Learning Nonnegative Factors From Tensor Data: Probabilistic Modeling and Inference Algorithm

Lei Cheng, Xueke Tong, Shuai Wang, Yik-Chung Wu, and H. Vincent Poor

***Abstract*—Tensor canonical polyadic decomposition (CPD) with nonnegative factor matrices, which extracts useful latent information from multidimensional data, has found wide-spread applications in various big data analytic tasks. Currently, the implementation of most existing algorithms needs the knowledge of tensor rank. However, this information is practically unknown and difficult to acquire. To address this issue, a probabilistic approach is taken in this paper. Different from previous works, this paper firstly introduces a sparsity-promoting nonnegative Gaussian-gamma prior, based on which a novel probabilistic model for CPD problem with nonnegative and continuous factors is established. This probabilistic model further enables the derivation of an efficient inference algorithm that accurately learns the nonnegative factors from the tensor data, along with an integrated feature of automatic rank determination. Numerical results using synthetic data and real-world applications are presented to show the remarkable performance of the proposed algorithm.**

***Index Terms*—Tensor decomposition, nonnegative factors, variational inference, automatic rank determination**

I. Introduction

Modern society generates large amounts of data, many of which have multiple attributes, and tensors provide a natural representation for them. Because these data often consist of latent components, tensor canonical polyadic decomposition (CPD) [1], which is defined as a linear combination of rank-1 tensors, has been very popular and successful in achieving state-of-the-art performance for various big data mining tasks including social group clustering [2]–[4], text mining [5], [6], drug discovery [7], environmental and biomedical analysis [8], [9], and channel estimation in the next generation communications [10], [11]. Compared to the standard matrix decomposition which provides a "flat-world view", tensor CPD retains the information along all the data dimensions by virtue of its multi-linear structure, and is capable of producing unique and physically meaningful latent components both theoretically [1], [12] and empirically [2]-[11]. That is why tensor CPD has witnessed increasing popularity and adoption in applications mentioned above and beyond.

Despite the inherent uniqueness of the basic CPD, incorporating side information into the CPD model could further improve its identifiability and interpretability [13]. This has triggered flourishing recent studies on developing constrained tensor CPD algorithms for various applications [14], [15], [24], wherein tensor CPD with nonnegative factors plays an important role [16], [17]. Due to the nonnegative constraint imposed on each element of the factor matrices, the combination of latent rank-1 components only allows addition but not subtraction. This leads to a parts-based representation of the data, in the sense that each extracted latent component is a part of the data, thus further enhancing the interpretability of various data analytic results [13]–[17].

However, imposing nonnegative constraints on the CPD model is not easy, since it further complicates the originally non-convex CPD problem [1]. To tackle this, the most popular solution is the nonnegative alternating least-squares (NALS) method [16]. NALS algorithm iteratively optimizes one factor matrix at a time while holding other factor matrices fixed, and in each iteration, the optimization with respect to a single factor matrix is a nonnegative least-squares problem [16], for which recent parallel algorithms [18], [19] have been proposed. In addition to the NALS framework, recent derivative-based approaches [20], [21], which update all the nonnegative factor matrices in each iteration via first-order optimization methods, were also developed to achieve scalability.

While deriving tensor CPD algorithms with nonnegative factors is possible from a nonlinear programming perspective, their implementations require the knowledge of tensor rank. The tensor rank is defined as the smallest number of rank-one tensors that could be linearly combined to generate the original tensor [1]. Its physical meaning is the number of latent components inside the data. For example, in social group analysis [2], the tensor rank corresponds to the number of clusters of people; in the biomedical data analysis [9], the tensor rank represents how many different types of chemical species are in the given sample. This knowledge might be obtained from problem-specific domain information in some cases, but most of the time it is unknown and has to be estimated.

Unfortunately, determining the tensor rank is known to be non-deterministic polynomial-time hard (NP-hard) [1]. A common approach is to run multiple algorithms assuming different tensor ranks, and choose the best one in data interpretation.

This work was supported in part by the National Key R&D Program of China (No. 2018YFB1800800), in part by Shenzhen Peacock Plan under Grant KQTD2015033114415450, in part by the Open Research Fund from Shenzhen Research Institute of Big Data under Grant No. 2019ORF01012, in part by the General Research Fund from the Hong Kong Research Grant Council through Project 17207018, and in part by the U.S. National Science Foundation under Grant CCF-1908308.

Lei Cheng is with the Shenzhen Research Institute of Big Data, Shenzhen, Guangdong, P. R. China (e-mail: leicheng@sribd.cn).

Xueke Tong and Yik-Chung Wu are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: xktong@eee.hku.hk, ycwu@eee.hku.hk).

Shuai Wang is with the Department of Electrical and Electronic Engineering, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: wangs3@sustech.edu.cn).

H. Vincent Poor is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (email: poor@princeton.edu).

Although this manual tuning procedure is widely adopted in tensor data mining research, heavy computational burden is inevitable. To learn the tensor rank automatically, recent advances in probabilistic inference [5], [6], [22]–[26], [31] integrate the tensor rank learning into its hyper-parameter inference steps, and the Bayesian theory provides a natural recipe for automatic rank determination. However, existing probabilistic CPD models [23]–[26], [31] only considered factor matrices with no constraints or with orthogonal constraints. Despite recent works [5], [6], [22] have considered the tensor CPD with nonnegative constraints, their algorithms are tailored to handle count-valued data and factor matrices, which frequently occurs in text mining and link prediction. However, continuous data and factor matrices are prevalent in many applications such as biomedical data analysis [8], [9] and Gaussian noise corrupted image/video/speech processing [23], [24], [36]. Even some datasets are originally count-valued, any pre-processing procedure such as normalization and centralization would destroy the discrete nature [2]–[4]. Therefore, it is essential to design a probabilistic tensor CPD model that is tailored to continuous data and incorporating nonnegative constraints on its continuous factor matrices.

