Electronic Journal of Statistics
Vol. 16 (2022) 2175–2205
ISSN: 1935-7524
https://doi.org/10.1214/22-EJS1978

Tensor factorization recommender systems with dependency[*]

Jiuchen Zhang, Yubai Yuan and Annie Qu[†]

Department of Statistics,
University of California, Irvine
e-mail: jiuchez@uci.edu*;* yubaiy@uci.edu*;* aqu2@uci.edu

Abstract: Dependency structure in recommender systems has been widely adopted in recent years to improve prediction accuracy. In this paper, we propose an innovative tensor-based recommender system, namely, the Tensor Factorization with Dependency (TFD). The proposed method utilizes shared factors to characterize the dependency between different modes, in addition to pairwise additive tensor factorization to integrate information among multiple modes. One advantage of the proposed method is that it provides flexibility for different dependency structures by incorporating shared latent factors. In addition, the proposed method unifies both binary and ordinal ratings in recommender systems. We achieve scalable computation for scarce tensors with high missing rates. In theory, we show the asymptotic consistency of estimators with various loss functions for both binary and ordinal data. Our numerical studies demonstrate that the proposed method outperforms the existing methods, especially on prediction accuracy.

MSC2020 subject classifications: Primary 62H25, 62G05; secondary 62P20.
Keywords and phrases: Context-aware recommender system, dependency among modes, shared latent factor, parsimonious tensor decomposition.

Received August 2021.

1. Introduction

A recommender system aims to provide recommendations for items users might prefer, which has been widely used in market boosting by targeting different people for different products and giving personalized recommendations. Traditionally, a recommender system can be formulated into a matrix of user-item interactions, such as movie ratings, sales of products, and the frequency of users visiting certain locations.

Nowadays, high-order information beyond user-item pairs is collected, such as time, location, and product features [34, 25, 5], and incorporating such information can increase predictive power. The additional information defining the underlying situation in which a recommendation is provided is referred to as a

[*] This work is supported by NSF grant DMS 1952406.
[†] Corresponding author

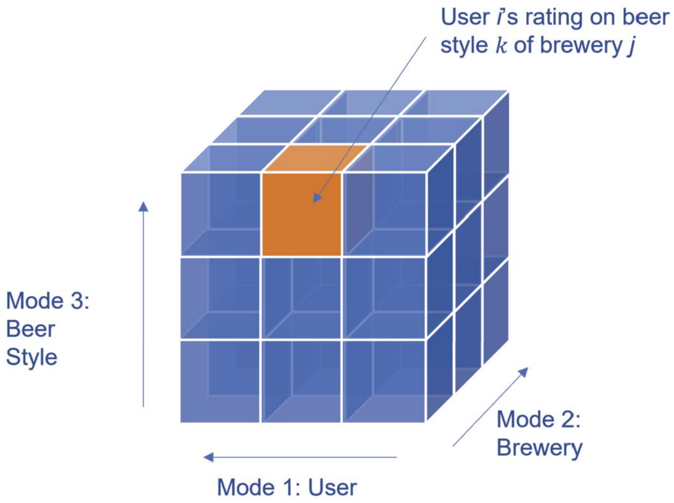


Fig 1. *An illustration of a third-order tensor. This tensor represent the brewery rating under different beer style. The element X_{ijk} represents the rating on beer style k of brewery j given by user i.*

context. Recommender systems with contextual information are called context-aware recommender systems (CARS) [2]. The applications of CARS include personalized marketing strategies for retail stores, product recommendations for different seasonalities, and individualized medical therapies. Ignoring context information may result in a loss of predictive power. For example, the preferences of customers buying new clothes can be significantly different between summer and winter [2].

Tensor representation and decomposition are newly developed tools to handle context information in recommender systems [17, 4, 36, 37]. Tensors are generalizations of matrices for higher-order relational data and provide useful high-order data representation formats [4]. In recommender systems, a tensor is effective and efficient for incorporating or utilizing contextual information for CARS. Specifically, in addition to user and item, CARS can be formulated as a tensor $\boldsymbol{\mathcal{X}}$ illustrated in Figure 1.

One advantage to utilizing tensor representation is capturing complex user-item-context interactions through tensor decompositions such as CANDECOMP/PARAFAC (CP) decomposition [16] and Tucker decomposition [33]. Tensor decompositions can be viewed as higher-order latent factor modeling, where a set of latent factor vectors is used to encode users, items and contexts, and capture the relations in a latent space.

The existence of dependency structures is quite common in recommender systems [4, 36]. A dependency may be induced from the tensor entries which are not necessarily independent, as users may be influenced by other users who share similar preferences. Existing methods mainly focus on a specific dependency structure. For example, the recommendation engine of multi-layers [4] and the double core tensor factorization model [32] consider the heterogeneity across subgroup structures in each mode. For other example when time is regarded as contextual information, the recommender system often exhibits strong temporal patterns, which also introduces a dependency structure among user and item interactions over time. Accordingly, the temporal dynamics factor models [18] and the Bayesian probabilistic tensor factorization method [36] aim to solve this issue. However, in many real-world applications, dependence can also exist among users, items, and other contextual variables. For example, in beer recommendation, the beer style is entirely controlled by the brewery, and users' preference of brewery highly depends on the beer styles the brewery produces. In tag recommendation, tags are highly associated with items, as users select similar tags for a specific item. Therefore, between-modes dependence occurs frequently, and developing a tensor-based method to capture dependency among user, item, and context is very necessary in practice.

A key feature of recommender systems is that the utilities of matrix or tensor are commonly in a binary or ordinal form to represent yes or no, or ratings of products. Studies considering high-order binary or rating data include link function approaches [20, 24, 35, 12, 19], or Boolean tensor decomposition, through a binary tensor decomposition into a series of binary factors without a population assumption [22, 11, 28, 13, 10]. However, for a model-free approach such as the Boolean tensor decomposition, no distribution assumption is specified for the tensor entries. In contrast to the Boolean tensor decomposition, methods utilizing link function take advantage of a population model assumption [35, 28], which can distinguish the algorithm error from statistical error [35].

In this paper, we propose a new tensor factorization with dependency (TFD) model. The main novelty of our method is to incorporate the dependency structures among modes by utilizing shared latent factors across pairwise interactions. Furthermore, the proposed method unifies the binary and ordinal ratings in recommender systems and is scalable in computing through imposing the scarcity of the observed utility tensor. In addition, we establish the asymptotic consistency and convergence rate of the proposed estimator for both binary and ordinal cases.

The proposed tensor factorization method has the following advantages. First, our method is able to incorporate a wide range of dependency structures and requires no additional assumptions on latent factors or variables. In contrast, existing methods generally target a specific dependency structure such as the Markov chain on temporal data or the multivariate copula [36, 15]. In addition, by introducing the shared latent factor, the proposed model can accommodate a weak dependency among modes, therefore improving on the prediction accuracy. Here, weak dependency indicates that there are only a few latent factors shared between user-item interaction and item-context interaction. This also implies that the latent factors corresponding to user-item interaction would not affect the item-context interaction.

Second, the proposed model is effective for high-order CARS with a high observation missing rate. By utilizing a summation of pairwise interactions, we can capture most of the information of the user-item interaction and dependency with other interactions via a parsimonious model. Our method is less stringent on sample size requirements for recovering the tensor compared to existing tensor decomposition methods under the same framework. Pairwise interaction tensor factorization (PITF) has been applied in tag recommendations and movie recommendations [27, 8]. However, the PITF is not able to model the dependency structure among interactions, where the latent factors modeling interactions across different modes are assumed to be independent with each other.

In addition, we propose a mode-wise coordinate descent (MCD) algorithm to reduce the computational cost via utilizing the sparsity of the observed tensor. In both of our simulations and real data applications, the proposed method outperforms the competing methods on prediction accuracy.

The rest of the paper is organized as follows. Section 2 provides the background of CARS and tensor factorization. Sections 3 and 4 introduce the proposed method and the optimization algorithm. Theoretical properties are provided in Section 5. Section 6 presents simulation studies to investigate the performance of our method. In Section 7, we apply the proposed method to two real data applications: User-Location-Activity data and Beer Review data. Section 8 concludes with a discussion.

2. Background and notation

In this section, we introduce the notation, the background of tensor representation and the related tensor decompositions.

A tensor is a multidimensional array, a generalization of vector and matrix, where the order of a tensor is the number of dimensions of the array, and is also known as the mode. For example, the vector and matrix can be viewed as a 1-order and 2-order tensor, respectively. In this paper, vectors are denoted by boldface letters, e.g., $\boldsymbol{a}$, matrices are denoted by boldface capital letters, e.g., $\boldsymbol{A}$, tensors are denoted by boldface Euler script letters, e.g., $\boldsymbol{\mathcal{X}}$.

Traditional recommender systems can be formulated by a utility matrix $\boldsymbol{X} \in \mathbb{R}^{n_1 \times n_2}$ representing user-item interactions, such as ratings or purchase status, where the element X_{ij} indicates the interaction between user i and item j. For example, in brewery recommendations, the element X_{ij} represents the rating of brewery j given by user i. However, more sophisticated recommender systems also collect other potential useful contextual information, such as time of year or beer style. Instead of using a utility matrix $\boldsymbol{X}$ to represent user-item interactions, a tensor structure $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is widely adopted to incorporate additional contextual information, as illustrated in Figure 1. The element X_{ijk} of a third-order tensor indicates the interaction among user i, item j, and context k. For instance, in previous brewery recommendations, the element X_{ijk} represents the rating on beer style k of brewery j given by user i. By fixing the last index,

$$\boldsymbol{X}_k = \begin{pmatrix} X_{11k} & \dots & X_{1n_2k} \\ & \dots & \\ X_{n_11k} & \dots & X_{n_1n_2k} \end{pmatrix}_{n_1 \times n_2}$$

represents the utility matrix under context k.

Tensor representation and decomposition are newly developed tools to deal with context-aware recommender systems (CARS) [2, 4, 6], due to its capability of reducing model complexity through utilizing a low-rank structure. One commonly used tensor decomposition is Canonical Polyadic Decomposition (CPD) [16], which factorizes a tensor into a sum of r rank-one tensors. For example, the CPD for a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is:

$$\boldsymbol{\mathcal{X}} = \sum_{r=1}^{R} \boldsymbol{a_r^{(1)}} \circ \boldsymbol{a_r^{(2)}} \circ \boldsymbol{a_r^{(3)}} + \boldsymbol{\mathcal{E}}, \tag{1}$$

where $\circ$ represents the vector outer product, $\boldsymbol{\mathcal{E}} = \{\varepsilon_{ijk}\}$ is a noise tensor with independent and identical distributed (i.i.d.) entries, and the latent factor $\boldsymbol{a_r^{(k)}} \in \mathbb{R}^{n_k \times 1}$ $(r = 1, ..., R; k = 1, 2, 3)$ corresponds to the k-th mode. The rank of a tensor is defined as $rank(\boldsymbol{\mathcal{X}}) = \min\{R : \boldsymbol{\mathcal{X}} \approx \sum_{r=1}^{R} \boldsymbol{a_r^{(1)}} \circ \boldsymbol{a_r^{(2)}} \circ \boldsymbol{a_r^{(3)}}\}$. We denote $\boldsymbol{A^{(k)}} = (\boldsymbol{a_1^{(k)}}, ..., \boldsymbol{a_r^{(m)}})_{n_m \times R}, m = 1, 2, 3$ as factor matrices where each row of $\boldsymbol{A^{(1)}}$ represents the r-dimentional latent factor for each user, and each row of $\boldsymbol{A^{(2)}}$ and $\boldsymbol{A^{(3)}}$ is a latent factor for an item and a contextual variable, respectively.

