Supervised Tensor Decomposition with Interactive Side Information

Jiaxin Hu, Chanwoo Lee, and Miaoyan Wang*
Department of Statistics, University of Wisconsin-Madison

May 24, 2021

Abstract

Higher-order tensors have received increased attention across science and engineering. While most tensor decomposition methods are developed for a single tensor observation, scientific studies often collect side information, in the form of node features and interactions thereof, together with the tensor data. Such data problems are common in neuroimaging, network analysis, and spatial-temporal modeling. Identifying the relationship between a high-dimensional tensor and side information is important yet challenging. Here, we develop a tensor decomposition method that incorporates multiple side information as interactive features. Unlike unsupervised tensor decomposition, our supervised decomposition captures the effective dimension reduction of the data tensor confined to feature space on each mode. An efficient alternating optimization algorithm is further developed. Our proposal handles a broad range of data types, including continuous, count, and binary observations. We apply the method to diffusion tensor imaging data from human connectome project and multi-relational political network data. We identify the key global connectivity pattern and pinpoint the local regions that are associated with available features. Our method will help the practitioners efficiently analyze tensor datasets in various areas. Toward this end, the package and data used are available at `https://CRAN.R-project.org/package=tensorregress`.

Keywords: Tensor data analysis, Supervised dimension reduction, Exponential family distribution, Generalized multilinear model, Alternating optimization

*The authors gratefully acknowledge NSF grant DMS-1915978 and funding from the Wisconsin Alumni Research Foundation.

1 Introduction

Multi-dimensional arrays, known as tensors, are often collected with side information on multiple modes in modern scientific and engineering studies. A popular example is in neuroimaging (Zhou et al., 2013). The brain connectivity networks are collected from a sample of individuals, accompanied by individual characteristics such as age, gender, and diseases status (see Figure 1a). Another example is in network analysis (Berthet and Baldin, 2020; Hoff, 2005). A typical social network consists of nodes that represent people and edges that represent friendships. Side information such as people's demographic information and friendship types are often available. In both examples, it is of keen scientific interest to identify the variation in the data tensor (e.g., brain connectivities, social community patterns) that is affected by available features. These seemingly different scenarios pose a common yet challenging problem for tensor data modeling.

In addition to the aforementioned challenges, many tensor datasets consist of non-Gaussian measurements. Examples include the political interaction dataset (Nickel et al., 2011; Hu et al., 2015) which measures action counts between countries under various relations, and the brain connectivity network dataset (Zhang et al., 2018; Wang and Li, 2020; Wang et al., 2019a) which is a collection of binary adjacency matrices. Classical tensor decomposition methods are based on minimizing the Frobenius norm of deviation, leading to suboptimal predictions for binary- or count-valued response variables. A number of supervised tensor methods have been proposed (Narita et al., 2012; Lock and Li, 2018; Rai et al., 2014) to address the tensor regression problem in various forms, such as scalar-to-tensor regression and tensor-response regression. These methods often assume Gaussian distribution for the tensor entries, or impose random designs for the feature matrices, both of which are less suitable for applications of our interest. The gap between theory and practice means a great opportunity to modeling paradigms and better capture the complexity in tensor data.

We present a general model and associated method for decomposing a data tensor whose entries are from exponential family with interactive side information. We formulate the learning task as a structured regression problem, with tensor observation serving as the

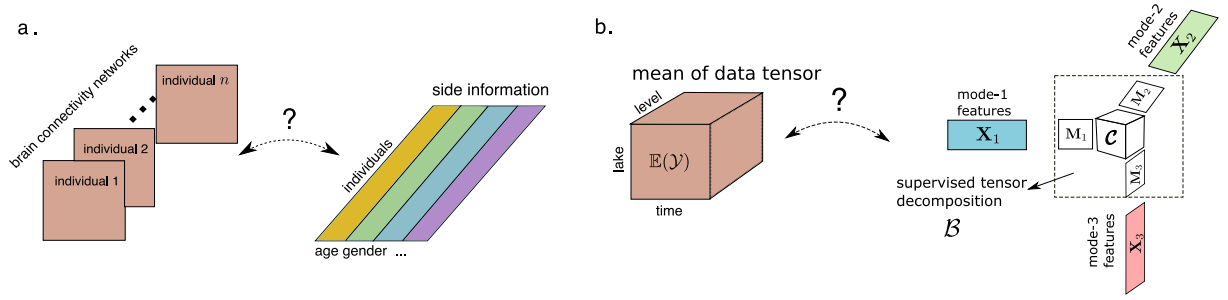

Figure 1: Examples of supervised tensor decomposition with interactive side information. (a) Network population model. (b) Spatio-temporal growth model.

response, and the multiple side information as interactive features. Figure 1b illustrates our model in the special case of order-3 tensors. A low-rank structure is imposed to the conditional mean of tensor observation, where unlike classical decomposition, the tensor factors $\boldsymbol{X}_k\boldsymbol{M}_k \in \mathbb{R}^{d_k \times r_k}$ belong to the space spanned by features $\boldsymbol{X}_k \in \mathbb{R}^{d_k \times p_k}$ for $k = 1, 2, 3$. The unknown matrices $\boldsymbol{M}_k \in \mathbb{R}^{p_k \times r_k}$ (referred to as "dimension reduction matrices") link the conditional mean to the feature spaces, thereby allowing the identification of variations in the tensor data attributable to the side information.

Our proposal blends the modeling power of generalized linear model (GLM) and the exploratory capability of tensor dimension reduction in order to take the best out of both worlds. We leverage GLM to allow heteroscedacity due to the mean-variance relationship in the non-Gaussian data. This flexibility is important in practice. Furthermore, our low-rank model on the (transformed) conditional mean tensor effectively mitigates the curse of high dimensionality. In classical GLM, the sample size and feature dimension are well defined; however, in the tensor data analysis, we observe only one realization of an order-K tensor and up to K interactive feature matrices. Both the number of tensor entries and feature dimension grow exponentially in K. Dimension reduction is therefore crucial for prediction and interpretability. We establish the statistical convergence of our estimator, and we quantify the gain in prediction through simulations and case studies.

Our work is closely related to but also clearly distinctive from several lines of previous work. The first line is a class of *unsupervised* tensor decomposition such as Tucker and CP decomposition (De Lathauwer et al., 2000; Kolda and Bader, 2009; Hong et al., 2020; Wang

and Song, 2017; Bi et al., 2018; Chi and Kolda, 2012) that aims to find the best low-rank representation of a data tensor. In contrast, our model is a *supervised* tensor learning, which aims to identify the association between a data tensor and multiple features. The low-rank factorization is determined jointly by the tensor data and feature matrices in our model.

The second line of work studies the tensor-on-tensor regression (Raskutti et al., 2019; Lock, 2018; Gahrooei et al., 2020). Our model shares a common ground with earlier approaches, but we provide more efficient solutions to new settings that have more practical significance. As we show in Section 3.3, the supervised tensor decomposition has an interesting connection with tensor-on-tensor regression. Previous methods (Lock, 2018; Lock and Li, 2018) mainly focus on Gaussian tensors. The Frobenius norm used in the objective function is statistical suboptimal for general exponential family tensors. Maximum likelihood estimator (MLE) is studied in Raskutti et al. (2019) and a convex relaxation algorithm is proposed to solve for low-rank tensor coefficients. However, in the tensor case, convex MLE suffers from both computational intractability and statistical suboptimality. We advocate a non-convex approach and provide strong evidence for its success in our setting. Most previous tensor regression focuses on prediction (Lock, 2018; Raskutti et al., 2019), and we go step further by finding the sufficient dimension reduction (Adragni and Cook, 2009), Span($\boldsymbol{M}_k$), that facilitates the identification of interaction effects in features (see Figure 1b). The latter approach greatly improves the *interpretability* in prediction. In this regards, our method opens up new opportunities for tensor data analysis in a wider range of applications.

The third line of work uses side information for various tensor learning tasks, such as for completion (Narita et al., 2012; Song et al., 2019; Cao et al., 2016) and for recommendation system (Ioannidis et al., 2019; Farias and Li, 2019). These methods also study tensors with side information, but they take regularization approaches to penalize predictions that are distant from side information (Cao et al., 2016; Song et al., 2019). One important difference is that their goal is prediction but not parameter estimation. The effects of features and their interactions are not estimated in these data-driven approaches. In contrast, our

goal is interpretable prediction, and we estimate factor matrices $\boldsymbol{M}_k$ using a model-based approach. Estimating $\boldsymbol{M}_k$ allows us to identify sufficient features and the interactions thereof. We numerically compare these two approaches in Section 5.

The remainder of the paper is organized as follows. Section 2 introduces tensor preliminaries. Section 3 presents the main model and three motivating examples for supervised tensor decomposition. We describe the quasi-likelihood estimation and alternating optimization algorithm in Section 4. In Section 5, we present numerical experiments and assess the performance in comparison to alternative methods. In Section 6, we apply the method to diffusion tensor imaging data from human connectome project and multi-relational social network data. We conclude in Section 7 with discussions about our findings and avenues of future work. All proofs are deferred to Supplementary Materials.

2 Preliminaries

We introduce the basic tensor properties (Kolda and Bader, 2009) used in the paper. We use lower-case letters (e.g., a, b, c) for scalars and vectors, upper-case boldface letters (e.g., $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$) for matrices, and calligraphy letters (e.g., $\mathcal{A}, \mathcal{B}, \mathcal{C}$) for tensors of order three or greater. Let $\mathcal{Y} = [\![y_{i_1,\ldots,i_K}]\!] \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ denote an order-K $(d_1, \ldots, d_K)$-dimensional tensor, where K is the number of modes and also called the order. The multilinear multiplication of a tensor $\mathcal{Y} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ by matrices $\boldsymbol{X}_k = [\![x^{(k)}_{i_k, j_k}]\!] \in \mathbb{R}^{p_k \times d_k}$ is defined as

$$\mathcal{Y} \times_1 \boldsymbol{X}_1 \times \cdots \times_K \boldsymbol{X}_K = [\![\sum_{i_1,\ldots,i_K} y_{i_1,\ldots,i_K} x^{(1)}_{j_1,i_1} \cdots x^{(K)}_{j_K,i_K}]\!],$$

which results in an order-K $(p_1, \ldots, p_K)$-dimensional tensor. For ease of presentation, we use the shorthand $\mathcal{Y} \times \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_K\}$ to denote the tensor-by-matrix product. For any two tensors $\mathcal{Y} = [\![y_{i_1,\ldots,i_K}]\!]$, $\mathcal{Y}' = [\![y'_{i_1,\ldots,i_K}]\!]$ of identical order and dimensions, their inner product is defined as

$$\langle \mathcal{Y},\ \mathcal{Y}' \rangle = \sum_{i_1,\ldots,i_K} y_{i_1,\ldots,i_K} y'_{i_1,\ldots,i_K}.$$

The tensor Frobenius norm and maximum norm are defined as

$$\|\mathcal{Y}\|_F = \langle \mathcal{Y},\ \mathcal{Y} \rangle^{1/2}, \quad \text{and} \quad \|\mathcal{Y}\|_\infty = \max_{i_1,\ldots,i_K} y_{i_1,\ldots,i_K}.$$

When a is a vector, we use $\|a\|_2 = \langle a, a \rangle^{1/2}$ to denote the vector 2-norm.