Designing a probabilistic model is an art. It needs to trade off the expressive power of the model and the tractability of the inference algorithm. A good model should be flexible enough to incorporate information of the problem while simple enough to enable efficient inference algorithm, and this forms the core in modern research of probabilistic inference [22]–[33], [38]–[40], [49]–[51]. In previous probabilistic CPD models [23]–[26], the Gaussian-gamma prior distribution is used as the primary building block due to its appealing sparsity-promoting and exponentially conjugacy property, which has enabled the automatic relevance determination in relevance vector machine [27], [28] and low-rank matrix decomposition [29], [30]. However, it is with an unbounded support and thus cannot model non-negativeness. In this paper, we inspect its nonnegative variant, i.e., the nonnegative Gaussian-gamma prior distribution, and show that it inherits all the desired properties of the Gaussian-gamma prior distribution. Using the nonnegative Gaussian-gamma prior, a novel probabilistic CPD model with explicit nonnegative constraints on its continuous factor matrices is proposed. Then, under the framework of variational inference [32], [33], an efficient inference algorithm with no high-dimensional multiple integration is developed. The resultant algorithm, which includes the NALS algorithm as a special case, is convergence guaranteed and is very flexible in incorporating the most recent advances in optimization [18], [41] for scalability improvement.

The remainder of this paper is organized as follows. Section II presents the formulation for the CPD problem with nonnegative factors and the challenges ahead. In Section III, a probabilistic CPD model with nonnegative factors is established, based on which an efficient inference algorithm is derived in Section IV. Numerical results with synthetic data and real-world data are reported in Section V. Finally, conclusions are drawn in Section VI.

Notation: Boldface lowercase and uppercase letters will be used for vectors and matrices, respectively. Tensors are written as calligraphic letters. $\mathbb{E}[\,\cdot\,]$ denotes the expectation of its argument. Superscript T denotes transpose, and the operator $\mathrm{Tr}\,(\boldsymbol{A})$ denotes the trace of a matrix $\boldsymbol{A}$. $\|\cdot\|_F$ represents the Frobenius norm of the argument. $\mathcal{N}(\boldsymbol{x}|\boldsymbol{u},\boldsymbol{R})$ stands for the probability density function of a Gaussian vector $\boldsymbol{x}$ with mean $\boldsymbol{u}$ and covariance matrix $\boldsymbol{R}$. The $N\times N$ diagonal matrix with diagonal components a_1 through a_N is represented as $\mathrm{diag}\{a_1,a_2,...,a_N\}$, while $\boldsymbol{I}_M$ represents the $M\times M$ identity matrix. The $(i,j)^{th}$ element, the i^{th} row, and the j^{th} column of a matrix $\boldsymbol{A}$ are represented by $\boldsymbol{A}_{i,j}$, $\boldsymbol{A}_{i,:}$ and $\boldsymbol{A}_{:,j}$, respectively.

II. CPD WITH NONNEGATIVE FACTORS AND CHALLENGES IN RANK ESTIMATION

Tensor CPD with nonnegative factors has found applications in many different fields [13]–[17]. For illustration, fluorescence data analysis [34], [35] and e-mail data mining applications [2]–[4] are presented in Appendix A. The general problem, which decomposes a N dimensional tensor $\mathcal{Y}\in\mathbb{R}^{J_1\times J_2\times\ldots\times J_N}$ into a set of nonnegative factor matrices $\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N$, is formulated as:

$$\begin{aligned}\min_{\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N}\ &\|\,\mathcal{Y}-\underbrace{\sum_{r=1}^{R}\boldsymbol{\Xi}^{(1)}_{:,r}\circ\boldsymbol{\Xi}^{(2)}_{:,r}\circ\cdots\circ\boldsymbol{\Xi}^{(N)}_{:,r}}_{\triangleq[\![\boldsymbol{\Xi}^{(1)},\boldsymbol{\Xi}^{(2)},\cdots,\boldsymbol{\Xi}^{(N)}]\!]}\|_F^2\\ \text{s.t.}\ \ &\boldsymbol{\Xi}^{(n)}\geq\boldsymbol{0}_{J_n\times R},\quad n=1,2,...,N,\end{aligned}\tag{1}$$

where symbol $\circ$ denotes vector outer product.

In problem (1), there are two significant challenges. Firstly, nonnegative factor matrices $\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N$ are complicatedly coupled, resulting in a difficult non-convex optimization problem. To tackle this challenge, alternating optimization is one of the most commonly used techniques. In each iteration, after fixing all but one factor matrices, problem (1) will become a standard nonlinear least-square (LS) subproblem, for which there are various off-the-shelf algorithms for solving it [42], including interior point methods and augmented Lagrangian methods. To scale the solution of each subproblem to handle big tensor data, first-order methods, such as Nesterov-type projected gradient descent [44, page 81 and 90], has been proposed to replace the interior point methods in [18], [19].

