor matrices in the CP decomposition of a centered logit parameter tensor. Our approach naturally extends the sparse tensor decomposition [20, 21] to binary data and also extends the sparse logistic PCA [24, 25] to higher-order data.

Rank-one components in the decomposition of the underlying logit tensor generally correspond to multiplicative interactions among different modes. For this reason, sparse factors are well suited for modeling more local patterns of interactions involving only a subset of features along each mode. Such patterns can reveal interesting co-clustering structures between different modes. For binary matrices, [25] demonstrated the idea of co-clustering with a biclustering algorithm. Recently, [26] proposed a more general co-clustering analysis framework for exponential family tensor data.

Computationally, binary tensor decomposition entails maximization of the likelihood of a logit parameter tensor under a Bernoulli distribution assumption on the binary entries. Incorporating an ℓ_1-norm penalty or ℓ_0-norm constraint on the factors in the decomposition for encouraging sparsity leads to regularization of the likelihood. To solve the resulting optimization problems, we develop several novel computational algorithms. In a nutshell, we combine the strengths of tensor power method for tensor decompositions and majorization-

minimization (MM) algorithms, which have been successfully applied in logistic PCA and exponential family PCA for matrix data. By majorizing the negative log likelihood with a quadratic function, we turn the logistic CP decomposition problems with binary data into iterative applications of a plain CP decomposition with real-valued data. Thereby, we could make use of the tensor power method and its adaptations to sparse tensor decompositions for analysis of binary data. In particular, we adopt the tensor power method with alternating rank-one updates [27] in the MM approach to logistic tensor decomposition. Further, we incorporate the truncated power method [28, 21] for the ℓ_0-norm constrained logistic tensor decomposition and the soft-thresholding power method [29, 20] for the ℓ_1-norm penalized logistic tensor decomposition. We illustrate the utility of the proposed algorithms for analysis of binary tensor data.

The rest of the paper is organized as follows. Section 2 reviews tensor decomposition for real-valued tensor data. In Section 3, we introduce logistic CP decomposition for binary data using a Bernoulli likelihood and present sparse logistic CP decomposition using a regularized likelihood in Section 4. In addition, Sections 3 and 4 include MM-based computational algorithms for logistic tensor decomposition and sparse counterpart, respectively. Section 5 regards an extension of logistic CP decomposition for handling missing data and tensor completion. In Section 6, we discuss several criteria for choosing the rank of tensor decomposition and tuning parameters. We present simulation studies in Section 7 and demonstrate the effectiveness of sparse logistic CP decomposition with an application to nations data in Section 8. We summarize our contributions and list several directions for further investigation in Section 9.

2. Preliminaries

This section provides a technical background of tensor decomposition. Throughout the paper we focus on third-order tensor data, which are common in many applications. Methods for higher-order tensors can be developed similarly.

2.1. Notation

For $p \in \mathbb{N}$, we use $[p]$ to denote the index set $\{1, \ldots, p\}$. For two tensors $\mathcal{X}$ and $\mathcal{Y} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$, the inner product of $\mathcal{X}$ and $\mathcal{Y}$ is defined as $\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{\omega \in [p_1] \times [p_2] \times [p_3]} \mathcal{X}(\omega)\mathcal{Y}(\omega)$. This induces the Frobenius norm of $\mathcal{X}$ as $\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$, similar to the Frobenius norm of a matrix.

A tensor can be transformed into a matrix or *matricized* by unfolding it in a given mode. The mode-n matricization of a tensor $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ is denoted by $X_{(n)}$ for $n \in [3]$. For example, $X_{(1)} \in \mathbb{R}^{p_1 \times p_{-1}}$ with $p_{-1} = p_2 p_3$ is a matrix whose columns are the mode-1 fibers of $\mathcal{X}$. Multiplication of a tensor by a matrix in mode n is called the mode-n matrix product and denoted by $\times_n$. For example, the mode-1 matrix product of a tensor $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ and a matrix $U \in \mathbb{R}^{q \times p_1}$ is denoted by $\mathcal{X} \times_1 U \in \mathbb{R}^{q \times p_2 \times p_3}$.

The outer product of vectors $\boldsymbol{a} \in \mathbb{R}^I$ and $\boldsymbol{b} \in \mathbb{R}^J$ is denoted by $\boldsymbol{a} \circ \boldsymbol{b} \in \mathbb{R}^{I \times J}$. For a vector $\boldsymbol{v} \in \mathbb{R}^p$, $\|\boldsymbol{v}\|_2$ refers to the Euclidean norm, $\|\boldsymbol{v}\|_1$ the ℓ_1-norm, and $\|\boldsymbol{v}\|_0$ the number of non-zero entries in $\boldsymbol{v}$. The Kronecker product of matrices $A \in \mathbb{R}^{I \times J}$ and $B \in \mathbb{R}^{K \times L}$ is denoted by $A \otimes B \in \mathbb{R}^{IK \times JL}$ with $a_{ij}B$ in the ijth block. The Khatri-Rao product of matrices $A \in \mathbb{R}^{I \times K}$ and $B \in \mathbb{R}^{J \times K}$ is the columnwise Kronecker product of A and B denoted by $A \odot B = [\boldsymbol{a}_1 \otimes \boldsymbol{b}_1 \ \boldsymbol{a}_2 \otimes \boldsymbol{b}_2 \ \ldots \ \boldsymbol{a}_K \otimes \boldsymbol{b}_K] \in \mathbb{R}^{IJ \times K}$. The Hadamard product of matrices $A \in \mathbb{R}^{I \times J}$ and $B \in \mathbb{R}^{I \times J}$ is the elementwise product of A and B denoted by $A * B = [a_{ij}b_{ij}] \in \mathbb{R}^{I \times J}$. The Hadamard product of two tensors can be defined analogously.

The following property of the Kronecker product will be useful. Let $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \cdots \times p_N}$ and $A^{(n)} \in \mathbb{R}^{q_n \times p_n}$ for each $n \in [N]$. Then $\mathcal{Y} = \mathcal{X} \times_1 A^{(1)} \times_2 A^{(2)} \cdots \times_N A^{(N)}$ is equivalent to $Y_{(n)} = A^{(n)} X_{(n)} (A^{(N)} \otimes \cdots A^{(n+1)} \otimes A^{(n-1)} \cdots A^{(1)})^\top$ for every $n \in [N]$.

2.2. Tensor Decomposition

We briefly review tensor decomposition for real-valued tensor data. The idea of a CP decomposition [7, 6] is to factorize a tensor $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ into a sum

of rank-one component tensors of the form:

$$\mathcal{X} \approx \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r,$$

where $\boldsymbol{u}_r \in \mathbb{R}^{p_1}$, $\boldsymbol{v}_r \in \mathbb{R}^{p_2}$, $\boldsymbol{w}_r \in \mathbb{R}^{p_3}$, $d_{\max} = d_1 \geq \cdots \geq d_R = d_{\min} > 0$, and $\boldsymbol{u}_r^\top \boldsymbol{u}_r = 1$, $\boldsymbol{v}_r^\top \boldsymbol{v}_r = 1$, $\boldsymbol{w}_r^\top \boldsymbol{w}_r = 1$ for $r \in [R]$. Here R is the rank of the tensor $\mathcal{X}$. This can be formulated as a minimization problem with squared error loss:

$$\min_{\boldsymbol{d},U,V,W} \quad \|\mathcal{X} - \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r\|_F^2,$$

where $\boldsymbol{d} = (d_1, \ldots, d_R)^\top \in \mathbb{R}^R$ is a vector of weight parameters, and $U = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_R] = [u_{ir}] \in \mathbb{R}^{p_1 \times R}$, $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_R] = [v_{jr}] \in \mathbb{R}^{p_2 \times R}$, and $W = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_R] = [w_{kr}] \in \mathbb{R}^{p_3 \times R}$ are the factor matrices. It's worth noting that this CP decomposition has the property of essential uniqueness. That is, the columns of U, V and W are determined up to joint permutation.

[30, 31] provided a sufficient condition for the uniqueness of a three-way CP decomposition up to permutation and rescaling of rank-one tensors. Kruskal's condition is

$$k_U + k_V + k_W \geq 2R + 2,$$

where k_U, k_V and k_W are the Kruskal ranks of the matrices U, V and W.

The Tucker decomposition [8] aims at approximating a tensor $\mathcal{X} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ with a reduced core tensor $\mathcal{S} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ and factor matrices $U \in \mathbb{R}^{p_1 \times R_1}$, $V \in \mathbb{R}^{p_2 \times R_2}$, and $W \in \mathbb{R}^{p_3 \times R_3}$ as follows:

$$\mathcal{X} \approx \mathcal{S} \times_1 U \times_2 V \times_3 W.$$

Again this can be formulated as an optimization problem with squared error loss:

$$\min_{\mathcal{S},U,V,W} \quad \|\mathcal{X} - \mathcal{S} \times_1 U \times_2 V \times_3 W\|_F^2,$$

where $U^\top U = I_{R_1}$, $V^\top V = I_{R_2}$, and $W^\top W = I_{R_3}$. Here R_1, R_2 and R_3 are the numbers of components in the factor matrices U, V and W, respectively. The CP decomposition can be viewed as a special case of the Tucker decomposition

when the core tensor $\mathcal{S}$ is super-diagonal and $R_1 = R_2 = R_3 = R$. However, the Tucker decomposition doesn't have uniqueness since we can multiply factor matrices by nonsingular matrices and define a new core tensor and new factor matrices.

A comprehensive review of tensor decomposition is available in [5]. In this paper, we focus on the CP decomposition for tensors as it is a more natural choice for defining latent factors, and it is also more amenable to computation. The use of squared error loss can be regarded as an implicit normal distribution assumption for real-valued tensor data. We will employ an alternative loss for binary tensor data.

3. CP Decomposition for Binary Data

3.1. Logistic CP Decomposition

To handle tensors with dichotomous outcomes in many applications, we consider a binary tensor $\mathcal{X} = (x_{ijk}) \in \{0,1\}^{p_1 \times p_2 \times p_3}$, where each entry x_{ijk} encodes one of the two types of outcomes (e.g., absence or presence) with 0 or 1. We posit a generative model for the tensor and assume that x_{ijk} are realizations of mutually independent Bernoulli random variables with probability p_{ijk}, or $x_{ijk} \sim \text{Bernoulli}(p_{ijk})$.

For a Bernoulli random variable with probability parameter p, the probability mass function is $\Pr(X = x) = p^x(1-p)^{1-x}$, and $\log \Pr(X = x) = x \log \frac{p}{1-p} + \log(1-p)$. Reparametrizing with the logit parameter $\theta := \log \frac{p}{1-p}$, the log likelihood of θ based on x is given by $\ell(x;\theta) = x\theta - \log(1+\exp(\theta))$.

Letting $\theta_{ijk} := \log \frac{p_{ijk}}{1-p_{ijk}}$ for individual data entries x_{ijk} in the binary tensor $\mathcal{X}$, we derive the log likelihood of the logit parameter tensor $\Theta = (\theta_{ijk}) \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ as follows:

$$\begin{aligned} \ell(\mathcal{X};\Theta) &= \sum_{i,j,k} \left\{ x_{ijk}\theta_{ijk} - \log(1+\exp(\theta_{ijk})) \right\} \\ &= \langle \mathcal{X}, \Theta \rangle - \langle \mathbf{1}_{p_1 p_2 p_3}, \log(\mathbf{1}_{p_1 p_2 p_3} + \exp(\Theta)) \rangle, \end{aligned}$$

where $\mathbf{1}_{p_1p_2p_3} := \mathbf{1}_{p_1} \circ \mathbf{1}_{p_2} \circ \mathbf{1}_{p_3} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ is the tensor with all entries equal to one, and $\log(\cdot)$ and $\exp(\cdot)$ are taken as element-wise operators with tensors. Naturally extending the exponential family PCA for a data matrix in [9] to a higher-order tensor, we consider a CP decomposition of the logit parameter tensor Θ rather than the binary data tensor $\mathcal{X}$ itself and call it *logistic tensor decomposition*.

We include an offset term $\mu \in \mathbb{R}$ in logistic CP decomposition taken as an overall logit parameter value and consider the following decomposition:

$$\Theta = \mu \mathbf{1}_{p_1p_2p_3} + \sum_{r \in [R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r \tag{1}$$

or $\theta_{ijk} = \mu + \sum_{r \in [R]} d_r u_{ir} v_{jr} w_{kr}$, where the multiplicative part has rank R. Standard logistic CP decomposition in the literature assumes $\mu = 0$. For notational convenience, we use Θ_c to refer to $\sum_{r \in [R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r$, the portion of Θ adjusted by the offset.

