Received: 16 October 2016 | Revised: 13 June 2017 | Accepted: 8 November 2017

DOI: 10.1002/nla.2136

RESEARCH ARTICLE

WILEY

Low-rank approximation of tensors via sparse optimization

Xiaofei Wang[1] | Carmeliza Navasca[2]

[1]Key Laboratory for Applied Statistics of MOE, School of Mathematics and Statistics, Northeast Normal University, Changchun 5268, China

[2]Department of Mathematics, University of Alabama at Birmingham, Birmingham AL, USA

Correspondence
Carmeliza Navasca, Department of Mathematics, University of Alabama at Birmingham, 1300 University Boulevard, Birmingham AL, USA.
Email: cnavasca@uab.edu

Funding information
National Natural Foundation of China, Grant/Award Number: 11401092

Summary

The goal of this paper is to find a low-rank approximation for a given *n*th tensor. Specifically, we give a computable strategy on calculating the rank of a given tensor, based on approximating the solution to an NP-hard problem. In this paper, we formulate a sparse optimization problem via an l_1-regularization to find a low-rank approximation of tensors. To solve this sparse optimization problem, we propose a rescaling algorithm of the proximal alternating minimization and study the theoretical convergence of this algorithm. Furthermore, we discuss the probabilistic consistency of the sparsity result and suggest a way to choose the regularization parameter for practical computation. In the simulation experiments, the performance of our algorithm supports that our method provides an efficient estimate on the number of rank-one tensor components in a given tensor. Moreover, this algorithm is also applied to surveillance videos for low-rank approximation.

KEYWORDS

l_1-regularization, low-rank approximation, proximal alternating minimization, sparsity

1 | INTRODUCTION

We have seen the success of the matrix SVD for several decades. However, in the advent of modern and massive data sets, even SVD has its limitation. Because tensors have been known to be a natural representation of higher-order and hierarchical dimensional data sets, we focus on the extension of low-rank matrix approximation to tensors. Tensors have received much attention in the recent years in the areas of signal processing,[1–3] computer vision,[4–7] neuroscience,[8,9] data science, and machine learning.[4,7,10,11] Most of these applications rely on decomposing a tensor data into its low-rank form to be able to perform efficient computing and to reduce memory requirements. This type of tensor decomposition into a sum of rank-one tensor terms is called the canonical polyadic (CP) decomposition; thus, it is viewed as a generalization of the matrix SVD. The generalization of matrix SVD to tensors is not unique. Another tensor decomposition is called the higher-order SVD,[10,12,13] which is a product of orthogonal matrices with a dense core tensor. A reduced higher-order SVD[13] computes for the low-rank approximation to the *R*-term input with a large number of terms such that the SVD applies only to the factor matrices. Higher-order SVD is considered as another extension of the matrix SVD.

Unlike the matrix case where the low-rank matrix approximation is afforded by truncating away *small* rank-one matrix terms,[14] discarding negligible rank-one tensor terms does not necessarily provide the best low-rank tensor approximation.[15] Moreover, most low-rank tensor algorithms do not provide an estimation on the tensor rank; an a priori tensor rank is often required to find the decomposition. Several theoretical results[16,17] on tensor rank can help, but they are limited to low multidimensional and low-order tensors so they are inapplicable to tensors in real-life applications. In fact, for a real data set, tensor rank is important. In a source apportionment data problem,[18] the tensor rank of the data provides the number of pollution source profiles to be identified. In this work, the focus is on finding an estimation of the

tensor rank and its rank-one tensor decomposition (CP) of a given tensor. There are several numerical techniques[1,2,10,19,20] for approximating a kth rank tensor into its CP decomposition, but they do not give an approximation of the minimum rank. There are algorithms[21,22] that give a tensor rank, but they are specific to symmetric tensor decomposition over the complex field using algebraic geometry tools.

Our proposed algorithm addresses two difficult problems for the CP decomposition: (a) one is that finding the rank of tensors is an NP-hard problem,[23] and (b) the other is that tensors can be ill posed[24] and have failed to have their best low-rank approximations.

The tensor rank problem is formulated as an l_0 minimization problem, that is,

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 \quad \text{subject to} \quad \mathcal{A} = [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R, \tag{1.1}$$

where $\sum_{r=1}^{R} \alpha_r \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r = [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R$ represents a sum of the outer products of the vectors $\mathbf{x}_r, \mathbf{y}_r$, and $\mathbf{z}_r$ for $r = 1, \ldots, R$. Here, $\|\boldsymbol{\alpha}\|_0$ corresponds to the number of nonzero coefficients in the sum. However, this problem formulation (1.1) is NP hard. Inspired by the techniques in compressive sensing,[25–27] we then consider an l_1-regularization formulation for tensor rank, as follows:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \quad \text{subject to} \quad \mathcal{A} = [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R. \tag{1.2}$$

Here, we denote $\|\boldsymbol{\alpha}\|_1 = \sum_{i=1}^{R} |\alpha_i|$. It is well known in the compressed sensing community that minimizing the ℓ_1 norm of the vector $\boldsymbol{\alpha}$ recovers the sparse solution of the linear system. In the presence of noise, the constraint, $\mathcal{A} = [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R$, is replaced with $\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F \leq \varepsilon$, where $\|\cdot\|_F$ is the Frobenius norm with $\|\mathcal{A}\|_F = (\sum \mathcal{A}_{ijk}^2)^{\frac{1}{2}}$. Moreover, to achieve a tensor decomposition and a tensor rank, we minimize over the factor matrices, $\mathbf{X}, \mathbf{Y}$, and $\mathbf{Z}$; thus, this minimization problem is considered as follows:

$$\min_{\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}} \frac{1}{2} \|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda \|\boldsymbol{\alpha}\|_1. \tag{1.3}$$

The ℓ_1-regularization achieves a good approximation of tensor rank due to the sparsity structure and its tractability. The regularization parameter λ in (1.3) can control the sparsity of the estimated coefficients.[28,29] In contrast to (1.1), the optimization problem (1.3) is a quadratic program. Moreover, the convex property on $\boldsymbol{\alpha}$ of (1.3) could make the computation more tractable than (1.1). In addition, the l_1-regularization term provides a restriction on the boundedness of the variables, thereby ameliorating the ill-posedness of the best low-rank approximation of tensors. For more tractable computing, an alternative multiblock constraint optimization[30] is implemented, which is similar to the technique discussed by Xu et al.[7] Because (1.3) is a minimization of the sum of a smooth term and a nonsmooth term, we consider the following optimization problem with smooth and nonsmooth terms:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \left\langle \mathbf{x} - \mathbf{x}^k, \nabla_{\mathbf{x}} f(\mathbf{x}^k) \right\rangle + \frac{t}{2} \|\mathbf{x} - \mathbf{x}^k\|^2 + g(\mathbf{x}) \right\}, \tag{1.4}$$

where f and g are the smooth and nonsmooth functions, respectively. Here, f is approximated at a given point $\mathbf{x}^k$.

1.1 | Contributions

Here, we list our contributions in this paper:

1. We develop an iterative technique for tensor rank approximation, given that the main objective function contains a nonsmooth l_1-regularization term. The proximal alternating minimization technique[7,30] has been adapted and rescaled for our tensor rank minimization problem. The solution of the optimization method, $\{\hat{\boldsymbol{\alpha}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}, \hat{R}\}$, generates a by-product $[\hat{\boldsymbol{\alpha}}; \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}]_{\hat{R}}$ as a low-rank approximation of $\mathcal{A}$; it provides a CP decomposition for an estimated rank $\hat{R}$.
2. We provide some theoretical results on the convergence of our algorithm. We show that the objective function satisfies a descent property in Lemma 3 and a subdifferential lower bound.[30] A monotonically decreasing objective function is ensured on the sequence generated by the algorithm. Furthermore, we point out that the sequence generated by the algorithm converges to a critical point of the objective function with indicator functions on the normalization constraint that all the columns of the factor matrices have length one.
3. For practical implementation, we provide a technique (as well as theoretical results) to find a suitable choice on the regularization parameter directly from the data. The regularization parameter choice has remained a very challenging problem[31–34] in applied inverse problems. Our technique is based on the probabilistic consistency of the sparsity in the classical model found in other works,[35,36] as follows:

$$\mathbf{b} = \mathbf{B}\boldsymbol{\theta}^* + \boldsymbol{\varepsilon},$$

where θ^* is a sparse signal, $\mathbf{B}$ is a design matrix, and ϵ is a vector of independent sub-Gaussian entries with mean zero and parameter σ^2. We show that to find the true sparsity structure with a high probability, the regularization parameter relies on two intrinsic parameters σ^2 and γ of the models, where σ^2 represents the variance of noise, and γ is the incoherence parameter[35] on design matrix $\mathbf{B}$. The relationship between the regularization and intrinsic parameters actually provides us a suggestion on how to choose a reasonable regularization parameter for practical computation. To illustrate the performance of this low-rank approximation method, our experiment consists of four parts. In the first part, we show the relationship between the regularization parameter and the estimated rank. In the second part, we estimate the number of rank-one components for given tensors by adaptively selecting the regularization parameter λ. In the third one, we compare our algorithm with a modified alternating least-squares algorithm. In the last one, we handle the real surveillance video data.

1.2 | Organization

Our paper is organized as follows. The discussion is limited to third-order tensors, but the formulation works in any nth order tensors. In Section 2, we provide some notations and terminologies used throughout this paper. In Section 3, we formulate an l_1-regularization optimization to the low-rank approximation of tensors. In Section 4, we propose an algorithm to solve this l_1-regularization optimization by using a rescaling version of the proximal alternating minimization technique. In Section 5, we discuss the probabilistic consistency of the sparse optimal solution and give a suggestion on how to choose the regularization parameter. The numerical experiments in Section 6 consist of simulated and real data sets. Finally, our conclusion and future work are given in Section 7.

2 | NOTATION

We denote a vector by a bold lowercase letter $\mathbf{a}$. The bold uppercase letter $\mathbf{A}$ represents a matrix, and the symbol of tensor is a calligraphic letter $\mathcal{A}$. Throughout this paper, we focus on third-order tensors $\mathcal{A} = (a_{ijk}) \in \mathbb{R}^{I\times J\times K}$ of the three indices $1 \le i \le I, 1 \le j \le J$, and $1 \le k \le K$, but all the methods proposed here can be also applied to tensors of arbitrary high order.

A third-order tensor $\mathcal{A}$ has column, row, and tube fibers, which are defined by fixing every index but one and are denoted by $\mathbf{a}_{:jk}$, $\mathbf{a}_{i:k}$, and $\mathbf{a}_{ij:}$, respectively. Correspondingly, we can obtain three kinds $\mathbf{A}_{(1)}$, $\mathbf{A}_{(2)}$, and $\mathbf{A}_{(3)}$ of matricization of $\mathcal{A}$ according to respectively arranging the column, row, and tube fibers to be columns of matrices. We can also consider the vectorization for $\mathcal{A}$ to obtain a row vector $\mathbf{a}$ such that the elements of $\mathcal{A}$ are arranged according to k varying faster than j and j varying faster than i, that is, $\mathbf{a} = (a_{111}, \dots, a_{11K}, a_{121}, \dots, a_{12K}, \dots, a_{1J1}, \dots, a_{1JK}, \dots)$.