Although pioneering works [18], [19] allow the learning of nonnegative factors from big multidimensional data, they still face the second critical challenge of problem (1): how to automatically learn the tensor rank R from the data? With the physical meaning of tensor rank being the number of components/groups inside the data (see Appendix A), this value is usually unknown in practice and its estimation has been shown to be NP-hard [1]. For tensor CPD with no constraints on at least one factor matrix [23]–[26], this problem has been well solved via a probabilistic inference approach by the employment of the Gaussian-gamma prior distribution. In particular, without loss of generality, consider a machine learning model with parameter $\boldsymbol{w}\in\mathbb{R}^{M\times 1}$. The model parameter $\boldsymbol{w}$ consists of S non-overlapped blocks, each of

which is denoted as $\boldsymbol{w}_s \in \mathbb{R}^{M_s \times 1}$. The Gaussian-gamma prior can be expressed as [27]–[30]:

$$p(\boldsymbol{w}|\{\alpha_s\}_{s=1}^S) = \prod_{s=1}^S p(\boldsymbol{w}_s|\alpha_s) = \prod_{s=1}^S \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{0}_{M_s\times 1}, \alpha_s^{-1}\boldsymbol{I}_{M_s}), \tag{2}$$

$$p(\{\alpha_s\}_{s=1}^S) = \prod_{s=1}^S \text{gamma}(\alpha_s|a_s, b_s), \tag{3}$$

where α_s is the precision parameter (i.e., the inverse of variance, also called weight decay rate) that controls the relevance of model block $\boldsymbol{w}_s$ in data interpretation, and $\{a_s, b_s\}_{s=1}^S$ are pre-determined hyper-parameters. There are two important properties of Gaussian-gamma prior that leads to its success and prevalence in a variety of applications [23]–[30], [49]–[51]. Firstly, after integrating the gamma hyper-prior, the marginal distribution of model parameter $p(\boldsymbol{w})$ is a student's t distribution, which is strongly peaked at zero and with heavy tails, thus promoting sparsity. Secondly, the gamma hyper-prior (3) is conjugate[1] to the Gaussian prior (2). This conjugacy permits the closed-form solution of the variational inference [32], [33], which has recently come up as a major tool in inferring complicated probabilistic models with inexpensive computations.

However, the Gaussian-gamma prior in (2) and (3) cannot be used in the CPD problem with nonnegative factors, since the support of Gaussian probability density function (pdf) in (2) is not restricted to the nonnegative region. This calls for a different prior distribution modeling. An immediate idea might be to replace the Gaussian distribution in Gaussian-gamma prior by the truncated Gaussian distribution with a nonnegative support. However, a closer inspection is needed since there is no existing work discussing whether a gamma distribution is conjugate to a truncated Gaussian distribution with a nonnegative support (see Appendix B for related discussions).

III. Probabilistic modeling for CPD with nonnegative Factors

A. Properties of nonnegative Gaussian-gamma prior

The support of the Gaussian pdf in (2) is unbounded, and thus cannot model non-negativeness. On the other hand, the truncated Gaussian prior with a nonnegative support for each model block $\boldsymbol{w}_s$ can be written as:

$$\begin{aligned} p^+(\boldsymbol{w}|\{\alpha_s\}_{s=1}^S) &= \prod_{s=1}^S p^+(\boldsymbol{w}_s|\alpha_s) \\ &= \prod_{s=1}^S \frac{\mathcal{N}(\boldsymbol{w}_s|\boldsymbol{0}_{M_s\times 1}, \alpha_s^{-1}\boldsymbol{I}_{M_s})}{\int_{\boldsymbol{0}_{M_s\times 1}}^{\infty} \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{0}_{M_s\times 1}, \alpha_s^{-1}\boldsymbol{I}_{M_s})} \mathrm{U}\left(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}\right), \end{aligned} \tag{4}$$

where the function $\mathrm{U}\left(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}\right)$ is a unit-step function with value one when $\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}$, and with value zero otherwise. Together with the gamma distributions (3) for modeling the precision parameters $\{\alpha_s\}_{s=1}^S$, we have the nonnegative Gaussian-gamma prior. Even though it is clear that nonnegative Gaussian-gamma prior can model the non-negativeness of model parameters due to the unit-step function $\mathrm{U}\left(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}\right)$, whether it enjoys the advantages of the vanilla Gaussian-gamma prior needs further inspection. In the following, two properties of the nonnegative Gaussian-gamma prior are presented.

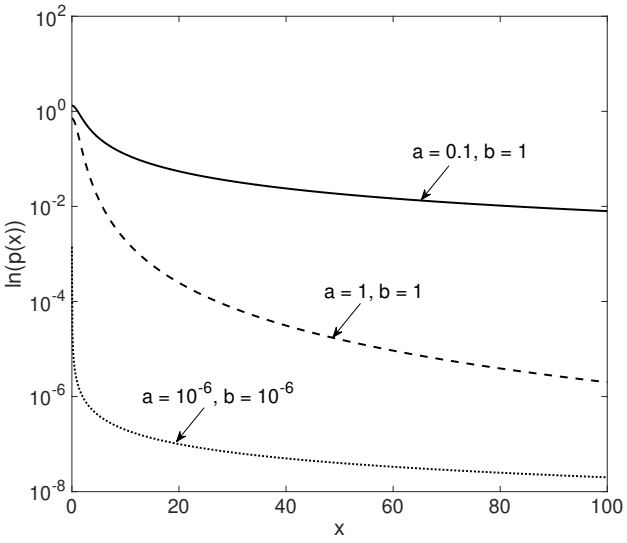


Figure 1: Univariate marginal probability density functions in (5) with different parameters

Property 1. The gamma distribution in (3) is conjugate to the nonnegative Gaussian distribution in (4).

Proof: See Appendix B. ∎

Property 2. After integrating out the precision parameters $\{\alpha_s\}_{s=1}^S$, the marginal pdf of model parameter $\boldsymbol{w}$ is

$$\begin{aligned} p^+(\boldsymbol{w}) &= \int p^+(\boldsymbol{w}|\{\alpha_s\}_{s=1}^S) p(\{\alpha_s\}_{s=1}^S) d\{\alpha_s\}_{s=1}^S \\ &= \prod_{s=1}^S 2^{M_s} \left(\frac{1}{\pi}\right)^{\frac{M_s}{2}} \frac{\Gamma(a_s + M_s/2)}{(2b_s)^{-a_s}\Gamma(a_s)} (2b_s + \boldsymbol{w}_s^T\boldsymbol{w}_s)^{-a_s - M_s/2} \\ &\quad \times \mathrm{U}\left(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}\right). \end{aligned} \tag{5}$$

It is a product of multivariate truncated student's t distributions, each of which is with a nonnegative support.