To find a CP decomposition of Θ of rank R given the binary tensor $\mathcal{X}$, we maximize the log likelihood or equivalently minimize the negative log likelihood and formulate a *logistic CP decomposition* problem as follows:

$$\begin{aligned} \min_{\mu, \boldsymbol{d}, U, V, W} \quad & -\langle \mathcal{X}, \Theta \rangle + \langle \mathbf{1}_{p_1p_2p_3}, \log(\mathbf{1}_{p_1p_2p_3} + \exp(\Theta)) \rangle \\ \text{s.t.} \quad & \Theta = \mu \mathbf{1}_{p_1p_2p_3} + \sum_{r \in [R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r, \\ & \boldsymbol{u}_r^\top \boldsymbol{u}_r = 1, \boldsymbol{v}_r^\top \boldsymbol{v}_r = 1, \boldsymbol{w}_r^\top \boldsymbol{w}_r = 1, \text{ and } d_r > 0 \text{ for } r \in [R], \end{aligned} \tag{2}$$

where $\boldsymbol{d} \in \mathbb{R}^r$, $U \in \mathbb{R}^{p_1 \times R}, V \in \mathbb{R}^{p_2 \times R}$ and $W \in \mathbb{R}^{p_3 \times R}$ are the weight vector and factor matrices as defined before. While the objective function in (2), $-\ell(\mathcal{X}; \Theta)$, is convex in Θ, it is not convex in the factor matrices jointly, and this leads to a non-convex optimization problem with possibly multiple local optima. Further, the objective function in (2) is convex in each factor when the other two factors are fixed, but the unit norm constraint on each factor makes the problem non-convex.

3.2. Majorization-Minimization Approach

For logistic PCA and exponential family PCA involving similar optimization problems, Majorization-Minimization (MM) algorithms [32] have been used successfully. See [10, 24, 25, 12] for example. To solve the logistic CP decomposition problem in (2), we propose to majorize the objective function with a quadratic loss function and apply state-of-the-art algorithms for CP decomposition iteratively.

To majorize the negative likelihood, we first rewrite $\Pr(X = x) = p^x(1-p)^{1-x} = \sigma(q\theta)$ with the sigmoid function $\sigma(x) = \text{logit}^{-1}(x) = \{1+\exp(-x)\}^{-1}$ and $q = 2x-1$, and use the following tight and uniform quadratic majorization of $-\log\sigma(x)$ from [33, 10]:

$$\begin{aligned} -\log\sigma(x) &\leq -\log\sigma(y) + (\sigma(y)-1)(x-y) + \frac{2\sigma(y)-1}{4y}(x-y)^2 \\ &\leq -\log\sigma(y) + (\sigma(y)-1)(x-y) + \frac{1}{8}(x-y)^2, \end{aligned}$$

where the equalities hold when $x = y$. We will focus on the uniform bound (the second inequality) for computational convenience and leave the tight bound (the first inequality) for future study.

Let $\Theta^{[m]}$ be the estimate of Θ obtained in the mth step of the MM algorithm. Then, applying the above majorization to $-\log\sigma(q_{ijk}\theta_{ijk})$ at $\theta_{ijk}^{[m]}$ with $q_{ijk} = 2x_{ijk}-1$, we have

$$-\ell(\mathcal{X};\Theta) = -\sum_{i,j,k}\log\sigma(q_{ijk}\theta_{ijk}) \leq -\sum_{i,j,k}\log\sigma(q_{ijk}\theta_{ijk}^{[m]}) + \frac{1}{8}\sum_{i,j,k}(\theta_{ijk}-z_{ijk}^{[m]})^2,$$

where $z_{ijk}^{[m]} = \theta_{ijk}^{[m]} + 4(x_{ijk} - \sigma(\theta_{ijk}^{[m]}))$ or in the form of tensor

$$\mathcal{Z}^{[m]} = \Theta^{[m]} + 4(\mathcal{X} - \sigma(\Theta^{[m]})). \tag{3}$$

In other words, our problem turns into a simple CP decomposition problem with $\mathcal{Z}^{[m]}$ in the next iteration:

$$\begin{aligned} \min_{\mu,\boldsymbol{d},U,V,W} \quad & \frac{1}{8}\|\mathcal{Z}^{[m]} - \Theta\|_F^2 \\ \text{s.t.} \quad & \Theta = \mu\mathbf{1}_{p_1p_2p_3} + \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r, \\ & \boldsymbol{u}_r^\top\boldsymbol{u}_r = 1, \boldsymbol{v}_r^\top\boldsymbol{v}_r = 1, \boldsymbol{w}_r^\top\boldsymbol{w}_r = 1, \text{ and } d_r > 0 \text{ for } r \in [R]. \end{aligned} \tag{4}$$

The following theorem describes the convergence of the MM iterates of logit parameter tensor, $\Theta^{[m]}$, to a local minimum of $-\ell(\mathcal{X};\Theta)$, the loss function for logistic tensor decomposition. Proof of the theorem can be found in Appendix.

Theorem 1. *In the updating scheme of the MM algorithm,*

(i) The function $\frac{1}{8}\|\mathcal{Z}^{[m]} - \Theta\|_F^2$ *majorizes* $-\ell(\mathcal{X};\Theta)$ *at* $\Theta^{[m]}$ *up to a constant depending on* $\Theta^{[m]}$.

(ii) Let $\Theta^{[m]}$ *be a sequence obtained by minimizing the majorizing function. As the number of iterations grows,* $-\ell(\mathcal{X};\Theta^{[m]})$ *decreases, and* $\Theta^{[m]}$ *converges to a local minimum of* $-\ell(\mathcal{X};\Theta)$ *as* $m \to \infty$.

There exist various approaches for a CP decomposition of real-valued tensors. Among those, we will consider the Alternating Least Squares (ALS) method [6, 7] and the Tensor Power (TP) method [20] to solve the CP decomposition problem (4) in each iteration.

3.2.1. Alternating Least Squares Method

The ALS approach optimizes one factor matrix while treating all the other factor matrices as constants and alternates this optimization procedure over each of the factor matrices repeatedly until some convergence criterion is satisfied.

To solve the CP decomposition problem (4) at the mth step, we update parameters $(\mu, \boldsymbol{d}, U, V, W)$ in a block coordinate-wise manner. Given $(\boldsymbol{d}, U, V, W)$ at step m, we compute $\Theta_c^{[m]} = \sum_{r\in[R]} \widehat{d}_r^{[m]} \cdot \widehat{\boldsymbol{u}}_r^{[m]} \circ \widehat{\boldsymbol{v}}_r^{[m]} \circ \widehat{\boldsymbol{w}}_r^{[m]}$ first and update μ by taking the average of $\mathcal{Z}^{[m]} - \Theta_c^{[m]}$. With this updated $\widehat{\mu}^{[m+1]}$, we define

$$\mathcal{Z}_c^{[m]} = \mathcal{Z}^{[m]} - \widehat{\mu}^{[m+1]} \mathbf{1}_{p_1 p_2 p_3} \tag{5}$$

as offset adjusted working variables. To update $\boldsymbol{d}, U, V$, and W, we minimize

$$\|\mathcal{Z}_c^{[m]} - \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r\|_F^2. \tag{6}$$

Given factor matrices V and W, we define $A := U\mathrm{diag}(\boldsymbol{d})$, and rewrite the above problem in matrix form as a linear least squares problem for A as follows:

$$\min_A \quad \|Z_{c(1)}^{[m]} - A(W \odot V)^\top\|_F^2.$$

Then it can be shown that $\widehat{A} = Z_{c(1)}^{[m]}[(W \odot V)^\top]^\dagger$, where $[(W \odot V)^\top]^\dagger$ indicates the Moore-Penrose pseudo-inverse of $(W \odot V)^\top \in \mathbb{R}^{R \times p_2 p_3}$ [34]. To avoid the pseudo-inverse of a large matrix, we can rewrite the above solution as $\widehat{A} = Z_{c(1)}^{[m]}[(W \odot V)^\top]^\dagger = Z_{c(1)}^{[m]}(W \odot V)(W^\top W * V^\top V)^\dagger$, where $W^\top W * V^\top V \in \mathbb{R}^{R \times R}$ is typically smaller. To obtain $\widehat{U}$ and $\widehat{\boldsymbol{d}}$, we let $\widehat{d}_r = \|\widehat{\boldsymbol{a}}_r\|_2$ and $\widehat{\boldsymbol{u}}_r = \widehat{\boldsymbol{a}}_r / \|\widehat{\boldsymbol{a}}_r\|_2$ for $r \in [R]$. Similarly, we could define $B := V\text{diag}(\boldsymbol{d})$ and $C := W\text{diag}(\boldsymbol{d})$ and obtain $\widehat{B} = Z_{c(2)}^{[m]}[(U \odot W)^\top]^\dagger = Z_{c(2)}^{[m]}(U \odot W)(U^\top U * W^\top W)^\dagger$ and $\widehat{C} = Z_{c(3)}^{[m]}[(U \odot V)^\top]^\dagger = Z_{c(3)}^{[m]}(U \odot V)(U^\top U * V^\top V)^\dagger$ given other factor matrices. We normalize each column of $\widehat{A}, \widehat{B}$, and $\widehat{C}$ to unit length and update $\widehat{U}, \widehat{V}$, and $\widehat{W}$ alternatingly. Here we use Normalize(U) to denote the matrix with normalized columns of matrix U.

This MM approach with the ALS method (MM-ALS) is summarized in Algorithm 2 of Supplementary Section A. While this approach is easy to implement, it may take many iterations to converge, and there is no guarantee for convergence to a global minimum or even a stationary point of problem (4) according to [5]. Moreover, inversions of $R \times R$ matrices appear many times in Algorithm 2, so the ALS method will be computational expensive for large rank R. The results of MM-ALS method may contain local optima. We make comparisons with other methods in a simulation study later.

3.2.2. Tensor Power Method with Clustering

As a related approach to ALS, we consider iterative rank-one approximations of $\mathcal{Z}_c^{[m]}$ known as the tensor power method [20] to solve the rank-R tensor decomposition problem (6). It is related to the power method for eigendecomposition [34]. For a rank-one problem, the logit parameter tensor has representation of

$$\Theta = \mu \mathbf{1}_{p_1 p_2 p_3} + d \cdot \boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}. \tag{7}$$

Focusing on rank-one vectors, we aim to solve the following approximation problem:

$$\begin{aligned} \min_{d, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}} \quad & \|\mathcal{Z}_c^{[m]} - d \cdot \boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}\|_F^2 \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1, \boldsymbol{v}^\top \boldsymbol{v} = 1, \boldsymbol{w}^\top \boldsymbol{w} = 1, \text{ and } d > 0. \end{aligned} \tag{8}$$

Given $\boldsymbol{u} \in \mathbb{R}^{p_1}, \boldsymbol{v} \in \mathbb{R}^{p_2}$ and $\boldsymbol{w} \in \mathbb{R}^{p_3}$, the minimizer $d \in \mathbb{R}$ is analytically identified as $d = \mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^\top \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top$, and this allows us to rewrite the squared error objective function as $\|\mathcal{Z}_c^{[m]}\|_F^2 - \|\mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^\top \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top\|_F^2$. Then we can recast the above problem (8) as

$$\begin{aligned} \max_{\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}} \quad & \mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^\top \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1, \boldsymbol{v}^\top \boldsymbol{v} = 1, \boldsymbol{w}^\top \boldsymbol{w} = 1. \end{aligned} \tag{9}$$

See [5] for reference.

Again alternating among the three factors, we can update one factor at a time given the other factors to maximize the objective function. For instance, we first update $\boldsymbol{u}$ given $\boldsymbol{v}$ and $\boldsymbol{w}$, and then update $\boldsymbol{v}$ and $\boldsymbol{w}$ respectively in a similar way. Each subproblem in matricized form can be solved explicitly, and the following theorem states the solutions to the subproblems. Then d is updated with $\widehat{d} = \mathcal{Z}_c^{[m]} \times_1 \widehat{\boldsymbol{u}}^\top \times_2 \widehat{\boldsymbol{v}}^\top \times_3 \widehat{\boldsymbol{w}}^\top$. Finally, the logit parameter tensor is updated with $\Theta^{[m+1]} = \widehat{\mu}^{[m+1]} \mathbf{1}_{p_1 p_2 p_3} + \widehat{d} \cdot \widehat{\boldsymbol{u}} \circ \widehat{\boldsymbol{v}} \circ \widehat{\boldsymbol{w}}$, which then gives $\mathcal{Z}^{[m+1]}$ in (3).