The outer product $\mathbf{x} \circ \mathbf{y} \circ \mathbf{z} \in \mathbb{R}^{I\times J\times K}$ of three nonzero vectors $\mathbf{x}, \mathbf{y}$, and $\mathbf{z}$ is a rank-one tensor with elements $x_i y_j z_k$ for all the indices. A CP decomposition of $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$ expresses $\mathcal{A}$ as a sum of rank-one outer products, as follows:

$$\mathcal{A} = \sum_{r=1}^{R} \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r, \tag{2.1}$$

where $\mathbf{x}_r \in \mathbb{R}^I, \mathbf{y}_r \in \mathbb{R}^J, \mathbf{z}_r \in \mathbb{R}^K$ for $1 \le r \le R$. Every outer product $\mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$ is called as a rank-one component, and the integer R is the number of rank-one components in tensor $\mathcal{A}$. The minimal number R such that the decomposition (2.1) holds is the rank of tensor $\mathcal{A}$, which is denoted by $\text{rank}(\mathcal{A})$. For any tensor $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$, $\text{rank}(\mathcal{A})$ has an upper bound $\min\{IJ, JK, IK\}$.[16]

The CP decomposition (2.1) can be also written as follows:

$$\mathcal{A} = \sum_{r=1}^{R} \alpha_r \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r, \tag{2.2}$$

where $\alpha_r \in \mathbb{R}$ is a rescaling coefficient of rank-one tensor $\mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$ for $r = 1, \dots, R$. For convenience, we let $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_R) \in \mathbb{R}^R$ and $[\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R = \sum_{r=1}^{R} \alpha_r \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$ in (2.2), where $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_R) \in \mathbb{R}^{I\times R}, \mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_R) \in \mathbb{R}^{J\times R}$, and $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_R) \in \mathbb{R}^{K\times R}$ are called the factor matrices of tensor $\mathcal{A}$. We impose a normalization constraint on factor matrices such that each column is normalized to length one,[10,37] which is denoted by $\mathbf{N(X,Y,Z)} = 1$. For most alternating optimization algorithms for tensors, *flattening* the tensor (matricization) is necessary to be able to break down the problem into several subproblems. Here, we describe a standard approach for a matricizing of a tensor. The Khatri–Rao product

of two matrices $\mathbf{X} \in \mathbb{R}^{I\times R}$ and $\mathbf{Y} \in \mathbb{R}^{J\times R}$ is defined as follows:

$$\mathbf{X} \odot \mathbf{Y} = (\mathbf{x}_1 \otimes \mathbf{y}_1, \ \dots \ , \mathbf{x}_R \otimes \mathbf{y}_R) \in \mathbb{R}^{IJ\times R},$$

where the symbol "$\otimes$" denotes the Kronecker product, as follows:

$$\mathbf{x} \otimes \mathbf{y} = (x_1 y_1, \ \dots \ , x_1 y_J, \ \dots \ , x_I y_1, \ \dots \ , x_I y_J)^T.$$

Using the Khatri–Rao product, the decomposition (2.2) can be written in three different matrix forms of tensor $\mathcal{A}$,[38] as follows:

$$\mathbf{A}_{(1)} = \mathbf{X}\mathbf{D}(\mathbf{Z} \odot \mathbf{Y})^T, \mathbf{A}_{(2)} = \mathbf{Y}\mathbf{D}(\mathbf{Z} \odot \mathbf{X})^T, \mathbf{A}_{(3)} = \mathbf{Z}\mathbf{D}(\mathbf{Y} \odot \mathbf{X})^T, \tag{2.3}$$

where the matrix $\mathbf{D}$ is diagonal with elements of $\boldsymbol{\alpha}$.

3 | SPARSE OPTIMIZATION FOR LOW-RANK APPROXIMATION

The main goal of this study is to find a low-rank tensor of the original tensor efficiently and accurately. We first formulate a tensor rank optimization problem, as follows:

$$\min_{\mathcal{B}} \operatorname{rank}(\mathcal{B}) \quad \text{subject to} \quad \|\mathcal{A} - \mathcal{B}\|_F^2 < \varepsilon.$$

For any given error ε, the minimal rank of $\mathcal{B}$ such that $\|\mathcal{A} - \mathcal{B}\|_F^2 \leq \varepsilon$ is no larger than rank($\mathcal{A}$). The optimal solution $\hat{\mathcal{B}}$ is a low-rank approximation of $\mathcal{A}$ with error ε.

We represent the tensor $\mathcal{B}$ as $\sum_{r=1}^{R} \alpha_r \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r = [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R$, where R is an upper bound of the rank of $\mathcal{A}$, and the columns of **X,Y,Z** satisfy the normalization constraint $\mathbf{N(X,Y,Z)} = 1$. Rescaling the columns of the matrices **X,Y,Z** is a standard technique.[11,37] It is implemented in practice for CP tensor decomposition to prevent the norm of the approximated matrices from blowing up to infinity, whereas another factor matrix tends to zero while keeping the residual small.

The tensor rank minimization is equivalent to the following constraint optimization problem with l_0-norm:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 \quad \text{s.t.} \quad \|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 \leq \varepsilon, \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1. \tag{3.1}$$

The problem (3.1) is equivalent to that of finding the rank of tensors when $\varepsilon = 0$, whose decision version is NP hard.[23]

To make it more tractable, we turn to an optimization problem with the following l_1-norm:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 \leq \varepsilon, \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1. \tag{3.2}$$

Furthermore, we then solve the following:

$$\min_{\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}} \frac{1}{2} \|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \ \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1, \tag{3.3}$$

an l_1-regularization optimization problem in which it includes the factor matrices as primal variables. These optimization formulations are common in compressed sensing.[25–27,39–41] By introducing the indicator function, we switch the constrained optimization problem (3.3) into the following unconstrained form:

$$\min_{\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}} \frac{1}{2} \|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 + \delta_{S_1}(\mathbf{X}) + \delta_{S_2}(\mathbf{Y}) + \delta_{S_3}(\mathbf{Z}), \tag{3.4}$$

where $S_1 = \{\mathbf{X} | \|\mathbf{x}_r\| = 1, r = 1, \ \dots \ , R\}, S_2 = \{\mathbf{Y} | \|\mathbf{y}_r\| = 1, r = 1, \ \dots \ , R\}$, and $S_3 = \{\mathbf{Z} | \|\mathbf{z}_r\| = 1, r = 1, \ \dots \ , R\}$.

Remark 1. Here, there is no simple manner to compute the relationship between ϵ and λ without already knowing the optimal solutions of formulations (3.2) and (3.3). In the matrix versions of Basis Pursuit,

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\|_1, \ \text{s.t.} \quad \|\mathbf{b} - \mathbf{B}\boldsymbol{\theta}\| \leq \varepsilon$$

and

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\|\mathbf{b} - \mathbf{B}\boldsymbol{\theta}\|^2 + \lambda\|\boldsymbol{\theta}\|_1,$$

it is possible to create a mapping between the two parameters through a Pareto curve to estimate the relationship from the support of few solutions.[42]

Our algorithm is tailored for solving the problem (3.4). Let the objective function in (3.4) be as follows:

$$\Psi(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) : \mathbb{R}^{I\times R}\times\mathbb{R}^{J\times R}\times\mathbb{R}^{K\times R}\times\mathbb{R}^{R}\to\mathbb{R}^{+},$$

where

$$\Psi(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) = f(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) + g(\boldsymbol{\alpha}) + \delta_{S_1}(\mathbf{X}) + \delta_{S_2}(\mathbf{Y}) + \delta_{S_3}(\mathbf{Z})$$

with the approximation term $f(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) = \frac{1}{2}\|\mathcal{A}-[\boldsymbol{\alpha};\mathbf{X},\mathbf{Y},\mathbf{Z}]_R\|_F^2$, the regularized penalty term $g(\boldsymbol{\alpha}) = \lambda\|\boldsymbol{\alpha}\|_1$, and three indicator functions $\delta_{S_1}(\mathbf{X})$, $\delta_{S_2}(\mathbf{Y})$, $\delta_{S_3}(\mathbf{Z})$. The function $f(\bullet)$ is a real polynomial function on $(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha})$, and the function $g(\bullet)$ is a nondifferential continuous function on $\boldsymbol{\alpha}$. Because S_1, S_2, S_3 are closed sets, indicator functions $\delta_{S_1}(\mathbf{X})$, $\delta_{S_2}(\mathbf{Y})$, and $\delta_{S_3}(\mathbf{Z})$ are proper and lower semicontinuous. Moreover, because $\delta_{S_1}(\mathbf{X})$, $\delta_{S_2}(\mathbf{Y})$, and $\delta_{S_3}(\mathbf{Z})$ are three semi-algebraic functions; thus, the objective function is also a semi-algebraic function. So, it is a Kurdyka–Łojasiewicz (KL) function.[30] For a point $\boldsymbol{\omega} = (\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) \in \mathbb{R}^{I\times R}\times\mathbb{R}^{J\times R}\times\mathbb{R}^{K\times R}\times\mathbb{R}^{R}$, if its (limiting) subdifferential,[30] denoted by $\partial\Psi(\boldsymbol{\omega})$, contains $\mathbf{0}$, we call it a critical point of $\Psi(\bullet)$. The set of critical points of $\Psi(\bullet)$ is denoted by C_Ψ.

Due to the ill-posedness[24,43] of the best low-rank approximation of tensors, it is known that the problem of finding a best rank-R approximation for tensors of order 3 or higher has no solution in general. However, after introducing the l_1 penalty term $\lambda\|\boldsymbol{\alpha}\|_1$ to the low-rank approximation term $f(\bullet)$, it is always attainable for the minimization of the objective function in (3.4). Thus, we have the following theorem to show the existence of the global optimal solution of problem (3.4).

Theorem 1. *The global optimal solution of problem (3.4) exists.*

Proof. For any tensor $\mathcal{A}\in\mathbb{R}^{I\times J\times K}$, the objective function $\frac{1}{2}\|\mathcal{A}-[\boldsymbol{\alpha};\mathbf{X},\mathbf{Y},\mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 + \delta_{S_1}(\mathbf{X}) + \delta_{S_2}(\mathbf{Y}) + \delta_{S_3}(\mathbf{Z})$ is denoted as $\Psi(\mathbf{X,Y,Z,}\boldsymbol{\alpha})$. Notice that all the columns of $\mathbf{X,Y,Z}$ in problem (3.4) are constrained to have length one. We define the $d-$ dimensional unit sphere as $\Delta^d = \{\mathbf{v}\in\mathbb{R}^d | \|\mathbf{v}\|_2 = 1\}$, and a set $S = \{(\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}) \in (\Delta^I)^R\times(\Delta^J)^R\times(\Delta^K)^R\times\mathbb{R}^R\}$. Because this function $\Psi(\bullet)$ is continuous on S, we only need to show that there is a point $s\in S$ such that $\Psi(s) = \inf\{\Psi(x)|x\in S\}$, that is, the minimization of low-rank approximation with l_1 penalty is attainable.

For a scalar $\xi > \inf\{\Psi(x)|x\in S\}$, we will show that the level set $L = \{x\in S|\Psi(x)\le\xi\}$ is compact. Because $\Psi(\bullet)$ is continuous on S, the set L is closed, and we only need to prove that L is bounded. Actually, it is guaranteed by the l_1 penalty term $\lambda\|\boldsymbol{\alpha}\|_1$ of $\Psi(\bullet)$. Otherwise, unbounded points will take the penalty term to infinity contrary to the boundedness of $\Psi(\bullet)$ on L. From the compactness of the level set L, the infimum $\inf\{\Psi(x)|x\in L\}$ is attainable because $\Psi(\bullet)$ is continuous on L. Furthermore, because $\inf\{\Psi(x)|x\in S\} = \inf\{\Psi(x)|x\in L\}$, there exists a point $s\in S$ such that $\Psi(s) = \inf\{\Psi(x)|x\in S\}$. □

4 | LOW-RANK APPROXIMATION OF TENSOR

In this section, we first describe an algorithm of low-rank approximation of tensor (LRAT) for computing the solution of problem (3.4) and then show some theoretical guarantees on the convergence of LRAT: (1) The sequence $\{(\mathbf{X}^n,\mathbf{Y}^n,\mathbf{Z}^n,\boldsymbol{\alpha}^n)\}_{n\in\mathbb{N}}$ generated by LRAT converges to a critical point of $\Psi(\bullet)$. (2) The limit point of $\{(\mathbf{X}^n,\mathbf{Y}^n,\mathbf{Z}^n,\boldsymbol{\alpha}^n)\}_{n\in\mathbb{N}}$ is a KKT point of problem (3.3).