Proof: See Appendix B. ∎

The shape of the marginal distribution $p^+(\boldsymbol{w})$ is determined by the hyper-parameters $\{a_s, b_s\}_{s=1}^S$. Usually, their values are chosen to be a very small value (e.g., $\epsilon = 10^{-6}$), since as $a_s \to 0$ and $b_s \to 0$, a Jeffrey's non-informative prior $p(\alpha_s) \propto \alpha_s^{-1}$ [37] can be obtained. After letting the hyper-parameters $\{a_s, b_s\}_{s=1}^S$ in (5) go to zero, it is easy to obtain the following property.

Property 3. If $a_s \to 0$ and $b_s \to 0$, the marginal pdf of the model parameter $\boldsymbol{w}$ becomes

$$p^+(\boldsymbol{w}) \propto \prod_{s=1}^S 2^{M_s} \left(\frac{1}{||\boldsymbol{w}_s||_2}\right)^{M_s} \mathrm{U}\left(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}\right), \tag{6}$$

[1] In Bayesian theory, a probability density function (pdf) $p(x)$ is said to be conjugate to a conditional pdf $p(y|x)$ if the resulting posterior pdf $p(x|y)$ is in the same distribution family as $p(x)$.

which is highly peaked at zeros.

As an illustration for **Property 2 and 3**, univariate marginal pdfs with different hyper-parameters $\{a_s, b_s\}_{s=1}^S$ are plotted in Figure 1, from which it is clear that the nonnegative Gaussian-gamma prior is sparsity-promoting. Further with the conjugacy property revealed in **Property 1**, the nonnegative Gaussian-gamma prior is a good candidate for probabilistic modeling with nonnegative model parameters.

B. Probabilistic modeling of CPD with nonnegative factors

In the CPD problem with nonnegative factors in (1), the l^{th} column group $\{\boldsymbol{\Xi}_{:,l}^{(n)}\}_{n=1}^N$, which consists of the l^{th} column of all the factor matrices, can be treated as a model block, since their outer product contributes a rank-1 tensor. Therefore, with the principle of nonnegative Gaussian-gamma prior in the previous subsection, the l^{th} column group $\{\boldsymbol{\Xi}_{:,l}^{(n)}\}_{n=1}^N$ can be modeled using (4), but with $\boldsymbol{w}_s$ replaced by $\{\boldsymbol{\Xi}_{:,l}^{(n)}\}_{n=1}^N$ and α_s replaced by γ_l. Considering the independence among different column groups in $\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N$, we have

$$\begin{aligned} &p(\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N|\{\gamma_l\}_{l=1}^L) \\ &= \prod_{n=1}^N\prod_{l=1}^L \frac{\mathcal{N}\left(\boldsymbol{\Xi}_{:,l}^{(n)}|\mathbf{0}_{J_n\times 1}, \gamma_l^{-1}\boldsymbol{I}_L\right)}{\int_{\mathbf{0}_{J_n\times 1}}^{\infty}\mathcal{N}\left(\boldsymbol{\Xi}_{:,l}^{(n)}|\mathbf{0}_{J_n\times 1}, \gamma_l^{-1}\boldsymbol{I}_L\right)d\boldsymbol{\Xi}_{:,l}^{(n)}} \\ &\qquad \times \mathrm{U}\left(\boldsymbol{\Xi}_{:,l}^{(n)} \geq \mathbf{0}_{J_n\times 1}\right), \end{aligned} \tag{7}$$

$$p(\{\gamma_l\}_{l=1}^L|\boldsymbol{\lambda}_{\boldsymbol{\gamma}}) = \prod_{l=1}^L \mathrm{gamma}(\gamma_l|c_l^0, d_l^0), \tag{8}$$

where the precision γ_l is modeled as a gamma distribution. From discussions below **Property 2**, c_l^0 and d_l^0 can be chosen to be a very small value (e.g., $\epsilon = 10^{-6}$) to approach a non-informative prior of precision parameter γ_l.

On the other hand, the least-square objective function in the nonnegative tensor CPD problem (1) motivates the use of a Gaussian likelihood [27]–[30]:

$$\begin{aligned} &p\left(\mathcal{Y} \mid \boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, ..., \boldsymbol{\Xi}^{(N)}, \beta\right) \\ &\propto \exp\left(-\frac{\beta}{2} \parallel \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, ..., \boldsymbol{\Xi}^{(N)}]\!] \parallel_F^2\right), \end{aligned} \tag{9}$$

in which the parameter β represents the inverse of noise power. Since there is no information about it, a gamma distribution $p(\beta|\boldsymbol{\alpha}_\beta) = \mathrm{gamma}(\beta|\epsilon, \epsilon)$ with very small ϵ is employed, making $p(\beta|\boldsymbol{\alpha}_\beta)$ approache Jeffrey's non-informative prior.