A higher-order tensor can be reshaped into a lower-order object. We use $\text{vec}(\cdot)$ to denote the operation that reshapes the tensor into a vector, and $\text{Unfold}_k(\cdot)$ to denote the operation that reshapes the tensor along mode k into a matrix of size d_k-by-$\prod_{i \neq k} d_i$. The multilinear rank of an order-K tensor $\mathcal{Y}$ is defined as a length-K vector $\boldsymbol{r} = (r_1, \ldots, r_K)$, where r_k is the rank of matrix $\text{Unfold}_k(\mathcal{Y})$, $k = 1, \ldots, K$. We use $\sigma_{\min}(\cdot)$ and $\sigma_{\max}(\cdot)$ denote the smallest and largest matrix singular values. We let $\boldsymbol{I}_d$ denote the $d \times d$ identity matrix and $[d]$ denote the d-set $\{1, \ldots, d\}$. We use $\mathbb{O}_{d,r}$ to denote the collection of all d-by-r matrices with orthogonal columns; i.e., $\mathbb{O}_{d,r} = \{\boldsymbol{P} \in \mathbb{R}^{d \times r} \colon \boldsymbol{P}^T \boldsymbol{P} = \boldsymbol{1}_r\}$. For ease of notation, we allow the basic arithmetic operators (e.g., $+, -, \geq$) and univariate functions $f \colon \mathbb{R} \to \mathbb{R}$ to be applied to tensors in an element-wise manner.

3 Motivation and model

3.1 General framework for supervised tensor decomposition

We begin with a general framework for supervised tensor decomposition and then discuss its implication in three concrete examples. Let $\mathcal{Y} = [\![y_{i_1,\ldots,i_K}]\!] \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ denote an order-K data tensor. Suppose the side information is available on each of the K modes. Let $\boldsymbol{X}_k = [\![x_{ij}]\!] \in \mathbb{R}^{d_k \times p_k}$ denote the feature matrix on the mode $k \in [K]$, where x_{ij} denotes the j-th feature value for the i-th tensor entity, for $(i, j) \in [d_k] \times [p_k]$, $p_k \leq d_k$. We propose a multilinear conditional mean model between the data tensor and feature matrices. Assume that, conditional on the features $\boldsymbol{X}_k$, the entries of tensor $\mathcal{Y}$ are independent realizations from an exponential family distribution, and the conditional mean tensor admits the form

$$\mathbb{E}(\mathcal{Y} | \boldsymbol{X}_1, \ldots, \boldsymbol{X}_K) = f(\Theta), \quad \text{with} \quad \Theta = \mathcal{B} \times \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_K\}, \tag{1}$$

where $\Theta \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ is the multilinear predictor, $\mathcal{B} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ is the unknown parameter tensor, $f(\cdot)$ is a known link function whose form depending on the data type of $\mathcal{Y}$, and $\times$ denotes the tensor-by-matrix product. The choice of link function is based on the assumed distribution family of tensor entries. Common choices of link functions include identity link for Gaussian distribution, logistic link for Bernoulli distribution, and $\exp(\cdot)$ link for Poisson distribution. In general, dispersion parameters can also be included in the model. Because our main focus is the tensor decomposition under the mean model, we omit the dispersion parameter in this section for ease of presentation.

In classical tensor decomposition, tensor factorization is performed on either data tensor $\mathcal{Y}$ or mean tensor $\mathbb{E}(\mathcal{Y})$. In the context of supervised tensor decomposition, we propose to factorize the latent parameter tensor $\mathcal{B}$,

$$\mathcal{B} = \mathcal{C} \times \{\boldsymbol{M}_1, \ \ldots, \ \boldsymbol{M}_K\}, \tag{2}$$

where $\mathcal{C} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ is a full-rank core tensor, and $\boldsymbol{M}_k \in \mathbb{R}^{p_k \times r_k}$ are factor matrices consisting of orthonormal columns, where $r_k \leq p_k$ for all $k \in [K]$. By the definition of multilinear rank, model equations (1) and (2) imply the low-rankness $\boldsymbol{r} = (r_1, \ldots, r_K)$ of the conditional mean tensor under the link function. We now reach our final model for supervised tensor decomposition,

$$\begin{aligned} \mathbb{E}(\mathcal{Y}|\boldsymbol{X}_1, \ldots, \boldsymbol{X}_K) &= f\left(\mathcal{C} \times \{\boldsymbol{X}_1\boldsymbol{M}_1, \ldots, \boldsymbol{X}_K\boldsymbol{M}_K\}\right), \\ &\text{with} \quad \boldsymbol{M}_k^T\boldsymbol{M}_k = \boldsymbol{I}_{r_k}, \ \boldsymbol{M}_k \in \mathbb{R}^{p_k \times r_k} \quad \text{for all } k = 1, \ldots, K, \end{aligned} \tag{3}$$

where the parameters of interest are $\boldsymbol{M}_k$ and $\mathcal{C}$. Note that model (3) assumes a fixed, known rank $\boldsymbol{r} = (r_1, \ldots, r_K)$; the adaptation to unknown rank will be addressed in Section 4.3. Figure 1b provides a schematic illustration of our model. The features $\boldsymbol{X}_k$ affect the distribution of tensor entries in $\mathcal{Y}$ through the form $\boldsymbol{X}_k\boldsymbol{M}_k$, which are r_k linear combinations of features on mode k. We call $\boldsymbol{X}_k\boldsymbol{M}_k$ the "supervised tensor factors" or "sufficient features" (Adragni and Cook, 2009), and call $\boldsymbol{M}_k$ the "dimension reduction matrix." The core tensor $\mathcal{C}$ collects the interaction effects between sufficient features across K modes.

Our goal is to find $\boldsymbol{M}_k$ and the corresponding $\mathcal{C}$. Note that $\boldsymbol{M}_k$ and $\mathcal{C}$ are identifiable only up to orthonormal transformations.

3.2 Three examples

We give three seemingly different examples that can all be formulated as our supervised tensor decomposition model (3).

Example 1 (Spatio-temporal growth model)**.** The growth curve model (Gabriel, 1998; Srivastava et al., 2008) was originally proposed as an example of bilinear model for matrix data, and we adopt its higher-order extension here. Let $\mathcal{Y} = [\![y_{ijk}]\!] \in \mathbb{R}^{d\times m\times n}$ denote the pH measurements of d lakes at m levels of depth and for n time points. Suppose the sampled lakes belong to q types, with p lakes in each type. Let $\{\ell_j\}_{j\in[m]}$ denote the sampled depth levels and $\{t_k\}_{k\in[n]}$ the time points. Assume that the expected pH trend in depth is a polynomial of order at most r and that the expected trend in time is a polynomial of order s. Then, the conditional mean model for the spatio-temporal growth can be represented as

$$\mathbb{E}(\mathcal{Y}|\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3) = \mathcal{C} \times \{\boldsymbol{X}_1\boldsymbol{M}_1,\ \boldsymbol{X}_2\boldsymbol{M}_2,\ \boldsymbol{X}_3\boldsymbol{M}_3\}, \tag{4}$$

where $\boldsymbol{X}_1 = \text{blockdiag}\{\mathbf{1}_p, \ldots, \mathbf{1}_p\} \in \{0,1\}^{d\times q}$ is the design matrix for lake types, and

$$\boldsymbol{X}_2 = \begin{pmatrix} 1 & \ell_1 & \cdots & \ell_1^r \\ 1 & \ell_2 & \cdots & \ell_2^r \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \ell_m & \cdots & \ell_m^r \end{pmatrix}, \quad \boldsymbol{X}_3 = \begin{pmatrix} 1 & t_1 & \cdots & t_1^s \\ 1 & t_2 & \cdots & t_2^s \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & \cdots & t_n^s \end{pmatrix}$$

are the design matrices for spatial and temporal effects, respectively, $\mathcal{C} \in \mathbb{R}^{r_1\times r_2\times r_3}$ is the unknown core tensor, and $\boldsymbol{M}_k$ are unknown dimension reduction matrices on each mode. The factors $\boldsymbol{X}_k\boldsymbol{M}_k$ are sufficient features in the mean model (4). The spatial-temporal model is a special case of our supervised tensor decomposition model (3), with features available on each of the three modes.

Example 2 (Network population model)**.** Network response model (Rabusseau and Kadri, 2016) is recently developed for neuroimaging analysis. The goal is to study the relationship between brain network connectivity pattern and features of individuals. Suppose we have a sample of n observations, $\{(\boldsymbol{Y}_i, \boldsymbol{x}_i)\colon i = 1, \ldots, n\}$, where for each individual $i \in [n]$, $\boldsymbol{Y}_i \in \{0,1\}^{d\times d}$ is the undirected adjacency matrix whose entries indicate presences/absences of connectivities between d brain nodes, and $\boldsymbol{x}_i \in \mathbb{R}^p$ is the individual's feature such as age, gender, cognition score, etc. The network-response model has the conditional mean

$$\text{logit}(\mathbb{E}(\boldsymbol{Y}_i|\boldsymbol{x}_i)) = \mathcal{B} \times_3 \boldsymbol{x}_i, \quad \text{for } i = 1, \ldots, n, \tag{5}$$

where $\mathcal{B} \in \mathbb{R}^{d\times d\times p}$ is a rank-(r_1, r_1, r_2) coefficient tensor, and $\mathcal{B}$ is assumed to be symmetric in the first two modes.

The model (5) is a special case of our supervised tensor decomposition, with feature matrix on the last mode of the tensor. Specifically, we stack the network observations $\{\boldsymbol{Y}_i\}$ together and obtain an order-3 response tensor $\mathcal{Y} \in \{0,1\}^{d\times d\times n}$. Define a feature matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^T \in \mathbb{R}^{n\times p}$. Then, the model (5) has the equivalent representation of supervised tensor decomposition,

$$\text{logit}(\mathbb{E}(\mathcal{Y}|\boldsymbol{X})) = \mathcal{C} \times \{\boldsymbol{M},\ \boldsymbol{M},\ \boldsymbol{X}\boldsymbol{M}'\},$$

where $\mathcal{C} \in \mathbb{R}^{r_1\times r_1\times r_2}$ is the core tensor, $\boldsymbol{M} \in \mathbb{R}^{d\times r}$ is the dimension reduction matrix on the first two modes, and $\boldsymbol{M}' \in \mathbb{R}^{p\times r_2}$ is for the last mode.