4.1 | The algorithm

As in (2.3), the matricizations of tensor $\mathcal{B} = [\boldsymbol{\alpha};\mathbf{X},\mathbf{Y},\mathbf{Z}]_R$ via Khatri–Rao products are as follows:

$$\mathbf{B}_{(1)} = \mathbf{X}\mathbf{D}(\mathbf{Z}\odot\mathbf{Y})^T, \mathbf{B}_{(2)} = \mathbf{Y}\mathbf{D}(\mathbf{Z}\odot\mathbf{X})^T, \mathbf{B}_{(3)} = \mathbf{Z}\mathbf{D}(\mathbf{Y}\odot\mathbf{X})^T,$$

where $\mathbf{D} = diag(\alpha_1,\ \dots\ ,\alpha_R)$. We introduce the following three matrices for updating in the Algorithm 1:

$$\mathbf{U} = \mathbf{D}(\mathbf{Z}\odot\mathbf{Y})^T, \mathbf{V} = \mathbf{D}(\mathbf{Z}\odot\mathbf{X})^T, \mathbf{W} = \mathbf{D}(\mathbf{Y}\odot\mathbf{X})^T. \tag{4.1}$$

Algorithm 1 Low-Rank Approximation Of Tensors (LRAT)

Input: A third order tensor $\mathcal{A}$, an upper bound R of rank($\mathcal{A}$), a penalty parameter λ and a scale $s > 1$;

Output: An approximated tensor $\hat{\mathcal{B}}$ with an estimated rank $\hat{R}$;

1: Give an initial tensor $\mathcal{B}^0 = [\boldsymbol{\alpha}^0; \mathbf{X}^0, \mathbf{Y}^0, \mathbf{Z}^0]_R$.

2: Update step:

b. Update matrices $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$:
Compute $\mathbf{U}^n$ by (4.1) and let $d_n = \max\left\{\|\mathbf{U}^n\mathbf{U}^{n^T}\|_F, 1\right\}$.
Compute $\mathbf{D}^n$ and $\mathbf{X}^{n+1}$ by

$$\mathbf{D}^n = \mathbf{X}^n - \frac{1}{sd_n}\nabla_{\mathbf{X}} f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n),$$
$$\mathbf{X}^{n+1} = \mathbf{D}^n \mathrm{diag}(\|\mathbf{d}_1^n\|, \cdots, \|\mathbf{d}_R^n\|)^{-1}$$

where $\mathbf{d}_i^n$ is the i-th column of $\mathbf{D}^n$ for $i = 1, \cdots, R$.
Compute $\mathbf{V}^n$ by (4.1) and let $e_n = \max\{\|\mathbf{V}^n\mathbf{V}^{n^T}\|_F, 1\}$.
Compute $\mathbf{E}^n$ and $\mathbf{Y}^{n+1}$ by

$$\mathbf{E}^n = \mathbf{Y}^n - \frac{1}{se_n}\nabla_{\mathbf{Y}} f(\mathbf{X}^{n+1}, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n),$$
$$\mathbf{Y}^{n+1} = \mathbf{E}^n \mathrm{diag}\left(\left\|\mathbf{e}_1^n\right\|, \cdots, \left\|\mathbf{e}_R^n\right\|\right)^{-1}$$

where $\mathbf{e}_i^n$ is the i-th column of $\mathbf{E}^n$ for $i = 1, \cdots, R$.
Compute $\mathbf{W}^n$ by (4.1) and let $f_n = \max\{\|\mathbf{W}^n\mathbf{W}^{n^T}\|_F, 1\}$.
Compute $\mathbf{F}^n$ and $\mathbf{Z}^{n+1}$ by

$$\mathbf{F}^n = \mathbf{Z}^n - \frac{1}{sf_n}\nabla_{\mathbf{Z}} f(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^n, \boldsymbol{\alpha}^n),$$
$$\mathbf{Z}^{n+1} = \mathbf{F}^n \mathrm{diag}\left(\left\|\mathbf{f}_1^n\right\|, \cdots, \left\|\mathbf{f}_R^n\right\|\right)^{-1}$$

where $\mathbf{f}_i^n$ is the i-th column of $\mathbf{F}^n$ for $i = 1, \cdots, R$.

c. Update the row vector $\boldsymbol{\alpha}$:
Compute $\mathbf{Q}^{n+1}$ by (4.3) and let $\eta_n = \max\{\|\mathbf{Q}^{n+1}\mathbf{Q}^{n+1^T}\|_F, 1\}$.
Compute $\boldsymbol{\beta}^{n+1}$ by (4.5) and use the soft thresholding:

$$\boldsymbol{\alpha}^{n+1} = \mathcal{S}_{\frac{\lambda}{s\eta_n}}(\boldsymbol{\beta}^{n+1}).$$

3: Denote the limitations by $\hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\alpha}}$, compute $\hat{\mathcal{B}} = [\hat{\boldsymbol{\alpha}}; \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}]_R$ and count the number $\hat{R}$ of nonzero entries in $\hat{\boldsymbol{\alpha}}$.

4: **return** The tensor $\hat{\mathcal{B}}$ with the estimated rank $\hat{R}$.

It follows that $\mathbf{B}_{(1)} = \mathbf{XU}$, $\mathbf{B}_{(2)} = \mathbf{YV}$, and $\mathbf{B}_{(3)} = \mathbf{ZW}$. Thus, the function $f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha})$ can be written in three equivalent forms: $\frac{1}{2}\|\mathbf{A}_{(1)} - \mathbf{XU}\|_F^2 = \frac{1}{2}\|\mathbf{A}_{(2)} - \mathbf{YV}\|_F^2 = \frac{1}{2}\|\mathbf{A}_{(3)} - \mathbf{ZW}\|_F^2$. Furthermore, we have the gradients of $f(\bullet)$ on $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$, as follows:

$$\begin{aligned}
\nabla_{\mathbf{X}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}) &= \left(\mathbf{XU} - \mathbf{A}_{(1)}\right)\mathbf{U}^T,\\
\nabla_{\mathbf{Y}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}) &= \left(\mathbf{YV} - \mathbf{A}_{(2)}\right)\mathbf{V}^T,\\
\nabla_{\mathbf{Z}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}) &= \left(\mathbf{ZW} - \mathbf{A}_{(3)}\right)\mathbf{W}^T.
\end{aligned} \tag{4.2}$$

Using the vectorization of tensors,[44] we can vectorize every rank-one tensor of outer product $\mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$ into a row vector $\mathbf{q}_r$ for $1 \le r \le R$. We denote a matrix consisting of all $\mathbf{q}_r$ for $1 \le r \le R$ by the following:

$$\mathbf{Q} = \left(\mathbf{q}_1^T, \ldots, \mathbf{q}_R^T\right)^T. \tag{4.3}$$

Thus, the function $f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha})$ can be also written as $\frac{1}{2}\|\mathbf{a} - \boldsymbol{\alpha}\boldsymbol{Q}\|_F^2$, where $\mathbf{a}$ is a vectorization for tensor $\mathcal{A}$. Furthermore, the gradient of $f(\bullet)$ on $\boldsymbol{\alpha}$ is as follows:

$$\nabla_{\boldsymbol{\alpha}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}) = (\boldsymbol{\alpha}\mathbf{Q} - \mathbf{a})\mathbf{Q}^T. \tag{4.4}$$

Our algorithm starts from $(\mathbf{X}^k, \mathbf{Y}^k, \mathbf{Z}^k, \boldsymbol{\alpha}^k)$ and iteratively updates variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ and then $\boldsymbol{\alpha}$ in each loop. Inspired by the Equation (1.4), the update of $\mathbf{X}$ is based on the following constraint optimization problem:

$$\begin{aligned}
&\arg\min_{\mathbf{X}} \left\{ \langle \mathbf{X} - \mathbf{X}^n, \nabla_{\mathbf{X}} f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)\rangle + \frac{sd_n}{2}\|\mathbf{X} - \mathbf{X}^n\|_F^2 \right\}\\
&\quad \text{s.t. } \|\mathbf{x}_i\| = 1, i = 1, \ldots, R,
\end{aligned}$$

where $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_R) \in \mathbb{R}^{I\times R}$, $d_n = \max\{\|\mathbf{U}^n\mathbf{U}^{nT}\|_F, 1\}$, and $\mathbf{U}^n$ is computed from $\boldsymbol{\alpha}^n, \mathbf{Y}^n, \mathbf{Z}^n$ by (4.1). This problem is equivalent to the following:

$$\arg\min_{\mathbf{X}} \left\{\|\mathbf{X} - \mathbf{D}^n\|_F^2\right\} \quad \text{s.t.} \quad \|\mathbf{x}_i\| = 1, i = 1, \ldots, R,$$

where $\mathbf{D}^n = \mathbf{X}^n - \frac{1}{sd_n}\nabla_{\mathbf{X}} f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$. So, we obtain the update of X, as follows:

$$\mathbf{x}_i^{n+1} = \mathbf{d}_i^n / \|\mathbf{d}_i^n\|, i = 1, \ldots, R,$$

where $\mathbf{x}_i^{n+1}$ and $\mathbf{d}_i^n$ are the ith columns of $\mathbf{X}^{n+1}$ and $\mathbf{D}^n$.

Similarly, the update of $\mathbf{Y}$ is based on the following optimization problem:

$$\arg\min_{\mathbf{Y}} \left\{ \left\langle \mathbf{Y} - \mathbf{Y}^n, \nabla_{\mathbf{Y}} f(\mathbf{X}^{n+1}, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)\right\rangle + \frac{se_n}{2}\|\mathbf{Y} - \mathbf{Y}^n\|_F^2 \right\}$$
$$\text{s.t.} \quad \|\mathbf{y}_i\| = 1, i = 1, \ldots, R,$$

where $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_R) \in \mathbb{R}^{J\times R}$, $e_n = \max\{\|\mathbf{V}^n\mathbf{V}^{nT}\|_F, 1\}$, and $\mathbf{V}^n$ is computed from $\boldsymbol{\alpha}^n, \mathbf{X}^{n+1}, \mathbf{Z}^n$ by (4.1). So, we obtain the update of Y, as follows:

$$\mathbf{y}_i^{n+1} = \mathbf{e}_i^n / \|\mathbf{e}_i^n\|, i = 1, \ldots, R,$$

where $\mathbf{y}_i^{n+1}$ and $\mathbf{e}_i^n$ are the ith columns of $\mathbf{Y}^{n+1}$ and $\mathbf{E}^n = \mathbf{Y}^n - \frac{1}{se_n}\nabla_{\mathbf{Y}} f(\mathbf{X}^{n+1}, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$.

The update of $\mathbf{Z}$ is based on the following constraint optimization problem:

$$\arg\min_{\mathbf{Z}} \left\{ \left\langle \mathbf{Z} - \mathbf{Z}^n, \nabla_{\mathbf{Z}} f(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^n, \boldsymbol{\alpha}^n)\right\rangle + \frac{sf_n}{2}\|\mathbf{Z} - \mathbf{Z}^n\|_F^2 \right\}$$
$$\text{s.t.} \quad \|\mathbf{z}_i\| = 1, i = 1, \ldots, R,$$

where $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_R) \in \mathbb{R}^{K\times R}$, $f_n = \max\{\|\mathbf{W}^n\mathbf{W}^{nT}\|_F, 1\}$, and $\mathbf{W}^n$ is computed from $\boldsymbol{\alpha}^n, \mathbf{X}^{n+1}, \mathbf{Y}^{n+1}$ by (4.1). The update of Z is as follows:

$$\mathbf{z}_i^{n+1} = \mathbf{f}_i^{n+1} / \left\|\mathbf{f}_i^{n+1}\right\|, i = 1, \ldots, R,$$

where $\mathbf{z}_i^{n+1}$ and $\mathbf{f}_i^{n+1}$ are the ith columns of $\mathbf{Z}^{n+1}$ and $\mathbf{F}^n = \mathbf{Z}^n - \frac{1}{sf_n}\nabla_{\mathbf{Z}} f(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$.