Theorem 2. *The solutions to the subproblems of* (9) *are given by* $\widehat{\boldsymbol{u}} = \mathit{Normalize}(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top)$, $\widehat{\boldsymbol{v}} = \mathit{Normalize}(\mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^\top \times_3 \boldsymbol{w}^\top)$, $\widehat{\boldsymbol{w}} = \mathit{Normalize}(\mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^\top \times_2 \boldsymbol{v}^\top)$. *Each factor-wise update monotonically decreases the objective of* (8)*, and when iterated, the sequence of updates converges to a local solution of* (8).

For rank R component tensors, we repeat this rank-one decomposition multiple times with different initializations and produce R rank-one component tensors to combine. By repeating the tensor power method with differential initializations for L times, we can obtain L tuples stored in $S = \{(\widehat{d}_\tau, \widehat{\boldsymbol{u}}_\tau, \widehat{\boldsymbol{v}}_\tau, \widehat{\boldsymbol{w}}_\tau), \tau \in [L]\}$. Focusing on large estimates of $\widehat{d}_\tau$ and removing too similar tuples, we further cluster those L tuples into R clusters to produce R distinct rank-one component tensors. Then we reorder the R components with decreasing magnitude of $\widehat{d}_j$, which are the final output of our algorithm. We summarize this MM approximation with Tensor Power method (MM-TP) in Algorithm 1.

The Tensor Power method (TP) can be viewed as a rank-one version of ALS.

It only updates one column of each factor matrix in each iteration and does not require matrix inversion, which greatly reduces computational complexity compared to ALS [27]. Also, as a greedy method, the first few estimated factors by the tensor power method typically explain more deviance than ALS method [20].

Initialization is important for non-convex problems. In order to avoid local optima, Algorithm 1 contains a loop running for L different initializations. Because good initial values are not known in advance, we need to identify them. As suggested in [27], we develop an algorithm which clusters L tuples in $S = \{(\widehat{d}_\tau, \widehat{\boldsymbol{u}}_\tau, \widehat{\boldsymbol{v}}_\tau, \widehat{\boldsymbol{w}}_\tau), \tau \in [L]\}$ into R clusters $\{(\widehat{d}_j, \widehat{\boldsymbol{u}}_j, \widehat{\boldsymbol{v}}_j, \widehat{\boldsymbol{w}}_j), j \in [R]\}$ to obtain the final estimates. This clustering algorithm is described in Algorithm 5. Defining $\widehat{\Theta}_c = \sum_{j\in[R]} \widehat{d}_j \cdot \widehat{\boldsymbol{u}}_j \circ \widehat{\boldsymbol{v}}_j \circ \widehat{\boldsymbol{w}}_j$ and fixing this portion of Θ in (2), we can obtain the final estimate $\widehat{\mu}$ as the solution to (2).

4. Sparse CP Decomposition for Binary Data

4.1. Sparse Logistic CP Decomposition

Based on the formulation of logistic tensor decomposition, we consider two approaches which can produce sparse factor matrices with many zero entries. Similar to sparse logistic PCA and sparse tensor decomposition, when appropriate, sparse factor matrices can describe the latent structure more concisely, and nonzero entries can indicate important variables in each mode. To obtain sparse factor matrices for logistic tensor decomposition, we could add a penalty or constraint on the factor matrices. For example, the ℓ_1-norm penalty and ℓ_0-norm penalty have been successfully applied in the problems of penalized matrix decomposition [29, 28] and penalized tensor decomposition [20, 21].

Based on the logistic CP decomposition (2), we propose the following *sparse*

Algorithm 1 MM-Tensor Power algorithm for logistic CP decomposition

input: tensor $\mathcal{X}$, number of initializations L, and rank R.

Initialize with $\widehat{\mu}_\tau^{[0]}$ and $(\widehat{d}_\tau^{[0]}, \widehat{\boldsymbol{u}}_\tau^{[0]}, \widehat{\boldsymbol{v}}_\tau^{[0]}, \widehat{\boldsymbol{w}}_\tau^{[0]})$ where $\tau \in [L]$. Set $m = 0$.

for $\tau = 1$ **to** L **do**

  repeat

    Compute $\mathcal{Z}^{[m]}$ in (3).

    Update $\widehat{\mu}^{[m+1]}$.

    Compute $\mathcal{Z}_c^{[m]} = \mathcal{Z}^{[m]} - \widehat{\mu}^{[m+1]} \mathbf{1}_{p_1 p_2 p_3}$.

    repeat

      $\boldsymbol{u}_\tau = \text{Normalize}(\mathcal{Z}_c^{[m]} \times_2 (\boldsymbol{v}_\tau)^\top \times_3 (\boldsymbol{w}_\tau)^\top)$

      $\boldsymbol{v}_\tau = \text{Normalize}(\mathcal{Z}_c^{[m]} \times_1 (\boldsymbol{u}_\tau)^\top \times_3 (\boldsymbol{w}_\tau)^\top)$

      $\boldsymbol{w}_\tau = \text{Normalize}(\mathcal{Z}_c^{[m]} \times_1 (\boldsymbol{u}_\tau)^\top \times_2 (\boldsymbol{v}_\tau)^\top)$

    until converge

    Update $(\widehat{d}_\tau^{[m+1]}, \widehat{\boldsymbol{u}}_\tau^{[m+1]}, \widehat{\boldsymbol{v}}_\tau^{[m+1]}, \widehat{\boldsymbol{w}}_\tau^{[m+1]})$.

    $m \leftarrow m + 1$

  until converge

  Return $\widehat{\mu}_\tau$ and $(\widehat{d}_\tau, \widehat{\boldsymbol{u}}_\tau, \widehat{\boldsymbol{v}}_\tau, \widehat{\boldsymbol{w}}_\tau)$.

end for

Cluster $\{(\widehat{d}_\tau, \widehat{\boldsymbol{u}}_\tau, \widehat{\boldsymbol{v}}_\tau, \widehat{\boldsymbol{w}}_\tau), \tau \in [L]\}$ into R clusters $\{(\widehat{d}_j, \widehat{\boldsymbol{u}}_j, \widehat{\boldsymbol{v}}_j, \widehat{\boldsymbol{w}}_j), j \in [R]\}$ by Algorithm 5.

output: $\widehat{\mu}$ and R clusters $\{(\widehat{d}_j, \widehat{\boldsymbol{u}}_j, \widehat{\boldsymbol{v}}_j, \widehat{\boldsymbol{w}}_j), j \in [R]\}$.

logistic CP decomposition (SLCPD):

$$\begin{aligned}\min_{\mu,\boldsymbol{d},U,V,W} \quad & -\langle \mathcal{X}, \Theta \rangle + \langle \mathbf{1}_{p_1p_2p_3}, \log(\mathbf{1}_{p_1p_2p_3} + \exp(\Theta)) \rangle \\ \text{s.t.} \quad & \Theta = \mu \mathbf{1}_{p_1p_2p_3} + \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r, \\ & \boldsymbol{u}_r^\top \boldsymbol{u}_r = 1, \boldsymbol{v}_r^\top \boldsymbol{v}_r = 1, \boldsymbol{w}_r^\top \boldsymbol{w}_r = 1, \\ & p_1(\boldsymbol{u}_r) \le t_{1r}, p_2(\boldsymbol{v}_r) \le t_{2r}, p_3(\boldsymbol{w}_r) \le t_{3r}, \text{ and } d_r > 0 \text{ for } r \in [R].\end{aligned} \tag{10}$$

where $p_i(\cdot)$ for $i = 1, 2, 3$ are penalty functions for factors, and t_{ir} are tuning parameters. We consider two types of sparsity inducing penalty for each factor: the ℓ_1-norm and ℓ_0-norm penalty functions for $p_i(\cdot)$. For the ℓ_1-norm constrained formulation, $p_1(\boldsymbol{u}_r) = \|\boldsymbol{u}_r\|_1$, $p_2(\boldsymbol{v}_r) = \|\boldsymbol{v}_r\|_1$ and $p_3(\boldsymbol{w}_r) = \|\boldsymbol{w}_r\|_1$. For the ℓ_0-norm constrained formulation, $p_1(\boldsymbol{u}_r) = \|\boldsymbol{u}_r\|_0$, $p_2(\boldsymbol{v}_r) = \|\boldsymbol{v}_r\|_0$ and $p_3(\boldsymbol{w}_r) = \|\boldsymbol{w}_r\|_0$. This formulation naturally extends sparse logistic PCA [24] and binary matrix biclustering [25] to higher-order binary tensors.

To solve problem (10), we could update U, V and W in an alternative manner similar to ALS. When V and W are fixed, we could solve a regularized non-convex problem for U. And we could solve for V and W in an analogous manner. However, this regularized alternating least squares approach cannot guarantee that the solution is the global minimizer of the problem [20]. Instead, we consider a tensor power method and update each factor in an iterative block-wise manner.

4.2. Majorization-Minimization Approach with ℓ_1-norm Constraint

In order to simplify problem (10) with the ℓ_1-norm constraints and obtain a simple analytic solution, we relax the original non-convex equality constraints (e.g., $\boldsymbol{u}^\top \boldsymbol{u} = 1$), and consider the tensor decomposition problem with convex inequality constraints (e.g., $\boldsymbol{u}^\top \boldsymbol{u} \le 1$) [20]. Although the objective function is not convex in factor matrices jointly, it is convex in each factor matrix individually with all other factor matrices fixed.

Similar to logistic CP decomposition, we consider a rank-one problem to avoid local minima and the MM algorithm. For a rank-one problem, in the mth step of MM algorithm with $\mathcal{Z}_c^{[m]}$ defined in (5), we have the following relaxation:

$$\begin{aligned} \min_{d,\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}} \quad & \|\mathcal{Z}_c^{[m]} - d \cdot \boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}\|_F^2 \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} \le 1, \boldsymbol{v}^\top \boldsymbol{v} \le 1, \boldsymbol{w}^\top \boldsymbol{w} \le 1, d > 0, \\ & \|\boldsymbol{u}\|_1 \le c_1, \|\boldsymbol{v}\|_1 \le c_2, \|\boldsymbol{w}\|_1 \le c_3, \end{aligned} \tag{11}$$

where $c_i \ge 0$ for $i \in [3]$ are tuning parameters.

The constrained formulation (11) produces a feasible solution if $1 \le c_i \le \sqrt{p_i}$ and reduces to the un-regularized version when $c_i = \sqrt{p_i}$. If c_i are chosen appropriately, the solution to the relaxed problem still solves the original problem with the ℓ_2-norm constraints. To solve (11), we consider subproblems for each factor alternating among the three factors. Each time we update one factor given the other two factors. The following theorem states the solutions to the subproblems using the soft-thresholding operator $S(\cdot, \lambda) = \text{sign}(\cdot)(|\cdot| - \lambda)_+$ for $\lambda \ge 0$.

Theorem 3. *The solutions to the subproblems of* (11) *are given by* $\widehat{\boldsymbol{u}} = \textit{Normalize}(S(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top, \lambda_1))$, $\widehat{\boldsymbol{v}} = \textit{Normalize}(S(\mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^T \times_3 \boldsymbol{w}^T, \lambda_2))$, $\widehat{\boldsymbol{w}} = \textit{Normalize}(S(\mathcal{Z}_c^{[m]} \times_1 \boldsymbol{u}^T \times_2 \boldsymbol{v}^T, \lambda_3))$, *where* λ_1 *is the smallest nonnegative value such that* $\|\widehat{\boldsymbol{u}}\|_1 \le c_1$, *and* λ_2 *and* λ_3 *are defined analogously for* $\widehat{\boldsymbol{v}}$ *and* $\widehat{\boldsymbol{w}}$.

The values of λ_j can be chosen by a binary search [29]. This MM approximation with Tensor Soft-thresholding Power method (MM-TSP) is summarized in Algorithm 3 of Supplementary Section A.

4.3. Majorization-Minimization Approach with ℓ_0-norm Constraint

We propose to solve problem (10) with the ℓ_0-norm constraints in a manner similar to the tensor power method, and consider iterative rank-one sparse approximations of $\mathcal{Z}_c^{[m]}$ in (5). For a rank-one problem, in the mth step of MM

algorithm, we have the following problem:

$$\begin{aligned} \min_{d,\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}} \quad & \|\mathcal{Z}_c^{[m]} - d \cdot \boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}\|_F^2 \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1, \boldsymbol{v}^\top \boldsymbol{v} = 1, \boldsymbol{w}^\top \boldsymbol{w} = 1, d > 0, \\ & \|\boldsymbol{u}\|_0 \le s_1, \|\boldsymbol{v}\|_0 \le s_2, \|\boldsymbol{w}\|_0 \le s_3, \end{aligned} \tag{12}$$

where $s_i \le p_i$ for $i \in [3]$ are tuning parameters.