Pairwise interaction tensor factorization (PITF) has gained considerable attention due to its simplicity and high quality in prediction, and has been applied in tag recommendation, sequential recommender systems and movie recommendation [26, 27, 8]. Here, tag recommendation refers to a recommender system that suggests useful tags on a specific item, while a sequential recommender system predicts what a user would purchase based on their previous purchases experience. PITF models multi-way interactions through a summation of series of two-way interactions. For example, to model a user's ratings for brewery over different beer styles, the PITF model assumes that each rating is determined by three two-way interactions: the user's inherent preference on a certain brewery, the brewery's specific recipes for different beer styles, and the user's preference over different beer styles. That is, each entry x_{ijk} of tensor $\boldsymbol{\mathcal{X}}$ is modeled by:

$$x_{ijk} = \sum_{r=1}^{R} a_{ir}^{(b)} b_{jr}^{(a)} + \sum_{r=1}^{R} b_{jr}^{(c)} c_{kr}^{(b)} + \sum_{r=1}^{R} a_{ir}^{(c)} c_{kr}^{(a)} + \varepsilon_{ijk}. \tag{2}$$

For each mode, two factors are modeled to count for interactions with other two modes. For example, latent factor $a_{ir}^{(b)}$ characterizes the i-th user's interaction with items, while $a_{ir}^{(c)}$ represents i-th user's interactions with contexts. One advantage of the PITF is that it only requires $O\{nr\log^2(n)\}$ observations to fully recover the tensor [8], while CP tensor factorization requires $O\{r^5 n^{3/2} \log^4(n)\}$, where n is the maximum size of mode dimensions and r is the dimension of latent vectors [14]. The explicit form of tensor structure in (2) ensures that the parameter space of the latent factor is smaller than the regular CP decomposition, and therefore the prediction accuracy can be improved [27]. However, the PITF does not consider the dependency among pairwise interactions of user, item and context since two latent factors are used for each mode. Therefore, the user-item interaction is independent from the user-context interaction. However, in practice, the user-item interaction is highly associated with item-context interaction. For example, in brewery recommendation, the user's preference for a certain brewery also depends on the main beer style of the brewery.

Another well-known tensor factorization method is the high-order singular value decomposition, also referred as Tucker decomposition (TD) [33], which decomposes a tensor into a core tensor corresponding to each mode. In contrast to CP decomposition, the rank of tensor in the Tucker decomposition is not as well defined as in the matrix rank for singular value matrix decomposition. However, in the recommender systems application, the tensor rank is essential in achieving dimension reduction of original tensor data and providing interpretability on latent factors [4]. Another drawback of the TD is that the theoretical computational complexity of TD is $O(R^3)$, while the CP decomposition is only $O(R)$. Thus, CP decompostiion is more popular in recommender systems [22, 4, 12]. More details on the tensor can be found in [17, 4].

3. General methodology

3.1. Tensor factorization with dependency

In this section, we introduce the proposed tensor dependency modeling and the corresponding factorization method. The dependency between user-item relation and item-context relation is very common in practice. Therefore, we model a context-aware rating tensor via a series of two-way interactions among latent factors corresponding to user, item and context, respectively. Specifically, we propose

$$x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}, \tag{3}$$

where a_{ir} is the r-th latent factor for user i, and b_{jr} and c_{kr} are the corresponding latent factors for item j and context k.

The underlying dependency among different modes in the tensor is incorporated via sharing mode-specific latent factors across the two-way interactions, e.g., $\{b_{jr}\}$ are shared in both interaction terms in (3). The proposed method differs from existing tensor modeling of recommender systems such as the PITF [27], in that the proposed model imposes a latent-factor-sharing structure, which incorporates a multi-way interaction among user, item, and context through a summation of two-way interactions. In model (3), user i's preference under context k is implicitly encoded into the dependency between $a_{ir}b_{jr}$ and $b_{jr}c_{kr}$ via the shared latent factor b_{jr} of an item, which serves as a "bridge" between user i and context k. Sharing latent factors can accommodate different types of dependency. For example, we can incorporate dependency among users, items, or contexts from the same group, which has been studied before [4, 32].

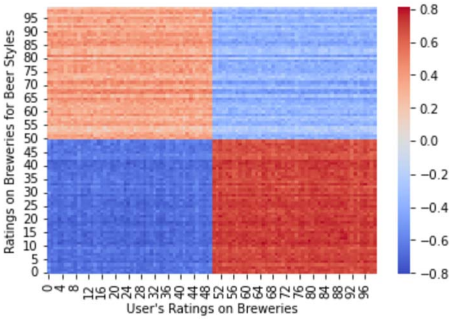


FIG 2. *Correlations between each row of $\boldsymbol{X}_{ij\cdot}$ and each column of $\boldsymbol{X}_{\cdot jk}$, which represent the correlation between user-brewery interaction and brewery-style interaction.*

For example, consider a third-order tensor $\boldsymbol{X} = \{x_{ijk}\} \in \mathbb{R}^{100\times100\times100}$ representing the user-brewery-style relations. We generate two groups of latent factor for each mode, where the first 50 users, breweries, and styles have latent factor $\boldsymbol{a}_1$, $\boldsymbol{b}_1$, and $\boldsymbol{c}_1$, respectively, while the last 50 users, breweries, and styles have latent factor $\boldsymbol{a}_2$, $\boldsymbol{b}_2$, and $\boldsymbol{c}_2$, respectively. Latent factors $\boldsymbol{a}_i$, $\boldsymbol{b}_i$, $\boldsymbol{c}_i \in \mathbb{R}^R$ for $i = 1, 2$ are generated randomly from $N(\boldsymbol{0}, \boldsymbol{I}_R)$, where R is the rank of latent space which is set as 3. Let $\boldsymbol{X}_{ij\cdot} = \{x_{ij\cdot}\}_{100\times100}$ and $\boldsymbol{X}_{\cdot jk} = \{x_{\cdot jk}\}_{100\times100}$, where $x_{ij\cdot} = \frac{1}{100}\sum_{k=1}^{100} x_{ijk}$ and $x_{\cdot jk} = \frac{1}{100}\sum_{i=1}^{100} x_{ijk}$. By averaging ratings over styles and users, each row of $\boldsymbol{X}_{ij\cdot}$ and each column of $\boldsymbol{X}_{\cdot jk}$ represent ratings on breweries by each user, and ratings on breweries for each style, respectively. Figure 2 shows the correlation between each row of $\boldsymbol{X}_{ij\cdot}$ and each column of $\boldsymbol{X}_{\cdot jk}$, representing the correlation between user-brewery interaction and brewery-style interaction. The top-left red block shows that the first group of users' ratings on breweries are positively correlated with the first group of styles' ratings for breweries, while the bottom left blue block shows that the second group of users' ratings on breweries are negatively correlated with the first group of styles' ratings for breweries. This type of dependency may be caused by users' preferences for different tastes of beer styles. If the first group of users like the sweet taste, and the first group of styles' high ratings are due to the sweet flavor of the beer, then the users' ratings on breweries are positively correlated with the styles' ratings for breweries. On the other hand, if the second group of users prefer the strong and malty taste, they may tend to give low ratings to breweries with high ratings for the first group of styles, which leads to negative correlations at the bottom left. The above simulation shows how the proposed method is able to capture the dependency between user-item interaction and item-context interaction under the beer review context.

For our the real data, Figure 3 shows the dependency between user-location interaction and location-activity interaction from user-location-activity data. There are only positive correlations or no correlation under this data. This can be explained in that people tend to go to places where there are activities they prefer.

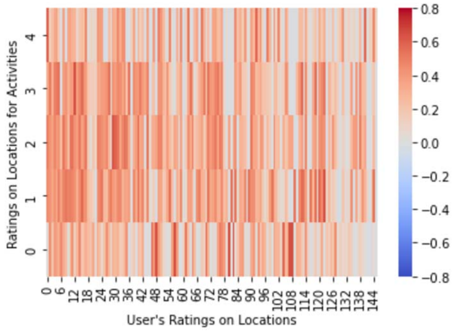


Fig 3. *Correlations between each row of $\boldsymbol{X}_{ij\cdot}$ and each column of $\boldsymbol{X}_{\cdot jk}$, representing the correlations between user-location interaction and location-activity interaction.*

Although one might capture the dependence of interaction through increasing the rank in tensor decomposition, such as the CP and the Tucker, this could be infeasible as non-identifiability, unstable estimation and higher computational cost could also occur if the rank in tensor increases. By specifying a more explicit structure of the tensor as in (3), we can drop the number of observations in order to recover the tensor, especially when the tensor size is large. In addition, a three-way interaction could introduce spurious correlation and noise, especially if there is no correlation between user and context variable. For example, in tag recommendation, a tag is used to describe the characteristics of items, which are not likely to be correlated with users.

In general, the user-context interaction can be ignored in a recommender system; that is, the interaction between user $\{a_{ir}\}$ and context $\{c_{kr}\}$ is less influential when a user's primary interest is the item ranking [27]. This can be explained from the Bayesian Personalized Ranking (BPR) perspective, where optimization is achieved through maximizing the likelihood of item ranking conditioning on user-context pairs (i, k). Specifically, let $j_1 >_{i,k} j_2$ denote that j_1 ranks higher than j_2, given a user-context pair (i, k), and matrix $\boldsymbol{R}^{(i,k)} \in \{0,1\}^{n_2 \times n_2}$ denote the rankings of all items for user i under context k, where each entry is $R^{(i,k)}_{j_1 j_2} = I(j_1 >_{i,k} j_2)$, and $I(\cdot)$ is the indicator function. The problem of finding the best ranking of item $\boldsymbol{R}^{(i,k)}$ given the observed data can be formulated as maximizing the following likelihood of independent observations:

$$\underset{\Theta}{\operatorname{argmax}} \prod_{(i,k)\in I\times K} P(\boldsymbol{R}^{(i,k)}|\boldsymbol{\Theta}), \tag{4}$$

where I, K are sets of users and contexts, respectively, and $\boldsymbol{\Theta}$ is a model parameter vector. Assume that $j_1 >_{i,k} j_2$ follows a Bernoulli distribution, we have:

$$\begin{aligned}\prod_{(i,k)\in I\times K} P(\boldsymbol{R}^{(i,k)}|\boldsymbol{\Theta}) &= \prod_{(i,k,j_1,j_2)\in I\times K\times J^2} \Big[P(j_1 >_{i,k} j_2|\boldsymbol{\Theta})^{I\{(i,k,j_1,j_2)\in D\}} \\ &\qquad \times \{1-P(j_1 >_{i,k} j_2|\boldsymbol{\Theta})\}^{1-I\{(i,k,j_1,j_2)\in D\}}\Big] \qquad (5) \\ &= \prod_{(i,k,j_1,j_2)\in D} P(j_1 >_{i,k} j_2|\boldsymbol{\Theta}),\end{aligned}$$

where $D = \{(i,k,j_1,j_2)\}$ is a set of indices such that $j_1 >_{i,k} j_2$ is observed.

Suppose that we have a model predicting a scoring function $\hat{y} : I\times J\times K \to \mathbb{R}$. We derive an estimator for $p(j_1 >_{i,k} j_2|\boldsymbol{\Theta})$ by plugging in $\hat{y}$:

$$p(j_1 >_{i,k} j_2|\boldsymbol{\Theta}) \coloneqq f(\hat{y}_{i,j_1,k} - \hat{y}_{i,j_2,k}), \qquad (6)$$

where $f(x) = \frac{\exp(x)}{1+\exp(x)}$ is the logistic function. Notice that $\hat{y}$ is is an arbitrary real-valued function obtained from a model parameter vector $\boldsymbol{\Theta}$, e.g., the proposed factorization model.

The Bayesian Personalized Ranking (BPR) perspective shows that if the primary interest is item ranking, then, when estimating the scoring $\hat{y}$, we only care about the difference between two items' scoring, $\hat{y}_{i,j_1,k} - \hat{y}_{i,j_2,k}$. Although ignoring the user-context interaction may lead to inaccurate scoring predictions, the ranking of items based on the estimated scoring will not be affected. For example, suppose $\hat{y}_{i,j,k} = \sum_{r=1}^{R} a_{ir}b_{jr} + \sum_{r=1}^{R} b_{jr}c_{kr} + \sum_{r=1}^{R} a_{ir}c_{kr}$, then $\hat{y}_{i,j_1,k} - \hat{y}_{i,j_2,k} = (\sum_{r=1}^{R} a_{ir}b_{j_1r} + \sum_{r=1}^{R} b_{j_1r}c_{kr}) - (\sum_{r=1}^{R} a_{ir}b_{j_2r} + \sum_{r=1}^{R} b_{j_2r}c_{kr})$, where the third term for the user-context interactions is canceled out. In addition, discarding user-context interaction does not lead to poor rating predictions, as shown in our simulations. This may be because the value-based loss functions for binary ratings and ordinal ratings are similar to the rank-based loss function.