Example 3 (Dyadic data with node attributes)**.** Dyadic dataset consists of measurements on pairs of objects. Common examples include graphs and networks. Let $\mathcal{G} = (V, E)$ denote a graph, where $V = [d]$ is the node set of the graph, and $E \subset V \times V$ is the edge set. Suppose that we also observe feature vector $\boldsymbol{x}_i \in \mathbb{R}^p$ associated to each node $i \in V$. A probabilistic model on the graph $\mathcal{G} = (V, E)$ can be described by the following matrix regression. The edge connects the two vertices i and j independently of other pairs, and

the probability of connection is modeled as

$$\text{logit}\left(\mathbb{P}\left((i,j)\in E\right)\right) = \boldsymbol{x}_i^T \boldsymbol{B}\boldsymbol{x}_j = \langle \boldsymbol{B},\ \boldsymbol{x}_i^T\boldsymbol{x}_j\rangle, \tag{6}$$

where $\boldsymbol{B}\in\mathbb{R}^{p\times p}$ is a symmetric rank-r matrix. The low-rankness in $\boldsymbol{B}$ has demonstrated its success in modeling transitivity, balance, and communities in networks (Hoff, 2005). We show that our supervised tensor decompostion (3) also incorporates the graph model as a special case. Let $\mathcal{Y} = [\![y_{ij}]\!]$ be a binary matrix where $y_{ij} = \mathbb{1}_{(i,j)\in E}$. Define $\boldsymbol{X} = [\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n]^T \in \mathbb{R}^{n\times p}$. Then, the graph model (6) can be expressed as

$$\text{logit}(\mathbb{E}(\boldsymbol{Y}|\boldsymbol{X})) = \boldsymbol{C}\times\{\boldsymbol{X}\boldsymbol{M},\ \boldsymbol{X}\boldsymbol{M}\},$$

where $\boldsymbol{C}\in\mathbb{R}^{r\times r}, \boldsymbol{M}\in\mathbb{R}^{p\times r}$ are from the singular value decomposition of $\boldsymbol{B} = \boldsymbol{M}\boldsymbol{C}\boldsymbol{M}^T$.

In the above three examples and many other studies, researchers are interested in uncovering the variation in the data tensor that can be explained by features. Our supervised tensor decomposition (3) allows arbitrary numbers of feature matrices. When certain mode k has no side information, we set $\boldsymbol{X}_k = \boldsymbol{I}_{d_k}$ in the model (3). In particular, our model (3) reduces to classical unsupervised tensor decomposition (De Lathauwer et al., 2000; Hong et al., 2020) when no side information is available; i.e., $\boldsymbol{X}_k = \boldsymbol{I}_{d_k}$ for all $k\in[K]$.

3.3 Connection to sufficient dimension reduction and tensor-on-tensor regression

One important implication of our method is that we allow high-dimensionality in both tensor dimension d_k and feature dimension p_k. Recall that the model rank r_k is typically smaller than d_k and p_k. In such a case, the matrices $\boldsymbol{M}_k$ serve the role of simultaneous dimension reduction of the data tensor and features. The role of the sufficient features $\boldsymbol{X}_k\boldsymbol{M}_k$ can be seen in the following two conditional independence assumptions in model (3),

$$\mathcal{Y} \perp\!\!\!\perp \{\boldsymbol{X}_k\} \,\big|\, \{\boldsymbol{X}_k\boldsymbol{M}_k\} \ \text{ (independence between the tensor and features)},$$

$$y_{i_1,\ldots,i_K} \perp\!\!\!\perp y_{i'_1,\ldots,i'_K} \mid \{\boldsymbol{X}_k\boldsymbol{M}_k\} \quad \text{(independence within the tensor)},$$

where the second line holds for all $(i_1,\ldots,i_K) \neq (i'_1,\ldots,i'_K) \in [d_1]\times\cdots\times[d_K]$, and $\perp\!\!\!\perp$ denotes the independence. The first property highlights the "decorrelation" role of $\boldsymbol{M}_k$, in the same spirit as the sufficient dimension reduction (Adragni and Cook, 2009) in supervised learning, whereas the second property highlights the tensor dimension reduction in the usual unsupervised sense (consider, for example, $\boldsymbol{X}_k = \mathbf{I}_{d_k}$ for all $k \in [K]$).

Our model also has a close connection with tensor-on-tensor regression (Raskutti et al., 2019; Lock, 2018; Gahrooei et al., 2020; Hao et al., 2020). Specifically, model (3) can be viewed as a multivariate regression model, where the response is the vectorized tensor and the covariates are interactions between sufficient features across modes. We take an order-3 tensor under the Gaussian model for illustration. Let $\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{W}$ denote the feature matrix on mode $k = 1, 2, 3$, respectively. Suppose that each mode has two-dimensional sufficient features, denoted $\boldsymbol{M}_1\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2]$, $\boldsymbol{M}_2\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2]$, $\boldsymbol{M}_3\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2]$. Here $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{w}_1, \boldsymbol{w}_2$ are column vectors. Then the model (3) is a regression model with across-mode interactions,

$$\mathbb{E}(y_{ijk}|\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{W}) = c_{111}\boldsymbol{x}_{1i}\boldsymbol{z}_{1j}\boldsymbol{w}_{1k} + c_{121}\boldsymbol{x}_{i1}\boldsymbol{z}_{j2}\boldsymbol{w}_{k1} + \cdots + c_{221}\boldsymbol{x}_{2i}\boldsymbol{z}_{2j}\boldsymbol{w}_{1k} + c_{222}\boldsymbol{x}_{2i}\boldsymbol{z}_{2j}\boldsymbol{w}_{2k}, \quad (7)$$

where $[\![c_{ijk}]\!] \in \mathbb{R}^{2\times2\times3}$ are unknown interaction effects, $\boldsymbol{x}_{1i}$ denotes the i-th entry in the feature vector $\boldsymbol{x}_1$, and similar notations apply to other features. Note that lower-order interactions are naturally incorporated in (7) if we include an intercept column in the sufficient feature matrices. Model (7) shows the benefit of our supervised tensor decomposition for identifying across-mode interactions.

4 Estimation

4.1 Rank-constrained M-estimator

We develop a likelihood-based procedure to estimate $\mathcal{C}$ and $\boldsymbol{M}_k$ in (3). We adopt the exponential family as a flexible framework for different data types. In a classical generalized linear model with a scalar response y and feature $\boldsymbol{x}$, the density is expressed as

$$p(y|\boldsymbol{x}, \boldsymbol{\beta}) = c(y, \phi) \exp\left(\frac{y\theta - b(\theta)}{\phi}\right) \text{ with } \theta = \boldsymbol{\beta}^T \boldsymbol{x},$$

where $b(\cdot)$ is a known function, θ is the linear predictor, $\phi > 0$ is the dispersion parameter, and $c(\cdot)$ is a known normalizing function. The choice of link functions depends on the data types and on the observation domain of y, denoted $\mathbb{Y}$. For example, the observation domain is $\mathbb{Y} = \mathbb{R}$ for continuous data, $\mathbb{Y} = \mathbb{N}$ for count data, and $\mathbb{Y} = \{0, 1\}$ for binary data. The canonical link function f is chosen to be $f(\cdot) = b'(\cdot)$, the first-order derivative of $b(\cdot)$. Table 1 summarizes the canonical link functions for common types of distributions.

Data type	Gaussian	Poisson	Bernoulli
Domain $\mathbb{Y}$	$\mathbb{R}$	$\mathbb{N}$	$\{0, 1\}$
$b(\theta)$	$\theta^2/2$	$\exp(\theta)$	$\log(1 + \exp(\theta))$
link $f(\theta)$	θ	$\exp(\theta)$	$(1 + \exp(-\theta))^{-1}$

Table 1: Canonical links for common distributions.

In our context, we model the tensor entries $y_{i_1,\ldots,i_K}$, conditional on $\theta_{i_1,\ldots,i_K}$, as independent draws from an exponential family. Ignoring constants that do not depend on Θ, the quasi log-likelihood of (3) is equal to Bregman distance between $\mathcal{Y}$ and $b'(\Theta)$:

$$\mathcal{L}_{\mathcal{Y}}(\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K) = \langle \mathcal{Y}, \Theta \rangle - \sum_{i_1,\ldots,i_K} b(\theta_{i_1,\ldots,i_K}),$$
$$\text{where} \quad \Theta = \mathcal{C} \times \{\boldsymbol{M}_1 \boldsymbol{X}_1, \ldots, \boldsymbol{M}_K \boldsymbol{X}_K\}.$$

We propose a constrained maximum quasi-likelihood estimator (M-estimator),

$$(\hat{\mathcal{C}}, \hat{\boldsymbol{M}}_1, \ldots, \hat{\boldsymbol{M}}_k) = \underset{(\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K) \in \mathcal{P}}{\arg\max} \mathcal{L}_{\mathcal{Y}}(\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K), \tag{8}$$

where the parameter space $\mathcal{P}$ is

$$\mathcal{P} = \left\{ (\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K) \;\middle|\; \boldsymbol{M}_k^T \boldsymbol{M}_k = \mathbf{I}_{r_k} \text{ for all } k \in [K],\ \|\Theta(\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K)\|_\infty \leq \alpha \right\}.$$

The maximum norm constraint on the linear predictor Θ is a technical condition to avoid the divergence in the non-Gaussian variance.

4.2 Alternating optimization

We propose an alternating optimization algorithm to solve (8). The decision variables in the objective function (8) consist of $K+1$ blocks of variables, one for the core tensor $\mathcal{C}$ and K for the factor matrices $\boldsymbol{M}_k$. We notice that, if any K out of the $K+1$ blocks of variables are known, then the optimization reduces to a simple GLM with respect to the last block of variables. This observation leads to an iterative updating scheme for one block at a time while keeping others fixed. After each iteration, we rescale the core tensor $\mathcal{C}^{(t+1)}$ subject to the maximum norm constraint. The full algorithm is described in Algorithm 1.

Algorithm 1 Supervised Tensor Decomposition with Interactive Side Information

Input: Response tensor $\mathcal{Y} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, feature matrices $\boldsymbol{X}_k \in \mathbb{R}^{d_k \times p_k}$ for $k = 1, \ldots, K$, target Tucker rank $\boldsymbol{r} = (r_1, \ldots, r_K)$, link function f, maximum norm bound α

Output: Estimated core tensor $\hat{\mathcal{C}} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ and factor matrices $\hat{\boldsymbol{M}}_k \in \mathbb{R}^{p_k \times r_k}$.