Finally, we consider to update $\boldsymbol{\alpha}$ by using the Equation (1.4), as follows:

$$\arg\min_{\boldsymbol{\alpha}} \left\{ \left\langle \boldsymbol{\alpha} - \boldsymbol{\alpha}^n, \nabla_{\boldsymbol{\alpha}} f(C^{n+1}, \mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^n)\right\rangle + \frac{s\eta_n}{2}\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^n\|^2 + \lambda\|\boldsymbol{\alpha}\|_1 \right\},$$

where $\eta_n = \max\{\|\mathbf{Q}^{n+1}\mathbf{Q}^{n+1^T}\|_F, 1\}$, and $\mathbf{Q}^{n+1}$ can be computed from $\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}$ by (4.3). This optimization problem is equivalent to the following:

$$\arg\min_{\boldsymbol{\alpha}} \frac{1}{2}\left\|\boldsymbol{\alpha} - \boldsymbol{\alpha}^n + \frac{1}{s\eta_n}\nabla_{\boldsymbol{\alpha}} f(C^{n+1}, \mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^n)\right\|^2 + \frac{\lambda}{s\eta_n}\|\boldsymbol{\alpha}\|_1.$$

So, we can obtain the updated form for $\boldsymbol{\alpha}$ in Algorithm 1 by using the separate soft thresholding as follows:

$$\boldsymbol{\alpha}^{n+1} = \mathcal{S}_{\frac{\lambda}{s\eta_n}}(\boldsymbol{\beta}^{n+1}),$$

where

$$\boldsymbol{\beta}^{n+1} = \boldsymbol{\alpha}^n - \frac{1}{s\eta_n}\nabla_{\boldsymbol{\alpha}} f(C^{n+1}, \mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^n). \tag{4.5}$$

It should be noted that if we set $\lambda = 0$, the LRAT algorithm turns into a modified alternative least-squares method (modALS). The modALS algorithm uses linearized iterative technique[7,30] to update variables in each step. Although the regularization parameter λ is fixed in Algorithm 1, we can adaptively choose it for practical computation, which will be shown in Section 5.

Remark 2. In our algorithm, the computational complexity mainly comes from matrix multiplications. The Update Step (2b) for updating $\boldsymbol{\alpha}$ in LRAT Algorithm requires more cpu time than the Update Step (2a) because of the large matrix dimension of $\mathbf{Q}$. The complexity of our algorithm is $O(NIJKR^2)$, where N is the total number of iteration.

4.2 | Convergence of algorithm

In this section, we illustrate the convergence mechanism of the LRAT algorithm, which is a rescaling version of the proximal alternating linear minimization algorithm.[30] The following Lemma 1 points out that for the function $f(\boldsymbol{\omega}) = \frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2$, the gradient $\nabla_{\boldsymbol{\omega}} f(\boldsymbol{\omega})$ of $f(\boldsymbol{\omega})$ is Lipschitz continuous on bounded subsets and that all the partial gradients of $f(\boldsymbol{\omega})$ are globally Lipschitz with modulus.

Lemma 1. *Let $f(\boldsymbol{\omega})$ be the approximation term $\frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2$, where $\boldsymbol{\omega} = (\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\alpha})$. We have that the gradient function ∇f is Lipschitz continuous on bounded subsets of $\mathbb{R}^{I\times R} \times \mathbb{R}^{J\times R} \times \mathbb{R}^{K\times R} \times \mathbb{R}^R$, that is, for any bounded subset $B \in \mathbb{R}^{I\times R} \times \mathbb{R}^{J\times R} \times \mathbb{R}^{K\times R} \times \mathbb{R}^R$, there exists $M > 0$ such that for any $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2 \in B$,*

$$\|\nabla_{\boldsymbol{\omega}} f(\boldsymbol{\omega}_1) - \nabla_{\boldsymbol{\omega}} f(\boldsymbol{\omega}_2)\|_F \leq M\|\boldsymbol{\omega}_1 - \boldsymbol{\omega}_2\|_F.$$

Moreover, for any fixed $\mathbf{X} \in \mathbb{R}^{I\times R}, \mathbf{Y} \in \mathbb{R}^{J\times R}, \mathbf{Z} \in \mathbb{R}^{K\times R}, \boldsymbol{\alpha} \in \mathbb{R}^R$, there exist four constants $c, d, e, \eta > 0$ such that:

$$\begin{aligned}
\|\nabla_{\mathbf{X}} f(\mathbf{X}_1, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}) - \nabla_{\mathbf{X}} f(\mathbf{X}_2, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha})\|_F &\leq d\|\mathbf{X}_1 - \mathbf{X}_2\|_F, \text{ for any } \mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^{I\times R}\\
\|\nabla_{\mathbf{Y}} f(\mathbf{X}, \mathbf{Y}_1, \mathbf{Z}, \boldsymbol{\alpha}) - \nabla_{\mathbf{Y}} f(\mathbf{X}, \mathbf{Y}_2, \mathbf{Z}, \boldsymbol{\alpha})\|_F &\leq e\|\mathbf{Y}_1 - \mathbf{Y}_2\|_F, \text{ for any } \mathbf{Y}_1, \mathbf{Y}_2 \in \mathbb{R}^{J\times R}\\
\|\nabla_{\mathbf{Z}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}_1, \boldsymbol{\alpha}) - \nabla_{\mathbf{Z}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}_2, \boldsymbol{\alpha})\|_F &\leq f\|\mathbf{Z}_1 - \mathbf{Z}_2\|_F, \text{ for any } \mathbf{Z}_1, \mathbf{Z}_2 \in \mathbb{R}^{K\times R}\\
\|\nabla_{\boldsymbol{\alpha}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}_1) - \nabla_{\boldsymbol{\alpha}} f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha}_2)\|_F &\leq \eta\|\boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2\|_F, \text{ for any } \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \in \mathbb{R}^R,
\end{aligned}$$

where $d = \|\boldsymbol{U}\boldsymbol{U}^{\mathbf{T}}\|_F, e = \|\boldsymbol{V}\boldsymbol{V}^{\mathbf{T}}\|_F, f = \|\boldsymbol{W}\boldsymbol{W}^{\mathbf{T}}\|_F, \eta = \|\boldsymbol{Q}\boldsymbol{Q}^{\mathbf{T}}\|_F$.

The proof has not been included because it relies on standard techniques. In our LRAT algorithm, those Lipschitz constants rely on the iterative number n and have a lower bound 1. Specifically, $d_n = \max\{\|\mathbf{U}^{n+1}\mathbf{U}^{n+1^T}\|_F, 1\}$, $e_n = \max\{\|\mathbf{V}^{n+1}\mathbf{V}^{n+1^T}\|_F, 1\}$, $f_n = \max\{\|\mathbf{W}^{n+1}\mathbf{W}^{n+1^T}\|_F, 1\}$, $\eta_n = \max\{\|\mathbf{Q}^{n+1}\mathbf{Q}^{n+1^T}\|_F, 1\}$.

Lemma 2. (See sufficient decrease property in the work of Bolte et al.[30])
Let $f : \mathbb{R}^m \to \mathbb{R}$ be continuously differentiable with gradient ∇f assumed to be L_f-Lipschitz continuous, and let $\sigma : \mathbb{R}^m \to (-\infty, +\infty]$ be a proper and lower semicontinuous function with $\inf_{\mathbb{R}^m} \sigma > -\infty$. For any $t > L_f$ and $u \in dom\, \sigma$, define

$$u^+ = \arg\min_x \left\{ \langle x - u, \nabla f(u)\rangle + \frac{t}{2}\|x - u\|^2 + \sigma(u) \right\}.$$

Then, we have that

$$f(u) + \sigma(u) - \left(f(u^+) + \sigma(u^+)\right) \geq \frac{1}{2}(t - L_f)\|u^+ - u\|^2. \tag{4.6}$$

Lemma 3. *Let $\Psi(\bullet)$ be the objective function in problem (3.4). If $(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)_{n\in\mathbb{N}}$ and $(d_n, e_n, f_n, \eta_n)_{n\in\mathbb{N}}$ are generated by our LRAT algorithm, we have that for any $s > 1$,*

$$\begin{aligned}
\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n) - \Psi(\mathbf{X}^{n+1}, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n) &\geq \frac{1}{2}(s-1)d_n\|\mathbf{X}^n - \mathbf{X}^{n+1}\|_F^2,\\
\Psi(\mathbf{X}^{n+1}, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n) - \Psi(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^n, \boldsymbol{\alpha}^n) &\geq \frac{1}{2}(s-1)e_n\|\mathbf{Y}^n - \mathbf{Y}^{n+1}\|_F^2,\\
\Psi(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^n, \boldsymbol{\alpha}^n) - \Psi(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^n) &\geq \frac{1}{2}(s-1)f_n\|\mathbf{Z}^n - \mathbf{Z}^{n+1}\|_F^2,\\
\Psi(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^n) - \Psi(\mathbf{X}^{n+1}, \mathbf{Y}^{n+1}, \mathbf{Z}^{n+1}, \boldsymbol{\alpha}^{n+1}) &\geq \frac{1}{2}(s-1)\eta_n\|\boldsymbol{\alpha}^n - \boldsymbol{\alpha}^{n+1}\|_F^2.
\end{aligned}$$

Proof. These four inequalities can be obtained by using Lemma 2. □

The following lemma shows that the value of $\Psi(\bullet)$ monotonically decreases on the sequence $(\boldsymbol{\omega}^n)_{n\in\mathbb{N}}$, which is generated by our algorithm.

Lemma 4. *Let $\Psi(\boldsymbol{\omega})$ be the objective function as follows:*

$$\frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 + \delta_{S_1}(\mathbf{X}) + \delta_{S_2}(\mathbf{Y}) + \delta_{S_3}(\mathbf{Z}),$$

where $\boldsymbol{\omega} = (\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\alpha})$, then

i. the sequence $\{\Psi(\boldsymbol{\omega}^n)\}_{n\in\mathbb{N}}$ is nonincreasing, and for any $n \in \mathbb{N}$, there is a scalar $\beta > 0$ such that $\Psi(\boldsymbol{\omega}^n) - \Psi(\boldsymbol{\omega}^{n+1}) \geq \beta\|\boldsymbol{\omega}^n - \boldsymbol{\omega}^{n+1}\|_F^2$.

ii. $\lim_{n\to\infty}\|\mathbf{X}^n-\mathbf{X}^{n+1}\|_F\to 0, \lim_{n\to\infty}\|\mathbf{Y}^n-\mathbf{Y}^{n+1}\|_F\to 0, \lim_{n\to\infty}\|\mathbf{Z}^n-\mathbf{Z}^{n+1}\|_F\to 0$ *and* $\lim_{n\to\infty}\|\boldsymbol{\alpha}^n-\boldsymbol{\alpha}^{n+1}\|_F\to 0$.
iii. the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ *is bounded.*

Proof. In our algorithm, all the Lipschitz constants $d_n, e_n, f_n, \eta_n \geq 1$. So, by Lemma 3, $\Psi(\boldsymbol{\omega}^n)-\Psi(\boldsymbol{\omega}^{n+1})\geq \beta\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n+1}\|_F^2$, where $\beta=\min\{(s-1)/2, 1/2\}$. We can obtain the first conclusion (i).

The second conclusion (ii) holds from the first one because the sum $\sum_{n=0}^{\infty}(\Psi(\boldsymbol{\omega}^n)-\Psi(\boldsymbol{\omega}^{n+1}))$ is finite.

If the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ is unbounded, it means that $\{\boldsymbol{\alpha}^n\}_{n\in\mathbb{N}}$ is unbounded because columns of $\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n$ are constrained to have length one. So the sequence $\{\Psi(\boldsymbol{\omega}^n)\}_{n\in\mathbb{N}}$ is unbounded since $\Psi(C, \mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha})\geq\lambda\|\boldsymbol{\alpha}\|_1$. From the conclusion (i), $\Psi(\boldsymbol{\omega}^n)$ is nonincreasing. Because $\Psi(\bullet)$ has a lower bound, the sequence $\{\Psi(\boldsymbol{\omega}_n)\}_{n\in\mathbb{N}}$ is not unbounded. It is a contradiction. So, the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ must be bounded. □

Furthermore, from Lemma 1 and the boundedness shown in Lemma 4, we can obtain the following Lipschitz upper bounds for subdifferentials.