Another advantage of the proposed model (3) is its flexibility in modeling different dependency structure among modes. Existing methods for characterizing dependency structure require a specific dependence structure, such as the Markov chain on temporal data or the multivariate copula [36, 15]. In the Markov chain model, the probability of each event depends only on the state of the previous event, while in the multivariate copula model, data are assumed to follow a specific multivariate distribution. These model assumptions could be violated. In contrast, the proposed model utilizes latent factor modeling and is able to approximate complex unstructured dependency between tensor modes.

In this paper, we consider the most common cases where x_{ijk} is binary or ordinal. We first introduce the proposed model for binary rating tensor as follows:

$$logit\{P(x_{ijk}=1)\} = \sum_{r=1}^{R} a_{ir}b_{jr} + \sum_{r=1}^{R} b_{jr}c_{kr}, \qquad (7)$$

where $P(x_{ijk} = 1)$ denotes the probability of x_{ijk} being 1, and $logit(p) = \log(\frac{p}{1-p})$.

To estimate the latent factor matrices $\boldsymbol{A} = \{a_{ir}\}$, $\boldsymbol{B} = \{b_{jr}\}$ and $\boldsymbol{C} = \{c_{kr}\}$, we minimize the following square loss on x_{ijk} with L_2 penalty [5, 3, 29], where the L_2 penalty controls the model complexity to avoid over-fitting and scale indeterminacy [1]:

$$\begin{aligned} L(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}|\boldsymbol{\mathcal{X}}) = \sum_{(i,j,k)\in\Omega} & \left(x_{ijk} - f(\sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}) \right)^2 \\ & + \lambda(||\boldsymbol{A}||_F^2 + ||\boldsymbol{B}||_F^2 + ||\boldsymbol{C}||_F^2), \end{aligned} \tag{8}$$

where Ω denotes the set of indices corresponding to observed ratings, $f(x) = \frac{\exp(x)}{1+\exp(x)}$ is the logistic function, λ is the penalty tuning parameter, and $||\cdot||_F$ denotes the Frobenius norm.

Note that minimizing the square loss is equivalent to maximizing the likelihood of observed $\{x_{ijk}\}$ if the data are fully observed. In recommender systems, the first square loss term and the remaining penalty term in (8) could be under different domain spaces, therefore (8) is more computationally efficient in practice.

3.2. Ordinal rating tensor

In the following, we extend the proposed tensor modeling to the case of ordinal rating. Given an ordinal ratings tensor $\boldsymbol{\mathcal{X}} = \{x_{ijk}\}$ with L levels, we propose a proportional-odds-based modeling on rating x_{ijk} as follows:

$$\begin{aligned} & logit\{P(x_{ijk} \leq l\} = \alpha_l - \theta_{ijk}, \quad l = 1, ..., L-1, \\ & \theta_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}, \end{aligned} \tag{9}$$

where $\alpha_1 \leq \alpha_2 \leq ... \leq \alpha_{L-1}$.

The model (9) has an equivalent representation as follows:

$$x_{ijk} = \begin{cases} 1, & x_{ijk}^* \in (-\infty, \alpha_1]\,, \\ 2, & x_{ijk}^* \in (\alpha_1, \alpha_2]\,, \\ \vdots & \vdots \\ L, & x_{ijk}^* \in (\alpha_{L-1}, \infty)\,, \end{cases} \tag{10}$$

where $\{x_{ijk}^*\}$ is a noisy version of $\{\theta_{ijk}\}$:

$$x_{ijk}^* = \theta_{ijk} + \epsilon_{ijk},$$

and $\{\epsilon_{ijk}\}$ are i.i.d. noises with a cumulative distribution function $\mathbb{P}(\epsilon) = \frac{1}{1+e^{-\epsilon}}$ [19]. According to (10), we assume that the ordinal outcome x_{ijk} can be formulated as a categorized version of a latent continuous variable x_{ijk}^*. For example, in a 5-point ordinal ranking, we can treat it as a discrete surrogate of the continuous variable indicating the degree of likeness.

In the following, we propose a proportional odds model where the distance between pairs of levels are the same for all entries, and therefore (9) implies L-1 parallel classification hyperplanes for the L-level ratings as follows:

$$\log \frac{\mathbb{P}\left(x_{ijk} \leq l\right)}{\mathbb{P}\left(x_{ijk} > l\right)} - \log \frac{\mathbb{P}\left(x_{ijk} \leq l-1\right)}{\mathbb{P}\left(x_{ijk} > l-1\right)} = \alpha_l - \alpha_{l-1}, \quad l \in \{1, \dots, L-1\}. \tag{11}$$

Note that the right hand side of (11) does not involve x_{ijk}, that is the log odds are the same across observations. If there is evidence showing that the log odds are heteroscedastic for post (i,j,k), then a multinomial logistic model is more desirable.

Let Ω represent the set of indices corresponding to the observed ratings. To simplify the notation, let $\alpha_0 = -\infty$, and $\alpha_L = \infty$. We estimate $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\alpha} = \{\alpha_l\}, l = 1, ..., L-1$ via minimizing the following penalized negative log likelihood,

$$\begin{aligned} & L(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\alpha} | \boldsymbol{\mathcal{X}}) \\ & = - \sum_{(i,j,k) \in \Omega} \sum_{l=1}^{L} I(X_{ijk} = l) \log \left\{ f(\alpha_l - \theta_{ijk}) - f(\alpha_{l-1} - \theta_{ijk}) \right\} \\ & + \lambda(||\boldsymbol{A}||_F^2 + ||\boldsymbol{B}||_F^2 + ||\boldsymbol{C}||_F^2), \end{aligned} \tag{12}$$

where $\theta_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}$.

Notice that we do not penalize $\boldsymbol{\alpha}$ as they are the cut-off points for each level of rating. When $L = 2$, the above proportional odds loss function (12) degenerates to a penalized logistic loss function for the binary ratings.

4. Computation

In this section, we propose a scalable algorithm to estimate the latent factors $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ and cutoff parameter $\boldsymbol{\alpha}$, and provide implementation strategies to improve the stability and efficiency of the optimization.

The proposed algorithm is based on a mode-wise gradient descent, and the gradient of the latent factors can be derived from loss functions (8) and (12). Let $\Omega := \{(i,j,k) : x_{ijk} \text{ is observed}\}$ be the set of indices corresponding to the observed ratings, and define $\Omega_{i\cdot\cdot} := \{(j,k) : (i,j,k) \in \Omega\}$ as a subset of Ω collecting entries whose first index is i. We similarly define $\Omega_{\cdot j \cdot} := \{(i,k) : (i,j,k) \in \Omega\}$ and $\Omega_{\cdot\cdot k} := \{(i,j) : (i,j,k) \in \Omega\}$ corresponding to the second and third modes. For the binary rating case, the gradients of the proposed loss function (8) with respect to the latent factors $\boldsymbol{A} = \{a_{ir}\}, \boldsymbol{B} = \{b_{ir}\}, \boldsymbol{C} = \{c_{ir}\}$ are

$$\begin{aligned}
&\frac{\partial L}{\partial a_{ir}} = \sum_{(j,k)\in\Omega_{i\cdot\cdot}} y_{ijk} b_{jk} + 2\lambda a_{ir}, \quad \frac{\partial L}{\partial b_{jr}} = \sum_{(i,k)\in\Omega_{\cdot j\cdot}} y_{ijk}\left(a_{ir} + c_{kr}\right) + 2\lambda b_{jr}, \\
&\frac{\partial L}{\partial c_{kr}} = \sum_{(i,j)\in\Omega_{\cdot\cdot k}} y_{ijk} b_{jk} + 2\lambda c_{kr},
\end{aligned} \tag{13}$$

where

$$y_{ijk} = 2\left(f(\widehat{x}_{ijk}) - x_{ijk}\right) f(\widehat{x}_{ijk})(1 - f(\widehat{x}_{ijk})), and \quad \widehat{x}_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}. \tag{14}$$

For the ordinal rating case, the gradients of the penalized negative log likelihood (12) with respect to the latent factors $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$, and cutoff $\boldsymbol{\alpha}$ are

$$\begin{aligned}
&\frac{\partial L}{\partial a_{ir}} = -\sum_{(j,k)\in\Omega_{i\cdot\cdot}} y_{ijk} b_{jk} + 2\lambda a_{ir}, \frac{\partial L}{\partial b_{jr}} = -\sum_{(i,k)\in\Omega_{\cdot j\cdot}} y_{ijk}\left(a_{ir} + c_{kr}\right) + 2\lambda b_{jr}, \\
&\frac{\partial L}{\partial c_{kr}} = -\sum_{(i,j)\in\Omega_{\cdot\cdot k}} y_{ijk} b_{jk} + 2\lambda c_{kr}, \quad \frac{\partial L}{\partial \alpha_l} = -\sum_{(i,j,k)\in\Omega} y_{ijk},
\end{aligned} \tag{15}$$

where

$$\begin{aligned}
y_{ijk} = \sum_{l=1}^{L} \Big\{ & I(X_{ijk} = l) \times \\
& \frac{f(\alpha_l - \widehat{x}_{ijk})\left\{1 - f(\alpha_l - \widehat{x}_{ijk})\right\} - f(\alpha_{l-1} - \widehat{x}_{ijk})\left\{1 - f(\alpha_{l-1} - \widehat{x}_{ijk})\right\}}{f(\alpha_l - \widehat{x}_{ijk}) - f(\alpha_{l-1} - \widehat{x}_{ijk})} \Big\}.
\end{aligned} \tag{16}$$

Instead of updating only one entry of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\alpha}$ at each iteration, we can utilize the matrix and tensor operations to jointly update each mode of the rating tensor for each iteration using parallel computing, thus reducing computation time [12].

Due to the non-convex nature of the proposed objective functions (8) and (12), the gradient-descent based optimization can only guarantee convergence to a stationary point, and is sensitive to the initial value. However, the proposed algorithm is not sensitive to the initial value since our method has a low model complexity compared to the standard tensor decomposition methods. This low model complexity has an advantage of ensuring the convergence of the proposed algorithm in that it is more stable and less likely to be trapped by the saddle points [9]. We set the initial values for the latent factors randomly as

$$\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i \sim N\left(0, I_R\right),$$

where I_R is a $R \times R$ identity matrix and

$$\boldsymbol{\alpha} \sim Uniform\,(0, 1)\,,$$

for a cutoff in the ordinal rating case. After initializing $\boldsymbol{\alpha}$, we sort the entries of $\boldsymbol{\alpha}$ in an ascending order. We summarize the proposed optimization in the following Algorithm 1.

Algorithm 1 Mode-wise Coordinate Descent

```
1. (Initialization) Set stopping error $\epsilon$, rank $R$, tuning parameter $\lambda$ and initial values:
     (i) Binary rating case: $\boldsymbol{A}^{(0)}, \boldsymbol{B}^{(0)}, \boldsymbol{C}^{(0)}$.
     (ii) Ordinal rating case: $\boldsymbol{A}^{(0)}, \boldsymbol{B}^{(0)}, \boldsymbol{C}^{(0)}, \boldsymbol{\alpha}^{(0)}$.
2. (Latent factor update) At the $s$-th iteration ($s \geq 1$):
     (i) Binary rating case: update $\boldsymbol{A}^{(s)}, \boldsymbol{B}^{(s)}, \boldsymbol{C}^{(s)}$ sequentially via (13).
3. Ordinal rating case: update $\boldsymbol{A}^{(s)}, \boldsymbol{B}^{(s)}, \boldsymbol{C}^{(s)}, \boldsymbol{\alpha}^{(s+1)}$ sequentially via (15).
4. Stop if $\frac{|L^{(s+1)} - L^{(s)}|}{L^{(s)}} < \epsilon$.
```

In addition, we tune rank r and λ via minimizing the mean square error (MSE) on a validation set, where the MSE on a set Ω is defined as $\frac{1}{|\Omega|}\sum_{\Omega}(x_{ijk} - \hat{x}_{ijk})^2$.