1: Random initialization of the core tensor $\mathcal{C}$ and factor matrices $\boldsymbol{M}_k$.
2: **while** Do until convergence **do**
3: **for** $k = 1$ to K **do**
4: Obtain the factor matrix $\tilde{\boldsymbol{M}}_k \in \mathbb{R}^{p_k \times r_k}$ by a GLM with link function f.
5: Perform QR factorization $\tilde{\boldsymbol{M}}_k = \boldsymbol{Q}\boldsymbol{R}$, where $\boldsymbol{Q} \in \mathbb{R}^{p_k \times r_k}$ consists of orthonormal columns.
6: Update $\boldsymbol{M}_k \leftarrow \boldsymbol{Q}$ and core tensor $\mathcal{C} \leftarrow \mathcal{C} \times_k \boldsymbol{R}$.
7: **end for**
8: Update the core tensor $\mathcal{C}$ by solving a GLM with $\mathrm{vec}(\mathcal{Y})$ as response, $\otimes_{k=1}^{K} [\boldsymbol{X}_k \boldsymbol{M}_k]$ as features, and f as link function. Here $\otimes$ denotes the Kronecker product of matrices.
9: Rescale the core tensor $\mathcal{C}$ subject to the maximum norm constraint.
10: **end while**

The optimization (8) is a non-convex problem due to the non-convexity in the feasible set $\mathcal{P}$. Under mild conditions, our algorithm enjoys global convergence; i.e. any sequence of iterates generated by the alternating algorithm converges to a stationary point of $\mathcal{L}_{\mathcal{Y}}(\cdot)$

module orthogonal transformation. To establish the convergence properties of Algorithm 1, we first introduce the equivalent relationship induced by orthogonal transformation.

Definition 1 (Equivalence class)**.** Let $\mathcal{A} = (\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K) \in \mathbb{R}^{d_{\text{total}}}$ denote the collection of decision variables in the alternating optimization, where $d_{\text{total}} = \prod_k r_k + \sum_k r_k d_k$. Two parameters $\mathcal{A}' = (\mathcal{C}', \boldsymbol{M}_1', \ldots, \boldsymbol{M}_k')$ and $\mathcal{A} = (\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_k)$ are called equivalent, denoted $\mathcal{A} \sim \mathcal{A}'$, if and only if there exist orthogonal matrices $\boldsymbol{P}_k \in \mathbb{O}_{d_k, r_k}$ such that

$$\boldsymbol{M}_k' \boldsymbol{P}_k^T = \boldsymbol{M}_k \quad \text{for all } k \in [K], \quad \text{and} \quad \mathcal{C}' \times_1 \boldsymbol{P}_1 \times_2 \cdots \times_K \boldsymbol{P}_K = \mathcal{C}. \tag{9}$$

Equivalently, two parameters $\mathcal{A}$, $\mathcal{A}'$ are equivalent if the corresponding Tucker tensors (2) are the same, $\mathcal{B}(\mathcal{A}) = \mathcal{B}'(\mathcal{A}')$.

Proposition 4.1 (Global convergence)**.** Assume the set $\{\mathcal{A} \mid \mathcal{L}_{\mathcal{Y}}(\mathcal{A}) \geq \mathcal{L}_{\mathcal{Y}}(\mathcal{A}^{(0)})\}$ is compact and the stationary points of $\mathcal{L}_{\mathcal{Y}}(\mathcal{A})$ are isolated module the equivalence defined in (9). Furthermore, assume that $\alpha = \infty$; i.e., we impose no entrywise bound constrains on the parameter space. Then any sequence $\mathcal{A}^{(t)}$ generated by Algorithm 1 converges to a stationary point of $\mathcal{L}_{\mathcal{Y}}(\mathcal{A})$ module equivalence class.

In theory, global optimality of non-convex optimization is often challenging to obtain. Fortunately, we will show in Section 4.4 that the desired statistical performance holds for all local optimizers satisfying $\mathcal{L}_{\mathcal{Y}}(\hat{\mathcal{C}}, \hat{\boldsymbol{M}}_1, \ldots, \hat{\boldsymbol{M}}_K) \geq \mathcal{L}_{\mathcal{Y}}(\mathcal{C}_{\text{true}}, \boldsymbol{M}_{1,\text{true}}, \ldots, \boldsymbol{M}_{K,\text{true}})$, where the subscript "true" denotes the true parameter in model (3). This result indicates the global optimality is not necessarily a serious concern in our context, as long as the convergent objective is large enough. In the experiments we considered, we find that Algorithm 1 typically gives satisfactory convergent points upon random initialization. Figure 2 shows the trajectories of objective function for order-3 tensors based on model (3), where $d_k = d \in \{25, 30\}$, $p_k = 0.4d$, $r_k = r \in \{3, 6\}$ for $k = 1, 2, 3$. We consider input tensors with Gaussian, Bernoulli, and Poisson entries. Under all combinations of the dimension d, rank r, and type of the entries, Algorithm 1 converges quickly in a few iterations, and the objective values at convergent points are close to or larger than the value at true parameters.

1. add a sentence (XX out of XX random initialization leads to desired objective value ("…."))
2. We have added a new theory (…warm start by spectral initialization. —> global optimum when SNR is large enough.)

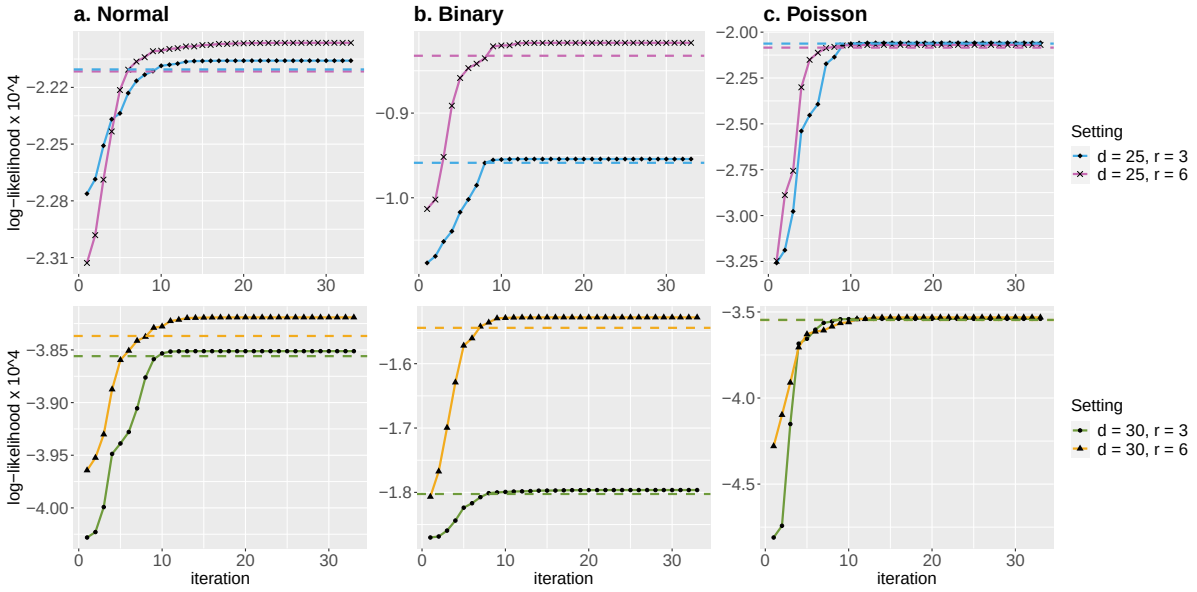

Figure 2: Trajectory of the objective function with various dimension d and rank r under (a) Gaussian (b) Bernoulli (c) Poisson models. The dashed line represents the objective value at true parameters.

4.3 Rank selection and computational complexity

Algorithm 1 assumes the rank $\boldsymbol{r}$ is given. In practice, the rank is often unknown and must be determined from the data. We propose to use Bayesian information criterion (BIC) and choose the rank that minimizes BIC, where

$$\mathrm{BIC}(\boldsymbol{r}) = -2\mathcal{L}_{\mathcal{Y}}(\hat{\mathcal{C}}, \hat{\boldsymbol{M}}_1, \ldots, \hat{\boldsymbol{M}}_K) + p_e(\boldsymbol{r}) \log(\prod\nolimits_k d_k). \tag{10}$$

Here, $p_e(\boldsymbol{r}) \stackrel{\text{def}}{=} \sum_k (p_k - r_k - 1) r_k + \prod_k r_k$ is the effective number of parameters in the model. We choose $\hat{\boldsymbol{r}}$ that minimizes $\mathrm{BIC}(\boldsymbol{r})$ via grid search. Our choice of BIC aims to balance between the goodness-of-fit for the data and the degree of freedom in the population model. We evaluate the empirical performance of BIC in Section 5.

The computational complexity of our Algorithm 1 is $O\left(d \sum_k p_k^3\right)$ for each loop of iterations, where $d = \prod_k d_k$ is the total size of the data tensor. More precisely, the update of core tensor costs $O(r^3 d)$, where $r = \prod_k r_k$ is the total size of the core tensor. The update of each factor matrix $\boldsymbol{M}_k$ involves a GLM with a d-length response, and d-by-$(r_k p_k)$ feature matrix. Solving such a GLM requires $O(d r_k^3 p_k^3)$, and therefore the cost for updating K

factors in total is $O(d\sum_k r_k^3 p_k^3)$. This complexity in tensor dimension matches with the classical tensor decomposition (Kolda and Bader, 2009).

4.4 Statistical properties

In this section, we provide the accuracy guarantee for the proposed M-estimator (8). Note that the factor matrices $\boldsymbol{M}_k$ are identifiable only up to orthogonal transformation. Therefore, we use angle distance to assess the accuracy in estimating the column space, $\mathrm{Span}(\boldsymbol{M}_k)$. For any two column-orthonormal matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{O}(d, r)$ of same dimension, the angle distance is defined as

$$\sin\Theta(\boldsymbol{A}, \boldsymbol{B}) = \|\boldsymbol{A}^T \boldsymbol{B}^\perp\|_\sigma = \max\left\{ \frac{\langle \boldsymbol{x}, \boldsymbol{y}\rangle}{\|\boldsymbol{x}\|_2 \|\boldsymbol{y}\|_2} : \ \boldsymbol{x} \in \mathrm{Span}(\boldsymbol{A}),\ \boldsymbol{y} \in \mathrm{Span}(\boldsymbol{B}^\perp) \right\}.$$

In modern applications, the tensor data and features are often large-scale. We are particularly interested in the high-dimensional regime in which both d_k and p_k diverge; i.e. $d_k \to \infty$ and $p_k \to \infty$, while $p_k/d_k \to \gamma_k \in [0, 1)$. As the size of problem grows, and so does the number of unknown parameters. The classical MLE theory does not directly apply. We leverage the recent development in random tensor theory and high-dimensional statistics to establish the error bounds of the estimation.