Lemma 5. *Let* $\boldsymbol{\omega}^n=(\boldsymbol{X}^n, \boldsymbol{Y}^n, \boldsymbol{Z}^n, \boldsymbol{\alpha}^n)$ *be the sequence generated by our LRAT algorithm. There exist four positive scales* L_1, L_2, L_3, *and* L_4 *such that the following inequalities hold for any* $n\in\mathbb{N}$.
There is some $\boldsymbol{\eta}_1^n\in\partial_{\mathbf{X}}\Psi(\boldsymbol{\omega}^n)$ *such that* $\|\boldsymbol{\eta}_1^n\|_F\leq L_1\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.
There is some $\boldsymbol{\eta}_2^n\in\partial_{\mathbf{Y}}\Psi(\boldsymbol{\omega}^n)$ *such that* $\|\boldsymbol{\eta}_2^n\|_F\leq L_2\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.
There is some $\boldsymbol{\eta}_3^n\in\partial_{\mathbf{Z}}\Psi(\boldsymbol{\omega}^n)$ *such that* $\|\boldsymbol{\eta}_3^n\|_F\leq L_3\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.
There is some $\boldsymbol{\eta}_4^n\in\partial_{\boldsymbol{\alpha}}\Psi(\boldsymbol{\omega}^n)$ *such that* $\|\boldsymbol{\eta}_4^n\|_F\leq L_4\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.

Proof. By the update of $\mathbf{X}$,

$$\mathbf{X}^n=\arg\min_{\mathbf{X}}\left\{\left\langle\mathbf{X}-\mathbf{X}^{n-1}, \nabla_{\mathbf{X}}f(C^n, \mathbf{X}^{n-1}, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1})\right\rangle+\frac{sd_n}{2}\|\mathbf{X}-\mathbf{X}^{n-1}\|_F^2+\delta_{S_1}(\mathbf{X})\right\}.$$

So, we have that $\nabla_{\mathbf{X}}f(\mathbf{X}^{n-1}, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1})+sd_n(\mathbf{X}^n-\mathbf{X}^{n-1})+\mathbf{u}_1^n=\mathbf{0}$, where $\mathbf{u}_1^n\in\partial_{\mathbf{X}}\delta_{S_1}(\mathbf{X}^n)$. Hence,

$$\mathbf{u}_1^n=sd_n(\mathbf{X}^{n-1}-\mathbf{X}^n)-\nabla_{\mathbf{X}}f(\mathbf{X}^{n-1}, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1}).$$

Because $\nabla_{\mathbf{X}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+\mathbf{u}_1^n\in\partial_{\mathbf{X}}\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$, we have that

$$\begin{aligned}\boldsymbol{\eta}_1^n&=\nabla_{\mathbf{X}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+sd_n(\mathbf{X}^{n-1}-\mathbf{X}^n)-\nabla_{\mathbf{X}}f(\mathbf{X}^{n-1}, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1})\\&=\nabla_{\mathbf{X}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+\mathbf{u}_1^n\\&\in\partial_{\mathbf{X}}\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n).\end{aligned}\tag{4.7}$$

By Lemma 1 and the boundedness of $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$, we have that there exists a constant L_1 such that $\|\boldsymbol{\eta}_1^n\|_F\leq L_1\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.

Similarly, we can choose the following:

$$\boldsymbol{\eta}_2^n=\nabla_{\mathbf{Y}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+se_n(\mathbf{Y}^{n-1}-\mathbf{Y}^n)-\nabla_{\mathbf{Y}}f(\mathbf{X}^n, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1})\tag{4.8}$$

and

$$\boldsymbol{\eta}_3^n=\nabla_{\mathbf{Z}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+sf_n(\mathbf{Z}^{n-1}-\mathbf{Z}^n)-\nabla_{\mathbf{Z}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1}).\tag{4.9}$$

So, $\boldsymbol{\eta}_2^n\in\partial_{\mathbf{Y}}\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$ and $\boldsymbol{\eta}_3^n\in\partial_{\mathbf{Z}}\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$. Furthermore, there exist constants L_2 and L_3 such that $\|\boldsymbol{\eta}_2^n\|_F\leq L_2\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$ and $\|\boldsymbol{\eta}_3^n\|_F\leq L_3\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F$.

By the update of $\boldsymbol{\alpha}$,

$$\nabla_{\boldsymbol{\alpha}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^{n-1})+s\eta_n(\boldsymbol{\alpha}^n-\boldsymbol{\alpha}^{n-1})+\mathbf{u}^n=0,\tag{4.10}$$

where $\mathbf{u}^n\in\partial_{\boldsymbol{\alpha}}g(\boldsymbol{\alpha}^n)$ and $g(\mathbf{x})=\lambda\|\mathbf{x}\|_1$. Denote $\boldsymbol{\eta}_4^n$ as $\nabla_{\boldsymbol{\alpha}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+\mathbf{u}^n$. Thus, we have that $\boldsymbol{\eta}_4^n\in\partial_{\boldsymbol{\alpha}}\Psi(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)$ and

$$\begin{aligned}\|\boldsymbol{\eta}_4^n\|_F&=\|\nabla_{\boldsymbol{\alpha}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)+\mathbf{u}^n\|_F\\&\leq\|\nabla_{\boldsymbol{\alpha}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^n)-\nabla_{\boldsymbol{\alpha}}f(\mathbf{X}^n, \mathbf{Y}^n, \mathbf{Z}^n, \boldsymbol{\alpha}^{n-1})\|_F+s\eta_n\|\boldsymbol{\alpha}^n-\boldsymbol{\alpha}^{n-1}\|_F\\&\leq L_4\|\boldsymbol{\omega}^n-\boldsymbol{\omega}^{n-1}\|_F.\end{aligned}$$

We also get the last inequality by using Lemma 1 and the boundedness of $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$. □

The following theorem shows that the sequence of the LRAT algorithm is convergent to a critical point of $\Psi(\bullet)$.

Theorem 2. *Let $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ be a sequence generated by the LRAT algorithm from a starting point $\boldsymbol{\omega}^0$. Then, the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ converges to a critical point $\boldsymbol{\omega}^*$ of $\Psi(\boldsymbol{\omega})$.*

Proof. By Lemma 3, the sufficient decrease property is satisfied that there is a constant $\beta > 0$ such that for any $n \in \mathbb{N}$,

$$\beta\|\boldsymbol{\omega}^n - \boldsymbol{\omega}^{n+1}\|_F^2 \leq \Psi(\boldsymbol{\omega}^n) - \Psi(\boldsymbol{\omega}^{n+1}).$$

By Lemma 5, the iterates' gap has a lower bound by the length of a vector in the subdifferential of Ψ. There is a constant $L > 0$ and $\{\boldsymbol{\eta}^n\}_{n\in\mathbb{N}}$ such that for any $n \in \mathbb{N}$,

$$\|\boldsymbol{\eta}^n\|_F \leq L\|\boldsymbol{\omega}^n - \boldsymbol{\omega}^{n-1}\|_F,$$

where $\boldsymbol{\eta}^n \in \partial\Psi(\boldsymbol{\omega}^n)$.

Furthermore, because $\Psi(\bullet)$ is a KL function, we complete the proof by using theorem 3.1 in the work of Bolte et al.[30] □

A point $\boldsymbol{\omega} = (\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \boldsymbol{\alpha})$ is called as a KKT point of problem (3.3) if there are three diagonal matrices $\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3 \in \mathbb{R}^{R\times R}$ and a vector $\mathbf{u} \in \partial_{\boldsymbol{\alpha}} g(\boldsymbol{\alpha})$ such that

$$\begin{aligned}\nabla_{\mathbf{X}} f(\boldsymbol{\omega}) + \mathbf{X}\mathbf{H}_1 = \mathbf{0}, \nabla_{\mathbf{Y}} f(\boldsymbol{\omega}) + \mathbf{Y}\mathbf{H}_2 = \mathbf{0}, \nabla_{\mathbf{Z}} f(\boldsymbol{\omega}) + \mathbf{Z}\mathbf{H}_3 = \mathbf{0}\\ \nabla_{\boldsymbol{\alpha}} f(\boldsymbol{\omega}) + \mathbf{u} = \mathbf{0}, \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1.\end{aligned} \tag{4.11}$$

In the following, we show that the limit point $\boldsymbol{\omega}^* = (C^*, \mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*, \boldsymbol{\alpha}^*)$ of the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ is a KKT point of problem (3.3).

Corollary 1. *Let $\boldsymbol{\omega}^* = (\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*, \boldsymbol{\alpha}^*)$ be the limit point of the sequence $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$ generated by the LRAT algorithm. If $\boldsymbol{X}^*, \boldsymbol{Y}^*$, and $\boldsymbol{Z}^*$ have a full column rank, the limit point $\boldsymbol{\omega}^*$ is a KKT point of problem (3.3).*

Proof. $\mathbf{N}(\mathbf{X}^*,\mathbf{Y}^*,\mathbf{Z}^*) = 1$ is obvious because $\mathbf{N}(\mathbf{X}^\mathbf{n},\mathbf{Y}^\mathbf{n},\mathbf{Z}^\mathbf{n}) = 1$ and the convergence of $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$. From (4.10), there exists a vector $\mathbf{u} \in \partial_{\boldsymbol{\alpha}} g(\boldsymbol{\alpha}^*)$ such that

$$\nabla_{\boldsymbol{\alpha}} f(\boldsymbol{\omega}^*) + \mathbf{u} = \mathbf{0}.$$

By the update of $\mathbf{X}$, there is a diagonal matrix $\mathbf{H}_1^n$ such that

$$\nabla_{\mathbf{X}} f(\mathbf{X}^{n-1}, \mathbf{Y}^{n-1}, \mathbf{Z}^{n-1}, \boldsymbol{\alpha}^{n-1}) + sd_n(\mathbf{X}^n - \mathbf{X}^{n-1}) + \mathbf{X}^n\mathbf{H}_1^n = \mathbf{0}.$$

By the convergency of $\{\boldsymbol{\omega}^n\}_{n\in\mathbb{N}}$, we have that $\mathbf{H}_1^n$ is convergent to some diagonal matrix $\mathbf{H}_1^*$ because $\mathbf{X}^*$ has a full column rank. Furthermore, we can obtain the following:

$$\nabla_{\mathbf{X}} f(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*, \boldsymbol{\alpha}^*) + \mathbf{X}^*\mathbf{H}_1^* = \mathbf{0}.$$

Similarly, we have

$$\nabla_{\mathbf{Y}} f(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*, \boldsymbol{\alpha}^*) + \mathbf{Y}^*\mathbf{H}_2^* = \mathbf{0}, \nabla_{\mathbf{Z}} f(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*, \boldsymbol{\alpha}^*) + \mathbf{Z}^*\mathbf{H}_3^* = \mathbf{0}.$$

This completes the proof of this corollary. □

5 | PROBABILISTIC CONSISTENCY OF THE SPARSITY

In this section, we will discuss the probabilistic consistency of the sparsity of the optimal solution to problem (3.3). We will see that under a suitable choice on the regularization parameter, the optimal solution can recover the true sparsity in a statistical model with a high probability.

For a given regularization parameter $\lambda > 0$, an optimal solution to problem (3.3) is denoted by the following:

$$(\hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\alpha}}) = \underset{\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}}{\arg\min} \frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1.$$

As shown in Section 4.1, we can construct a $R \times (I * J * K)$ matrix $\hat{\mathbf{Q}} = (\hat{\mathbf{q}}_1^T, \ldots, \hat{\mathbf{q}}_R^T)^T = ((\hat{\mathbf{X}} \odot \hat{\mathbf{Y}}) \odot \hat{\mathbf{Z}})^T$ from (4.3) and vectorize tensor $\mathcal{A}$ into a row vector $\mathbf{a}$.

For convenience, we introduce new variables: $\mathbf{b}, \theta, \mathbf{B}$ for $\mathbf{a}^T, \boldsymbol{\alpha}^T, \hat{\mathbf{Q}}^T$, respectively. Thus, $\mathbf{b}$ and $\boldsymbol{\theta}$ are column vectors with dimension $I * J * K$ and R, and $\mathbf{B}$ is a $(I * J * K) \times R$ matrix. Furthermore, we have the following equality:

$$\frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}]_R\|_F^2 + \|\boldsymbol{\alpha}\|_1 = \frac{1}{2}\|\mathbf{b} - \mathbf{B}\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1. \tag{5.1}$$

The optimal solution $\hat{\boldsymbol{\alpha}}^T$ for tensor approximation problem (3.3) is also an optimal solution $\hat{\boldsymbol{\theta}}$ of a standard l_1-regularized least-squares problem as follows:

$$\min_{\theta} \frac{1}{2}\|\mathbf{b} - \mathbf{B}\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1. \tag{5.2}$$

Assume that $\mathbf{b}$ and $\mathbf{B}$ have a sparse representation structure as follows:

$$\mathbf{b} = \mathbf{B}\boldsymbol{\theta}^* + \boldsymbol{\varepsilon}, \tag{5.3}$$

where all the columns of $\mathbf{B}$ are normalized to one. The variable $\boldsymbol{\theta}^*$ is a sparse signal with k nonzero entries ($k < R$), and $\boldsymbol{\varepsilon}$ is a vector with independent sub-Gaussian entries of mean zero and parameter σ^2.