The constrained formulation (12) produces a feasible solution if $1 \le s_i \le p_i$. It reduces to the un-regularized problem without any constraint in each factor when $s_i = p_i$. Inspired by [28, 21], we could apply the tensor truncated power method in solving the above problem. Given $\boldsymbol{v}$ and $\boldsymbol{w}$, the constrained problem can be rewritten as a subproblem for $\boldsymbol{u}$:

$$\begin{aligned} \max_{\boldsymbol{u}} \quad & \boldsymbol{u}^\top (\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top) \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1, \|\boldsymbol{u}\|_0 \le s_1, \end{aligned}$$

where s_1 denotes the number of non-zero entries. This subproblem has explicit solution of

$$\widehat{\boldsymbol{u}} = \text{Normalize}(T(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top, s_1)). \tag{13}$$

Here $T(\cdot, s)$ is the truncation operator which keeps the largest $\lfloor s \rfloor$ entries of a vector in the absolute value and truncates the remaining entries to zero. We can update $\boldsymbol{v}$ and $\boldsymbol{w}$ in a similar manner. This MM approximation with Tensor Truncated Power method (MM-TTP) is summarized in Algorithm 4 of Supplementary Section.

5. Missing Data and Tensor Completion

In practice, missing data is common. To handle missing data, we extend our algorithms. Given data tensor $\mathcal{X}$ of size $p_1 \times p_2 \times p_3$, we let $\Omega = \{(i,j,k) \in [p_1] \times [p_2] \times [p_3] | x_{ijk} \text{ is observed}\}$ denote the index set of observed entries. Given $\Omega \subseteq [p_1] \times [p_2] \times [p_3]$, we can define the projection operation $\mathcal{P}_\Omega : \mathbb{R}^{p_1 \times p_2 \times p_3} \mapsto$

$\mathbb{R}^{p_1 \times p_2 \times p_3}$ as follows:

$$\mathcal{P}_\Omega(\mathcal{X}) = \begin{cases} x_{ijk} & \text{if } (i,j,k) \in \Omega \\ 0 & \text{if } (i,j,k) \notin \Omega. \end{cases}$$

$\mathcal{P}_\Omega$ replaces the missing entries in the data tensor $\mathcal{X}$ with zeros, and leaves the observed entries unchanged. For tenor $\mathcal{X}$ with the index set of observed data Ω, the sparse logistic CP decomposition minimizes the following objective function:

$$-\ell_{obs}(\mathcal{X}; \Theta) = -\sum_{(i,j,k)\in\Omega} \log \sigma(q_{ijk}\theta_{ijk})$$

which can be viewed as the observed data log likelihood. Let $\mathcal{H} = (h_{ijk}) \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ be a masking tensor such that $\mathcal{P}_\Omega(\mathcal{X}) = \mathcal{H} * \mathcal{X}$, where $*$ denotes the Hadamard product of two tensors. Then for partially observed data, we can redefine the following rank-R logistic CP decomposition problem:

$$\begin{aligned} \min_{\mu, \boldsymbol{d}, U, V, W} \quad & -\langle \mathcal{H} * \mathcal{X}, \Theta \rangle + \langle \mathcal{H}, \log(\mathbf{1}_{p_1 p_2 p_3} + \exp(\Theta)) \rangle \\ \text{s.t.} \quad & \Theta = \mu \mathbf{1}_{p_1 p_2 p_3} + \sum_{r\in[R]} d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r, \\ & \boldsymbol{u}_r^\top \boldsymbol{u}_r = 1, \boldsymbol{v}_r^\top \boldsymbol{v}_r = 1, \boldsymbol{w}_r^\top \boldsymbol{w}_r = 1, \text{ and } d_r > 0 \text{ for } r \in [R]. \end{aligned} \tag{14}$$

To solve the above problem, we modify the previous algorithms by introducing new working variables. We define new working variables $\mathcal{Y}^{[m]} = (y_{ijk}^{[m]})$ by filling in the missing values with fitted values based on the current estimate of logit parameter tensor Θ as follows:

$$y_{ijk}^{[m]} = \begin{cases} z_{ijk}^{[m]} & (i,j,k) \in \Omega \\ \theta_{ijk}^{[m]} & (i,j,k) \notin \Omega, \end{cases} \tag{15}$$

where $\mathcal{Z}^{[m]} = \Theta^{[m]} + 4(\mathcal{X} - \sigma(\Theta^{[m]}))$. In the mth step of MM approximation, the objective function of problem (2) turns into $\frac{1}{8}\|\mathcal{Y}^{[m]} - \Theta\|_F^2$. For sparse logistic CP decomposition, we could also replace the working variables $\mathcal{Z}^{[m]}$ with $\mathcal{Y}^{[m]}$ in the MM approximation of the regularized problem (10). We extend Theorem 1 to the missing data case and provide the proof in Appendix.

Theorem 4. *In the updating scheme of the MM algorithm with missing data,*

(i) The function $\frac{1}{8}\|\mathcal{Y}^{[m]}-\Theta\|_F^2$ *majorizes* $-\ell_{obs}(\mathcal{X};\Theta)$ *at* $\Theta^{[m]}$ *up to a constant depending on* $\Theta^{[m]}$.

(ii) Let $\Theta^{[m]}$ *be a sequence obtained by minimizing the majorizing function. As the number of iterations grows,* $-\ell_{obs}(\mathcal{X};\Theta^{[m]})$ *decreases, and it converges to a local minimum of* $-\ell_{obs}(\mathcal{X};\Theta)$ *as* $m \to \infty$.

Once we have $\widehat{\Theta}$ from observed data, we can use it for missing value prediction. After estimating $\widehat{\Theta}$ from observed data as in (14), we could predict the missing entries of $\mathcal{X}$ by using $\widehat{\mathcal{P}} = \text{logit}^{-1}(\widehat{\Theta})$, where $\widehat{\mathcal{P}} = (\widehat{p}_{ijk})$ is the tensor with estimated probabilities of Bernoulli random variables. For probability $\widehat{p}_{ijk} \geq 0.5$, we could impute missing x_{ijk} with one and zero otherwise. Similar ideas of such tensor completion for continuous data and binary data have been investigated in [35] and [15].

6. Selecting Rank and Tuning Parameters

Selecting an appropriate rank for tensor decomposition is an issue of practical importance. However, there has been few discussions in the literature. [20, 21, 15] derived a BIC heuristic to select the rank and degree of sparsity for CP decomposition. The consistency of BIC in binary tensor decomposition is unknown, but similar problems have been investigated by [36] in the context of relational learning. Alternatively, we could use cross-validation to choose the rank and tuning parameters, but cross-validation can be slow to carry out for high-dimensional tensors. [19, 29, 11] have used cross-validation to select the rank and sparsity tuning parameters for matrix decomposition problems. In this section, we investigate AIC, BIC, cross-validation and explained deviance as possible approaches to select the rank and tuning parameters. These approaches are illustrated with simulated data in Supplementary Section B.

6.1. BIC and AIC

As the tuning parameters c_i or s_i decrease, the estimated factor matrices become sparser, and the model for the underlyng logit parameter tensor becomes

simpler and easier to interpret. To reach a balance between model complexity and goodness of fit, we adopt the Bayesian information criterion (BIC) to select the optimal penalty parameter in sparse logistic tensor decomposition.

Given a prespecified set of rank values and penalty parameter values c_i or cardinality values s_i, we choose the combination of parameters $(\widehat{R}, \widehat{c}_1, \widehat{c}_2, \widehat{c}_3)$ or $(\widehat{R}, \widehat{s}_1, \widehat{s}_2, \widehat{s}_3)$ which minimizes the BIC criterion for sparse logistic CP decomposition $\widehat{\Theta}$ in (10):

$$\text{BIC} := -2\ell(\mathcal{X}; \widehat{\Theta}) + \log(p_1 p_2 p_3) \times \text{df}. \tag{16}$$

where $\text{df} = 1 + \|\widehat{U}\|_0 + \|\widehat{V}\|_0 + \|\widehat{W}\|_0 - 2R$. Here $\|\widehat{U}\|_0$ is the number of nonzero entries in matrix $\widehat{U}$ when the penalty parameter is $\widehat{c}_1$ or cardinality parameter is $\widehat{s}_1$, and $\|\widehat{V}\|_0$ and $\|\widehat{W}\|_0$ are defined analogously. Note that $\hat{\mu}$ has $df = 1$, $\widehat{\boldsymbol{d}}$ has $df = R$ and there are $3R$ constraints on $(\widehat{U}, \widehat{V}, \widehat{W})$, so the overall df of the logit tensor model can be taken as $1 + \|\widehat{U}\|_0 + \|\widehat{V}\|_0 + \|\widehat{W}\|_0 - 2R$. This is analogous to the way the model degrees of freedom is defined for sparse logistic PCA.

For the case with missing data, letting Ω denote the index set of observed entries, we use $|\Omega|$ rather than $p_1 p_2 p_3$, and the log likelihood for the observed entries in Ω is defined as $\ell(\mathcal{X}_\Omega; \Theta) := \sum_{ijk \in \Omega} \ell(x_{ijk}; \theta_{ijk})$ in the BIC. This leads to the following extended BIC:

$$\text{BIC}_\Omega := -2\ell(\mathcal{X}_\Omega; \widehat{\Theta}) + \log(|\Omega|) \times \text{df}.$$

Note that for fully observed data, $|\Omega| = p_1 p_2 p_3$ and $\ell(\mathcal{X}_\Omega; \Theta) = \ell(\mathcal{X}; \Theta)$, and the extended BIC reduces to (16).

To select the optimal tuning parameters, we could also consider minimizing the Akaike information criterion (AIC) for tensor decomposition:

$$\text{AIC} := -2\ell(\mathcal{X}; \widehat{\Theta}) + 2 \times \text{df}. \tag{17}$$

Given a fixed rank R, we first search for the optimal tuning parameters $(\widehat{c}_1, \widehat{c}_2, \widehat{c}_3)$ or $(\widehat{s}_1, \widehat{s}_2, \widehat{s}_3)$ by BIC/AIC, and then given the tuning parameter values, we seek the best rank R which minimizes BIC/AIC.

6.2. Cross-validation

Cross-validation has been proven to be useful in selecting tuning parameters in many settings. We could also select the rank and sparsity penalty parameters by an approach similar to cross-validation since our algorithms can handle missing data. However, compared with BIC or AIC, cross-validation is computationally more expensive. Many metrics could be used for cross-validation in binary tensor decomposition. We cross-validate each tuning parameter value by minimizing the negative log likelihood, $-\ell(\mathcal{X};\Theta)$ in this paper.

For a 5-fold cross-validation of rank R, we randomly split binary tensor entries into 5 folds: 4 folds are used for training and 1 fold is used for testing, where nonzero entries and zero entries are split separately with the same ratio. For a fixed rank R, we treat the test data as missing data and estimate Θ by minimizing the negative log likelihood $-\ell(\mathcal{X}_{\text{train}};\Theta)$ with the training data $\mathcal{X}_{\text{train}}$ only. Then for evaluation of $\widehat{\Theta}$, we calculate the negative log likelihood $-\ell(\mathcal{X}_{\text{test}};\widehat{\Theta})$ using the test data $\mathcal{X}_{\text{test}}$. We repeat the above process for five times and obtain the average negative log likelihood for each rank. A similar process can be used to select the levels of sparsity in factor matrices.

6.3. Explained Deviance

Alternatively, we could also use the explained deviance for determining the rank and tuning parameter values analogous to the use of explained total variance in standard PCA. The deviance of estimated logit parameter tensor $\widehat{\Theta}$ based on data $\mathcal{X}$ is defined as $D(\mathcal{X};\widehat{\Theta}) = -2(\ell(\mathcal{X};\widehat{\Theta}) - \ell(\mathcal{X};\Theta_S))$, where Θ_S is the logit parameter tensor of the saturated model. For binary tensor $\mathcal{X}$, $\Theta_S = \text{logit}(\mathcal{X})$, where $\text{logit}(x) = \log(\frac{x}{1-x})$ is taken elementwise, and thus $\ell(\mathcal{X};\Theta_S) = 0$. This leads to $D(\mathcal{X};\widehat{\Theta}) = -2\ell(\mathcal{X};\widehat{\Theta})$, and we have

$$D(\mathcal{X};\widehat{\Theta}) := -2\langle\mathcal{X},\widehat{\Theta}\rangle + 2\langle\mathbf{1}_{p_1p_2p_3},\log(\mathbf{1}_{p_1p_2p_3}+\exp(\widehat{\Theta}))\rangle.$$

For partially observed data $\mathcal{X}_\Omega$, the deviance can be expressed as

$$D(\mathcal{X}_\Omega;\widehat{\Theta}) := -2\langle\mathcal{H}*\mathcal{X},\widehat{\Theta}\rangle + 2\langle\mathcal{H},\log(\mathbf{1}_{p_1p_2p_3}+\exp(\widehat{\Theta}))\rangle$$

using the masking tensor $\mathcal{H}$ defined previously.