Assumption 1. We make the following assumptions:

- A1. There exist two positive constants $c_1, c_2 > 0$ such that $c_1 \le \sigma_{\min}(\boldsymbol{X}_k) \le \sigma_{\max}(\boldsymbol{X}_k) \le c_2$ for all $k \in [K]$.
- A1'. The feature matrices $\boldsymbol{X}_k$ are Gaussian designs with i.i.d. $N(0, 1)$ entries.
- A2. There exist two positive constants $L, U > 0$, such that $L\phi \le \mathrm{Var}(y_{i_1,\ldots,i_K} | \theta_{i_1,\ldots,i_K}) \le U\phi$, or equivalently, $L \le b''(\theta_{i_1,\ldots,i_K}) \le U$, for all $|\theta_{i_1,\ldots,i_K}| \le \alpha$. Here α is the upper bound of the linear predictor in (8), and $b''(\cdot)$ denotes the second-order derivative.

The assumptions are fairly mild. Assumptions A1 and A1' consider two separate scenarios about feature matrices. Assumption A1 is applicable when feature matrix is asymptotically non-singular and has bounded spectral norm, whereas Assumption A1' imposes

the commonly-used Gaussian design (Raskutti et al., 2019). Assumption A2 ensures the log-likelihood $\mathcal{L}_{\mathcal{Y}}(\Theta)$ is strictly concave in the linear predictor Θ.

Theorem 4.1 (Statistical convergence)**.** Consider a data tensor generated from model (3), where the entries are conditionally independent realizations from an exponential family. Let $(\hat{\mathcal{C}}, \hat{\boldsymbol{M}}_1, \ldots, \hat{\boldsymbol{M}}_K)$ be the M-estimator in (8) and $\hat{\mathcal{B}} = \hat{\mathcal{C}} \times \hat{\boldsymbol{M}}_1 \times \cdots \times \hat{\boldsymbol{M}}_K$. Define $r_{\text{total}} = \prod_k r_k$ and $r_{\max} = \max_k r_k$. Under Assumptions A1 and A2 with scaled feature matrices $\check{\boldsymbol{X}}_k = \sqrt{d_k}\boldsymbol{X}_k$, or under Assumptions A1' and A2 with original feature matrices, there exist two positive constants $C_1 = C_1(\alpha, K), C_2 = C_2(\alpha, K) > 0$ independent of dimensions $\{d_k\}$ and $\{p_k\}$, such that, with probability at least $1 - \exp(-C_1 \sum_k p_k)$,

$$\|\mathcal{B}_{\text{true}} - \hat{\mathcal{B}}\|_F^2 \leq \frac{C_2 r_{\text{total}}}{r_{\max}} \frac{\sum_k p_k}{\prod_k d_k}. \tag{11}$$

Furthermore, if the unfolded core tensor has non-degenerate singular values at mode $k \in [K]$, i.e., $\sigma_{\min}(\text{Unfold}_k(\mathcal{C}_{\text{true}})) \geq c > 0$ for some constant c, then

$$\sin^2\Theta(\boldsymbol{M}_{k,\text{true}},\ \hat{\boldsymbol{M}}_k) \leq \frac{C_2 r_{\text{total}}}{r_{\max}\sigma_{\min}^2(\text{Unfold}_k(\mathcal{C}_{\text{true}}))} \frac{\sum_k p_k}{\prod_k d_k}.$$

Theorem 4.1 establishes the statistical convergence for the estimator (8). In fact, our proof shows that the desired convergence rate holds not only for the M-estimator, but also for any local optimizers satisfying $\mathcal{L}_{\mathcal{Y}}(\hat{\mathcal{C}}, \hat{\boldsymbol{M}}_1, \ldots, \hat{\boldsymbol{M}}_K) \geq \mathcal{L}_{\mathcal{Y}}(\mathcal{C}_{\text{true}}, \boldsymbol{M}_{1,\text{true}}, \ldots, \boldsymbol{M}_{K,\text{true}})$. Consider a special case when tensor dimensions are equal on each of the modes, i.e., $d_k = d$ for all $k \in [K]$, and feature dimension grows with tensor dimension, $p_k = \gamma d$, $\gamma \in [0, 1)$, for $k \in [K]$. The result in (11) implies that the estimation has a convergence rate $\mathcal{O}(d^{-(K-1)})$. Therefore, our estimation is consistent in high dimensional regimes, and the convergence becomes especially favorable as the order of tensor data increases.

As immediate applications, we obtain the convergence rate for the three examples mentioned in Section 3.

Example 1 (Spatio-temporal growth model)**.** The estimated type-by-time-by-space coefficient tensor converges at the rate $\mathcal{O}\left(\frac{p+r+s}{dmn}\right)$ where $p \leq d$, $r \leq m$ and $s \leq n$. The estimation

achieves consistency as long as the dimension grows along either of the three modes.

Example 2 (Network population model)**.** The estimated node-by-node-by-feature tensor converges at the rate $\mathcal{O}\left(\frac{2d+p}{d^2n}\right)$ where $p \leq n$. The estimation achieves consistency as the number of individuals or the number of nodes grows.

Example 3 (Dyadic data with node attributes)**.** The estimated feature-by-feature matrix converges at the rate $\mathcal{O}\left(\frac{p}{d^2}\right)$ where $p \leq d$. Again, our estimation achieves consistency as the number of nodes grows.

We conclude this section by providing the accuracy, measured in KL divergence, for the response distribution.

Corollary 4.1 (Prediction error)**.** Assume the same set-up as in Theorem 4.1. Let $\mathbb{P}_{\mathcal{Y}_{\text{true}}}$ and $\mathbb{P}_{\hat{\mathcal{Y}}}$ denote the distributions of $\mathcal{Y}$ given the true parameter $\mathcal{B}_{\text{true}}$ and estimated parameter $\hat{\mathcal{B}}$, respectively. Then, we have, with probability at least $1 - \exp(C_1 \sum_k p_k)$,

$$\text{KL}(\mathbb{P}_{\mathcal{Y}_{\text{true}}},\ \mathbb{P}_{\hat{\mathcal{Y}}}) \leq \frac{C_3 r_{\text{total}}}{r_{\max}} \sum_k p_k,$$

where $C_3 = C_3(\alpha, K) > 0$ is a constant independent of dimensions $\{d_k\}$ and $\{p_k\}$.

5 Numerical experiments

We evaluate the empirical performance of our supervised tensor decomposition (STD) through simulations. We consider order-3 tensors with a range of distribution types. Unless otherwise specified, the conditional mean tensor is generated form model (3), where the core tensor entries are i.i.d. drawn from Uniform[-1,1], the factor matrix $\boldsymbol{M}_k$ is uniformly sampled with respect to Haar measure from matrices with orthonormal columns. The feature matrix $\boldsymbol{X}_k$ is either an identity matrix (i.e., no feature available) or Gaussian random matrix with i.i.d. entries from $N(0,1)$. The linear predictor $\Theta = \mathcal{C} \times \{\boldsymbol{M}_1\boldsymbol{X}_1, \boldsymbol{M}_2\boldsymbol{X}_3, \boldsymbol{M}_3\boldsymbol{X}_3\}$ is scaled such that $\|\Theta\|_\infty = 1$. Conditional on the linear predictor $\Theta = [\![\theta_{ijk}]\!]$, the entries in the tensor $\mathcal{Y} = [\![y_{ijk}]\!]$ are drawn independently according to three probabilistic models:

True Rank $\boldsymbol{r}$	Dimension (Gaussian tensors)		Dimension (Poisson tensors)	
(3, 3, 3)	(**3.0**, **3.0**, **3.0**)	(**3.0**, **3.0**, **3.0**)	(**3.0**, **3.0**, **3.0**)	(**3.0**, **3.0**, **3.0**)
(4, 4, 6)	(3.0, 3.0, **4.6**)	(**4.0**, **4.0**, **5.3**)	(3.0, 3.0, **5.3**)	(**4.0**, **4.0**, **5.6**)
(6, 8, 8)	(5.0, 5.0, 5.0)	(**6.0**, **8.0**, **8.0**)	(5.0, 5.0, 6.7)	(**6.0**, **8.0**, **8.0**)

Table 2: Rank selection via BIC. The estimated ranks are averaged across 30 simulation. Bold number indicates the ground truth is within two standard deviations of the estimate.

(a) Gaussian model: continuous tensor entries $y_{ijk} \sim N(\alpha\theta_{ijk}, 1)$.

(b) Poisson model: count tensor entries $y_{ijk} \sim \text{Poisson}\left(e^{\alpha\theta_{ijk}}\right)$.

(c) Bernoulli model: binary tensor entries $y_{ijk} \sim \text{Bernoulli}\left(\frac{e^{\alpha\theta_{ijk}}}{1+e^{\alpha\theta_{ijk}}}\right)$.

Here $\alpha > 0$ is a scalar controlling the magnitude of the effect size. In each simulation study, we report the mean squared error (MSE), $\|\mathcal{B}_{\text{true}} - \hat{\mathcal{B}}\|_F^2$, averaged across 30 replications.

5.1 Finite-sample performance

The first experiment assesses the selection accuracy of our BIC criterion (10). We consider the balanced situation where $d_k = d$, $p_k = 0.4d_k$ for $k = 1, 2, 3$. We set $\alpha = 4$ and consider various combinations of dimension d and rank $\boldsymbol{r} = (r_1, r_2, r_3)$. For each combination, we minimize BIC using a grid search over three modes. The hyper-parameter α is set to infinity in the fitting, which essentially imposes no constraint on the tensor magnitude. Table 2 reports the selected rank averaged over $n_{\text{sim}} = 30$ replicates. We found that in the high-rank setting with $d = 20$, the selected rank slightly underestimates the true rank, and the accuracy immediately improves when either the dimension increases to $d = 40$ or the rank reduces to $\boldsymbol{r} = (3, 3, 3)$. This agrees with our expectation, because in the tensor decomposition, the sample size is related to the number of tensor entries. A larger d implies a larger sample size, so the BIC selection becomes more accurate.