Denote a subgradient vector in $\partial\|\boldsymbol{\theta}\|_1$ as $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_R)^T$. The entries of $\boldsymbol{\beta}$ satisfy that for any $1 \le i \le R$, $\beta_i = \mathrm{sgn}(\theta_i)$ if $\theta_i \neq 0$ and $\beta_i \in [-1, 1]$ if $\theta_i = 0$. As shown in the Lemma 1 of,[35] $\hat{\boldsymbol{\theta}}$ is an optimal solution to problem (5.2) if and only if there exists a subgradient vector $\hat{\boldsymbol{\beta}} \in \partial\|\hat{\boldsymbol{\theta}}\|_1$ such that

$$-\mathbf{B}^T(\mathbf{b} - \mathbf{B}\hat{\boldsymbol{\theta}}) + \lambda\hat{\boldsymbol{\beta}} = \mathbf{0} \tag{5.4}$$

if and only if there exists a subgradient vector $\hat{\boldsymbol{\beta}} \in \partial\|\hat{\boldsymbol{\theta}}\|_1$ such that

$$\mathbf{B}^T\mathbf{B}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) - \mathbf{B}^T\boldsymbol{\varepsilon} + \lambda\hat{\boldsymbol{\beta}} = \mathbf{0}. \tag{5.5}$$

Assume that $\mathbf{B}$ is a full column rank matrix. Then, the objective function in problem (5.2) is strictly convex, and the optimal solution $\hat{\boldsymbol{\theta}}$ to problem (5.2) is unique and exactly $\hat{\boldsymbol{\alpha}}^T$. Denote S and $\hat{S}$ as the index sets of nonzero entries in $\boldsymbol{\theta}^*$ and $\hat{\boldsymbol{\theta}}$, respectively. So, the sparse signal $\boldsymbol{\theta}^*$ can be rewritten as $(\boldsymbol{\theta}_S^{*T}, \mathbf{0}^T)^T$, and the cardinality of S is k. We will show in Theorem 3 that the optimal solution $\hat{\boldsymbol{\theta}}$, which is also the $\hat{\boldsymbol{\alpha}}^T$, of problem (5.2) may become a suitable approximation for the real sparse signal $\boldsymbol{\theta}^*$. Similar results shown in other works[35,36] consider the case $\mathbf{B}^T\mathbf{B}/n \to \mathbf{C}$ as $n \to \infty$ or $n^{-1/2}\max_{j\in S^c}\|\mathbf{B}_j\| \le 1$, where n is the number of rows in $\mathbf{B}$, whereas in this paper, all the $\mathbf{B}_j$ are normalized to one. We can further obtain a specific probability bound shown in Theorem 3, which relies only on two intrinsic parameters of model.

According to the unknown set S, we can separate columns of the design matrix $\mathbf{B}$ as two parts $(\mathbf{B}_S, \mathbf{B}_{S^c})$, where S^C is the complement of S. Moreover, because $\mathbf{B}_S$ also have a full column rank, there exists a unique solution $\hat{\boldsymbol{\theta}}_S$ by solving the following restricted Lasso problem:

$$\min_{\theta_S} \frac{1}{2}\|\mathbf{b} - \mathbf{B}_S\boldsymbol{\theta}_S\|_2^2 + \lambda\|\boldsymbol{\theta}_S\|_1. \tag{5.6}$$

Furthermore, if $(\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$ satisfies the equation (5.5), $(\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$ is thus the unique optimal solution $\hat{\boldsymbol{\theta}}$ to problem (5.2) because $\mathbf{B}$ has a full column rank. Moreover, we also obtain that the index set $\hat{S} \subseteq S$. From (5.5), if $\hat{\boldsymbol{\theta}}_S$ satisfies two equations, then

$$\mathbf{B}_S^T\mathbf{B}_S\left(\hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_S^*\right) - \mathbf{B}_S^T\boldsymbol{\varepsilon} + \lambda\hat{\boldsymbol{\beta}}_S = \mathbf{0} \tag{5.7}$$

and

$$\mathbf{B}_{S^c}^T\mathbf{B}_S\left(\hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_S^*\right) - \mathbf{B}_{S^c}^T\boldsymbol{\varepsilon} + \lambda\hat{\boldsymbol{\beta}}_{S^c} = \mathbf{0}, \tag{5.8}$$

where $\hat{\boldsymbol{\beta}}_S \in \partial\|\hat{\boldsymbol{\theta}}_S\|_1$ and $\|\hat{\boldsymbol{\beta}}_{S^c}\|_\infty = \max_{j\in S^c}|\hat{\boldsymbol{\beta}}_j| < 1$; we have that $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$ satisfies Equation (5.5) and $(\hat{\boldsymbol{\beta}}_S^T, \hat{\boldsymbol{\beta}}_{S^c}^T)^T \in \partial\|\hat{\boldsymbol{\theta}}\|_1$. Actually, because $\hat{\boldsymbol{\theta}}_S$ minimizes the problem (5.6), there exists $\hat{\boldsymbol{\beta}}_S \in \partial\|\hat{\boldsymbol{\theta}}_S\|_1$ such that Equation (5.7) holds. So, if it happens with a high probability that Equation (5.8) holds and $\|\hat{\boldsymbol{\beta}}_{S^c}\|_\infty < 1$, then the event $\boldsymbol{\Gamma} = \{(\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$ is the unique optimal solution $\hat{\boldsymbol{\theta}}$ to problem (5.2)$\}$ happens with a high probability. Furthermore, the event $\{\hat{S} \subseteq S\}$ also happens with a high probability. We are going to show these in the following part of this section.

From Equations (5.7) and (5.8), we have that

$$\hat{\boldsymbol{\beta}}_{S^c} = \mathbf{B}_{S^c}^T\mathbf{B}_S\left(\mathbf{B}_S^T\mathbf{B}_S\right)^{-1}\hat{\boldsymbol{\beta}}_S + \mathbf{B}_{S^c}^T\left(\mathbf{I} - \mathbf{B}_S\left(\mathbf{B}_S^T\mathbf{B}_S\right)^{-1}\mathbf{B}_S^T\right)\frac{\boldsymbol{\varepsilon}}{\lambda}, \tag{5.9}$$

$$\boldsymbol{\delta}_S = \hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_S^* = \left(\mathbf{B}_S^T\mathbf{B}_S\right)^{-1}\left(\mathbf{B}_S^T\boldsymbol{\varepsilon} - \lambda\hat{\boldsymbol{\beta}}_S\right). \tag{5.10}$$

For any $j \in S^C$, we have that

$$\hat{\beta}_j = \mathbf{B}_j^T \mathbf{B}_S \left(\mathbf{B}_S^T \mathbf{B}_S\right)^{-1} \hat{\boldsymbol{\beta}}_S + \mathbf{B}_j^T \left(\mathbf{I} - \mathbf{B}_S \left(\mathbf{B}_S^T \mathbf{B}_S\right)^{-1} \mathbf{B}_S^T\right) \frac{\boldsymbol{\epsilon}}{\lambda} = \mu_j + \omega_j.$$

We assume that there exists an incoherence parameter $\gamma \in (0, 1]$ such that $\|\mathbf{B}_{S^C}^T \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty \leq 1 - \gamma$, where matrix norm $\|M\|_\infty = \max_i \sum_j |M_{ij}|$. It is easy to obtain $|\mu_j| = |\mathbf{B}_j^T \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \hat{\boldsymbol{\beta}}_S| \leq \|\mathbf{B}_{S^C}^T \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty \leq 1 - \gamma$. Let us consider $\omega_j = \mathbf{B}_j^T (\mathbf{I} - \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T) \frac{\boldsymbol{\epsilon}}{\lambda} = \frac{1}{\lambda}(c_1 \varepsilon_1 + \cdots + c_n \varepsilon_n)$, where $(c_1, \ldots, c_n) = \mathbf{B}_j^T (\mathbf{I} - \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T)$. Thus, ω_j is a sub-Gaussian distribution with zero mean and parameter $\frac{\sigma^2}{\lambda}(c_1^2 + \cdots + c_n^2) = \frac{\sigma^2}{\lambda^2} \mathbf{B}_j^T (\mathbf{I} - \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T) \mathbf{B}_j$. Because $\mathbf{B}_j^T \mathbf{B}_j = 1$, this parameter is no more than $\frac{\sigma^2}{\lambda^2}$. So, $Pr(\max_{j \in S^C} |\omega_j| \geq t) \leq 2(R - k) \exp(-\frac{\lambda^2 t^2}{2\sigma^2})$, where k is the cardinality of S. By choosing $t = \frac{1}{2}\gamma$, we have that $Pr(\max_{j \in S^C} |\omega_j| \geq \frac{1}{2}\gamma) \leq 2(R - k) \exp(-\frac{\lambda^2 \gamma^2}{8\sigma^2})$. Thus, we have that

$$\Pr\left(\max_{j \in S^C} |\hat{\beta}_j| > 1 - \frac{\gamma}{2}\right) \leq \Pr\left(\max_{j \in S^C} |\omega_j| \geq \frac{1}{2}\gamma\right) \leq 2(R - k) \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right). \tag{5.11}$$

Now, let us consider the upper bound of $\boldsymbol{\delta}_S$: $\|\boldsymbol{\delta}_S\|_\infty \leq \|(\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T \boldsymbol{\epsilon}\|_\infty + \lambda \|(\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty$. Because $\lambda \|(\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty$ has a fixed value, we only need to consider the first term. For any $i \in S$, we have that $v_i = \mathbf{e}_i^T (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T \boldsymbol{\epsilon} = c_1 \varepsilon_1 + \cdots + c_n \varepsilon_n$, where $(c_1, \ldots, c_n) = \mathbf{e}_i^T (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{B}_S^T$. If we assume that $\lambda_{\min}(\mathbf{B}_S^T \mathbf{B}_S) \geq \mu$, then v_i is a sub-Gaussian distribution with zero mean and parameter $\frac{\sigma^2}{\lambda}(c_1^2 + \cdots + c_n^2) = \sigma^2 \mathbf{e}_i^T (\mathbf{B}_S^T \mathbf{B}_S)^{-1} \mathbf{e}_i \leq \frac{\sigma^2}{\mu}$. Thus, $Pr(\max_{i \in S} |v_i| > t) \leq 2k \exp(-\frac{t^2 \mu}{2\sigma^2})$. By choosing $t = \frac{\lambda}{2\sqrt{\mu}}$, we have that

$$\Pr\left(\max_{i \in S} |v_i| > \frac{\lambda}{2\sqrt{\mu}}\right) \leq 2k \exp\left(-\frac{\lambda^2}{8\sigma^2}\right) \leq 2k \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right). \tag{5.12}$$

By combining (5.11) and (5.12), we have the probability inequality $Pr(\{\max_{j \in S^C} |\hat{\beta}_j| > 1 - \frac{\gamma}{2}\} \cup \{\max_{i \in S} |v_i| > \frac{\lambda}{2\sqrt{\mu}}\}) \leq 2R \exp(-\frac{\lambda^2 \gamma^2}{8\sigma^2})$. Thus, the probability inequality on the complementary set is that

$$\Pr\left(\left\{\max_{j \in S^C} |\hat{\beta}_j| \leq 1 - \frac{\gamma}{2}\right\} \bigcap \left\{\max_{i \in S} |v_i| \leq \frac{\lambda}{2\sqrt{\mu}}\right\}\right) \geq 1 - 2R \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right).$$

Furthermore, we have that

$$\Pr\left(\Gamma \bigcap \left\{\|\boldsymbol{\delta}_S\|_\infty \leq \frac{\lambda}{2\sqrt{\mu}} + \lambda \|\left(\mathbf{B}_S^T \mathbf{B}_S\right)^{-1}\|_\infty\right\}\right) \geq 1 - 2R \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right), \tag{5.13}$$

where $\boldsymbol{\Gamma} = \{(\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$is the unique optimalsolution $\hat{\boldsymbol{\theta}}$ to problem(5.2)$\}$.