Let $\widehat{\Theta}_0 := \widehat{\mu}\mathbf{1}_{p_1p_2p_3}$ as an estimated tensor with offset term μ only, and for $r \in [R]$, let $\widehat{\Theta}_r := \widehat{\mu}\mathbf{1}_{p_1p_2p_3} + \sum_{i=1}^{r} \widehat{d}_i \cdot \widehat{\boldsymbol{u}}_i \circ \widehat{\boldsymbol{v}}_i \circ \widehat{\boldsymbol{w}}_i$ with the first r components. We call $D(\mathcal{X}_\Omega; \widehat{\Theta}_0)$ the null deviance and define the cumulative percentage of explained deviance of the first r components as

$$1 - \frac{D(\mathcal{X}_\Omega; \widehat{\Theta}_r)}{D(\mathcal{X}_\Omega; \widehat{\Theta}_0)}.$$

Similarly, we define the marginal percentage of explained deviance by the rth component as

$$\frac{D(\mathcal{X}_\Omega; \widehat{\Theta}_{r-1}) - D(\mathcal{X}_\Omega; \widehat{\Theta}_r)}{D(\mathcal{X}_\Omega; \widehat{\Theta}_0)}.$$

These criteria extend the proportion of total variance explained in real-valued tensors [20] to binary tensors. The same criteria have been considered in the context of binary matrix factorization [13].

We could also define the marginal deviance of the rth component as

$$D_r := D(\mathcal{X}; \widehat{\Theta}_{(r)}) = -2\langle \mathcal{X}, \widehat{\Theta}_{(r)} \rangle + 2\langle \mathbf{1}_{p_1p_2p_3}, \log(\mathbf{1}_{p_1p_2p_3} + \exp(\widehat{\Theta}_{(r)})) \rangle,$$

where $\widehat{\Theta}_{(r)} := \widehat{\mu}\mathbf{1}_{p_1p_2p_3} + \widehat{d}_r \cdot \widehat{\boldsymbol{u}}_r \circ \widehat{\boldsymbol{v}}_r \circ \widehat{\boldsymbol{w}}_r$. As index r corresponds to weight d_r ordered from largest to smallest, typically the rth marginal deviance will increase as r increases. Therefore the first component with largest weight d_1 will have the smallest marginal deviance D_1, and the last component with smallest weight d_R will have the largest marginal deviance D_R.

7. Simulation Study

We compare the proposed ℓ_0-norm constrained logistic tensor decomposition with Tensor Truncated Power (TTP) method, ℓ_1-norm constrained logistic tensor decomposition with Tensor Soft-thresholding Power (TSP) method, and un-regularized logistic tensor decomposition with Tensor Power (TP) method and Alternating Least Squares (ALS) method. We have implemented all methods in R [37] using the `rTensor` package [38] for efficient tensor computations.

Supplementary Section C describes implementational details including initialization and termination of the proposed algorithms as well as the clustering procedure.

7.1. Simulation Setup

To generate binary tensor data $\mathcal{X}$ with sparse logit parameters, we first specify the underlying logit parameter tensor Θ^* of size $p_1 \times p_2 \times p_3$. We consider the following four scenarios for the size and rank of Θ^*:

I. $p_1 = 1000, p_2 = 10, p_3 = 10,$ and $R = 1$; II. $p_1 = 1000, p_2 = 10, p_3 = 10,$ and $R = 2$;

III. $p_1 = 1000, p_2 = 100, p_3 = 10,$ and $R = 1$; IV. $p_1 = 1000, p_2 = 100, p_3 = 10,$ and $R = 2$.

In all simulation settings, we keep the level of sparsity equal in each dimension by setting the cardinality of nonzero entries as $p_{0j} = 0.2p_j$ for $j = 1, 2, 3$. With fixed dimensionality (p_1, p_2, p_3) and true rank R, we first generate independent and identically distributed standard Gaussian entries for three factor matrices $U \in \mathbb{R}^{p_1 \times R}, V \in \mathbb{R}^{p_2 \times R}$ and $W \in \mathbb{R}^{p_3 \times R}$. Then to induce sparsity in the factor matrices with fixed cardinality parameters (p_{01}, p_{02}, p_{03}), we truncate some entries in each column of U, V and W to zero. Finally, we normalize each column of U, V and W to get U^*, V^* and W^*.

To specify the weights $d_1^*, \ldots, d_R^*$ properly, we first consider their null values when $\Theta = (0)$ or $\mathcal{P} = (1/2)$, taken as the baseline noise level, and then determine their actual values proportionally. To find such null values, we first generate a $p_1 \times p_2 \times p_3$ binary tensor whose entries are mutually independent realizations from a Bernoulli distribution with $p = 1/2$. We carry out a rank-R logistic CP decomposition (2) of the binary tensor and calculate the average of R weights denoted by d_b. We repeat this process for 100 times and take the mean of d_b as the baseline noise level. Then using d_b, we could define the signal-to-noise ratio (SNR) as $\text{SNR}_r = d_r^*/d_b$ to determine the weights d_r^* for $r \in [R]$.

We consider different combinations of signal-to-noise ratio values: $\text{SNR} = (5, 3)$ when $R = 2$, and $\text{SNR} = 3$ when $R = 1$. With specified weights, we define

the logit parameter tensor as

$$\Theta^* = \mu^* \mathbf{1}_{p_1 p_2 p_3} + \sum_{r \in [R]} d_r^* \cdot \boldsymbol{u}_r^* \circ \boldsymbol{v}_r^* \circ \boldsymbol{w}_r^*,$$

which extends the spiked tensor model [39] to binary data. The overall logit parameter μ^* is set to zero. Because of the sparsity in $(\boldsymbol{u}_r^*, \boldsymbol{v}_r^*, \boldsymbol{w}_r^*)$ for $r \in [R]$, Θ^* is also sparse. Finally, we generate x_{ijk} from Bernoulli(p_{ijk}^*), where $p_{ijk}^* = \text{logit}^{-1}(\theta_{ijk}^*)$ for $i \in [p_1]$, $j \in [p_2]$, $k \in [p_3]$, and obtain a binary tensor $\mathcal{X} = (x_{ijk})$ with the corresponding probability tensor $\mathcal{P}^* = (p_{ijk}^*)$.

As for tuning parameters in this simulation study, we parameterize $c_j = \sqrt{p_j} \times c$ $(j = 1, 2, 3)$ for the ℓ_1-norm constraint and $s_j = p_j \times s$ for the ℓ_0-norm constraint. To make the ℓ_1-norm and ℓ_0-norm problems well-defined, we vary the ratio $c \in [\max_i \frac{1}{\sqrt{p_i}}, 1]$ for the ℓ_1-norm constraint and ratio $s \in [\max_i \frac{1}{p_i}, 1]$ for the ℓ_0-norm constraint. For the numerical results in Table 1, we considered a prespecified set of rank values $\{1, \ldots, 4\}$ and a range of values for the ratio parameters c and s and tuned the parameters using AIC. In simulation settings where true factors $\boldsymbol{u}^*, \boldsymbol{v}^*$ and $\boldsymbol{w}^*$ are known, we could set $(s_1, s_2, s_3) = (\|\boldsymbol{u}^*\|_0, \|\boldsymbol{v}^*\|_0, \|\boldsymbol{w}^*\|_0)$ and $(c_1, c_2, c_3) = (\|\boldsymbol{u}^*\|_1, \|\boldsymbol{v}^*\|_1, \|\boldsymbol{w}^*\|_1)$ as optimal tuning parameters in the ℓ_0-norm and ℓ_1-norm problems, respectively.

7.2. True Positive Rate and False Positive Rate

When the true logit parameters are sparse, we are interested in recovering the sparse pattern and selecting important nonzero features in the latent factors. The selection performance can be measured by the true positive rate (TPR): the proportion of correctly estimated non-zeros in the true parameter and the false positive rate (FPR): the proportion of true zeros that are incorrectly estimated to be nonzero. For an estimated factor matrix $\widehat{U}$, the TPR and FPR are defined as

$$\text{TPR}_{\widehat{U}} = \frac{1}{r} \sum_{r \in [R]} \frac{|\{i : (\widehat{\boldsymbol{u}}_r)_i \neq 0 \text{ and } (\boldsymbol{u}_r^*)_i \neq 0\}|}{|\{i : (\boldsymbol{u}_r^*)_i \neq 0\}|}$$

and

$$\text{FPR}_{\widehat{U}} = \frac{1}{r} \sum_{r \in [R]} \frac{|\{i : (\widehat{\boldsymbol{u}}_r)_i \neq 0 \text{ and } (\boldsymbol{u}_r^*)_i = 0\}|}{|\{i : (\boldsymbol{u}_r^*)_i = 0\}|},$$

respectively. For $\widehat{V}$ and $\widehat{W}$, $\mathrm{TPR}_{\widehat{V}}, \mathrm{TPR}_{\widehat{W}}, \mathrm{FPR}_{\widehat{V}}$ and $\mathrm{FPR}_{\widehat{W}}$ can be defined analogously. Then the overall TPR and FPR for $\widehat{\Theta}$ can be defined as $\mathrm{TPR}(\widehat{\Theta}) = (\mathrm{TPR}_{\widehat{U}} + \mathrm{TPR}_{\widehat{V}} + \mathrm{TPR}_{\widehat{W}})/3$ and $\mathrm{FPR}(\widehat{\Theta}) = (\mathrm{FPR}_{\widehat{U}} + \mathrm{FPR}_{\widehat{V}} + \mathrm{FPR}_{\widehat{W}})/3$.

7.3. Estimation Errors

To evaluate the accuracy of $\widehat{\Theta} = \widehat{\mu}\mathbf{1}_{p_1p_2p_3} + \sum_{r=1}^{R} \widehat{d}_r \cdot \widehat{\boldsymbol{u}}_r \circ \widehat{\boldsymbol{v}}_r \circ \widehat{\boldsymbol{w}}_r$ in recovering the true logit parameter tensor Θ^*, we look at its root mean squared error (RMSE) defined as

$$\mathrm{RMSE}(\widehat{\Theta}) = \frac{1}{\sqrt{p_1p_2p_3}} \|\widehat{\Theta} - \Theta^*\|_F.$$

To measure the quality of the estimated components and weights in tensor decomposition separately, we also calculate the mean vector estimation error and weight estimation error [27, 21]:

$$\text{Mean Error} = \frac{1}{3}\{\mathrm{ME}_{\widehat{U}} + \mathrm{ME}_{\widehat{V}} + \mathrm{ME}_{\widehat{W}}\},$$

and

$$\text{Weight Error} = \frac{\|\widehat{\boldsymbol{d}} - \boldsymbol{d}^*\|_2}{\|\boldsymbol{d}^*\|_2},$$

where $\mathrm{ME}_{\widehat{U}} = \frac{1}{R}\sum_{r\in[R]} \min\{\|\widehat{\boldsymbol{u}}_r - \boldsymbol{u}_r^*\|_2, \|\widehat{\boldsymbol{u}}_r + \boldsymbol{u}_r^*\|_2\}$, $\mathrm{ME}_{\widehat{V}}$ and $\mathrm{ME}_{\widehat{W}}$ are defined analogously. Operating characteristics of these evaluation metrics are illustrated with simulated data in Supplementary Section B.

7.4. Comparisons

We compare the proposed ℓ_0-norm logistic tensor decomposition with TTP method, ℓ_1-norm logistic tensor decomposition with TSP method, and unregularized logistic tensor decomposition method with ALS, TP and block relaxation (BR) [15] methods by calculating the average mean squared error, mean estimation error, weight estimation error and TPR/FPR over 20 random replicates simulated from the four scenarios. Table 1 presents the results with standard error in parentheses.

The columns for TPR and FPR indicate that the use of ℓ_1-norm and ℓ_0-norm constraints can lead to correct identification of nonzero entries in the logit tensor

with FPR close to 0 and TPR mostly 80% to 90%. Regularized estimates tend to have smaller errors on average in terms of RMSE, mean vector estimation error and weight estimation error. In particular, ℓ_1-norm regularized estimates with TSP method have minimum errors on the whole. Table 1 suggests that sparse logistic tensor decompositions indeed have better performance than their non-sparse counterpart when the true factor matrices are sparse.

Table 1: Comparisons of five logistic tensor decomposition methods under four simulation settings. The minimum value for each error measure is highlighted in bold, and the numbers in parentheses are standard errors.