The second experiment evaluates the accuracy when features are available on all modes. We set $\alpha = 10, d_k = d, p_k = 0.4d_k, r_k = r \in \{2, 4, 6\}$ and increase d from 30 to 60. Our theoretical analysis suggests that $\hat{\mathcal{B}}$ has a convergence rate $\mathcal{O}(d^{-2})$ in this setting. Figure 3 plots the estimation error versus the "effective sample size", d^2, under three different distribution models. We find that the empirical MSE decreases roughly at the rate of $1/d^2$, which is consistent with our theoretical results. We also observe that, tensors

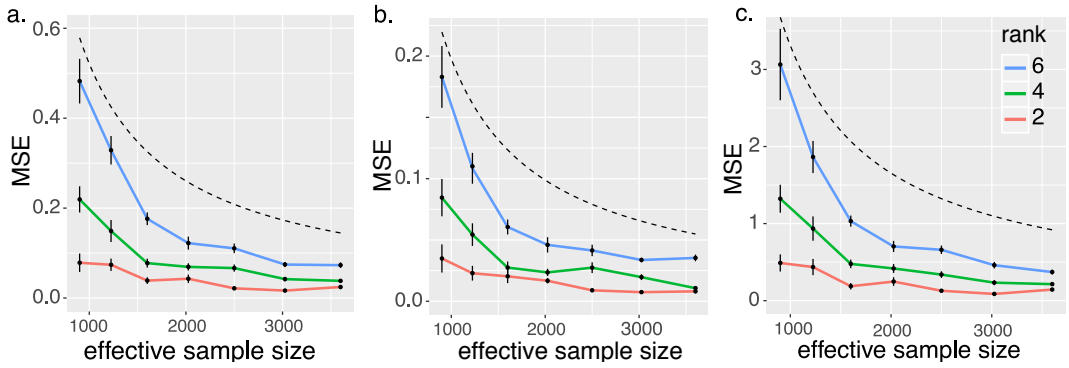

Figure 3: Estimation error against effective sample size. The three panels plot the MSE when the response tensors are generated from (a) Gaussian (b) Poisson and (c) Bernoulli models. The dashed curves correspond to $\mathcal{O}(1/d^2)$.

with higher rank tend to yield higher estimation errors, as reflected by the upward shift of the curves as r increases. Indeed, a larger r implies a higher model complexity and thus greater difficulty in the estimation.

5.2 Comparison with GLMs under stochastic block models

The third experiment investigates the performance of our model under correlated feature effects. We mimic the scenario of brain imaging analysis. A sample of $d_3 = 50$ networks are simulated, one for each individual. Each network measures the connections between $d_1 = d_2 = 20$ brain nodes. We simulate $p = 5$ features for the each of the 50 individuals. These features may represent, for example, age, gender, cognitive score, etc. Recent study has suggested that brain connectivity networks often exhibit community structure represented as a collection of subnetworks, and each subnetwork is comprised of a set of spatially distributed brain nodes. To accommodate this structure, we utilize the stochastic block model (Abbe, 2017) to generate the effect size. Specifically, we partition the nodes into r blocks by assigning each node to a block with uniform probability. Edges within a same block are assumed to share the same feature effects, where the effects are drawn i.i.d. from $N(0, 1)$. We then apply our tensor regression model to the network data using the BIC-selected rank. Note that in this case, the true model rank is unknown; the rank of a r-block network is not necessarily equal to matrix rank r (Wang and Zeng, 2019).

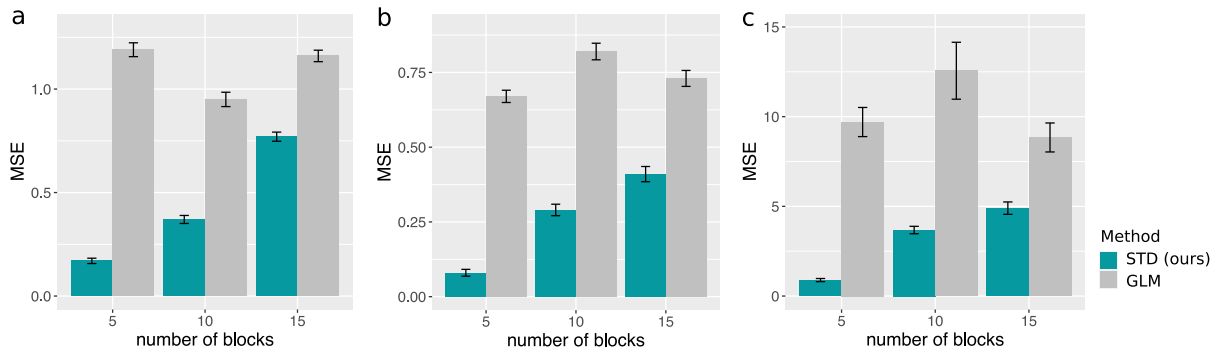

Figure 4: Performance comparison under stochastic block models. The three panels plot the MSE when the response tensors are generated from (a) Gaussian (b) Poisson and (c) Bernoulli models. The x-axis represents the number of blocks in the networks.

Figure 4 compares the MSE of our method with a multiple-response GLM approach. The multiple-response GLM is to regress the dyadic edges, one at a time, on the features, and this model is repeatedly fitted for each edge. As we find in Figure 4, our tensor regression method achieves significant error reduction in all three data types considered. The outperformance is substantial in the presence of large communities; even in the less structured case ($\sim 20/15 = 1.33$ nodes per block), our method still outer-performs GLM. The possible reason is that the multiple-response GLM approach does not account for the correlation among the edges, and suffers from overfitting. In contrast, the low-rankness in our modeling incorporates the shared information across entries. By selecting the rank in a data-driven way, our method achieves accurate estimation in a wide range of settings.

5.3 Comparison with other tensor methods

We compare our supervised tensor decomposition with three other tensor methods:

- Higher-order low-rank regression (**HOLRR**, Rabusseau and Kadri (2016)) is a least-square based tensor regression that allows features on a single mode.
- Higher-order partial least square (**HOPLS**, Zhao et al. (2012)) is a dimension-reduction method that jointly models a tensor response and a tensor feature.
- Subsampled tensor projected gradient (**TPG**, Yu and Liu (2016)) considers the same objective as **HOLRR** but instead uses a different algorithm to solve the problem.

These three methods are the closest algorithms to ours, in that they all relate a data

tensor to features using a low-rank structure. The three existing methods allow only Gaussian data, whereas ours is applicable to any exponential family distribution including Gaussian, Bernoulli, Poisson, etc. For fair comparison, we consider Gaussian tensors in the experiment. Because not every method returns the effect estimate $\hat{\mathcal{B}}$ as outputs, we measure the accuracy using mean squared prediction error, MSPE $= (\prod_k d_k)^{-1}\|\hat{\mathcal{Y}} - f(\Theta)\|_F^2$, where $f(\Theta)$ is the conditional mean of the tensor, and $\hat{\mathcal{Y}}$ is the fitted tensor from each method.

The comparison is assessed from three aspects: (a) benefit of incorporating features from multiple modes; (b) prediction error with respect to sample size; (c) sensitivity of accuracy with respect to model complexity. We use similar simulation setups as in our experiment II, but consider combinations of rank, $\boldsymbol{r} = (3, 3, 3)$ (low) vs. $(4, 5, 6)$ (high), signal $\alpha = 3$ (low) vs. 6 (high), dimension d ranging from 20 to 100 for modes with features, and $d = 20$ for modes without features. Two methods (**STD** and **HOLRR**) require the tensor rank as inputs. For fair comparison, we provide both algorithms the true rank. For algorithms (**HOPLS** and **TPG**) that use different notions of model rank, we use a grid search to set the hyperparameter that gives the best mean square prediction error.

Figures 5a-b show the averaged prediction error across 30 replicates. We see that our **STD** outperforms others, especially in the low-signal, high-rank setting. As the number of informative modes (i.e., modes with available features) increases, the **STD** exhibits a substantial reduction in error whereas others remain unchanged (Figure 5b). This showcases the benefit of incorporation of multiple features. Note that our method **STD** is most comparable to **HOLRR** when there is only a single informative mode. In such a case, both methods share a same cost function but have different algorithms. **STD** alternates between informative and non-informative modes, whereas **HOLRR** approximates the non-informative modes without alternating. The accuracy gain in Figure 5 demonstrates the benefit of alternating algorithm – incorporation of informative modes also improves the estimation in the non-informative modes.

Figures 5c-d compare the prediction error with respect to effective sample size when only one mode has side information. In the high-signal low-rank setting, our method has similar performance as **HOLRR**, and the improvement becomes more pronounced in the

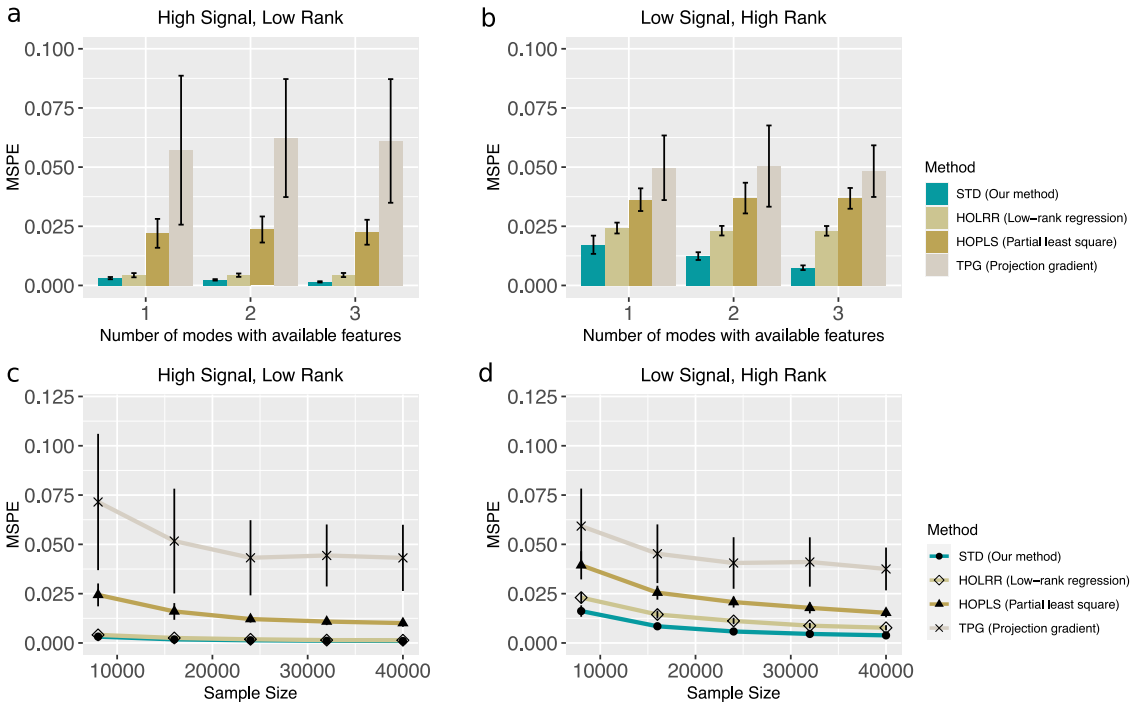

Figure 5: Comparison between different tensor methods. Panels (a) and (b) plot MSPE versus the number of modes with available features. Panels (c) and (d) plot MSPE versus the effective sample size d^2. We consider rank $\boldsymbol{r} = (3, 3, 3)$ (low), $\boldsymbol{r} = (4, 5, 6)$ (high), and signal $\alpha = 3$ (low), $\alpha = 6$ (high).

low-signal high-rank setting. The latter setting is harder because of the higher inter-mode complexity, and our **STD** method shows the advantage in addressing this challenge.