The Gaussian likelihood function in (9) is with an unbounded support over the real space, and thus it is suitable for applications such as fluorescence data analysis [34], [35] and blind speech separation [36], in which the observed data $\mathcal{Y}$ could be both positive and negative [34]–[36]. On the other hand, if the data $\mathcal{Y}$ are all nonnegative and continuous (e.g., the email dataset [2]–[4] after pre-processing), a truncated Gaussian likelihood can be used to model the data:

$$\begin{aligned} &p\left(\mathcal{Y} \mid \boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, ..., \boldsymbol{\Xi}^{(N)}, \beta\right) \\ &\propto \exp\left(-\frac{\beta}{2} \parallel \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, ..., \boldsymbol{\Xi}^{(N)}]\!] \parallel_F^2\right) \mathrm{U}(\mathcal{Y} \geq 0). \end{aligned} \tag{10}$$

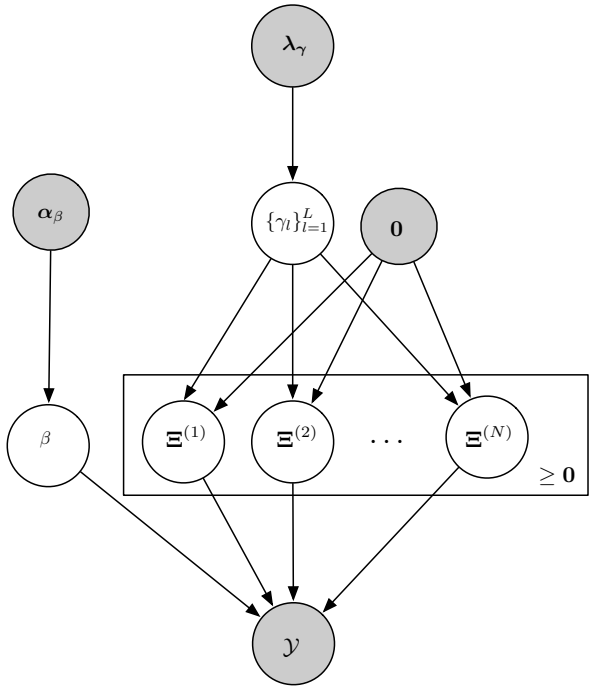


Figure 2: Probabilistic model for tensor CPD with nonnegative factors

Finally, the complete probabilistic model is a three-layer Bayes network and is illustrated in Figure 2.

Remark 1: For applications in Appendix A (i.e., fluorescence spectroscopy and social group clustering), the desired factor matrices are with continuous and nonnegative elements, and thus the prior distribution should have a nonnegative support. On other other hand, in order to automatically identify the inherent component/cluster number, the prior distribution needs to enjoy the sparsity promoting property. Therefore, the choice of nonnegative Gaussian pdf together with a gamma hyper-prior suits these applications.

IV. Inference algorithm for Tensor CPD with nonnegative factors

The unknown parameter set $\boldsymbol{\Theta}$ includes the factor matrices $\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^N$, the noise power β^{-1} and the precision parameter $\{\gamma_l\}_{l=1}^L$. The aim of Bayesian inference is to infer the posterior distribution $p(\boldsymbol{\Theta}|\mathcal{Y}) = p(\boldsymbol{\Theta}, \mathcal{Y})/\int p(\boldsymbol{\Theta}, \mathcal{Y})d\boldsymbol{\Theta}$. However, the proposed probabilistic model is too complicated to enable an analytical solution since multiple integrations are involved. To tackle this, variational inference [32], [33] has recently been widely used for inferring parameters of a complicated probabilistic model. The key idea is to approximate the true posterior distribution by a variational pdf $Q(\boldsymbol{\Theta})$ that minimizes the Kullback-Leibler (KL) divergence $\mathrm{KL}\big(Q\left(\boldsymbol{\Theta}\right) \parallel p\left(\boldsymbol{\Theta} \mid \mathcal{Y}\right)\big) \triangleq -\mathbb{E}_{Q(\boldsymbol{\Theta})}\left\{\ln \frac{p(\boldsymbol{\Theta}|\mathcal{Y})}{Q(\boldsymbol{\Theta})}\right\}$, thus recasting the probabilistic inference problem into a functional optimization problem. To facilitate the optimization, the variational pdf $Q(\boldsymbol{\Theta})$ is usually restricted to the mean-field family $Q(\boldsymbol{\Theta}) = \prod_{k=1}^K Q(\boldsymbol{\Theta}_k)$, where $\boldsymbol{\Theta}$ is partitioned into mutually disjoint non-empty subsets $\boldsymbol{\Theta}_k$ (i.e., $\boldsymbol{\Theta}_k$ is a part of $\boldsymbol{\Theta}$

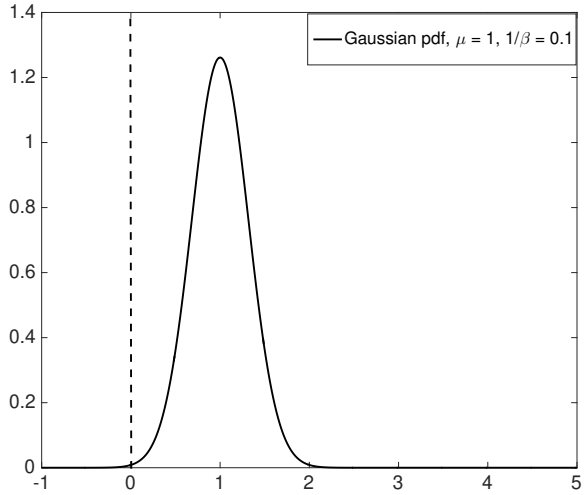


Figure 3: Illustration of a univariate Gaussian probability density function with its mean much larger than the variance

with $\cup_{k=1}^{K}\boldsymbol{\Theta}_k = \boldsymbol{\Theta}$ and $\cap_{k=1}^{K}\boldsymbol{\Theta}_k = \emptyset$). Under the mean-field assumption, each optimal variational pdf $Q^*(\boldsymbol{\Theta}_k)$ that minimizes the KL divergence is obtained by solving the following problem with other $\{Q(\boldsymbol{\Theta}_j)\}_{j\neq k}$ fixed [32]:

$$\min_{Q(\boldsymbol{\Theta}_k)} \int Q(\boldsymbol{\Theta}_k)(-\mathbb{E}_{\prod_{j\neq k} Q(\boldsymbol{\Theta}_j)}\left[\ln p(\boldsymbol{\Theta},\mathcal{Y})\right] + \ln Q(\boldsymbol{\Theta}_k))d\boldsymbol{\Theta}_k. \tag{11}$$