Scenario	Method	RMSE(Θ)	Mean Error	Weight Error	TPR	FPR
1	BR	0.3683 (0.0675)	0.8651 (0.3601)	0.5649 (0.1469)	1 (0)	1 (0)
	ALS	0.3255 (0.0159)	0.5418 (0.0465)	0.5600 (0.0272)	1 (0)	1 (0)
	TP	0.3450 (0.0146)	0.6279 (0.0413)	0.5590 (0.0338)	1 (0)	1 (0)
	TSP	**0.1744** (0.0312)	**0.2832** (0.0265)	**0.0347** (0.0255)	0.8916 (0.0083)	0.0208 (0.0041)
	TTP	0.2877 (0.0028)	0.3922 (0.0021)	0.5405 (0.0029)	0.9000 (0.0000)	0.0250 (0.0000)
2	BR	0.6815 (0.0592)	0.6911 (0.1768)	0.6266 (0.1362)	1 (0)	1 (0)
	ALS	0.6504 (0.0309)	0.6932 (0.1740)	0.6283 (0.0170)	1 (0)	1 (0)
	TP	0.7353 (0.0276)	0.5534 (0.0347)	0.6049 (0.0238)	1 (0)	1 (0)
	TSP	**0.6491** (0.0627)	**0.4150** (0.1095)	**0.1994** (0.0068)	0.8791 (0.0291)	0.0208 (0.0020)
	TTP	0.7195 (0.0507)	0.6478 (0.2754)	0.5629 (0.0413)	0.7083 (0.2250)	0.0729 (0.2250)
3	BR	0.1545 (0.0112)	0.5192 (0.0420)	0.2957 (0.0444)	1 (0)	1 (0)
	ALS	0.1842 (0.0102)	0.6017 (0.0286)	0.4970 (0.0209)	1 (0)	1 (0)
	TP	0.1914 (0.0123)	0.6456 (0.0351)	0.4920 (0.0296)	1 (0)	1 (0)
	TSP	**0.1451** (0.0180)	**0.4197** (0.0495)	**0.1225** (0.0981)	0.7900 (0.0200)	0.0137 (0.0070)
	TTP	0.1523 (0.0101)	0.4296 (0.0149)	0.4289 (0.0294)	0.8708 (0.0225)	0.0322 (0.0225)
4	BR	0.3951 (0.0142)	0.4248 (0.0164)	0.3081 (0.0168)	1 (0)	1 (0)
	ALS	0.3930 (0.0179)	0.5733 (0.0162)	0.5841 (0.0113)	1 (0)	1 (0)
	TP	0.4344 (0.0162)	0.5560 (0.0158)	0.4561 (0.0182)	1 (0)	1 (0)
	TSP	**0.3911** (0.0152)	**0.3590** (0.0120)	**0.1985** (0.0218)	0.8591 (0.0066)	0.0053 (0.0038)
	TTP	0.4155 (0.0145)	0.4088 (0.0172)	0.4207 (0.0371)	0.9358 (0.0025)	0.0160 (0.0025)

We also compare the five methods computationally in terms of the number

Table 2: Comparison of the average run time (in seconds) and number of iterations for five logistic tensor decomposition methods under scenario 3.

Method	Time	Iteration Number	Time per Iteration
BR	59.1640 (0.8690)		
ALS	8.9147 (1.4783)	12.3 (4.2322)	2.7853 (0.7180)
TP	6.3479 (2.1182)	25.1 (6.7797)	0.2336 (0.0221)
TSP	10.8628 (4.4506)	36.0 (11.6961)	0.2545 (0.0490)
TTP	10.9305 (4.3180)	12.6 (0.7023)	0.9083 (0.3570)

of iterations and run time. For comparison, we ran all methods on the same data simulated from scenario 3 with rank-one decomposition using the same initialization and repeated the process 10 times. Table 2 shows the average run time in seconds and number of iterations. Their standard errors are in parentheses. The time for clustering is ignored. Computing was done on a laptop with a 2.7 GHz processor and 8 GB of memory. The times for TSP and TTP methods correspond to the optimal tuning parameters. Note that the run time varies with different initializations, which result in relatively large standard errors. According to Table 2, TP is faster than ALS, and TSP and TTP take more time to converge than the un-regularized TP method. For time per iteration, we find all methods based on tensor power method are faster than ALS. And among all tensor power methods, TTP is the slowest due to truncation. Notably, the block relaxation approach to logistic tensor decomposition using iteratively reweighted least squares method takes significantly longer than the proposed MM approach.

8. Analysis of Nations Data

This section investigates the efficacy of our methods on transposable binary tensor data. The nations dataset [40] we consider includes 14 countries and 54 binary predicates (e.g. *treaties*, *exports*) representing interactions between

countries. [41] thresholded each continuous variable at its mean and created a binary tensor of size $14 \times 14 \times 56$. This tensor consists of 56 political relations of 14 countries between 1950 and 1965. Each entry in the tensor (*nation, nation, relation*) indicates the presence or absence of a political relation. If nations i and j have relation k, $x_{ijk} = 1$ and otherwise $x_{ijk} = 0$.

The relationship between a nation and itself is not well defined, so we exclude the diagonal elements x_{iik} and treat them as missing entries. Overall the missing rate is 11.1%. This dataset has been investigated by [41, 1, 15]. Different from the previous analysis, we incorporate an offset term μ for the logit parameter tensor and impose a sparsity penalty on factor matrices.

The goals for this data analysis are grouping nations and relations, and identifying potential blocks of nations and relations. For example, we are interested in finding relations that exist significantly for certain groups of nations, or that can help distinguish different groups of nations. Supplementary Section D provides more details of data analysis.

8.1. Visualization of Factors

Due to the special structure of the nations data, we consider a special logistic CP decomposition with the same first two modes $\boldsymbol{u}$ and $\boldsymbol{v}$. More specifically, we impose the additional constraints $\boldsymbol{u}_r = \boldsymbol{v}_r$ for $r \in [R]$ in standard logistic CP decomposition in (2). To maintain this special structure, we keep the original update of $\boldsymbol{u}$ and $\boldsymbol{w}$ but set $\boldsymbol{v} = \boldsymbol{u}$ for the update of $\boldsymbol{v}$ in the tensor power method.

To decide a proper rank of logistic CP decomposition, we fit a rank-14 logistic CP decomposition first. We find that there are 3 weights much larger than other weights as shown in Figure 1 and the offset term μ is estimated to be -1.69. In Figure 1, the scree plot of marginal explained deviance suggests $R = 4$. Based on the information, we conclude that a rank-4 logistic CP decomposition is reasonable for the nations data.

To get a better understanding of factors, we apply K-means clustering on the estimated factors for the nations and relations, and visualize them in Figures

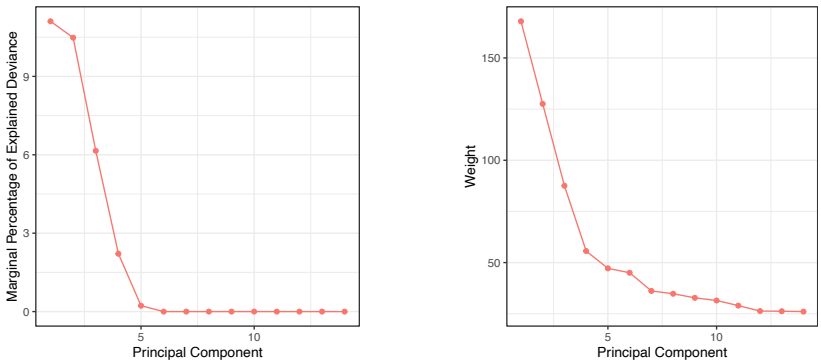

Figure 1: Explained deviance and weight versus the number of principal components for the nations data.

2 and 3. The estimated number of clusters can be determined by BIC criterion, where $K = 3$ is chosen for the nation factors, and $K = 5$ is chosen for the relation factors. The three clusters of nations contain communist countries (*USSR, Poland, Cuba, China*), western countries (*USA, UK, Netherlands, Brazil*), and neutral countries. The relations are grouped into five clusters. Three major clusters regard i) negative/hostile actions (e.g., *warning, protests, accusation, military actions*), ii) international partnerships through intergovernmental organizations and NGOs (e.g., *intergovorgs, relngo, ngo*), and iii) exports and population exchanges (e.g., *exportbook, exports, students, emigrants*).

8.2. Co-clustering of Nations and Relations

The two-way clustering methods in [42] and [25] have been proven to be successful for analysis of continuous and binary matrix data. The core idea of two-way clustering is imposing sparsity inducing penalties on the row score vector $\boldsymbol{u}_r$ and column loading vector $\boldsymbol{v}_r$ in the SVD of centered data matrix or centered logit parameter matrix. This could yield a checkerboard-like structure for each rank-one matrix $d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r$ for $r \in [R]$. By penalizing $\boldsymbol{u}_r$ and $\boldsymbol{v}_r$ in the rth component, the rows with nonzero u_{ir} are naturally clustered together, and the columns with nonzero v_{jr} are naturally clustered together. So penalization on both the score and loading vectors could simultaneously link sets of rows and

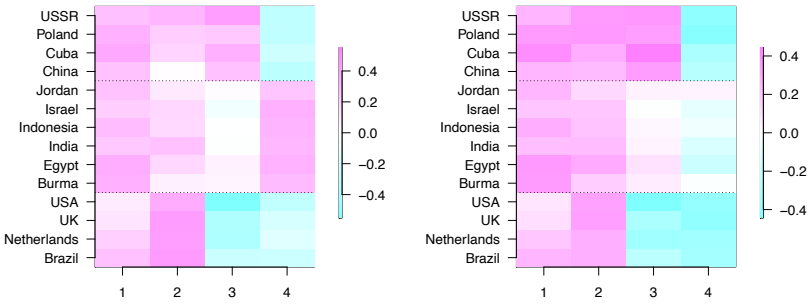

Figure 2: Heatmap of the estimated nation factors from the un-regularized model (left) and regularized model (right).

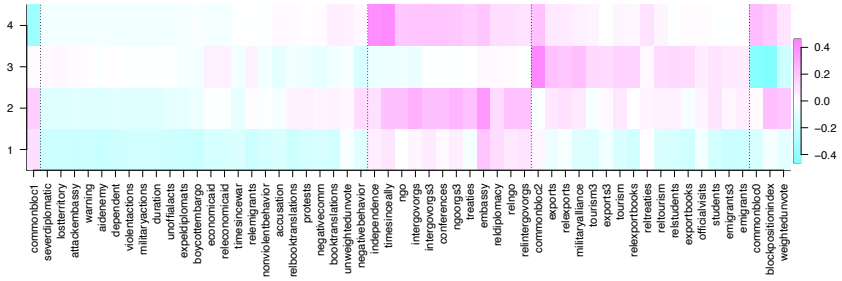

Figure 3: Heatmap of the estimated relation factors without regularization.

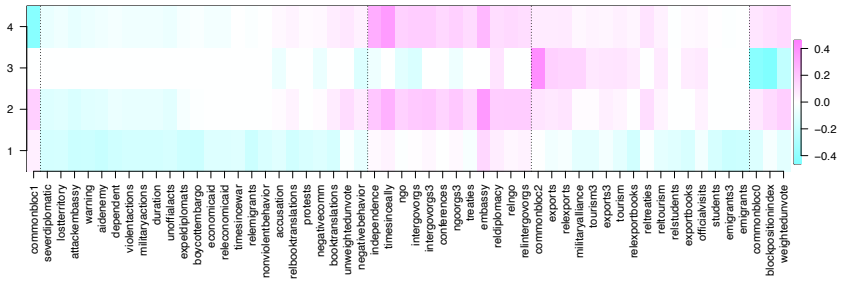

Figure 4: Heatmap of the estimated relation factors with regularization.

sets of columns together, and reveal some desirable row-column association. More generally, co-clustering methods could cluster related variables in each factor for tensor data.

The heatmap for the estimated nation factors is displayed in the left panel of Figure 2. It doesn't have a sparse pattern. By contrast, the heatmap for the relation factors in Figure 3 does suggest a potential benefit of sparsity because many entries are close to zero. Therefore we fit a rank-4 sparse logistic CP decomposition with an ℓ_0-norm constraint on the relation factors W. More specifically, this model imposes the constraints $\boldsymbol{u}_r = \boldsymbol{v}_r$ and $\|\boldsymbol{w}_r\|_0 \leq s_{3r}$, where tuning parameters s_{3r} control the number of nonzero entries in $\boldsymbol{w}_r$ for $r \in [R]$. The sparse estimated relation factors are presented in Figure 4, where the nonzero entries reveal important relations in each component.

Based on the estimated nation factors $\boldsymbol{u}_r$ and relation factors $\boldsymbol{w}_r$, we could build rank-one tensors $d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{v}_r \circ \boldsymbol{w}_r$ for $r \in [R]$. For visualization, we display rank-one matrices $d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{w}_r$ for $r \in [R]$. With given $r \in [R]$, $\boldsymbol{u}_r$ and $\boldsymbol{w}_r$ split the nations and relations into two or three clusters according to the sign of the entries, and therefore produce the clusters of nations and relations.