From the above discussion, we obtain the following Theorem 3, which illustrates the probabilistic consistency of the optimal solution $\hat{\boldsymbol{\theta}}$ to problem (5.2).

Theorem 3. *Suppose that the sparse structure (5.3) exists, the sparse signal $\boldsymbol{\theta}^* = (\boldsymbol{\theta}_S^{*T}, \mathbf{0}^T)^T$, and $\boldsymbol{B}$ has a full column rank. If there exist some parameters γ and μ, where $0 < \gamma < 1$ and $\mu > 0$ such that $\|\mathbf{B}_{S^C}^T \mathbf{B}_S (\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty \leq 1 - \gamma$ and $\lambda_{min}(\mathbf{B}_S^T \mathbf{B}_S) \geq \mu$, we have that*

$$Pr\left(\{\hat{S} \subseteq S\} \bigcap \left\{\|\boldsymbol{\delta}_S\|_\infty \leq \frac{\lambda}{2\sqrt{\mu}} + \lambda \|\left(\mathbf{B}_S^T \mathbf{B}_S\right)^{-1}\|_\infty\right\}\right) \geq 1 - 2R \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right), \tag{5.14}$$

where $\hat{S}$ is the index set of nonzero entries in $\hat{\boldsymbol{\theta}}$, and $\boldsymbol{\delta}_S = \hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_S^*$ and $\hat{\boldsymbol{\theta}}_S$ is the optimal solution of (5.6). Furthermore, if the lower bound of the absolute values of elements in $\boldsymbol{\theta}_S^*$ is larger than $\lambda(\frac{1}{2\sqrt{\mu}} + \|(\mathbf{B}_S^T \mathbf{B}_S)^{-1}\|_\infty)$, we have that

$$\Pr\left(\{\hat{S} = S\}\right) \geq 1 - 2R \exp\left(-\frac{\lambda^2 \gamma^2}{8\sigma^2}\right). \tag{5.15}$$

Proof. In terms of (5.13), the first inequality (5.14) follows from $\{\hat{S} \subseteq S\} \supseteq \boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma} = \{(\hat{\boldsymbol{\theta}}_S^T, \mathbf{0}^T)^T$ is the unique optimal solution $\hat{\boldsymbol{\theta}}$ to problem (5.2)$\}$.

If $\|\hat{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_S^*\|_\infty = \|\boldsymbol{\delta}_S\|_\infty \leq \frac{\lambda}{2\sqrt{\mu}} + \lambda\|(\mathbf{B}_S^T\mathbf{B}_S)^{-1}\|_\infty$ and the lower bound of the absolute values of elements in $\boldsymbol{\theta}_S^*$ is larger than $\frac{\lambda}{2\sqrt{\mu}} + \lambda\|(\mathbf{B}_S^T\mathbf{B}_S)^{-1}\|_\infty$, it can be checked that the entries in $\hat{\boldsymbol{\theta}}_S$ and $\boldsymbol{\theta}_S^*$ of the same index have the same sign. From (5.13), we can obtain the second inequality (5.15). □

Theorem 3 tells us that if we want to recover the sparsity in (5.3) with a probability p, we should choose a λ such that $1 - 2R\exp(-\frac{\lambda^2\gamma^2}{8\sigma^2}) > p$ when we know the intrinsic parameters γ and σ^2. So, to adaptively give a regularization parameter λ based on the data $\mathcal{A}$, we need to give two guesses on the intrinsic parameters γ and σ^2. We set λ to zero in Algorithm 1 and compute an estimated tensor $\hat{\mathcal{B}} = [\hat{\boldsymbol{\alpha}}; \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}]_R$ from the tensor data $\mathcal{A}$. The parameter σ^2 is estimated by using the variance $\hat{\sigma}^2$ of all the entries in the difference $\mathcal{A} - \hat{\mathcal{B}}$, and the parameter γ is set as $\hat{\gamma} = 1 - \max\{|\langle \mathbf{B}_i, \mathbf{B}_j\rangle| | i \neq j\}$, where $\mathbf{B}_i$ is the ith column in $\mathbf{B} = (\hat{\mathbf{X}} \odot \hat{\mathbf{Y}}) \odot \hat{\mathbf{Z}}$. With regularization parameter $\hat{\lambda} = \frac{2}{\hat{\gamma}}\sqrt{2\hat{\sigma}^2\log(200R)}$, the result of our algorithm is shown by using the simulated and real data in the next section.

6 | NUMERICAL EXPERIMENT

In this section, we have four types of numerical experiments for testing the performance of our algorithm. The codes of the first three experiments are written in Matlab with simulated data. In all the simulations, the initial guesses are randomly generated. The stopping criterion used in all experiments depends on two parameters: one is the upper bound of the number of iteration (e.g., iter_max= 10, 000), and the other is a tolerance to decide whether convergence has been reached (e.g., conv_tol= e^{-10}). The fourth numerical experiment is executed in C++ with OpenCV for surveillance video data. These experiments ran on a laptop computer with Intel i5 CPU 3.3 GHz and 8 GB memory.

6.1 | Estimated rank

We randomly create a tensor $\mathcal{A} \in \mathbb{R}^{10\times10\times10}$ with five rank-one components and then use LRAT to estimate the rank of $\mathcal{A}$ along with the increment of the regularization parameter. The upper bound R of rank($\mathcal{A}$) is fixed to 10 in the algorithm, whereas the regularization parameter λ varies from 0 to 0.1 by step 0.001. As shown in Figure 1, the estimated rank $\hat{R}$ has a decreasing trend as the parameter λ increases for these particular random tensor examples. Heuristically, the reason for this trend lies in the minimization the objective function in (3.3); an increase in λ reduces the value of $\|\hat{\boldsymbol{\alpha}}\|_1$ and thus the estimated rank $\hat{R}$.

6.2 | Accuracy of the estimated rank

We randomly generate three kinds of tensors with various dimensions and various rank-one component numbers (cn). The estimated rank $\hat{R}$ is calculated with the regularization parameter $\hat{\lambda} = \frac{2}{\hat{\gamma}}\sqrt{2\hat{\sigma}^2\log(200R)}$, where $\hat{\sigma}^2$ and $\hat{\gamma}$ are computed

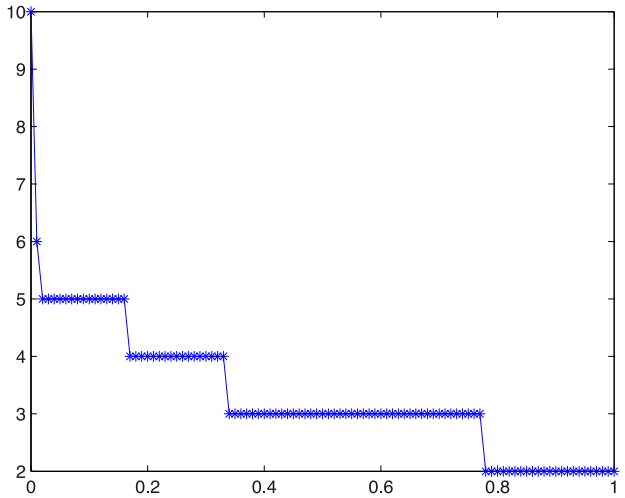

FIGURE 1 Trend of the estimated rank $\hat{R}$

TABLE 1 Mean and standard deviation of the estimated rank $\hat{R}$

	$I = J = K = 5$	$I = J = K = 10$	$I = J = K = 20$
$cn = 2$	2.28 (0.87)	3.25 (1.31)	5.41 (1.85)
$cn = 3$	3.15 (0.93)	4.49 (1.12)	7.2 (2.06)
$cn = 4$	3.6 (0.92)	5.18 (1.16)	8.35 (1.82)
$cn = 5$	n/a	5.77 (1.29)	9.98 (1.60)
$cn = 8$	n/a	7.52 (1.01)	10.88 (1.51)
$cn = 10$	n/a	n/a	11.69 (1.50)
$cn = 15$	n/a	n/a	14.11 (1.43)

Note. cn = component numbers.

as discussed in Section 5. Table 1 shows the mean and standard deviation of the estimated rank. Also, the experiments ran when cn is less than the mode size. Otherwise, no experiments (n/a) ran with cn larger than or equal to the mode size.

For each $cn(cn = 2, 3, 4)$ we randomly generate 100 tensors in $\mathbb{R}^{5\times5\times5}$ with $I = J = K = 5$ and then use the LRAT with the upper bound $R = 5$ to compute the estimated rank $\hat{R}$. As shown in Table 1, when the rank-one $cn = 3$, the average estimation difference of $\hat{R} - cn$ is 0.15 and the standard deviation of $\hat{R}$ is 0.93.

Similarly, for each cn, $cn = 2, 3, 4, 5, 8$, we randomly generate 100 tensors in $\mathbb{R}^{10\times10\times10}$, and for $cn = 2, 3, 4, 5, 8, 10, 15$, we randomly generate 100 tensors in $\mathbb{R}^{20\times20\times20}$. The upper bound R is set to $I = 10, 20$. The mean and standard deviation of $\hat{R}$ are shown in the last two columns of Table 1.

6.3 | Comparison between LRAT and modALS

In this subsection, we show the comparison between LRAT and modALS[7] on a toy model. A tensor $\mathcal{A}$ in $\mathbb{R}^{5\times5\times5}$ is randomly generated with three rank-one components for these experiments. Figure 2(a) demonstrates the residual function $\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2$ for the modALS and LRAT algorithms. The method modALS provided a better accuracy than LRAT; the residual errors are 10^{-4} (modALS) and 10^{-3} (LRAT). However, modALS was able only to generate a CP decomposition of an input rank of 5 with this accuracy. On the other hand, LRAT gave an estimated rank of 3 and a CP decomposition with a residual error of 10^{-3}. This is a sensible result that modALS has a low misfit because we know that it specializes on minimizing the residual function with a required rank input, whereas the LRAT algorithm is designed for the sparse optimization, as follows:

$$\min_{\mathbf{X},\mathbf{Y},\mathbf{Z},\boldsymbol{\alpha}} \frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \mathbf{N}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = 1.$$

The LRAT monotonically decreases $\frac{1}{2}\|\mathcal{A} - [\boldsymbol{\alpha}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}]_R\|_F^2 + \lambda\|\boldsymbol{\alpha}\|_1$ as shown in Figure 2(b) and provides an estimate on the number of rank-one components for any given tensor.