For this convex problem, the Karush-Kuhn-Tucker (KKT) condition gives the optimal solution as [32, page 132]:

$$Q^*(\boldsymbol{\Theta}_k) = \frac{\exp\left(\mathbb{E}_{\prod_{j\neq k} Q(\boldsymbol{\Theta}_j)}\left[\ln p(\boldsymbol{\Theta},\mathcal{Y})\right]\right)}{\int \exp\left(\mathbb{E}_{\prod_{j\neq k} Q(\boldsymbol{\Theta}_j)}\left[\ln p(\boldsymbol{\Theta},\mathcal{Y})\right]\right) d\boldsymbol{\Theta}_k}. \tag{12}$$

Nevertheless, even under the mean-field family assumption, the unknown parameter $\boldsymbol{\Xi}^{(k)}$ is still difficult to be inferred since its moments cannot be easily computed. In particular, in the proposed probabilistic model, if no functional assumption is made for variational pdf $Q(\boldsymbol{\Xi}^{(k)})$, after using (12), a multivariate truncated Gaussian distribution would be obtained, of which the moments are known to be very difficult to be computed due to the multiple integrations involved [46]. In this case, the variational pdf $Q\left(\boldsymbol{\Xi}^{(k)}\right)$ could be further restricted to be a Dirac delta function $Q\left(\boldsymbol{\Xi}^{(k)}\right) = \delta(\boldsymbol{\Xi}^{(k)} - \hat{\boldsymbol{\Xi}}^{(k)})$, where $\hat{\boldsymbol{\Xi}}^{(k)}$ is the point estimate of the parameter $\boldsymbol{\Xi}^{(k)}$. After substituting this functional form into problem (11), the optimal point estimate $\hat{\boldsymbol{\Xi}}^{(k)*}$ is obtained by [32, page 164]:

$$\hat{\boldsymbol{\Xi}}^{(k)*} = \arg\max \mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\ln p(\boldsymbol{\Theta},\mathcal{Y})\right]. \tag{13}$$

This is indeed the framework of variational expectation maximization (EM), in which the factor matrices $\{\boldsymbol{\Xi}^{(k)}\}_{k=1}^{N}$ are treated as global parameters and other variables are treated as latent variables.

In (12) and (13), the log of the joint pdf $\ln p(\boldsymbol{\Theta},\mathcal{Y})$ needs to be evaluated. If the Gaussian likelihood function (9) is adopted, it is expressed as

$$\begin{aligned}\ln p^{\dagger}(\boldsymbol{\Theta},\mathcal{Y}) &= \sum_{n=1}^{N}\ln\left(\mathrm{U}(\boldsymbol{\Xi}^{(n)} \geq \mathbf{0}_{J_n\times L})\right) + \frac{\prod_{n=1}^{N} J_n}{2}\ln\beta \\ &- \frac{\beta}{2}\parallel \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)},\boldsymbol{\Xi}^{(2)},\cdots,\boldsymbol{\Xi}^{(N)}]\!]\parallel_F^2 + \sum_{n=1}^{N}\frac{J_n}{2}\sum_{l=1}^{L}\ln\gamma_l \\ &- \sum_{n=1}^{N}\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{\Xi}^{(n)}\boldsymbol{\Gamma}\boldsymbol{\Xi}^{(n)T}\right) + \sum_{l=1}^{L}\left[(10^{-6}-1)\ln\gamma_l - 10^{-6}\gamma_l\right] \\ &+ (10^{-6}-1)\ln\beta - 10^{-6}\beta + \mathrm{const},\end{aligned} \tag{14}$$

where the term $\boldsymbol{\Gamma} = \mathrm{diag}\{\gamma_1,\gamma_2,...,\gamma_L\}$. On the other hand, if the truncated Gaussian likelihood (10) is used, the log of the joint pdf $\ln p^{\sharp}(\boldsymbol{\Theta},\mathcal{Y})$ takes the following form

$$\ln p^{\sharp}(\boldsymbol{\Theta},\mathcal{Y}) = \ln p^{\dagger}(\boldsymbol{\Theta},\mathcal{Y}) - \ln\Phi\left(\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^{N},\beta\right) + \mathrm{const}. \tag{15}$$

where

$$\begin{aligned}\Phi\left(\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^{N},\beta\right) =& \int_{\mathbf{0}}^{\infty}\left(\frac{\beta}{2\pi}\right)^{\prod_{n=1}^{N}\frac{J_n}{2}} \\ &\times\exp\left(\frac{-\beta}{2}\parallel \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)},\boldsymbol{\Xi}^{(2)},...,\boldsymbol{\Xi}^{(N)}]\!]\parallel_F^2\right)d\mathcal{Y}.\end{aligned} \tag{16}$$