Figure 5 shows the heatmap of $d_r \cdot \boldsymbol{u}_r \circ \boldsymbol{w}_r$ for component 3. In the heatmap, the entries of $\boldsymbol{u}_r$ and $\boldsymbol{w}_r$ are arranged in increasing order. For the x-axis of the heatmap, factor $\boldsymbol{w}_r$ is displayed with entries in increasing order from left to

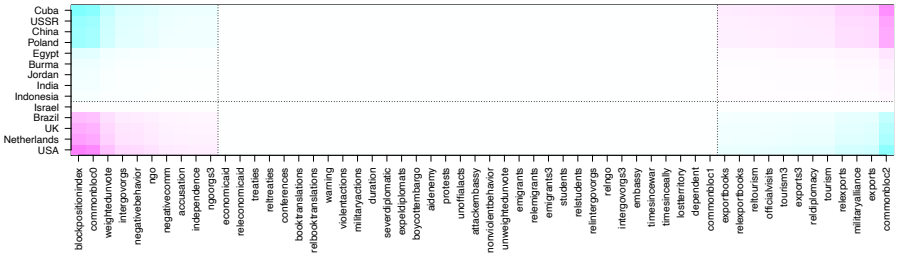

Figure 5: Heatmap of nation versus relation for component 3 with regularization.

right. For the y-axis of the heatmap, factor $\boldsymbol{u}_r$ is displayed with entries in increasing order from bottom to top.

For component 3 shown in Figure 5, negative values of the relation factor are associated with opposing common bloc membership, and positive values are associated with common bloc membership and resulting economic and cultural relations through exports and tourism. The nations are clearly separated into three groups. The nations with positive values are countries in the Communist bloc, and the nations with negative values are countries in the Western bloc. Neutral countries have almost zero values. We may as well consider imposing sparsity on this nations factor. As a form of interaction between nations and political relations, this component captures opposite political interactions between communist and western countries. It reveals a natural partition of the countries and clustering of relations as shown in Figure 5. This co-clustering suggested by the component is sensible, and the countries in the same cluster tend to share similar relation patterns.

While component 3 reflects a strong interaction between nations and political relations, the first two components mostly indicate main effects of the relations. The heatmaps of other components can be found in Supplementary Section D.

9. Conclusions and Discussion

In this paper, we have proposed several novel tensor decomposition methods for binary tensor data using the CP decomposition of a logit parameter tensor. We have mainly focused on three-way tensors in the paper, but similar methods can be developed for higher-order data. Starting with logistic CP decomposition, we have incorporated an ℓ_1-norm or ℓ_0-norm constraint on factors into the tensor decomposition formulation. To estimate factor matrices in logistic CP decomposition, we have developed computational algorithms that combine MM algorithm and variants of tensor power method. By imposing sparsity constraints on the factor matrices, we could identify and select important features in each factor. Sparse logistic CP decompositions can capture local multi-way interactions and therefore facilitate co-clustering of entities in different modes. Such co-clusters can reveal interesting associations between different modes.

There are several directions worth further investigation. As a structural element in logistic tensor decomposition, we have considered a constant offset term only. However, main effects along each mode are likely to be significant systematic elements in many applications as evidenced in the nations data analysis as well. From a modeling point of view, including additive main effects in the decomposition and using a small number of sparse rank-one tensors for multiplicative interactions will be a fruitful direction for extension. A similar logistic ANOVA model has been proposed for binary matrix data [43].

As another extension, we could generalize the current formulation with CP decomposition for binary data to a Tucker decomposition and develop a corresponding regularized version. Besides, we could replace the ℓ_0-norm and ℓ_1-norm penalties with general penalties such as fused lasso [44] in certain applications. For example, when one mode of a given tensor represents time points, smoothness in temporal factors might be desired.

Last but not least, we could develop similar methods for tensor decompositions in the natural parameter space for other types of exponential family data. For example, tensor data with counts or ratings as entries are common in rec-

ommender systems. Sparse Poisson or multinomial CP decompositions will be useful extensions of the current work.

Acknowledgments

This research was supported in part by the National Science Foundation Grants DMS-15-13566 and DMS-20-15490.

Appendix

Proof of Theorem 1. For part (i), we apply the uniform majorization of $-\log\sigma(x)$ to the negative log likelihood $-\ell(\mathcal{X};\Theta) = -\sum_{i,j,k}\log\sigma(q_{ijk}\theta_{ijk})$ at $\theta_{ijk}^{[m]}$ with $q_{ijk} = 2x_{ijk} - 1$ for each (i,j,k) and obtain

$$\sum_{i,j,k}\Big\{-\log\sigma(q_{ijk}\theta_{ijk}^{[m]}) - q_{ijk}(1-\sigma(q_{ijk}\theta_{ijk}^{[m]}))(\theta_{ijk}-\theta_{ijk}^{[m]}) + \frac{1}{8}(\theta_{ijk}-\theta_{ijk}^{[m]})^2\Big\}$$

as a uniform bound with tangency at $\theta_{ijk}^{[m]}$. By completing the squares and using $z_{ijk}^{[m]} = \theta_{ijk}^{[m]} + 4(x_{ijk} - \sigma(\theta_{ijk}^{[m]}))$, the majorizing function can be written as

$$-\ell(\mathcal{X};\Theta^{[m]}) - 2\sum_{i,j,k}\big(1-\sigma(q_{ijk}\theta_{ijk}^{[m]})\big)^2 + \frac{1}{8}\sum_{i,j,k}(\theta_{ijk}-z_{ijk}^{[m]})^2,$$

which equals $\frac{1}{8}\|\mathcal{Z}^{[m]}-\Theta\|_F^2$ up to an additive constant independent of Θ. This completes the proof of part (i).

For part (ii), we define the objective function $f(\Theta) := -\ell(\mathcal{X};\Theta)$ and its majorizing function $g(\Theta|\Theta^{[m]}) := -\ell(\mathcal{X};\Theta^{[m]}) - 2\|\mathbf{1}_{p_1p_2p_3} - \sigma(\mathcal{Q}*\Theta^{[m]})\|_F^2 + \frac{1}{8}\|\mathcal{Z}^{[m]}-\Theta\|_F^2$, where $\mathcal{Q}$ is a tensor with q_{ijk}. Then $f(\Theta) \le g(\Theta|\Theta^{[m]})$ for all Θ and $f(\Theta^{[m]}) = g(\Theta^{[m]}|\Theta^{[m]})$. Noting that $\Theta^{[m+1]} = \arg\min_{\Theta} g(\Theta|\Theta^{[m]})$, we have

$$f(\Theta^{[m+1]}) \le g(\Theta^{[m+1]}|\Theta^{[m]}) \le g(\Theta^{[m]}|\Theta^{[m]}) = f(\Theta^{[m]}),$$

which holds for each iteration m. Thus we conclude that the objective function $-\ell(\mathcal{X};\Theta^{[m]})$ decreases as the number of iterations grows.

The proof of local convergence relies on Ostrowski's theorem [45], which states that the sequence $\Theta^{[m+1]} = M(\Theta^{[m]})$ with an iteration map $M(\cdot)$ is locally

attracted to a local minimum $\Theta^{[\infty]}$ if the spectral radius of the differential of the iteration map $\rho[dM(\Theta^{[\infty]})]$ is strictly less than 1. The iteration map of the MM algorithm is defined as $M : \mathbb{R}^{p_1 \times p_2 \times p_3} \to \mathbb{R}^{p_1 \times p_2 \times p_3}$ with

$$M(\Theta) = \Theta - A(\Theta)^{-1} \nabla f(\Theta),$$

where $f(\Theta)$ is the objective function and $A(\Theta)$ is the Hessian of the surrogate function $g(\Theta|\Theta)$. At a local minimum $\Theta^{[\infty]}$, we calculate the differential of the MM algorithm map by vectorizing the tensor and get a diagonal matrix $dM(\Theta^{[\infty]}) = I_{p_1 p_2 p_3} - (\nabla^2 g(\Theta^{[\infty]}|\Theta^{[\infty]}))^{-1} \nabla^2 f(\Theta^{[\infty]})$. More specifically in the logistic tensor decomposition problem, we have $\nabla^2 f(\theta) = e^\theta/(1+e^\theta)^2 \in (0, 1/4]$ and $\nabla^2 g(\theta|\theta) = 1/4$, which lead to $1 - (\nabla^2 g(\theta|\theta))^{-1} \nabla^2 f(\theta) \in [0, 1)$. So $\rho[dM(\Theta^{[\infty]})] < 1$, and the sequence converges to a local minimum.

□

Proof of Theorem 2. Given $\boldsymbol{v}$ and $\boldsymbol{w}$, (9) can be rewritten as the following subproblem for $\boldsymbol{u}$:

$$\begin{aligned} \max_{\boldsymbol{u}} \quad & \boldsymbol{u}^\top (\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top) \\ \text{s.t.} \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1. \end{aligned}$$

The Lagrangian for the above problem is given by $L(\boldsymbol{u}, \gamma) = \boldsymbol{u}^\top (\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top) - \gamma(\boldsymbol{u}^\top \boldsymbol{u} - 1)$, where γ is a dual variable. The Karush–Kuhn–Tucker (KKT) conditions [46] imply that $\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top - 2\gamma \boldsymbol{u} = 0$ and $\boldsymbol{u}^\top \boldsymbol{u} - 1 = 0$. It's easy to verify that

$$\widehat{\boldsymbol{u}} = \text{Normalize}(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top)$$

satisfies the KKT conditions. We can update $\boldsymbol{v}$ and $\boldsymbol{w}$ in a similar manner.

When we optimize (9) in a block coordinate-wise manner, the objective function increases in each coordinate update. This implies that with the weight $d = \mathcal{Z}_c^{[m]} \times_1 \widehat{\boldsymbol{u}}^\top \times_2 \widehat{\boldsymbol{v}}^\top \times_3 \widehat{\boldsymbol{w}}^\top$, the objective of (8) monotonically decreases. Since it is bounded below by zero, the sequence of the values of the objective of (8) for the coordinatewise updates will converge. However, due to the non-convexity, this updating scheme can only produce a local optimum of (8).

□

Proof of Theorem 3. Using the same argument for deriving (9), given $\boldsymbol{v}$ and $\boldsymbol{w}$, the relaxed formulation in (11) can be rewritten as the following subproblem for $\boldsymbol{u}$:

$$\begin{aligned} \min_{\boldsymbol{u}} \quad & -\boldsymbol{u}^\top(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top) \\ \text{s.t.} \quad & \boldsymbol{u}^\top\boldsymbol{u} \le 1, \|\boldsymbol{u}\|_1 \le c_1. \end{aligned}$$

The Lagrangian for the subproblem is given by

$$L(\boldsymbol{u}, \lambda_1, \gamma) = -\boldsymbol{u}^\top(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top) + \lambda_1(\|\boldsymbol{u}\|_1 - c_1) + \gamma(\boldsymbol{u}^\top\boldsymbol{u} - 1),$$

where $\lambda_1 \ge 0$ and $\gamma \ge 0$ are dual variables. The KKT conditions imply that $-\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top + \lambda_1\Gamma + 2\gamma\boldsymbol{u} = 0$, $\lambda_1(\|\boldsymbol{u}\|_1 - c_1) = 0$ and $\gamma(\boldsymbol{u}^\top\boldsymbol{u} - 1) = 0$, where Γ is the subdifferential of $\|\boldsymbol{u}\|_1$ with $\Gamma_i = \text{sign}(u_i)$ if $u_i \neq 0$ and $\Gamma_i \in [-1, 1]$ otherwise. If $\gamma > 0$, then $\boldsymbol{u} = \frac{1}{2\gamma}S(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top, \lambda_1)$. From this expression of $\boldsymbol{u}$, we can verify that $\widehat{\boldsymbol{u}} = \text{Normalize}(S(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top, \lambda_1))$ and $\widehat{\gamma} = \|S(\mathcal{Z}_c^{[m]} \times_2 \boldsymbol{v}^\top \times_3 \boldsymbol{w}^\top, \lambda_1)\|_2/2$ simultaneously satisfy the KKT conditions. Unless $\lambda_1 = 0$ produces a feasible solution satisfying $\|\boldsymbol{u}\|_1 \le c_1$, we need to choose $\lambda_1 > 0$ such that $\|\boldsymbol{u}\|_1 = c_1$. Thus, λ_1 is the smallest nonnegative value such that $\|\boldsymbol{u}\|_1 \le c_1$. We can solve the subproblems for $\boldsymbol{v}$ and $\boldsymbol{w}$ in a similar manner.