6 Data analyses

We apply our supervised tensor decomposition to two datasets. The first application studies the brain networks in response to individual attributes (i.e., feature on one mode), and the second application focuses on multi-relational network analysis with dyadic attributes (i.e., features on two modes).

6.1 Application to human brain connection data

The Human Connectome Project (HCP) aims to build a network map that characterizes the anatomical and functional connectivity within healthy human brains (Geddes, 2016).

We follow the preprocessing procedure as in Zhang et al. (2018) and parcellate the brain into 68 regions of interest (Desikan et al., 2006). The dataset consists of 136 brain structural networks, one for each individual. Each brain network is represented as a 68-by-68 binary matrix, where the entries encode the presence or absence of fiber connections between the 68 brain regions. We consider four individual features: gender (65 females vs. 71 males), age 22-25 ($n = 35$), age 26-30 ($n = 58$), and age 31+ ($n = 43$). The preprocessed dataset is released in our R package `tensorregress`. The goal is to identify the connection edges that are affected by individual features. A key challenge in brain network is that the edges are correlated; for example, the nodes in edges may be from a same brain region, and it is of importance to take into account the within-dyad dependence.

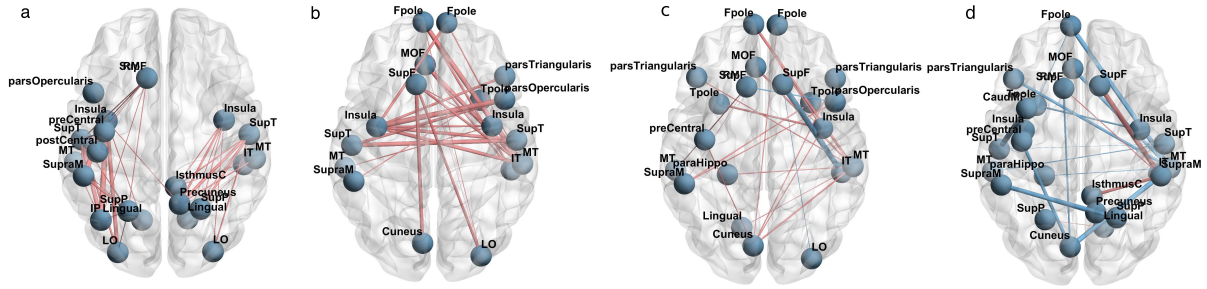

Figure 6: Top edges with large effects. (a) Global effect; (b) Female effect; (c) Age 22-25; (d) Age 31+. Red edges represent positive effects and blue edges represent negative effects. The edge-width is proportional to the magnitude of the effect size.

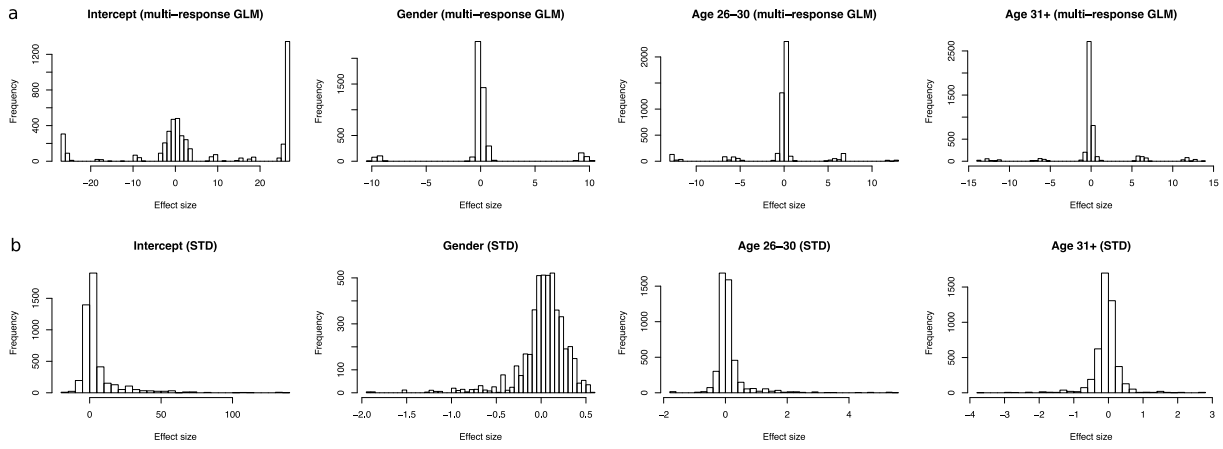

Figure 7: Comparison of estimated feature effects the HCP data using (a) multi-response GLM and (b) supervised tensor decomposition (STD).

We perform the supervised tensor decomposition to the HCP data. The BIC selection suggests a rank $\boldsymbol{r} = (10, 10, 4)$ with quasi log-likelihood $\mathcal{L}_{\mathcal{Y}} = -174654.7$. We utilize the sum-to-zero contrasts in coding the feature effects, and depict only the top 3% edges whose connections are non-constant across the sample. Figure 6 shows the top edges with high effect size, overlaid on the Desikan atlas brain template (Desikan et al., 2006). We find that the global connection exhibits clear spatial separation, and that the nodes within each hemisphere are more densely connected with each other (Figure 6a). In particular, the superior-temproal ($SupT$), middle-temporal (MT) and Insula are the top three popular nodes in the network. Interestingly, female brains display higher inter-hemispheric connectivity, especially in the frontal, parental and temporal lobes (Figure 6b). This is in agreement with a recent study showing that female brains are optimized for inter-hemispheric communication (Ingalhalikar et al., 2014). We find several edges with declined connection in the group Age 31+. Those edges involve Frontal-pole ($Fploe$), superior-frontal ($SupF$) and Cuneus nodes. The Frontal-pole region is known for its importance in memory and cognition, and the detected decline with age further highlights its biological importance.

Figure 7 compares the estimated coefficients from our method with those from multiple-response GLM approach. The multiple-response GLM is to regress the brain edges, one at a time, on the individual-level features, and this model is repeatedly fitted for every edge $\in [68] \times [68]$. As we can see in the figure, our tensor decomposition method shrinkages the coefficients towards center, thereby implicitly enforcing the sharing between feature effects.

6.2 Application to political relation data

The second application studies the multi-relational networks with node-level attributes. We consider *Nations* dataset (Nickel et al., 2011) which records 56 relations among 14 countries between 1950 and 1965. The multi-relational networks can be organized into a $14 \times 14 \times 56$ binary tensor, with each entry indicating the presence or absence of an action, such as "sending tourist to", "export", "import", between countries. The 56 relations span the fields of politics, economics, military, religion, etc. In addition, country-level attributes are also available, and we focus on the following six features: *constitutional, catholics, law ngo,*

political leadership, geography, and *medicine ngo*. The goal is to identify the variation in connections due to country-level attributes and their interactions. One of the key features is that the 56 relations are correlated, and we would like to take it into account in assessing the feature effects.

We apply our tensor model to the *Nations* data. The multi-relational network $\mathcal{Y}$ is a binary data tensor, and the country attributes $\boldsymbol{X} \in \mathbb{R}^{14\times 6}$ are features on both the 1st and 2nd modes. The BIC criterion suggests a rank $\boldsymbol{r} = (4, 4, 4)$ for the coefficient tensor $\mathcal{B} \in \mathbb{R}^{6\times 6\times 56}$. We perform the supervised tensor decomposition and obtain the dimension reduction matrices $\hat{\boldsymbol{M}}_k$ from the model. Then we apply K-mean clustering to dimension reduction matrix on each of the modes. Table 7 shows the K-means clustering of the 56 relations based on the dimension reduction matrix on the 3rd mode. We find that the relations reflecting the similar aspects of actions are grouped together. In particular, Cluster I consists of military relations such as *violentactions*, *warnings* and *militaryactions*; Clusters II and III capture the economic relations such as *economicaid, booktranslations, tourism*; and Cluster IV represents the political alliance and territory relations.

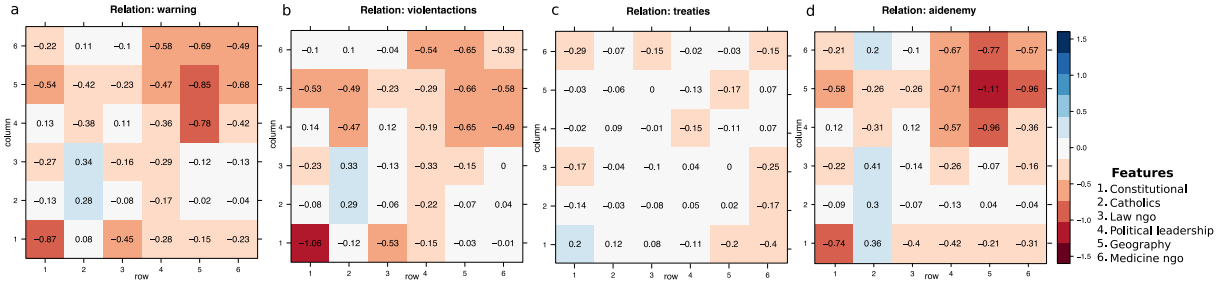

Figure 8: Estimated feature effects in the *Nations* data analysis. Panels (a)-(d) represent the estimated effects of country-level attributes towards the connection probability, for relations *warning*, *violentactions*, *treaties*, and *aidenemy*, respectively.

	Category	Relations
I	Military	warning, violentactions, militaryactions, duration, negativebehavior, protests, severdiplomatic timesincewar, commonbloc0, commonbloc1, blockpositionindex, expeldiplomats
II	Emigrant	emigrants, emigrants3, relemigrants, accusation, nonviolentbehavior, ngoorgs, commonbloc2, intergovorgs3 releconomicaid, relintergovorgs, relngo, students, relstudents, economicaid, negativecomm, militaryalliance
III	Economics	treaties, reltreaties, officialvisits, exportbooks, relexportbooks, booktranslations, relbooktranslations boycottembargo, weightedunvote, unweightedunvote, reltourism, tourism, tourism3, exports, exports3 relexports, intergovorgs, ngo, embassy, reldiplomacy, timesinceally, independence, conferences, dependent
IV	Territory	aidenemy, lostterritory, unoffialacts, attackembassy

Table 3: K-means clustering of relations based on dimension reduction on the 3rd mode.