High computational complexity is a common issue for tensor based modeling, especially when the dimension of the tensor is large. One difficulty arises from the trade-off between memory cost and computational efficiency. In additional to the rating tensor $\boldsymbol{\mathcal{X}}$, we also need to store $\boldsymbol{\mathcal{Y}} = \{y_{ijk}\}$ in (14) and (16) in computing gradients. When $\boldsymbol{\mathcal{X}}$ is sparse, we can store $\boldsymbol{\mathcal{X}}$ effciently by storing only its nonzero values and the corresponding indices. However, even when $\boldsymbol{\mathcal{X}}$ is sparse, $\boldsymbol{\mathcal{Y}}$ can still be dense and might cause memory issues.

The proposed algorithm achieves a higher computational efficiency when the dimension of each mode is large and $\boldsymbol{\mathcal{X}}$ is scarce [12]. In practice, $\boldsymbol{\mathcal{X}}$ is likely scarce as the vast majority of its entries are missing, due to the fact that tensor $\boldsymbol{\mathcal{Y}}$ becomes sparse when $\boldsymbol{\mathcal{X}}$ is scarce since each missing element in $\boldsymbol{\mathcal{X}}$ corresponds to a zero entry of $\boldsymbol{\mathcal{Y}}$. Our strategy is to store the nonzero entries of $\boldsymbol{\mathcal{Y}}$ only to save memory and therefore make our algorithm scalable.

5. Theory

In this section, we develop the theoretical properties for the proposed method. Specifically, we provide the consistency and convergence rate for the proposed estimator under the cases of binary and ordinal rating. In addition, we extend the convergence property of the proposed estimator under L_2 loss to a general loss function, given additional smoothness conditions.

We first provide the asymptotic properties for the proposed method for the binary rating case. Since our primary goal is the prediction of ratings, we focus on the convergence property of the rating values instead of the latent factor recovery. Suppose the entries $\boldsymbol{Y} = \{y_{i_1 i_2 i_3}\} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ are i.i.d. following a Bernoulli distribution with probability $\boldsymbol{\Theta} = \{\theta_{i_1 i_2 i_3}\}$, where $\theta_{i_1 i_2 i_3} =$

$f(\sum_{r=1}^{R} a_{i_1 r} b_{i_2 r} + \sum_{r=1}^{R} b_{i_2 r} c_{i_3 r})$. For element $y_{i_1 i_2 i_3}$ of $\boldsymbol{Y}$, we define the L_2-loss function as

$$l\left(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}\right) = \left(y_{i_1 i_2 i_3} - \theta_{i_1 i_2 i_3}\right)^2 . \tag{17}$$

Let $|\Omega|$ be the number of observed ratings and $J(\boldsymbol{\Theta})$ be a non-negative penalty function; for example, we have $J(\boldsymbol{\Theta}) = ||\boldsymbol{A}||_2^2 + ||\boldsymbol{B}||_2^2 + ||\boldsymbol{C}||_2^2$ for the L_2-penalty on the tensor factors. Then the overall object function is

$$L(\boldsymbol{\Theta} \mid \mathbf{Y}) = \sum_{(i_1, i_2, i_3) \in \Omega} l\left(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}\right) + \lambda_{|\Omega|} J(\boldsymbol{\Theta}), \tag{18}$$

where $\lambda_{|\Omega|}$ is a tuning parameter for the penalization. To establish the convergence rate, we introduce the following assumption:

Assumption 1. $\max\{||\boldsymbol{A}||_\infty, ||\boldsymbol{B}||_\infty, ||\boldsymbol{C}||_\infty\} \leq c_0$, *for a constant* c_0.

Assumption 1 assumes that the latent factors are bounded, since, in practice, the underlying probabilities of binary utilities are finite. We define the parameter space as

$$\mathcal{S}(k) = \{\boldsymbol{\Theta} : ||\boldsymbol{\Theta}||_\infty \leq c, J(\boldsymbol{\Theta}) \leq k^2\},$$

where k is a positive constant.

We assume that $k \sim O(\sqrt{\gamma})$, where $\gamma = \sum_{i=1}^{3} n_i R$ is the total number of parameters. We introduce $\mathcal{S}(k)$ to ensure that the proposed estimator can reach the best possible convergence rate. Let $\mathcal{S} \subseteq \mathbb{R}^{(n_1+n_2+n_3)R}$ be the true underlying parameter space. The estimator $\hat{\boldsymbol{\Theta}}_{|\Omega|}$ defined on $\mathcal{S}$ may not achieve the best possible convergence rate since the size of $\mathcal{S}$ is too large when n_i goes to infinity [31]. In contrast, $\mathcal{S}(k)$ imposes constraints on the parameters. When k increases, the constraint $J(\boldsymbol{\Theta}) \leq k^2$ becomes less stringent, and the parameter space $\mathcal{S}(k)$ converges to $\mathcal{S}$ when n_i goes to infinity [5, 30].

Let $\boldsymbol{\Theta}_0$ be the true probability of the binary ratings and $\widehat{\boldsymbol{\Theta}} = \arg\min_{\boldsymbol{\Theta} \in \mathcal{S}} L(\boldsymbol{\Theta} \mid \mathbf{Y})$. We denote $\hat{\boldsymbol{\Theta}}_{|\Omega|}$ as the sample estimator of $\boldsymbol{\Theta}_0$ satisfying:

$$L(\hat{\boldsymbol{\Theta}}_{|\Omega|} | \boldsymbol{Y}) \leq \inf_{\Theta \in \mathcal{S}(k)} L(\boldsymbol{\Theta} | \boldsymbol{Y}) + \tau_{|\Omega|}, \tag{19}$$

where $\lim_{|\Omega| \to \infty} \tau_{|\Omega|} = 0$. Since finding the global minimizer is not always feasible due to the non-convexity of L, we require that $\hat{\boldsymbol{\Theta}}_{|\Omega|}$ be close to a global minimizer of $L(\boldsymbol{\Theta}|\boldsymbol{Y})$ when $|\Omega| \to \infty$.

Let $l_\Delta(\boldsymbol{\Theta} \mid \cdot) = l(\boldsymbol{\Theta}, \cdot) - l\left(\boldsymbol{\Theta}_0, \cdot\right)$, and

$$K\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0\right) = \frac{1}{n_1 n_2 n_3} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{i_3=1}^{n_3} E\left\{l_\Delta\left(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}\right)\right\}, \tag{20}$$

which is the expected loss difference between $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}_0$. Since $\boldsymbol{\Theta}_0$ is the true parameter, we have $K\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}\right) \geq 0$ for all $\boldsymbol{\Theta} \in S$ and $K = 0$ only if $\boldsymbol{\Theta} = \boldsymbol{\Theta}_0$. Given $K\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0\right)$, the distance between $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}_0$ can be measured as $\rho\left(\mathbf{\Theta}_0, \mathbf{\Theta}\right) = K^{1/2}\left(\mathbf{\Theta}_0, \mathbf{\Theta}\right)$. Similarly, we quantify the variance of the loss difference l_Δ as:

$$V\left(\mathbf{\Theta}_0, \mathbf{\Theta}\right) = \frac{1}{n_1 n_2 n_3} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{i_3=1}^{n_3} \operatorname{Var}\left\{l_\Delta\left(\mathbf{\Theta}, y_{i_1 i_2 i_3}\right)\right\}.$$

Note that $K\left(\mathbf{\Theta}_0, \mathbf{\Theta}\right) = \frac{1}{n_1 n_2 n_3} ||\mathbf{\Theta}_0 - \mathbf{\Theta}\|^2$, where $\| \cdot \|$ stands for the Euclidean norm of the vectorized tensor. We establish the convergence of $\hat{\mathbf{\Theta}}_{|\Omega|}$ in the following Theorem 1.

Theorem 1. *Suppose $\hat{\mathbf{\Theta}}_{|\Omega|}$ is a sample estimator satisfying (19). Then under Assumption 1 we have:*

$$P\left(\rho(\hat{\mathbf{\Theta}}_{|\Omega|}, \mathbf{\Theta}_0) \geq \eta_{|\Omega|}\right) \leq 7 \exp\left(-c_1 |\Omega| \eta_{|\Omega|}^2\right),$$

where $c_1 \geq 0$ is a constant, $\eta_{|\Omega|} = \max\left(\varepsilon_{|\Omega|}, \lambda_{|\Omega|}^{1/2}\right)$, and $\varepsilon_{|\Omega|} \sim \frac{1}{|\Omega|^{1/2}}$ is the best possible rate that can be achieved when $\lambda_{|\Omega|} \sim \varepsilon_{|\Omega|}^2$.

Theorem 1 indicates that, if the penalty term shrinks to zero at an appropriate rate, the proposed method leads to a convergence rate of $\frac{1}{|\Omega|^{1/2}}$ for $\mathbf{\Theta}$.

Next, we extend Theorem 1 to a more general loss function. Notice that the loss function $l(.,.)$ in (17) can be replaced by a loss function other than the L_2 loss to model complex and nonlinear relations between the ratings and $\mathbf{\Theta}$. For example, the loss function $l(\cdot,\cdot)$ can be a log-likelihood loss, then $K(\cdot,\cdot)$ reduces to the Kullback-Leiber pseudo-distance. Let $W_p^\alpha[a,b]^{n_1 \times n_2 \times n_3}$ be a Sobolev space with a finite L_p norm, where a and b are constants and α is a parameter associated with the degree of smoothness of the loss difference l_Δ.

Assumption 2. *For each $y_{i_1 \cdots i_3}$, we assume*

$$\left|l\left(\mathbf{\Theta}_0, y_{i_1 i_2 i_3}\right) - l\left(\mathbf{\Theta}, y_{i_1 i_2 i_3}\right)\right| \leq g\left(y_{i_1 i_2 i_3}\right) \left\|\mathbf{\Theta}_0 - \mathbf{\Theta}\right\|,$$

where $g(\cdot)$ satisfies $\mathrm{E}\left[\exp\left\{t_0 g\left(y_{i_1 i_2 i_3}\right)\right\}\right] \leq c_2 < \infty$, for a constant c_2 and a constant t_0 around 0. In particular, there exists a constant $c_2' > 0$, such that $E\left\{g^2\left(y_{i_1 i_2 i_3}\right)\right\} \leq c_2'$ for all $y_{i_1 i_2 i_3}$.

Assumption 3. *Suppose there exist $\delta > 0$ and $\beta \in [0,1)$, such that for Θ within the δ-ball centered at $\mathbf{\Theta}_0$ under the metric ρ, we have $\rho\left(\mathbf{\Theta}_0, \mathbf{\Theta}\right) \geq c_3 \left\|\mathbf{\Theta}_0 - \mathbf{\Theta}\right\|^{\frac{1}{1+\beta}}$, where $c_3 \geq 0$ is a constant and $\| \cdot \|$ is the Euclidean norm.*

The regularity condition defined in Assumption 2 is a restriction on the smoothness of loss function $l(.,.)$. Assumption 3 requires that in the neighborhood of the true parameter $\mathbf{\Theta}_0$, a metric $\rho\left(\mathbf{\Theta}_0, \cdot\right)$ is larger than a certain order of the Euclidean distance in Theorem 2. In contrast, if $\rho\left(\mathbf{\Theta}_0, \cdot\right)$ is bounded by the Euclidean distance and Assumption 2 is satisfied, then Theorem 1 is still valid in terms of the metric $\rho\left(\mathbf{\Theta}_0, \cdot\right)$.

Theorem 2. *Let $\hat{\mathbf{\Theta}}_{|\Omega|}$ be a sample estimator satisfying (19). Assume that $l_\Delta \in W_p^\alpha[a,b]^{n_1\times n_2\times n_3}$, where $p > 2$, and that Assumptions 1, 2 and 3 hold. Then we have:*

$$P\left(\rho\left(\hat{\mathbf{\Theta}}_{|\Omega|}, \mathbf{\Theta}_0\right) \geq \eta_{|\Omega|}\right) \leq 7\exp\left(-c_4|\Omega|\eta_{|\Omega|}^2\right),$$

where $c_4 \geq 0$ is a constant, and $\eta_{|\Omega|} = \max\left(\varepsilon_{|\Omega|}, \lambda_{|\Omega|}^{1/2}\right)$ with

$$\varepsilon_{|\Omega|} \sim \begin{cases} \left(\frac{1}{|\Omega|^{1/2}}\right)^{\frac{2\omega}{2\omega+1}} & \textit{if } \omega > \frac{1}{2}, \\ \left(\frac{1}{|\Omega|^{1/2}}\right)^{\omega} & \textit{if } \omega \leq \frac{1}{2}, \end{cases}$$

being the best possible rate achieved when $\lambda_{|\Omega|} \sim \varepsilon_{|\Omega|}^2$. Here $\omega = \alpha/\gamma$, and $\gamma = \sum_{i=1}^3 n_i R$ is the total number of parameters.