In (15), the term $\ln\Phi\left(\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^{N},\beta\right)$, which arises from the truncated Gaussian likelihood in (10), is very difficult to evaluate and differentiate, due to the multiple integrations involved. Fortunately, for most applications in Bayesian nonnegative matrix/tensor decomposition [38]–[40], the confidence of the low-rank matrix/tensor model is relatively high, in the sense that the noise power $1/\beta$ is small compared to average element in signal tensor $[\![\boldsymbol{\Xi}^{(1)},\boldsymbol{\Xi}^{(2)},...,\boldsymbol{\Xi}^{(N)}]\!]$. Under this assumption, it is easy to see $\ln\Phi\left(\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^{N},\beta\right) \approx \ln 1 = 0$, since Gaussian pdf $p(\mathcal{Y}) = \left(\frac{\beta}{2\pi}\right)^{\frac{\prod_n J_n}{2}}\exp\left(-\beta\parallel \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)},\boldsymbol{\Xi}^{(2)},...,\boldsymbol{\Xi}^{(N)}]\!]\parallel_F^2\right)d\mathcal{Y}$ decays very rapidly and thus most densities are over the region $\mathcal{Y} \geq 0$. As an illustration, a univariate Gaussian pdf with its mean much larger than the variance is plotted in Figure 3, in which the probability density in the negative region is negligible. This suggests that the the log of the joint pdf $\ln p^{\sharp}(\boldsymbol{\Theta},\mathcal{Y})$ in (15) can be well approximated by $\ln p^{\dagger}(\boldsymbol{\Theta},\mathcal{Y})$ in (14), and therefore algorithm derivations are unified for the two likelihoods. This also explains why the previous Bayesian nonnegative matrix/tensor decompositions [38]–[40] all employ the Gaussian likelihood function.

A. Derivation for variational pdfs

As discussed in the first paragraph of this section, the mean-field approximation is employed to enable closed-form expression for each variational pdf. For the precision parameter γ_l, by substituting (14) into (12) and only taking the terms relevant to γ_l, the variational pdf $Q(\gamma_l)$ can be found to take

the same functional form as that of the gamma distribution, i.e., $Q(\gamma_l) = \text{gamma}(\gamma_l|c_l, d_l)$ with

$$c_l = \sum_{n=1}^{N} \frac{J_n}{2} + \epsilon, \tag{17}$$

$$d_l = \sum_{n=1}^{N} \frac{1}{2}\mathbb{E}_{Q(\boldsymbol{\Xi}^{(n)})}\left[\boldsymbol{\Xi}_{:,l}^{(n)T}\boldsymbol{\Xi}_{:,l}^{(n)}\right] + \epsilon. \tag{18}$$

Since the variational pdf $Q(\gamma_l)$ is determined by parameters c_l and d_l, its update is equivalent to the update of the two parameters in (17) and (18).

Similarly, using (12) and (14), the variational pdf $Q(\beta)$ can be found to be a gamma distribution $Q(\beta) = \text{gamma}(\beta|e, f)$, where

$$e = \frac{\prod_{n=1}^{N} J_n}{2} + \epsilon, \tag{19}$$

$$f = \frac{1}{2}\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \beta} Q(\boldsymbol{\Theta}_j)}\left[\| \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, \cdots, \boldsymbol{\Xi}^{(N)}]\!]\|_F^2\right] + \epsilon. \tag{20}$$

On the other hand, by substituting (14) into (13), the point estimate of $\hat{\boldsymbol{\Xi}}^{(k)}$ can be obtained via solving the following problem:

$$\max \mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\Bigg[\ln\Big(\mathrm{U}(\boldsymbol{\Xi}^{(k)} \geq \mathbf{0}_{J_k \times L})\Big) - \frac{\beta}{2}\| \mathcal{Y} - [\![\boldsymbol{\Xi}^{(1)}, \boldsymbol{\Xi}^{(2)}, \cdots, \boldsymbol{\Xi}^{(N)}]\!]\|_F^2 - \frac{1}{2}\mathrm{Tr}\Big(\boldsymbol{\Xi}^{(k)}\boldsymbol{\Gamma}\boldsymbol{\Xi}^{(k)T}\Big)\Bigg]. \tag{21}$$

After distributing the expectations, expanding the Frobenius norm, and utilizing the fact that $\ln(0) = -\infty$, problem (21) can be equivalently shown to be:

$$\begin{aligned} &\min f(\boldsymbol{\Xi}^{(k)}) \\ &\text{s.t.} \quad \boldsymbol{\Xi}^{(k)} \geq \mathbf{0}_{J_k \times L}, \end{aligned} \tag{22}$$

where

$$\begin{aligned} &f(\boldsymbol{\Xi}^{(k)}) \\ &= \frac{1}{2}\mathrm{Tr}\Big(\boldsymbol{\Xi}^{(k)}\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\beta\mathfrak{B}^{(k)}\mathfrak{B}^{(k)T} + \boldsymbol{\Gamma}\right]\boldsymbol{\Xi}^{(k)T} \\ &- 2\beta\boldsymbol{\Xi}^{(k)}\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\mathfrak{B}^{(k)}\right]\mathcal{Y}^{(k)T}\Big). \end{aligned} \tag{23}$$

In (23), the term $\mathfrak{B}^{(k)} = \Big(\overset{N}{\underset{n=1,n\neq k}{\diamond}} \boldsymbol{\Xi}^{(n)}\Big)^T$, with the multiple Khatri-Rao products $\overset{N}{\underset{n=1,n\neq k}{\diamond}} \boldsymbol{A}^{(n)} = \boldsymbol{A}^{(N)} \diamond \boldsymbol{A}^{(N-1)} \diamond \cdots \diamond \boldsymbol{A}^{(k+1)} \diamond \boldsymbol{A}^{(k-1)} \diamond \cdots \diamond \boldsymbol{A}^{(1)}$. It is easy to see that problem (22) is a quadratic programming (QP) problem with nonnegative constraints. Since each diagonal element γ_l in the diagonal matrix $\boldsymbol{\Gamma}$ is larger than zero, the Hessian matrix of the function $f(\boldsymbol{\Xi}^{(k)})$, with the expression being

$$\boldsymbol{H}^{(k)} = \mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\beta\mathfrak{B}^{(k)}\mathfrak{B}^{(k)T} + \boldsymbol{\Gamma}\right] \tag{24}$$

is positive definite. This implies that problem (22) is a convex problem, and its solutions have been investigated for decades [42, Chapter 2 and 4]. In particular, first-order methods have recently received much attention due to their scalability in big data applications. Within the class of first-order methods, a simple gradient projection method is introduced as follows.