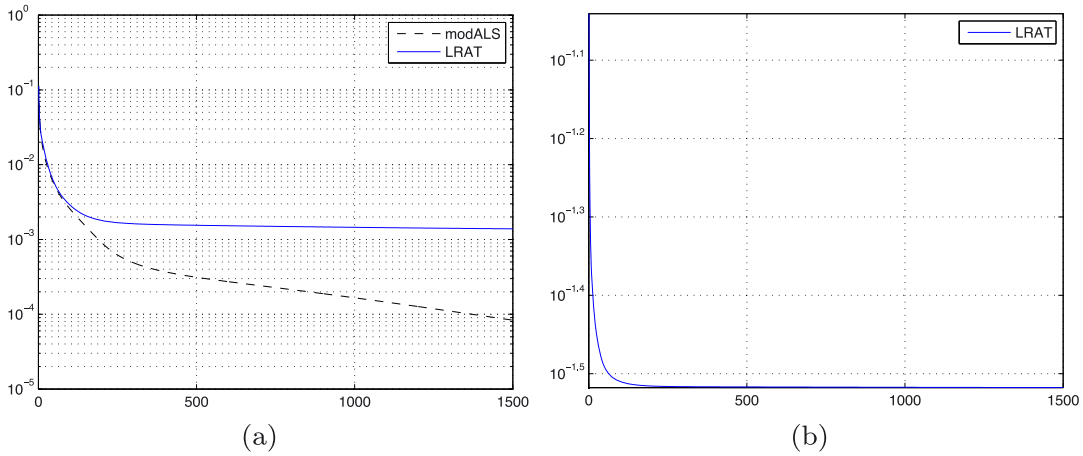

FIGURE 2 Comparison between LRAT and modALS. (a) Residual of LRAT and modALS; (b) objective function of LRAT

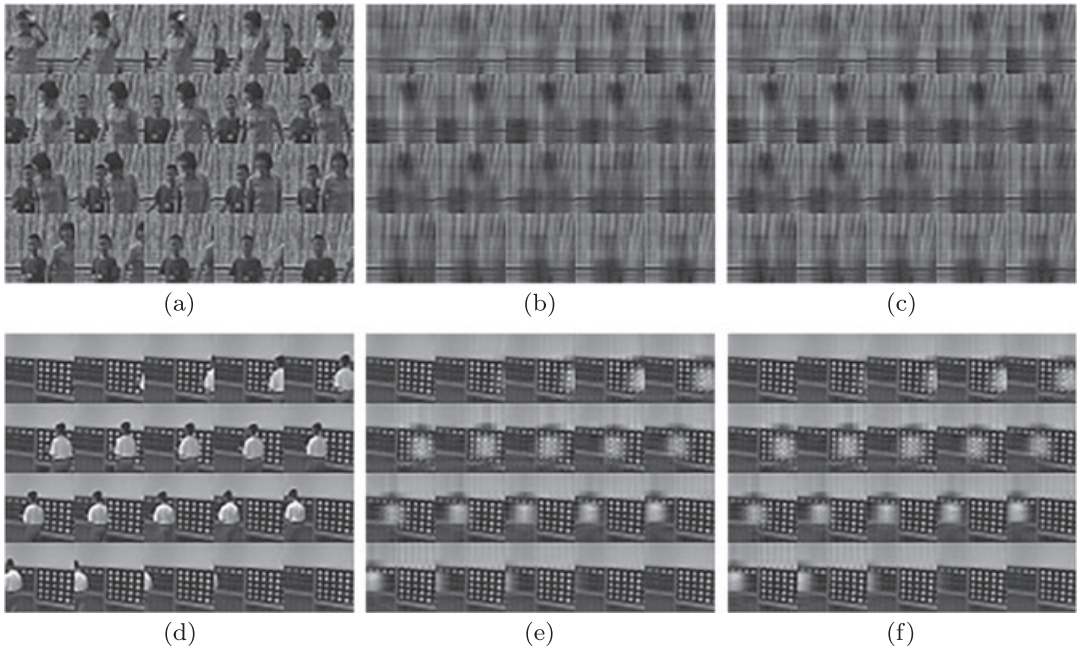

FIGURE 3 Computation results based on LRAT and modALS. (a) 20 frames for Fountain; (b) results from LRAT; (c) results from modALS; (d) 20 frames for Lobby; (e) results from LRAT; (f) results from modALS

6.4 | Application in surveillance video

Grayscale video data is a natural candidate for third-order tensors. Due to the correlation between the subsequent frames of the video, there exists some potential low-rank mechanism in the data. In this subsection, we apply the LRAT and the modALS to two surveillance videos* on Fountain and Lobby. For each video of 220 consecutive frames, we choose a region of interest with a resolution of 30×30.

Figure 3 demonstrates simulation results on the LRAT and the modALS. Here, the upper bound R is fixed to 400. Figure 3 shows 20 frames in the original video data $\mathcal{A}$ and those frames estimated by the LRAT and the modALS. The modALS provides an approximation with three factor matrices of $(30+30+220)\times 400$ elements. For the LRAT algorithm, the regularization parameter $\hat{\lambda}$ is set to $\frac{2}{\hat{\gamma}}\sqrt{2\hat{\sigma}^2\log(200R)}$, where $\hat{\sigma}^2$ and $\hat{\gamma}$ are computed as discussed in Section 5. The estimated number of rank-one components in $\hat{\mathcal{B}}$ is 378 for the Fountain video. The representation of $\hat{\mathcal{B}}$ with three factor matrices only needs $(30+30+220)\times 378$ elements. The estimated number of rank-one components is 392 for the Lobby video, and the representation with three factor matrices needs $(30+30+220)\times 392$ elements. Compared with the modALS algorithm, the LRAT has a smaller estimated rank but sacrifices more cpu time, because the LRAT algorithm requires an instructive (a starter) choice on $\hat{\lambda}$. For this case, we used the modALS algorithm to obtain a starter choice $\hat{\lambda}$ for LRAT.

7 | CONCLUSION AND FUTURE WORK

We propose an algorithm based on the proximal alternating minimization to detect the rank of tensors. This algorithm comes from the understanding of the low-rank approximation of tensors from sparse optimization. We also provide some theoretical guarantees on the convergence of this algorithm and a probabilistic consistency of the approximation result. Moreover, we suggest a way to choose a regularization parameter for practical computation. The simulation studies suggested that our algorithm can be used to detect the number of rank-one components in tensors.

The works presented in this paper have potential applications and extensions, especially in video processing and latent cn estimation. The ongoing work is to apply this low-rank approximation method to moving object detection and video data compression.

*The original data is from http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

ACKNOWLEDGEMENTS

This work is in part supported by the National Natural Foundation of China (Grant 11401092).

ORCID

Carmeliza Navasca http://orcid.org/0000-0001-9540-5366

REFERENCES

1. Comon P. Tensor decompositions. In: McWhirter JG, Proudler IK, editors. Mathematics in signal processing V. Oxford, UK: Clarendon Press, 2002; p. 1–24.
2. Sidiropoulos N, Bro R, Giannakis G. Parallel factor analysis in sensor array processing. IEEE Trans Signal Process. 2000;48:2377–2388.
3. Sorensen M, De Lathauwer L. Blind signal separation via tensor decomposition with vandermonde factor: canonical polyadic decomposition. IEEE Trans Signal Process. 2013;61:5507–5519.
4. Hao N, Kilmer ME, Braman K, Hoover RC. Facial recognition using tensor-tensor decompositions. SIAM J Imaging Sci. 2013;6:437–463.
5. Martin CD, Shafer R, Larue B. An order-p tensor factorization with applications in imaging. SIAM J Sci Comput. 2013;35:A474–A490.
6. Savas B, Elden L. Handwritten digit classification using higher order singular value decomposition. Pattern Recogn. 2007;40:993–1003.
7. Xu Y, Yin W. A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion. SIAM J Imaging Sci. 2013;6:1758–1789.
8. Beckmann C, Smith S. Tensorial extensions of the independent component analysis for multisubject FMRI analysis. NeuroImage. 2005;25:294–311.
9. Martínez-Montes E, Valdés-Sosa P, Miwakeichi F, Goldman R, Cohen M. Concurrent EEG/fMRI analysis by multiway partial least squares. NeuroImage. 2004;22:1023–1034.
10. Kolda TG, Bader BW. Tensor decompositions and applications. SIAM Rev. 2009;51:455–500.
11. Acar E, Dunlavy DM, Kolda T. A scalable optimization approach for fitting canonical tensor decompositions. J Chemometr. 2011;25(2):67–86.
12. De Lathauwer L, De Moor B, Vandevalle J. A multilinear singular value decomposition. SIAM J Matrix Anal Appl. 2000;21(4):1253–1278.
13. Khoromskij BN, Khoromskaia V. Multigrid accelerated tensor approximation of function related multi-dimensional arrays. SIAM J Sci Comput. 2009;31(4):3002–3026.
14. Eckart C, Young G. The approximation of one matrix by another of lower rank. Psychometrika. 1936;1:211–218.
15. Kolda TG. Orthogonal tensor decompositions. SIAM J Matrix Anal Appl. 2001;23(1):243–255.
16. Kruskal JB. Three-way arrays: rank and uniqueness of trilinear decompositions, with applications to arithmetic complexity and statistics. Linear Algebra Appl. 1977;18:95–138.
17. Landsberg JM. Tensors: geometry and applications. Graduate studies in mathematics. Vol. 128. Providence, RI: American Mathematical Society; 2012.
18. Li N, Hopke P, Kumar P, Cliff SC, Zhao Y, Navasca C. Source apportionment of time and size resolved ambient particulate matter. J Chemom Intell Lab Syst. 2013;129:15–20.
19. Domanov I, Lathauwer LD. Canonical polyadic decomposition of third-order tensors: reduction to generalized eigenvalue decomposition. SIAM J Matrix Anal Appl. 2014;35:636–660.
20. Navasca C, Lathauwer LD, Kindermann S. Swamp reducing technique for tensor decomposition. Paper presented at: Proceedings of the 16th European Signal Processing Conference; Lausanne, Switzerland; 2008.
21. Comon P, Golub G, Lim L-H, Mourrain B. Symmetric tensors and symmetric tensor rank. SIAM J Matrix Anal Appl. 2008;30(3):1254–1279.
22. Brachat J, Comon P, Mourrain B, Tsigaridas E. Symmetric tensor decomposition. Linear Algebra Appl. 2010;11-12:1851–1872.
23. Hillar CJ, Lim L-H. Most tensor problems are NP-hard. J ACM. 2013;60:1851–1872. Art. 45.
24. De Silva V, Lim L-H. Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM J Matrix Anal Appl. 2008;30:1084–1127.
25. Candès EJ, Plan Y. Near-ideal model selection by l1 minimization. Ann Stat. 2007;37: 2145–2177.
26. Candès EJ, Wakin M, Boyd S. Enhancing sparsity by reweighted l1 minimization. J Fourier Anal Appl. 2007;14:877–905.
27. Foucart S, Rauhut H. A mathematical introduction to compressive sensing. Basel, Switzerland: Birkhäuser; 2013.
28. Tibshirani R, Regression shrinkage and selection via the lasso. J Roy Statist Soc Ser B. 1996;58:267–288.
29. Chen S, Donoho D, Saunders M. Atomic decomposition by basis pursuit. SIAM J Sci Comp. 1998;20:33–61.
30. Bolte J, Sabach S, Teboulle M. Proximal alternating linearized minimization nonconvex and nonsmooth problems. Math Program. 2014;146:459–494.
31. Novati P, Russo MR. A GCV based Arnoldi-Tikhonov regularization method. BIT Numer Math. 2014;54:501–521.
32. Schwarz G. Estimating the dimension of a model. Ann Stat. 1978;6:461–464.

33. Kilmer M, O'Leary D. Choosing regularization parameters in iterative methods for ill-posed problems. SIAM J Matrix Anal Appl. 2001;22(4):1204–1221.
34. Morozov VA. Regularization methods for ill-posed problems. Boca Raton, FL: CRC Press; 1993.
35. Wainwright MJ. Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (Lasso). IEEE Trans Inf Theory. 2009;55:2183–2202.
36. Zhao P, Yu B. On model selection consistency of Lasso. J Mach Learn Res. 2006;7:2541–2567.
37. Uschmajew A. Local convergence of the alternating least squares algorithm for canonical tensor approximation. SIAM J Matrix Anal Appl. 2012;33(2):639–652.
38. Bro R. PARAFAC Tuturial and applications. Chemom Intell Lab Syst. 1997;38:149–171.
39. Candès EJ, Tao T. Decoding by linear programming. IEEE Trans Inform Theory. 2004;51:4203–4215.
40. Donoho D. Compressed sensing. IEEE Trans Inf Theory. 2006;52(4):1289–1306.
41. Candès EJ, Romberg J, Tao T. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans Inform Theory. 2004;52:489–509.
42. van den Berg E, Friedlander MP. Probing the Pareto frontier for basis pursuit solutions. SIAM J Sci Comput 2008;31:890–912.
43. Lim L-H, Comon P. Nonnegative approximations of nonnegative tensors. J Chemometr. 2009;23:432–441.
44. Golub G, Van Loan CF. Matrix computations. 4th ed. Baltimore, MD: Johns Hopkins University Press; 2013.
45. Karimi S, Vavasis S. IMRO: a proximal quasi-Newton method for solving l_1-regularized least squares problem. SIAM J Optim. 2017;27(2): 583–615.

How to cite this article: Wang X, Navasca C. Low-rank approximation of tensors via sparse optimization. *Numer Linear Algebra Appl*. 2018;25:e2136. https://doi.org/10.1002/nla.2136