Theorem 2 shows that the convergence rate under a general loss is still $\frac{1}{|\Omega|^{1/2}}$ when the loss function is infinitely differentiable, i.e. $\omega = \infty$. When the smoothness of the loss function decreases, the convergence rate of the estimator $\hat{\mathbf{\Theta}}_{|\Omega|}$ is slower.

We also consider the ordinal rating case. Let $\mathbf{\Theta}$ denote the vectorization of $(\boldsymbol{\alpha}, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$. We represent the penalized likelihood-based loss function (11) as:

$$\mathcal{L}(\mathbf{\Theta}|\boldsymbol{Y}) = -\sum_{(i_1,i_2,i_3)\in\Omega} \log\{P(y_{i_1i_2i_3}|\mathbf{\Theta})\} + \lambda_{|\Omega|}J(\mathbf{\Theta}).$$

Then Assumption 1 and the definition of $\mathcal{S}(k)$ are the same as in the binary case, except that the total number of parameters γ becomes $(n_1+n_2+n_3)R+L$.

To measure the difference between the two parameters $\hat{\mathbf{\Theta}}_{|\Omega|}$ and $\mathbf{\Theta}_0$, we introduce the Hellinger metric $h(\cdot,\cdot)$ on $\mathcal{S}(k)$ as:

$$h(\mathbf{\Theta}_0, \hat{\mathbf{\Theta}}_{|\Omega|}) = \left[\frac{1}{|\Omega|}\sum_{(i_1,i_2,i_3)\in\Omega}\int\left\{P^{1/2}\left(y_{i_1i_2i_3} \mid \mathbf{\Theta}_0\right) - P^{1/2}\left(y_{i_1i_2i_3} \mid \hat{\mathbf{\Theta}}_{|\Omega|}\right)\right\}^2 dy_{i_1i_2i_3}\right]^{1/2}.$$

Theorem 3. *Under Assumption 1 and suppose $\lambda_{|\Omega|} < \frac{1}{2k}\epsilon_{|\Omega|}^2$, the best possible convergence rate of $\hat{\mathbf{\Theta}}_{|\Omega|}$ is*

$$\epsilon_{|\Omega|} \sim \frac{\sqrt{\gamma}}{|\Omega|^{1/2}}\left\{\log\left(\frac{L|\Omega|}{\sqrt{n_1n_2n_3\gamma}}\right)\right\}^{1/2},$$

and there exists a constant $c_5 > 0$, such that

$$P\left(h(\mathbf{\Theta}_0, \hat{\mathbf{\Theta}}_{|\Omega|}) \geq \epsilon_{|\Omega|}\right) \leq 7\exp\left(-c_5|\Omega|\epsilon_{|\Omega|}^2\right).$$

Theorem 3 establishes the convergence rate of $\hat{\boldsymbol{\Theta}}_{|\Omega|}$ with the Hellinger distance as a metric for the parameter space. This result still holds if the L_2 distance is used and additional assumptions on the local and global behavior of $Var\{\mathcal{L}(\hat{\boldsymbol{\Theta}}_{|\Omega|}|\boldsymbol{Y}) - \mathcal{L}(\boldsymbol{\Theta}_0|\boldsymbol{Y})\}$ are needed. We can then apply Corollary 2 in [30] to prove it. Notice that the number of levels L also affects the convergence rate, and $\epsilon_{|\Omega|}$ increases as L grows.

6. Simulation studies

In this section, we conduct simulations to compare the proposed method (TFD) with existing tensor factorization methods for both binary and ordinal rating cases.

6.1. Binary rating tensor

We first focus on binary rating tensors and compare the proposed method (TFD) with five competing methods; namely, the Boolean Tensor Decomposition (BTD) [28], Canonical Polyadic Decomposition (CP) [16], Generalized Canonical Polyadic Decomposition (GCP) [12], Bayesian probabilistic tensor factorization (BPTF) [36], and the pairwise interaction tensor factorization (PITF) [27]. Among these models, CP and PITF are considered as baseline tensor factorization methods which model multi-way interactions and two-way interactions, respectively. Both the BTD and GCP are able to deal with binary data, and the BPTF is a tensor factorization method that can characterize dependency across subjects within the last mode.

In the first experiment, we evaluate the performance of different methods via prediction accuracy under different ranks and tensor sizes. We consider a 3-order tensor, whose size is $n_1 \times n_2 \times n_3 = 200 \times 100 \times 50$ or $2000 \times 1000 \times 50$ and the rank of the latent factors is set as $R = 3, 5$ or 8. Each latent factor is generated as $\boldsymbol{a}_i, \boldsymbol{b}_j, \boldsymbol{c}_k \overset{\text{iid}}{\sim} N(0, 4 \cdot \boldsymbol{I}_R)$ for $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, and $k = 1, \ldots, n_3$, and the underlying rating probability is formulated as:

$$p_{ijk} = P(x_{ijk} = 1) = f(\sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr}),$$

and the binary rating tensor is generated via $x_{ijk} \sim Bernoulli(p_{ijk})$. We denote the missing rate of the observation in the rating tensor as π_0. For small tensor size $200 \times 100 \times 50$, the missing rate $\pi_0 = 90\%$, while for large tensor size $2000 \times 1000 \times 50$, $\pi_0 = 99\%$. For the observed rating, we assign 60%, 20% and 20% of the data into training, validation and testing sets, respectively. Here, all simulations are replicated 100 times, and the true rank R is applied.

Table 1 provides the comparisons between the proposed method and competing methods, where the performance is measured by the mean square error (MSE) and the area under the receiver operating characteristic curve (AUC).

Table 1

The MSE and AUC of the proposed method and competing methods under the first simulation setting in Section 6.1. Standard errors are reported in parentheses.

Rank	Method	MSE		AUC	
		$200 \times 100 \times 50$	$2000 \times 1000 \times 50$	$200 \times 100 \times 50$	$2000 \times 1000 \times 50$
R=3	**TFD**	**0.038 (0.002)**	**0.123 (0.022)**	**0.995 (0.001)**	**0.906 (0.025)**
	PITF	0.176 (0.009)	0.190 (0.011)	0.816 (0.019)	0.808 (0.012)
	CP	0.455 (0.005)	0.495 (0.001)	0.902 (0.008)	0.801 (0.006)
	GCP	0.206 (0.002)	0.249 (0.000)	0.923 (0.001)	0.843 (0.003)
	BTD	0.310 (0.004)	–	0.689 (0.004)	–
	BPTF	0.505 (0.006)	0.501 (0.001)	0.622 (0.002)	0.562 (0.004)
R=5	**TFD**	**0.049 (0.002)**	**0.153 (0.023)**	**0.988 (0.004)**	0.877 (0.025)
	PITF	0.199 (0.008)	0.255 (0.057)	0.771 (0.017)	0.780 (0.012)
	CP	0.448 (0.004)	0.495 (0.001)	0.929 (0.001)	**0.906 (0.003)**
	GCP	0.233 (0.036)	0.249 (0.000)	0.904 (0.004)	0.765 (0.006)
	BTD	0.294 (0.002)	–	0.705 (0.010)	–
	BPTF	0.504 (0.001)	0.500 (0.001)	0.585 (0.003)	0.524 (0.004)
R=8	**TFD**	**0.068 (0.009)**	**0.128 (0.006)**	**0.971 (0.009)**	**0.907 (0.006)**
	PITF	0.238 (0.003)	0.318 (0.007)	0.717 (0.007)	0.713 (0.012)
	CP	0.456 (0.005)	0.495 (0.002)	0.935 (0.002)	0.907 (0.003
	GCP	0.256 (0.002)	0.249 (0.000)	0.797 (0.004)	0.817 (0.002)
	BTD	0.305 (0.004)	–	0.694 (0.003)	–
	BPTF	0.498 (0.005)	0.504 (0.001)	0.543 (0.003)	0.581 (0.002)

The proposed method performs the best in all cases except for the large tensor of $2000 \times 1000 \times 50$ with $R = 5$ in terms of AUC. This is because the observed sample size for the large tensor is sufficient for recovering the tensor using CP decomposition. Compared to a small tensor, a large tensor has 100 times more entries, but the missing rate is only 10 times smaller, resulting in 10 times more observations for a large tensor. The improvements of the proposed method compared to the other methods are at least 78% in the MSE. Note that the performance of all methods becomes worse when the rank increases in general, due to the increasing number of total parameter. The Boolean Tensor Decomposition (BTD) is not able to handle the large tensor of $2000 \times 1000 \times 50$, as there is no sparsity constraint. The CP method provides a comparable performance in terms of the AUC, but the MSE is rather poor. This is because CP is directly applied to a binary rating, and the estimation of each rating given by CP is around 0.5, which leads to a high MSE. However, the AUC is mainly used to distinguish the high and low binary ratings, and the CP can learn such information from binary ratings, thus gives a comparable AUC. In contrast, the proposed method is able to improve the performance in terms of MSE by utilizing the link function, and transforming a binary rating to a continuous outcome.

The second experiment investigates the prediction performance of binary ratings when no dependency exists among pairwise interactions of latent factors. Specifically, we generate two separate latent factors of item $\boldsymbol{b}_j^{(1)}$ for user-item interaction and $\boldsymbol{b}_j^{(2)}$ for item-context interaction from two normal distributions,

Table 2
The MSE and AUC of the proposed method and competing methods when no dependency between interactions exists. Standard errors are reported in parentheses.

	TFD	PITF	CP	GCP	BTD	BPTF
MSE	**0.157 (0.005)**	0.189 (0.008)	0.463 (0.004)	0.260 (0.001)	0.326 (0.007)	0.496 (0.005)
AUC	**0.853 (0.009)**	0.802 (0.029)	0.833 (0.006)	0.788 (0.005)	0.676 (0.003)	0.621 (0.005)

$N(0, 4 \cdot \boldsymbol{I}_R)$. Thus, the true probability tensor becomes:

$$p_{ijk} = P(x_{ijk} = 1) = f(\sum_{r=1}^{R} a_{ir} b_{jr}^{(1)} + \sum_{r=1}^{R} b_{jr}^{(2)} c_{kr}).$$

All other simulation settings are similar to the first experiment.

Table 2 provides the performance of different methods when no dependency exists between interactions. The proposed model is still the best compared with competing models, indicating that the proposed method is quite robust against model misspecification when no dependency among pairwise interactions exists. There are two major reasons for improved performance of the proposed method over the PITF. First, we treat binary response as a random variable following an underlying probability, while the PITF treats it as a continuous variable. Second, the proposed loss function includes penalty on the latent factors while the PITF does not. Therefore, the PITF might suffer from low prediction accuracy due to overfitting and scale indeterminacy of latent factors.

6.2. *Ordinal rating tensor*

In the following simulation, we investigate the performance of the proposed method (TFD) with five competing methods on the ordinal rating prediction. Three of them are CP, PITF and BPTF. The other two are extensions of the CP and PITF methods, which are not designed for ordinal rating, but can be plugged into the proportional odds model by replacing θ_{ijk} in (12) with the CP and PITF output. We refer to these two as the Ordinal Canonical Polyadic Decomposition (OCP) and Ordinal Pairwise Interaction Tensor Factorization (OPI).

We generate each latent factor $\boldsymbol{a_i}, \boldsymbol{b_j}, \boldsymbol{c_k} \overset{\text{iid}}{\sim} N(0, 4 \cdot \boldsymbol{I_r})$ for $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, $k = 1, \ldots, n_3$, and cutoff parameters $\alpha_l \overset{\text{iid}}{\sim} Uniform(0, 1)$ for $l = 1, \ldots, L - 1$. The tensor size is $200 \times 100 \times 50$, or $2000 \times 1000 \times 50$, and the number of rating levels is $L = 3, 5$ or 7.