□

Proof of Theorem 4. For part (i), first note that the complete data log likelihood is given by

$$\ell_{com}(\mathcal{X}; \Theta) = \sum_{(i,j,k)\in\Omega} \log\sigma(q_{ijk}\theta_{ijk}) + \sum_{(i,j,k)\notin\Omega} \log\sigma(q_{ijk}\theta_{ijk}),$$

and its conditional expectation given the observed data and the current parameter estimates is

$$Q(\Theta|\Theta^{[m]}) = \sum_{(i,j,k)\in\Omega} \log\sigma(q_{ijk}\theta_{ijk}) + \sum_{(i,j,k)\notin\Omega} \mathbb{E}[\log\sigma(q_{ijk}\theta_{ijk})|\mathcal{X}_{obs}, \Theta^{[m]}],$$

where $\mathcal{X}_{obs}$ denotes the observed tensor data. According to the standard EM algorithm, we could define

$$-\tilde{\ell}_{obs}(\mathcal{X};\Theta) := -Q(\Theta|\Theta^{[m]}) - \ell_{obs}(\mathcal{X};\Theta^{[m]}) + Q(\Theta^{[m]}|\Theta^{[m]})$$

and show that $-\tilde{\ell}_{obs}(\mathcal{X};\Theta)$ majorizes $-\ell_{obs}(\mathcal{X};\Theta)$ at $\Theta^{[m]}$.

In $-\tilde{\ell}_{obs}(\mathcal{X};\Theta)$, $-Q(\Theta|\Theta^{[m]})$ is the only term that depends on the unknown parameters, and it is decomposed into a term with the observed data and the other term with missing data. For $(i,j,k) \in \Omega$, $-\log\sigma(q_{ijk}\theta_{ijk})$ is majorized by $\frac{1}{8}(\theta_{ijk} - z_{ijk}^{[m]})^2$ up to an additive constant as shown before. For $(i,j,k) \notin \Omega$,

$$\begin{aligned}\mathbb{E}[\log\sigma(q_{ijk}\theta_{ijk})|\mathcal{X}_{obs},\Theta^{[m]}] &= \sigma(\theta_{ijk}^{[m]})\log\sigma(\theta_{ijk}) + (1-\sigma(\theta_{ijk}^{[m]}))\log(1-\sigma(\theta_{ijk})) \\ &= \sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]})\log(\sigma(q_{ijk}\theta_{ijk}))\end{aligned}$$

by the independence of the missing data and the observed data. By using the uniform quadratic majorization of $-\log\sigma(x)$, we have

$$\begin{aligned}&-\mathbb{E}[\log\sigma(q_{ijk}\theta_{ijk})|\mathcal{X}_{obs},\mathcal{X};\Theta^{[m]}] \\ &\leq \sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]})\Big[-\log\sigma(q_{ijk}\theta_{ijk}^{[m]}) - q_{ijk}(1-\sigma(q_{ijk}\theta_{ijk}^{[m]}))(\theta_{ijk}-\theta_{ijk}^{[m]}) + \frac{1}{8}(\theta_{ijk}-\theta_{ijk}^{[m]})^2\Big] \\ &= -\sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]})\log\sigma(q_{ijk}\theta_{ijk}^{[m]}) + (\theta_{ijk}^{[m]}-\theta_{ijk})\sum_{q_{ijk}=\pm 1}q_{ijk}\sigma(q_{ijk}\theta_{ijk}^{[m]})(1-\sigma(q_{ijk}\theta_{ijk}^{[m]})) \\ &\quad + \frac{1}{8}(\theta_{ijk}-\theta_{ijk}^{[m]})^2\sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]}) \\ &= -\sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]})\log\sigma(q_{ijk}\theta_{ijk}^{[m]}) + \frac{1}{8}(\theta_{ijk}-\theta_{ijk}^{[m]})^2.\end{aligned}$$

The last equality holds because $\sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]}) = 1$ and $\sum_{q_{ijk}=\pm 1}q_{ijk}\sigma(q_{ijk}\theta_{ijk}^{[m]})(1-\sigma(q_{ijk}\theta_{ijk}^{[m]})) = 0$ from the fact that $\sigma(-\theta) = 1-\sigma(\theta)$. Since $\sum_{q_{ijk}=\pm 1}\sigma(q_{ijk}\theta_{ijk}^{[m]})\log\sigma(q_{ijk}\theta_{ijk}^{[m]})$ is independent of Θ, we conclude that $-Q(\Theta|\Theta^{[m]})$ is majorized by $\frac{1}{8}\sum_{(i,j,k)\in\Omega}(\theta_{ijk} - z_{ijk}^{[m]})^2 + \frac{1}{8}\sum_{(i,j,k)\notin\Omega}(\theta_{ijk}-\theta_{ijk}^{[m]})^2$ up to an additive constant independent of Θ. So $\frac{1}{8}\|\mathcal{Y}^{[m]}-\Theta\|_F^2$ majorizes $-\tilde{\ell}_{obs}(\mathcal{X};\Theta)$ at $\Theta^{[m]}$. By the transitivity of majorization relation, we finally conclude that $\frac{1}{8}\|\mathcal{Y}^{[m]}-\Theta\|_F^2$ majorizes $-\ell_{obs}(\mathcal{X};\Theta)$ at $\Theta^{[m]}$.

The proof of part (ii) is similar to the arguments without missing data and thus omitted here.

□

References

[1] M. Nickel, V. Tresp, H.-P. Kriegel, A three-way model for collective learning on multi-relational data, in: Proceedings of the 28th International Conference on Machine Learning, 2011, pp. 809–816.

[2] X. Bi, A. Qu, X. Shen, Multilayer tensor factorization with applications to recommender systems, The Annals of Statistics 46 (6B) (2018) 3308–3333.

[3] M. Wang, J. Fischer, Y. S. Song, Three-way clustering of multi-tissue multi-individual gene expression data using semi-nonnegative tensor decomposition, The Annals of Applied Statistics 13 (2) (2019) 1103–1127.

[4] R. Bro, PARAFAC. Tutorial and applications, Chemometrics and Intelligent Laboratory Systems 38 (2) (1997) 149–171.

[5] T. G. Kolda, B. W. Bader, Tensor decompositions and applications, SIAM Review 51 (3) (2009) 455–500.

[6] R. Harshman, Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis, UCLA Working Papers in Phonetics 16 (1970) 1–84.

[7] J. D. Carroll, J.-J. Chang, Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition, Psychometrika 35 (3) (1970) 283–319.

[8] L. R. Tucker, Some mathematical notes on three-mode factor analysis, Psychometrika 31 (3) (1966) 279–311.

[9] M. Collins, S. Dasgupta, R. E. Schapire, A generalization of principal components analysis to the exponential family, in: Advances in Neural Information Processing Systems, 2002, pp. 617–624.

[10] J. de Leeuw, Principal component analysis of binary data by iterated singular value decomposition, Computational Statistics & Data Analysis 50 (1) (2006) 21 – 39, 2nd Special issue on Matrix Computations and Statistics. doi:https://doi.org/10.1016/j.csda.2004.07.010.

[11] M. Udell, C. Horn, R. Zadeh, S. Boyd, Generalized low rank models, Foundations and Trends® in Machine Learning 9 (1) (2016) 1–118.

[12] A. J. Landgraf, Y. Lee, Generalized principal component analysis: Projection of saturated model parameters, Technometrics 62 (4) (2020) 459–472.

[13] A. J. Landgraf, Y. Lee, Dimensionality reduction for binary data through the projection of natural parameters, Journal of Multivariate Analysis 180 (2020) 104668.

[14] J. Mažgut, P. Tiňo, M. Bodén, H. Yan, Dimensionality reduction and topographic mapping of binary tensors, Pattern Analysis and Applications 17 (3) (2014) 497–515.

[15] M. Wang, L. Li, Learning from binary multiway data: Probabilistic tensor decomposition and its statistical optimality, Journal of Machine Learning Research 21 (154) (2020) 1–38.

[16] D. Hong, T. G. Kolda, J. A. Duersch, Generalized canonical polyadic tensor decomposition, SIAM Review 62 (1) (2020) 133–163.

[17] I. T. Jolliffe, N. T. Trendafilov, M. Uddin, A modified principal component technique based on the lasso, Journal of Computational and Graphical Statistics 12 (3) (2003) 531–547.

[18] H. Zou, T. Hastie, R. Tibshirani, Sparse principal component analysis, Journal of Computational and Graphical Statistics 15 (2) (2006) 265–286.

[19] H. Shen, J. Z. Huang, Sparse principal component analysis via regularized low rank matrix approximation, Journal of Multivariate Analysis 99 (6) (2008) 1015–1034.

[20] G. Allen, Sparse higher-order principal components analysis, in: Artificial Intelligence and Statistics, 2012, pp. 27–36.

[21] W. W. Sun, J. Lu, H. Liu, G. Cheng, Provable sparse tensor decomposition, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 79 (3) (2017) 899–916.

[22] O. H. Madrid-Padilla, J. Scott, Tensor decomposition with generalized lasso penalties, Journal of Computational and Graphical Statistics 26 (3) (2017) 537–546.

[23] A. Zhang, R. Han, Optimal sparse singular value decomposition for high-dimensional high-order data, Journal of the American Statistical Association 114 (528) (2019) 1708–1725.

[24] S. Lee, J. Z. Huang, J. Hu, Sparse logistic principal components analysis for binary data, The Annals of Applied Statistics 4 (3) (2010) 1579.

[25] S. Lee, J. Z. Huang, A biclustering algorithm for binary matrices based on penalized Bernoulli likelihood, Statistics and Computing 24 (3) (2014) 429–441.

[26] G. Li, Generalized co-clustering analysis via regularized alternating least squares, Computational Statistics & Data Analysis (2020) 106989.

[27] A. Anandkumar, R. Ge, M. Janzamin, Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates, arXiv preprint arXiv:1402.5180.

[28] X.-T. Yuan, T. Zhang, Truncated power method for sparse eigenvalue problems, Journal of Machine Learning Research 14 (Apr) (2013) 899–925.

[29] D. M. Witten, R. Tibshirani, T. Hastie, A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis, Biostatistics 10 (3) (2009) 515–534.

[30] J. B. Kruskal, Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics, Linear Algebra and its Applications 18 (2) (1977) 95–138.

[31] J. B. Kruskal, Rank, decomposition, and uniqueness for 3-way and N-way arrays, Multiway Data Analysis (1989) 7–18.

[32] D. R. Hunter, K. Lange, A tutorial on MM algorithms, The American Statistician 58 (1) (2004) 30–37.

[33] T. S. Jaakkola, M. I. Jordan, Bayesian parameter estimation via variational methods, Statistics and Computing 10 (1) (2000) 25–37.

[34] G. H. Golub, C. F. van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, USA, 1996.

[35] E. Acar, D. M. Dunlavy, T. G. Kolda, M. Mørup, Scalable tensor factorizations for incomplete data, Chemometrics and Intelligent Laboratory Systems 106 (1) (2011) 41–56.

[36] C. Shi, W. Lu, R. Song, Determining the number of latent factors in statistical multi-relational learning, The Journal of Machine Learning Research 20 (1) (2019) 809–846.

[37] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2019).
URL https://www.R-project.org/

[38] J. Li, J. Bien, M. T. Wells, rTensor: An R package for multidimensional array (tensor) unfolding, multiplication, and decomposition, Journal of Statistical Software 87 (10) (2018) 1–31.

[39] A. Montanari, E. Richard, A statistical model for tensor PCA, in: Advances in Neural Information Processing Systems, 2014, pp. 2897–2905.

[40] R. J. Rummel, Dimensionality of nations project, Tech. rep., Department of Political Science, Hawaii University, Honolulu. (1968).

[41] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, N. Ueda, Learning systems of concepts with an infinite relational model, in: AAAI, Vol. 3, 2006, p. 5.

[42] M. Lee, H. Shen, J. Z. Huang, J. Marron, Biclustering via sparse singular value decomposition, Biometrics 66 (4) (2010) 1087–1095.

[43] Y. Jung, J. Z. Huang, J. Hu, Biomarker detection in association studies: modeling SNPs simultaneously via logistic ANOVA, Journal of the American Statistical Association 109 (508) (2014) 1355–1367.

[44] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, K. Knight, Sparsity and smoothness via the fused lasso, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67 (1) (2005) 91–108.

[45] K. Lange, Numerical analysis for statisticians, Springer Science & Business Media, 2010.

[46] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[47] H. Zhou, L. Li, H. Zhu, Tensor regression with applications in neuroimaging data analysis, Journal of the American Statistical Association 108 (502) (2013) 540–552.