To investigate the effects of dyadic attributes towards connections, we depict the estimated coefficients $\hat{\mathcal{B}} = [\![\hat{b}_{ijk}]\!]$ for several relation types (Figure 8). Note that the each entry $\hat{b}_{ijk}$ estimates the contribution, at the logit scale, of feature pair (i, j) (ith feature for the "sender" country and jth feature for the "receiver" country) towards the connection of relation k. Several interesting findings emerge from the observation. We find that relations belonging to a same cluster tend to have similar feature effects. For example, the relations "warning" and "violentactions" are classified into Cluster I, and both exhibit similar effect patterns (Figures 8a-b). Moreover, the feature *constitutional* has a strong effect in the relation "violentactions" and "warning", whereas the effect is weaker in the relation "treaties". The result is plausible because the constitutional attributes affect political actions more often than economical actions. The entries in $\mathcal{B}$ are useful for revealing interaction effects in a context-specific way. From Figure 8, we find a strong interaction between *geography* and *political leadership* in the relation "warning", and a strong interaction between *geogrphy* and *medicine ngo* in the relation "aidenemy". The relation-specific effect pattern showcases the applicability of our method in revealing complex interactions.

7 Discussion and future work

We have developed a supervised tensor decomposition method with side information on multiple modes. One important challenge of tensor data analysis is the complex interdependence among tensor entries and between multiple features. Our approach incorporates side information as feature matrices in the conditional mean tensor. The empirical results demonstrate the improved interpretability and accuracy over previous approaches. Applications to the brain connection and political relationship datasets yield conclusions with sensible interpretations, suggesting the practical utility of the proposed approach.

There are several possible extensions from the work. We have provided accuracy guarantees for parameter estimation in the supervised tensor model. Statistical inference based on tensor decomposition is an important future direction. Measures of uncertainty, such as confidence envelope for space estimation, would be useful. One possible approach would be performing parametric bootstrap (Tibshirani and Efron, 1993) to assess the uncertainty

in the estimation. For example, one can simulate tensors from the fitted low-rank model based on the estimates, and then assess the empirical distribution of the estimates. While being simple, bootstrap approach is often computationally expensive for large-scale data. Another possibility is to leverage recent development in debiased inference with distributional characterization (Chen et al., 2019; Xia, 2019). This approach has led to fruitful results for matrix data analysis. Uncertainly quantification involving tensors are generally harder than matrices, and establishing distribution theory for tensor estimation remains an open problem.

One assumption made by our method is that tensor entries are conditionally independent given the linear predictor Θ. This assumption can be extended by introducing a more general mixed-effect tensor model. For example, in the special case of Gaussian model, we can model the first two moments of data tensor using

$$\mathbb{E}(\mathcal{Y}|\boldsymbol{X}_1,\ldots,\boldsymbol{X}_K) = \mathcal{C} \times \boldsymbol{M}_1 \times \cdots \times \boldsymbol{M}_K,$$
$$\mathrm{Var}(\mathcal{Y}|\boldsymbol{X}_1,\ldots,\boldsymbol{X}_K) = \boldsymbol{\Phi}_1 \otimes \cdots \otimes \boldsymbol{\Phi}_K,$$

where $\boldsymbol{\Phi}_k \in \mathbb{R}^{d_k \times d_k}$ is the unknown covariance matrix on the mode $k \in [K]$. For general exponential family, an additional mean-variance relationship should also be considered. The joint estimation of mean model Θ and variance model $\boldsymbol{\Phi}_k$ will lead to more efficient estimation in the presence of unmeasured confounding effects. However, the introduction of unknown covariance matrices $\boldsymbol{\Phi}_k$ dramatically increases the number of parameters in the problem. Suitable regularization such as graphical lasso or specially-structured covariance assumptions should be considered. The extension of tensor modeling with heterogeneous mixed-effects will be an important future direction.

Although we have presented the supervised tensor decomposition in the context of order-3 data tensors, the framework applies to more general multi-way datasets. One possible extension is the integrative analysis of omics data, in which multiple types of omics measurements (gene expression, DNA methylation, microRNA) are collected in the same set of individuals (Lock et al., 2013; Wang et al., 2019b). In such a case, the observation is a stack of data matrices, which may not be of the same dimension. Other applications

include time-series tensor data with multiple side information. Exploiting the benefits and properties of the supervised tensor decomposition in specialized task will boost scientific discoveries.

SUPPLEMENTARY MATERIALS

Supplementary notes: technical proofs and additional simulation results.

Data and software: Our simulation code, R-package `tensorregress`, and datasets used in the paper are available at `https://CRAN.R-project.org/package=tensorregress` We also provide R scripts for reproducing all numbers in Figures 2-8 and Tables 2-3 in the paper.

References

Abbe, E. (2017). Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531.

Adragni, K. P. and Cook, R. D. (2009). Sufficient dimension reduction and prediction in regression. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4385–4405.

Berthet, Q. and Baldin, N. (2020). Statistical and computational rates in graph logistic regression. *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, 108:2719–2730.

Bi, X., Qu, A., and Shen, X. (2018). Multilayer tensor factorization with applications to recommender systems. *Ann. Statist.*, 46(6B):3308–3333.

Cao, B., Lu, C.-T., Wei, X., Philip, S. Y., and Leow, A. D. (2016). Semi-supervised tensor factorization for brain network analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 17–32. Springer.

Chen, Y., Fan, J., Ma, C., and Yan, Y. (2019). Inference and uncertainty quantification for noisy matrix completion. *Proceedings of the National Academy of Sciences*, 116(46):22931–22937.

Chi, E. C. and Kolda, T. G. (2012). On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272–1299.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278.

Desikan, R. S., Ségonne, F., Fischl, B., Quinn, B. T., Dickerson, B. C., Blacker, D., Buckner, R. L., Dale, A. M., Maguire, R. P., Hyman, B. T., et al. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *Neuroimage*, 31(3):968–980.

Farias, V. F. and Li, A. A. (2019). Learning preferences with side information. *Management Science*, 65(7):3131–3149.

Gabriel, K. R. (1998). Generalised bilinear regression. *Biometrika*, 85(3):689–700.

Gahrooei, M. R., Yan, H., Paynabar, K., and Shi, J. (2020). Multiple tensor-on-tensor regression: An approach for modeling processes with heterogeneous sources of data. *Technometrics*, pages 1–23.

Geddes, L. (2016). Human brain mapped in unprecedented detail. *Nature*.

Hao, B., Zhang, A., and Cheng, G. (2020). Sparse and low-rank tensor estimation via cubic sketchings. *IEEE Transactions on Information Theory*.

Hoff, P. D. (2005). Bilinear mixed-effects models for dyadic data. *Journal of the American Statistical Association*, 100(469):286–295.

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

Hu, C., Rai, P., Chen, C., Harding, M., and Carin, L. (2015). Scalable bayesian non-negative tensor factorization for massive count data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 53–70. Springer.

Ingalhalikar, M., Smith, A., Parker, D., Satterthwaite, T. D., Elliott, M. A., Ruparel, K., Hakonarson, H., Gur, R. E., Gur, R. C., and Verma, R. (2014). Sex differences in the structural connectome of the human brain. *Proceedings of the National Academy of Sciences*, 111(2):823–828.

Ioannidis, V. N., Zamzam, A. S., Giannakis, G. B., and Sidiropoulos, N. D. (2019). Coupled graph and tensor factorization for recommender systems and community detection. *IEEE Transactions on Knowledge and Data Engineering*.

Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM review*, 51(3):455–500.

Lock, E. F. (2018). Tensor-on-tensor regression. *Journal of Computational and Graphical Statistics*, 27(3):638–647.

Lock, E. F., Hoadley, K. A., Marron, J. S., and Nobel, A. B. (2013). Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *The Annals of Applied Statistics*, 7(1):523.

Lock, E. F. and Li, G. (2018). Supervised multiway factorization. *Electronic journal of statistics*, 12(1):1150.

Narita, A., Hayashi, K., Tomioka, R., and Kashima, H. (2012). Tensor factorization using auxiliary information. *Data Mining and Knowledge Discovery*, 25(2):298–324.

Nickel, M., Tresp, V., and Kriegel, H.-P. (2011). A three-way model for collective learning on multi-relational data. In *International Conference on Machine Learning*, volume 11, pages 809–816.

Rabusseau, G. and Kadri, H. (2016). Low-rank regression with tensor responses. In *Advances in Neural Information Processing Systems*, pages 1867–1875.

Rai, P., Wang, Y., Guo, S., Chen, G., Dunson, D., and Carin, L. (2014). Scalable bayesian low-rank decomposition of incomplete multiway tensors. In *International Conference on Machine Learning*, pages 1800–1808.

Raskutti, G., Yuan, M., Chen, H., et al. (2019). Convex regularization for high-dimensional multiresponse tensor regression. *The Annals of Statistics*, 47(3):1554–1584.

Song, Q., Ge, H., Caverlee, J., and Hu, X. (2019). Tensor completion algorithms in big data analytics. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(1):1–48.

Srivastava, M. S., von Rosen, T., and Von Rosen, D. (2008). Models with a kronecker product covariance structure: estimation and testing. *Mathematical Methods of Statistics*, 17(4):357–370.

Tibshirani, R. J. and Efron, B. (1993). An introduction to the bootstrap. *Monographs on statistics and applied probability*, 57:1–436.

Wang, L., Zhang, Z., Dunson, D., et al. (2019a). Common and individual structure of brain networks. *The Annals of Applied Statistics*, 13(1):85–112.

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

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

Wang, M. and Song, Y. (2017). Tensor decompositions via two-mode higher-order SVD (HOSVD). In *Artificial Intelligence and Statistics*, pages 614–622.

Wang, M. and Zeng, Y. (2019). Multiway clustering via tensor block models. *Advances in Neural Information Processing Systems 32 (NeurIPS 2019). arXiv:1906.03807*.

Xia, D. (2019). Confidence region of singular subspaces for low-rank matrix regression. *IEEE Transactions on Information Theory*, 65(11):7437–7459.

Yu, R. and Liu, Y. (2016). Learning from multiway data: Simple and efficient tensor regression. In *International Conference on Machine Learning*, pages 373–381.

Zhang, Z., Descoteaux, M., Zhang, J., Girard, G., Chamberland, M., Dunson, D., Srivastava, A., and Zhu, H. (2018). Mapping population-based structural connectomes. *NeuroImage*, 172:130–145.

Zhao, Q., Caiafa, C. F., Mandic, D. P., Chao, Z. C., Nagasaka, Y., Fujii, N., Zhang, L., and Cichocki, A. (2012). Higher order partial least squares (HOPLS): a generalized multilinear regression method. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1660–1673.

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