The underlying probability of the rating being l is

$$p_{ijk}^{l} = P(x_{ijk} = l) = f(\alpha_l - \theta_{ijk}) - f(\alpha_{l-1} - \theta_{ijk}), \quad l = 1, ..., L,$$

$$\theta_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} + \sum_{r=1}^{R} b_{jr} c_{kr},$$

Table 3

The MSE of the proposed method and competing methods for ordinal ratings under different number of levels. Standard errors are reported in parentheses.

Rank	Method	Tensor size	
		200 × 100 × 50	2000 × 1000 × 50
L=3	**TFD**	**0.073 (0.048)**	**0.183 (0.043)**
	PITF	0.253 (0.062)	4.975 (0.283)
	OPI	0.105 (0.087)	0.995 (0.184)
	CP	1.621 (0.991)	4.822 (3.069)
	OCP	0.551 (0.323)	1.059 (0.418)
	BPTF	0.220 (0.091)	1.175 (0.118)
L=5	**TFD**	**0.216 (0.088)**	**0.931 (0.178)**
	PITF	3.203 (0.454)	9.035 (1.061)
	OPI	1.973 (0.103)	3.015 (0.915)
	CP	7.117 (1.654)	9.423 (3.294)
	OCP	2.277 (0.654)	3.607 (0.642)
	BPTF	1.085 (0.109)	2.109 (0.354)
L=7	**TFD**	**0.513 (0.040)**	**1.852 (0.404)**
	PITF	5.055 (1.259)	13.891 (1.188)
	OPI	4.510 (0.703)	7.009 (1.125)
	CP	10.602 (6.528)	16.494 (3.130)
	OCP	9.618 (5.517)	7.620 (0.659)
	BPTF	3.008 (0.534)	3.796 (0.931)

where f is the logistic function and the ordinal rating tensor follows

$$x_{ijk} \sim Multinomial(p_{ijk}^1, \ldots, p_{ijk}^L).$$

We set the missing rate $\pi_0 = 90\%$ for the small tensor size, and $\pi_0 = 99\%$ for the large tensor size. All simulations are replicated 100 times, and use the true rank R.

Table 3 shows that for both small and large scale tensors, the proposed method outperforms other methods by more than 30% in the MSE when the number of levels is 3, while the improvement increases to more than 88% when the number of levels is 5 or 7. As the number of levels increases, the MSE of the proposed model increases as well, which is consistent with Theorem 3.

We also compare the performance of the proposed TFD model with PITF, OPI, CP, OCP and BPTF given different observation missing rates. Specifically, we fix the tensor size at 2000 × 1000 × 50, the rank of the latent factor at $R = 5$, and the number of rating levels at $L = 3$. We consider two missing rates at $\pi_0 = 0.95$ and 0.97. Table 4 shows that although all methods perform worse as π_0 increases, the proposed method and the PITF-based methods are most robust against observation-missing than the CP-based methods. This is because the proposed method pursues parsimonious tensor modeling, and is less demanding on the sample size to recover data information compared with CP-based modeling.

TABLE 4
The MSE of the proposed method and competing methods for ordinal ratings under different missing rates. Standard errors are reported in parentheses.

Method	Missing rate	
	95%	97%
TFD	**0.172 (0.076)**	**0.223 (0.096)**
PITF	2.636 (0.435)	3.150 (0.314)
OPI	1.088 (0.318)	1.196 (0.512)
CP	1.558 (0.470)	2.082 (0.415)
OCP	1.272 (0.301)	1.835 (0.397)
BPTF	0.925 (0.121)	0.966 (0.111)

7. Real data application

In this section, we apply the proposed method to two real application datasets, the User-Location-Activity dataset [38] and the Beer Review dataset [21].

7.1. User-location-activity data

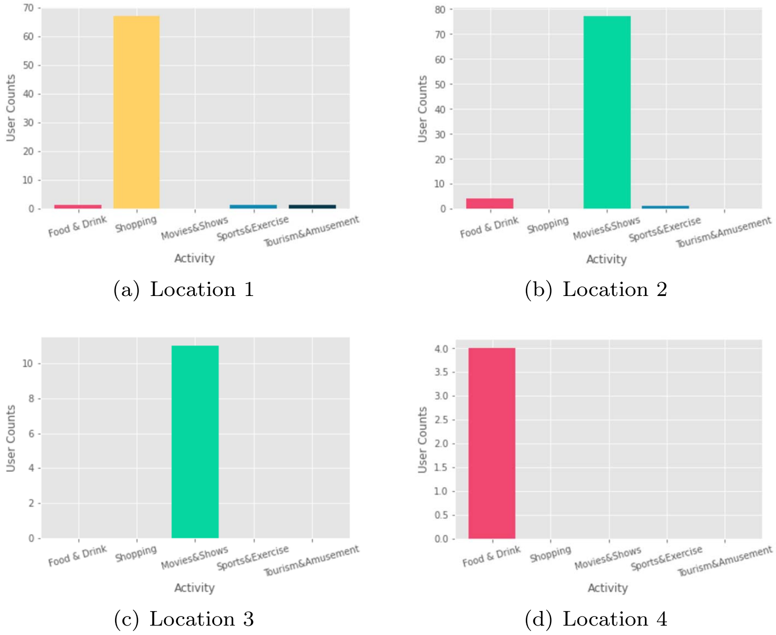
The User-Location-Activity dataset [38] contains 164 users, 168 locations and 5 different types of activities. The study asked 164 users to carry GPS devices to record their movements and activities from April 2007 to October 2009 in a city of China. The raw GPS points are labelled as 168 locations for recommendation. The activities are divided into 5 categories, including "Food & Drink", "Shopping", "Movies & Shows", "Sports & Exercise" and "Tourism & Amusement." The original data provides the counts of each activity at a specific location. After removing the users with no records and the locations with no counts, there are 146 users and 85 locations remaining, where only 2% of the entries have counts larger than 0. This is because each users only went to a few locations and had few specific activities during the experiment. Therefore, most entries are missing, which result in sparse tensor data.

Figure 4 illustrates the dependency between location and activity based on the data. Specifically, people doing specific activities are correlated with their locations. Thus, it is reasonable to consider a dependence between location and activity when build the recommender system.

We define the tensor $\mathcal{X}$ as:

$$x_{ijk} = \begin{cases} 1 & \text{if user } i \text{ does activity } k \text{ at location } j \\ 0 & \text{otherwise.} \end{cases}$$

The goal of our study is to predict the user's probability of choosing a certain activity at a specific location. Through the prediction result, we can recommend location-activity pairs to each user. The preprocessed data are randomly split into training, validation and testing sets with proportions of 60%, 20%, 20%, and the experiments are replicated 100 times. We compare the proposed method with competing methods in Section 6.1. Each method's rank is tuned via minimizing



(a) Location 1

(b) Location 2

(c) Location 3

(d) Location 4

Fig 4. *User count of five activities on four sample locations.*

Table 5

MSE, AUC, and resulting rank R of User-Location-Activity dataset. Standard errors are reported in parentheses.

	TFD	PITF	CP	GCP	BTD	BPTF
MSE	**0.016 (0.001)**	0.436 (0.035)	0.018 (0.001)	0.254 (0.003)	0.018 (0.003)	0.982 (0.002)
AUC	**0.928 (0.009)**	0.512 (0.042)	0.807 (0.036)	0.869 (0.007)	0.521 (0.043)	0.539 (0.010)
R	9	8	2	2	9	9

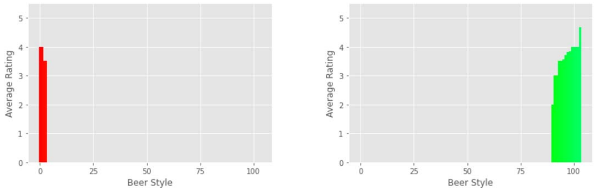
the mean square errors (MSE) from the validation set, and the largest possible rank is 10.

Table 5 shows that the proposed model outperforms the other methods in AUC and MSE. The improvement from the proposed method in the AUC is more than 6 % compared with the second-best model, the GCP method. The proposed model, CP and BTD perform similarly on the MSE of predictions due to the imbalanced data.

7.2. Beer review data

In this subsection, we apply the proposed method to Beer Review Dataset [21] and recommend breweries with preferred beer styles to customers.

This dataset consists of 1.5 million beer reviews from 2002 to 2011 collected



(a) Brewery1 (b) Brewery 2

FIG 5. *Average ratings of different beer styles on two sample breweries.*

by BeerAdvocate. Each review includes a rating from 0 to 5 with a 0.5 incremental point scale, thus consisting of 11 levels in total. The dataset includes a total of 33,388 users, 5,743 breweries and 104 beer styles. Note that for each brewery, there is only one beer for each beer style. Here, we consider the brewery as an item and the beer style as the contextual variable. For the dependence structure, the beer style is entirely controlled by the brewery, and users' preference of brewery highly depends on the features of different beer styles in different breweries. As illustrated in Figure 5, the users' ratings on different beer styles vary for different breweries, indicating the dependency between user-brewery interaction and brewery-beer style interaction. To be more specific, the brewery affects the distribution of ratings on beer styles.

We randomly split the data into training, validation and testing sets with proportions of 60%, 20% and 20%, respectively. We adopt the sparse tensor strategy in Section 4 to handle the large tensor size. We compare our method with the CP decomposition, Ordinal CP decomposition, PITF, Ordinal PITF and BPTF. Each method's rank is tuned via minimizing the mean square errors (MSE) from the validation set, and the largest possible rank is 10.

Table 6 shows that the proposed model outperforms other methods with respect to prediction MSE. Specifically, neither the CP or PITF can handle ordinal ratings with 11 levels well with large MSE. However, their modified versions, the OCP and OPI, perform better. The comparison between the proposed model and the Ordinal PITF indicates that incorporating dependency structure improves the prediction performance by 40% on the MSE. The proposed model also performs better than the BPTF, showing the advantages of incorporating the dependency between interactions of modes. Without any additional optimization of the implementation, the proposed algorithm has similar computation time compared to existing methods. Note that, except for the BPTF model, all the other methods including the proposed method are implemented in Python. Due to memory limitations, we store a large tensor in a sparse tensor data structure where only observed entries' information are stored. To further optimize the implementation, we can implement parallel computing in calculating gradients for different factors.

Table 6

MSE, selected rank R, and running time in minutes of the proposed method and competing methods for the Beer Review dataset. Standard errors are reported in parentheses.

	TFD	CP	OCP	PITF	OPI	BPTF
MSE	**1.064 (0.385)**	15.056 (0.060)	4.436 (1.267)	6.585 (1.258)	1.772 (0.148)	2.243 (0.245)
R	10	9	8	8	10	10
Time (mins)	39.753 (2.635)	21.801 (1.541)	87.078 (19.202)	33.059 (2.143)	90.421 (3.852)	80.702 (29.020)

8. Discussion

In this paper, we propose a new tensor-based recommender system to incorporate the dependency among modes, and to achieve tensor completion through utilizing shared factors in addition to the pairwise interactions. The proposed decomposition is capable of capturing the dependency between user-item and item-context interactions, and leads to significant advantages in incorporating context information that introduces dependency among modes. In addition, the proposed method is capable of dealing with ordinal ratings in additional to binary recommendations. We also propose a mode-wise coordinate descent (MCD) algorithm to accelerate computation and reduce memory storage. We demonstrate the superiority of the proposed method on both simulation and real data applications. In theory, we show that the estimated parameter achieves asymptotic consistency in both binary and ordinal rating cases.

There are several potential research directions to extend our method. One is to develop a unified framework for different types of utility jointly, such as binary, ordinal, count data and non-negative continuous data. Another possible direction is to extend the proposed method to a higher-order tensor, and develop an inference procedure to test whether certain dependencies between two specific interactions are needed or not.

Appendix A: Appendix section

A.1. Proof of Theorem 1

For any $k_i \geq 0$, let $A\left(k_1, k_2\right) = \left\{\boldsymbol{\Theta} \in \mathcal{S} : k_1 \leq \rho\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}\right) \leq 2k_1, J(\boldsymbol{\Theta}) \leq k_2\right\}$, and $\mathcal{F}\left(k_1, k_2\right) = \left\{l_\Delta(\boldsymbol{\Theta} \mid \cdot) : \boldsymbol{\Theta} \in A\left(k_1, k_2\right)\right\}$.

We verify several conditions of Corollary 2 in [30]. First, we verify Assumption B. By definition, we have

$$\operatorname{Var}\left\{l_\Delta\left(\Theta, y_{i_1 i_2 \cdots i_d}\right)\right\} = 4(\theta_{i_1 i_2 \cdots i_d} - \theta_{0 i_1 i_2 \cdots i_d})\theta_{0 i_1 i_2 \cdots i_d}(1 - \theta_{0 i_1 i_2 \cdots i_d}),$$

$$\sup_{A(k_1,k_2)} V\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}\right) \leq c_6 k_1^2 = c_6 k_1^2 \left\{1 + \left(k_1^2 + k_2\right)^{\beta_1}\right\},$$

and hence $\beta_1 = 0$. In the rest of this section, all c_i's with $i \in \mathbb{N}$ are assumed to be non-negative constants.

Second, for Assumption C, recall that $\theta_{0,i_1\cdots i_d}$ and $\theta_{i_1\cdots i_d}$ are between 0 and 1 by property of sigmoid function for a given $y_{i_1\cdots i_d}$. Thus, we have

$$\begin{aligned}\operatorname{Var}\left\{l_\Delta\left(\Theta, y_{i_1 i_2\cdots i_d}\right)\right\} &= 4(\theta_{i_1 i_2\cdots i_d} - \theta_{0 i_1 i_2\cdots i_d})\theta_{0 i_1 i_2\cdots i_d}(1-\theta_{0 i_1 i_2\cdots i_d})\\ &\leq c_7(\theta_{i_1 i_2\cdots i_d} - \theta_{0 i_1 i_2\cdots i_d}).\end{aligned}$$

Furthermore, $\left|l\left(\Theta, y_{i_1\cdots i_d}\right) - l\left(\boldsymbol{\Theta}_0, y_{i_1\cdots i_d}\right)\right| = \left|\theta_{0,i_1\cdots i_d} - \theta_{i_1\cdots i_d}\right| \cdot \left|2y_{i_1\cdots i_d} - \theta_{0,i_1\cdots i_d} - \theta_{i_1\cdots i_d}\right|$. Define a new random variable $w = \left|2y_{i_1\cdots i_d} - \theta_{0,i_1\cdots i_d} - \theta_{i_1\cdots i_d}\right|$, then we have $\mathrm{E}\left\{\exp\left(t_0 w\right)\right\} < \infty$ for t_0 at an open interval containing 0.

Now we verify that for a constant $c_8 > 0$, we have $\sup_{A(k_1,k_2)} \|\boldsymbol{\Theta}_0 - \boldsymbol{\Theta}\|_{\sup} \leq c_8\left(k_1^2 + k_2\right)^{\beta_2}$ for $\beta_2 \in [0,1)$. Define $f_0 = f_0(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}) = \boldsymbol{\Theta} - \boldsymbol{\Theta}_0$. Recall that $\|(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})\|_\infty \leq c$ and $\gamma = \sum_{k=1}^3 (n_k)\, r$ is the total number of parameters. Since f_0 is a quadratic function of elements of $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})$, we have $f_0 \in W_2^\infty[-c,c]^\gamma$ where W_2^∞ is a Sobolev space, and $\|f_0\|_2 = \rho\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}\right) \leq c_9$ for a constant $c_9 > 0$. In addition, we have $f_0^{(\alpha)} = 0$ for $\alpha = \infty$. Therefore, based on Lemma 2 of [30], we get

$$\|f_0\|_\infty = \|\boldsymbol{\Theta}_0 - \boldsymbol{\Theta}\|_\infty \leq 2c_4.$$

The required conditions are fulfilled by defining $c_3 = 2c_4$ and $\beta_2 = 0$.

Next, we verify the Assumption D. Let

$$\mathcal{N}(\varepsilon, n) = \left\{g_1^l, g_1^u, \ldots, g_n^l, g_n^u\right\},$$

be a set of functions from the L_2 space, where $\max_{1\leq i\leq n}\left\|g_i^u - g_i^l\right\|_2 \leq \varepsilon$. Suppose for any function $l_\Delta \in \mathcal{F}\left(k_1,k_2\right)$, there exists $i \in \{1,\ldots,n\}$ such that $g_i^l \leq l_\Delta \leq g_i^u$ almost surely. Then the Hellinger metric entropy is defined as $H(\varepsilon,\mathcal{F}) = \log\{n : \min\mathcal{N}(\varepsilon,n)\}$. Let $\omega = \frac{\alpha}{\gamma} = \infty$, then $p\omega = \infty > 1$. Define

$$\psi\left(k_1,k_2\right) = \int_{L_0}^{U_0} H^{1/2}(u,\mathcal{F})du/L_0,$$

where $L_0 = c_{10}\lambda_{|\Omega|}\left(k_1^2 + k_2\right)$ and $U_0 = c_{11}\varepsilon_{|\Omega|}\left(k_1^2+k_2\right)^{(1+\max(\beta_1,\beta_2))/2}$. Based on Theorem 5.2 of [7], the Hellinger metric entropy is controlled by

$$H\left(\varepsilon_{|\Omega|}, \mathcal{F}\right) \leq c_7\varepsilon_{|\Omega|}^{-0} = c_{12}.$$

Recall that $\beta_1 = \beta_2 = 0$. Then for fixed k_1 and k_2, we have $\psi\left(k_1,k_2\right) = \sqrt{c_{12}}\frac{U_0 - L_0}{L_0} \sim \frac{\varepsilon_{|\Omega|} - \lambda_{|\Omega|}}{\lambda_{|\Omega|}}$. Given that $\psi \sim |\Omega|^{1/2}$, the best possible rate is achieved at $\varepsilon_{|\Omega|} \sim \lambda_{|\Omega|}^{1/2}$, that is,

$$\varepsilon_{|\Omega|} \sim \frac{1}{|\Omega|^{1/2}}.$$

The result in Theorem 1 then follows by applying Corollary 2 of [30].

A.2. Proof of Theorem 2

We define $A(k_1, k_2)$ and $\mathcal{F}(k_1, k_2)$ the same as in the proof of Theorem 1 and start by verifying conditions of Corollary 2 of [30].

First,for Assumption B, based on the definition of $V(\cdot,\cdot)$ and Assumption 2 we have

$$\begin{aligned} V(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}) &= \frac{1}{n_1 n_2 n_3} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{i_3=1}^{n_3} \operatorname{Var}\left\{ l(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}) - l(\boldsymbol{\Theta}_0, y_{i_1 i_2 i_3}) \right\} \\ &\leq \frac{1}{n_1 n_2 n_3} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{i_3=1}^{n_3} \mathrm{E}\left\{ \left| l(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}) - l(\boldsymbol{\Theta}_0, y_{i_1 i_2 i_3}) \right|^2 \right\} \\ &\leq \frac{1}{n_1 n_2 n_3} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{i_3=1}^{n_3} \mathrm{E}\left\{ g^2(y_{i_1 i_2 i_3}) \right\} \left\| \boldsymbol{\Theta}_0 - \boldsymbol{\Theta} \right\|^2 \\ &\leq c_2' \left\| \boldsymbol{\Theta}_0 - \boldsymbol{\Theta} \right\|^2 . \end{aligned}$$

Therefore, we have $\sup_{A(k_1,k_2)} V(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}) \leq 4 c_2' k_1^2 \left\{ 1 + \left(k_1^2 + k_2 \right)^{\beta_1} \right\}$ with $\beta_1 = 0$.

Second, we verify Assumption C. By Assumptions 2 and 3, we have

$$\begin{aligned} \left\| l_\Delta(\boldsymbol{\Theta} \mid y_{i_1 i_2 i_3}) \right\|_2 &= \left[\mathrm{E}\left\{ \left| l(\boldsymbol{\Theta}, y_{i_1 i_2 i_3}) - l(\boldsymbol{\Theta}_0, y_{i_1 i_2 i_3}) \right|^2 \right\} \right]^{1/2} \\ &\leq c_2' \mid \boldsymbol{\Theta}_0 - \boldsymbol{\Theta} \| \\ &\leq c_{13} \rho(\boldsymbol{\Theta}_0, \boldsymbol{\Theta})^{1+\beta} , \end{aligned}$$

for $c_{13} = c_2' / c_3^{1+\beta}$, a given element $y_{i_1 i_2 i_3}$ in the tensor, and $\delta > 0$ such that $\Theta \in B_\delta(\boldsymbol{\Theta}_0)$. Then by Lemma 2 of [30], we have

$$\begin{aligned} \left\| l_\Delta(\boldsymbol{\Theta} \mid y_{i_1 i_2 i_3}) \right\|_\infty &\leq c_{14} \left\| l_\Delta(\boldsymbol{\Theta} \mid y_{i_1 i_2 i_3}) \right\|_2^{\left(\alpha - p^{-1}\right) / \left(\alpha - p^{-1} + 1/2\right)} \\ &\leq c_{13} c_{14} \rho(\boldsymbol{\Theta}_0, \boldsymbol{\Theta})^{2\beta_2} , \end{aligned}$$

where $\beta_2 = \frac{1+\beta}{2} \frac{\alpha - p^{-1}}{\alpha - p^{-1} + 1/2}$. Since $\beta \in [0, 1)$, we have $\beta_2 \in [0, 1)$.

We now verify Assumption D, the condition on the Hellinger metric entropy. Recall that $\omega = \frac{\alpha}{\gamma}$. Therefore, based on Theorem 5.2 of [7], the Hellinger metric entropy is upper-bounded by $H\left(\varepsilon_{|\Omega|}, \mathcal{F}\right) \leq c_{15} \varepsilon_{|\Omega|}^{-1/\omega}$. Then we have:

$$\begin{aligned} \psi(k_1, k_2) &= \int_{L_0}^{U_0} u^{-\frac{1}{2\omega}} du / L_0 \\ &\sim C(k_1, k_2) \frac{\varepsilon_{|\Omega|}^{-\frac{1}{2\omega}+1} - \lambda_{|\Omega|}^{-\frac{1}{2\omega}+1}}{\lambda_{|\Omega|}} . \end{aligned}$$

The best possible rate is provided by setting $\psi(k_1, k_2) \sim |\Omega|^{1/2}$ and $\lambda_{|\Omega|} \sim \varepsilon^2_{|\Omega|}$. Hence

$$\varepsilon_{|\Omega|}^{-\frac{1}{2\omega}-1} - \varepsilon_{|\Omega|}^{-\frac{1}{\omega}} \sim |\Omega|^{1/2}.$$

That is,

$$\varepsilon_{|\Omega|} \sim \begin{cases} \left(\frac{1}{|\Omega|^{1/2}}\right)^{\frac{2}{2\omega+1}} & \text{if } \omega > \frac{1}{2} \\ \left(\frac{1}{|\Omega|^{1/2}}\right)^{\omega} & \text{if } \omega \le \frac{1}{2} \end{cases}$$

Then the result follows by applying Corollary 2 of [30].

A.3. Proof of Theorem 3

Here the density is

$$p(y, \mathbf{\Theta}) = \sum_{l=1}^{L} \left\{ I(y = l) \left(f(\alpha_l - \theta) - f(\alpha_{l-1} - \theta) \right) \right\}.$$

where $\theta = \sum_r A^{(1)}_{ir} A^{(2)}_{jr} + \sum_r A^{(2)}_{jr} A^{(3)}_{kr}$.

Note that

$$\begin{aligned} & \left| p^{1/2}(y, \mathbf{\Theta}_1) - p^{1/2}(y, \mathbf{\Theta}_2) \right|^2 \\ \le & c_{16}(\left| f^{1/2}(\alpha_1 - \theta_1) - f^{1/2}(\alpha_1 - \theta_2) \right|^2 \\ & + \sum_{l=2}^{L-1} \left(\left| f^{1/2}(\alpha_l - \theta_1) - f^{1/2}(\alpha_l - \theta_2) \right|^2 \right. \\ & \left. + \left| (1 - f(\alpha_{l-1} - \theta_1))^{1/2} - (1 - f(\alpha_{l-1} - \theta_2))^{1/2} \right|^2 \right) \\ & + \left| (1 - f(\alpha_{L-1} - \theta_1))^{1/2} - (1 - f(\alpha_{L-1} - \theta_2))^{1/2} \right|^2) \\ \le & c_{17} L ||\mathbf{\Theta}_1 - \mathbf{\Theta}_2||_2^2, \end{aligned}$$

where c_{16} and c_{17} are some constant. In the last step, the mean value theorem and the boundness of function f is used.

We then verify the condition of Lemma 2.1 of [23]. Based on the above inequality,

$$\begin{aligned} & \left\{ \frac{1}{n_1 n_2 n_3} \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} E \left(\sup_{\hat{\mathbf{\Theta}} \in B_d(\mathbf{\Theta})} \left| p^{1/2}(y_{ijk}, \hat{\mathbf{\Theta}}_{ijk}) - p^{1/2}(y_{ijk}, \mathbf{\Theta}_{ijk}) \right|^2 \right) \right\}^{1/2} \\ = & \left\{ \frac{1}{n_1 n_2 n_3} \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \right. \end{aligned}$$

$$
\begin{aligned}
&\int \sup_{\hat{\boldsymbol{\Theta}} \in B_d(\boldsymbol{\Theta})} \left| p^{1/2}(y_{ijk}, \hat{\boldsymbol{\Theta}}_{ijk}) - p^{1/2}(y_{ijk}, \boldsymbol{\Theta}_{ijk}) \right|^2 d\nu(y) \Bigg\}^{1/2} \\
&\leq \left\{ \frac{c_{17} L}{n_1 n_2 n_3} \sup_{\hat{\boldsymbol{\Theta}} \in B_d(\boldsymbol{\Theta})} \|\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_2^2 \right\}^{1/2} \\
&\leq \sqrt{\frac{c_{17} L}{n_1 n_2 n_3}} d \\
&:= g(d).
\end{aligned}
$$

Hence for $u > 0$,

$$
H^B(u, \mathcal{S}(k), \rho) \leq H\left(g^{-1}(u/2), \mathcal{S}(k), \rho\right),
$$

where H^B is the metric entropy of $\mathcal{S}(k)$ with bracketing of $f^{1/2}$, H is the ordinary metric entropy of $\mathcal{S}(k)$, and ρ is the L_2 -norm. Next we provide an upper bound for $H\left(g^{-1}(u/2), \mathcal{S}(k), \rho\right)$. Since $g^{-1}(u/2) = \frac{\sqrt{n_1 n_2 n_3}}{2\sqrt{c_{17}}} u$, and $\|\boldsymbol{\Theta}\|_\infty \leq c_0$, we have

$$
\begin{aligned}
0 &\leq H^B(u, \mathcal{S}(k), \rho) \\
&\leq H\left(g^{-1}(u/2), \mathcal{S}(k), \rho\right) \\
&\leq \log\left[\max\left\{ \left(\frac{c_0 \sqrt{(n_1+n_2+n_3)R + L}}{\frac{\sqrt{n_1 n_2 n_3}}{2\sqrt{c_{17} L}} u} \right)^{(n_1+n_2+n_3)R+L}, 1 \right\} \right] \\
&\leq \max\left\{ ((n_1+n_2+n_3)R + L) \log\left(\frac{2c_0 \sqrt{c_{17} L((n_1+n_2+n_3)R + L)}}{\sqrt{n_1 n_2 n_3} u} \right), 0 \right\} \\
&= \max\left\{ ((n_1+n_2+n_3)R + L) \log\left(\frac{C \sqrt{L((n_1+n_2+n_3)R + L)}}{\sqrt{n_1 n_2 n_3} u} \right), 0 \right\},
\end{aligned}
$$

for $u \geq \epsilon_{|\Omega|}^2$ and $C = 2c_0\sqrt{c_17}$.

We now find the convergence rate $\epsilon_{|\Omega|}$, the smallest ϵ that satisties Assumption A of Theorem 1 of [30].

Note that $\psi_1 \leq 0 \leq c_{18}|\Omega|^{1/2}$ when $x \geq 1$, so we only consider the case when $0 < x < 1$. Assume that $n_1 n_2 n_3 > ((n_1 + n_2 + n_3)R + L)$, then:

$$
\begin{aligned}
\psi_1(\epsilon, k) &= \int_x^{x^{1/2}} \left\{ H^B(u, \mathcal{F}(k)) \right\}^{1/2} du/x \\
&\leq ((n_1+n_2+n_3)R + L)^{1/2} \\
&\int_x^{x^{1/2}} \left(\log\left(\frac{C\sqrt{L((n_1+n_2+n_3)R + L)}}{\sqrt{n_1 n_2 n_3}} \right) - \log u \right)^{1/2} du/x \\
&\leq ((n_1+n_2+n_3)R + L)^{1/2} \left(x^{-1/2} - 1 \right) \\
&\left\{ \log\left(\frac{C\sqrt{L((n_1+n_2+n_3)R + L)}}{\sqrt{n_1 n_2 n_3}} \right) + \log\left(x^{-1}\right) \right\}^{1/2}.
\end{aligned}
$$

For the best possible rate of convergence, we have $\lambda_{|\Omega|} = O\left(\epsilon^2_{|\Omega|}\right)$. Therefore, we solve

$$\begin{aligned}
&\sup_{k\geq k0} \psi_1(\epsilon, k) = \psi_1(\epsilon, k_0) \\
&\sim \sqrt{((n_1+n_2+n_3)R+L}\frac{1}{\epsilon_{|\Omega|}}\left\{\log\left(\frac{\sqrt{L((n_1+n_2+n_3)R+L)}}{\sqrt{n_1n_2n_3}\epsilon^2_{|\Omega|}}\right)\right\}^{1/2} \\
&= c_{18}|\Omega|^{1/2}.
\end{aligned}$$

Then we have

$$\epsilon_{|\Omega|} \sim \frac{\sqrt{\gamma}}{|\Omega|^{1/2}}\left\{\log\left(\frac{L|\Omega|}{\sqrt{n_1n_2n_3\gamma}}\right)\right\}^{1/2}.$$

where $\gamma = (n_1+n_2+n_3)R+L$ is the total number of parameters.

With $\epsilon_{|\Omega|}$ and $\lambda_{|\Omega|}$ above, the Assumption A of [30] is satisfied. The result then follows the Corollary 1.

Acknowledgments

The authors thank the editor, the associate editor, and reviewers for providing thoughtful comments and suggestions.

References

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

[2] Adomavicius, G. and Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender Systems Handbook* 217–253. Springer.

[3] Agarwal, D. and Chen, B.-C. (2009). Regression-based latent factor models. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 19–28.

[4] Bi, X., Qu, A., Shen, X. et al. (2018). Multilayer tensor factorization with applications to recommender systems. *The Annals of Statistics* **46** 3308–3333. MR3852653

[5] Bi, X., Qu, A., Wang, J. and Shen, X. (2017). A group-specific recommender system. *Journal of the American Statistical Association* **112** 1344–1353. MR3735382

[6] Bi, X., Tang, X., Yuan, Y., Zhang, Y. and Qu, A. (2021). Tensors in statistics. *Annual Review of Statistics and Its Application* **8** 345–368. MR4243551

[7] Birman, M. S. and Solomyak, M. Z. (1967). Piecewise-polynomial approximations of functions of the classes W_p^α. *Matematicheskii Sbornik* **115** 331–355. MR0217487

[8] Chen, S., Lyu, M. R., King, I. and Xu, Z. (2013). Exact and stable recovery of pairwise interaction tensors. *Advances in Neural Information Processing Systems* **26** 1691–1699.

[9] Chen, Y., Chi, Y., Fan, J. and Ma, C. (2019). Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval. *Mathematical Programming* **176** 5–37. MR3960803

[10] Diop, M., Miron, S., Larue, A. and Brie, D. (2019). Binary Matrix Factorization applied to Netflix dataset analysis. *IFAC-PapersOnLine* **52** 13–17.

[11] Erdos, D. and Miettinen, P. (2013). Discovering facts with boolean tensor tucker decomposition. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management* 1569–1572.

[12] Hong, D., Kolda, T. G. and Duersch, J. A. (2020). Generalized canonical polyadic tensor decomposition. *SIAM Review* **62** 133–163. MR4064532

[13] Hu, Q., Li, G., Wang, P., Zhang, Y. and Cheng, J. (2018). Training binary weight networks via semi-binary decomposition. In *Proceedings of the European Conference on Computer Vision (ECCV)* 637–653.

[14] Jain, P. and Oh, S. (2014). Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems* 1431–1439.

[15] Jaworski, P., Durante, F., Hardle, W. K. and Rychlik, T. (2010). *Copula theory and its applications* **198**. Springer. MR3075361

[16] Kiers, H. A. (2000). Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics: A Journal of the Chemometrics Society* **14** 105–122.

[17] Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM review* **51** 455–500. MR2535056

[18] Koren, Y. (2009). Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 447–456.

[19] Lee, C. and Wang, M. (2020). Tensor denoising and completion based on ordinal observations. In *International Conference on Machine Learning* 5778–5788. PMLR.

[20] Mažgut, J., Tiňo, P., Bodén, M. and Yan, H. (2014). Dimensionality reduction and topographic mapping of binary tensors. *Pattern Analysis and Applications* **17** 497–515. MR3227590

[21] McAuley, J., Leskovec, J. and Jurafsky, D. (2012). Learning attitudes and attributes from multi-aspect reviews. In *2012 IEEE 12th International Conference on Data Mining* 1020–1025. IEEE.

[22] Miettinen, P. (2011). Boolean tensor factorizations. In *2011 IEEE 11th International Conference on Data Mining* 447–456. IEEE.

[23] Ossiander, M. et al. (1987). A central limit theorem under metric entropy with L_2 bracketing. *Annals of Probability* **15** 897–919. MR0893905

[24] Rai, P., Hu, C., Harding, M. and Carin, L. (2015). Scalable probabilistic tensor factorization for binary and count data. In *IJCAI* 3770–3776. Citeseer.

[25] Rendle, S. (2010). Factorization machines. In *2010 IEEE International*

Conference on Data Mining 995–1000. IEEE.

[26] Rendle, S., Freudenthaler, C. and Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web* 811–820.

[27] Rendle, S. and Schmidt-Thieme, L. (2010). Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining* 81–90.

[28] Rukat, T., Holmes, C. and Yau, C. (2018). Probabilistic boolean tensor decomposition. In *International Conference on Machine Learning* 4413–4422.

[29] Salakhutdinov, R., Mnih, A. and Hinton, G. (2007). Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th International Conference on Machine Learning* 791–798.

[30] Shen, X. (1998). On the method of penalization. *Statistica Sinica* 337–357. MR1624410

[31] Shen, X. and Wong, W. H. (1994). Convergence rate of sieve estimates. *The Annals of Statistics* 580–615. MR1292531

[32] Tarzanagh, D. A. and Michailidis, G. (2019). Regularized and smooth double core tensor factorization for heterogeneous data. *arXiv preprint arXiv:1911.10454*.

[33] Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika* **31** 279–311. MR0205395

[34] Verbert, K., Manouselis, N., Ochoa, X., Wolpers, M., Drachsler, H., Bosnic, I. and Duval, E. (2012). Context-aware recommender systems for learning: a survey and future challenges. *IEEE Transactions on Learning Technologies* **5** 318–335.

[35] Wang, M. and Li, L. (2020). Learning from binary multiway data: Probabilistic tensor decomposition and its statistical optimality. *J. Mach. Learn. Res.* **21** 154–1. MR4209440

[36] Xiong, L., Chen, X., Huang, T.-K., Schneider, J. and Carbonell, J. G. (2010). Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the 2010 SIAM International Conference on Data Mining* 211–222. SIAM.

[37] Zhang, Y., Bi, X., Tang, N. and Qu, A. (2021). Dynamic tensor recommender systems. *Journal of Machine Learning Research* **22** 1–35. MR4253758

[38] Zheng, V. W., Cao, B., Zheng, Y., Xie, X. and Yang, Q. (2010). Collaborative filtering meets mobile recommendation: a user-centered approach. In *AAAI* **10** 236–241. Citeseer.