In each iteration of the gradient projection method, the update equation is of the form [42, page 224]:

$$\boldsymbol{\Xi}^{(k,t+1)} = \left[\boldsymbol{\Xi}^{(k,t)} - \alpha_t \nabla f(\boldsymbol{\Xi}^{(k,t)})\right]_+, \tag{25}$$

where the gradient $\nabla f(\boldsymbol{\Xi}^{(k,t)})$ is computed as

$$\begin{aligned} \nabla f(\boldsymbol{\Xi}^{(k,t)}) &= \boldsymbol{\Xi}^{(k,t)}\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\beta\mathfrak{B}^{(k)}\mathfrak{B}^{(k)T} + \boldsymbol{\Gamma}\right] \\ &- \mathcal{Y}^{(k)}\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\beta\mathfrak{B}^{(k)T}\right]. \end{aligned} \tag{26}$$

In (26), the symbol $[\cdot]_+$ denotes projecting each element of $\boldsymbol{\Xi}^{(k)}$ to $[0, \infty)$ (i.e., $[x]_+ = x$ if $x \geq 0$ and $[x]_+ = 0$ otherwise) and $\alpha_t \geq 0$ is the step size. During the inference, due to the sparsity-promoting property of the nonnegative Gaussian-gamma prior, some of the precision parameters will go to very large numbers while some of them will tend to be zero. This will result in a very large condition number of the Hessian matrix $\boldsymbol{H}^{(k)}$. In this case, applying the diminishing rule[2] to the step size α_t still enjoys a good convergence performance [42, page 228] and thus is set as the default step-size rule in the proposed algorithm.

B. Summary of the inference algorithm

From equations (17)-(26), it can be seen that we need to compute various expectations. In particular, for expectations $\mathbb{E}_{Q(\boldsymbol{\Xi}^{(n)})}[\boldsymbol{\Xi}^{(n)}]$, $\mathbb{E}_{Q(\gamma_l)}[\gamma_l]$ and $\mathbb{E}_{Q(\beta)}[\beta]$, their computations are very straightforward, i.e., $\mathbb{E}_{Q(\boldsymbol{\Xi}^{(n)})}[\boldsymbol{\Xi}^{(n)}] = \hat{\boldsymbol{\Xi}}^{(n)}$, $\mathbb{E}_{Q(\gamma_l)}[\gamma_l] = \frac{c_l}{d_l}$ and $\mathbb{E}_{Q(\beta)}[\beta] = \frac{e}{f}$. However, when updating $\hat{\boldsymbol{\Xi}}^{(n)}$ using (22) and (23), there is one complicated expectation $\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\mathfrak{B}^{(k)}\mathfrak{B}^{(k)T}\right]$. Fortunately, it can be shown that $\mathbb{E}_{\prod_{\boldsymbol{\Theta}_j \neq \boldsymbol{\Xi}^{(k)}} Q(\boldsymbol{\Theta}_j)}\left[\mathfrak{B}^{(k)}\mathfrak{B}^{(k)T}\right] = \overset{N}{\underset{n=1,n\neq k}{\odot}} \hat{\boldsymbol{\Xi}}^{(n)T}\hat{\boldsymbol{\Xi}}^{(n)}$, where the multiple Hadamard products $\overset{N}{\underset{n=1,n\neq k}{\odot}} \boldsymbol{A}^{(n)} = \boldsymbol{A}^{(N)} \odot \boldsymbol{A}^{(N-1)} \odot \cdots \odot \boldsymbol{A}^{(k+1)} \odot \boldsymbol{A}^{(k-1)} \odot \cdots \odot \boldsymbol{A}^{(1)}$. Since the computation of one variational pdf needs the statistics of other variational pdfs, alternating update is needed, resulting in the iterative algorithm summarized in **Algorithm 1**.

C. Discussions and insights

To gain further insight from the proposed inference algorithm, discussions of its convergence property, automatic rank determination, relationship to the NALS algorithm, computational complexity, and scalability improvement are presented in the following.

1) Convergence property: The proposed algorithm is derived under the framework of mean-field variational inference, where a variational pdf $Q(\boldsymbol{\Theta}) = \prod_k Q(\boldsymbol{\Theta}_k)$ is sought that minimizes the KL divergence $\mathrm{KL}\left(Q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta}|\mathcal{Y})\right)$. Even though this problem is known to be non-convex due to the non-convexity of the mean-field family set, it is convex with respect

[2]In the diminishing rule [42, page 227], the step size α_t needs to satisfy $\alpha_t \to 0$ and $\sum_{t=0}^{\infty} \alpha_t = \infty$.

Algorithm 1 Probabilistic Tensor CPD with Nonnegative Factors

Initializations: Choose $L > R$ and initial values $\{\hat{\boldsymbol{\Xi}}^{(n,0)}\}_{n=1}^{N}$, ϵ.

Iterations: For the s^{th} iteration ($s \geq 0$),

Update the parameter of $Q(\boldsymbol{\Xi}^{(k)})^{(s+1)}$

Set initial value $\boldsymbol{\Xi}^{(k,0)} = \hat{\boldsymbol{\Xi}}^{(k,s)}$.

Iterations: For the t^{th} iteration ($t \geq 0$), compute

$$\boldsymbol{\Xi}^{(k,t+1)} = \Big[\boldsymbol{\Xi}^{(k,t)} - \alpha_t \nabla f(\boldsymbol{\Xi}^{(k,t)})\Big]_+$$

where

$$\nabla f(\boldsymbol{\Xi}^{(k,t)}) = \boldsymbol{\Xi}^{(k,t)}\Big[\frac{e^s}{f^s} \overset{N}{\underset{n=1,n\neq k}{\odot}} \hat{\boldsymbol{\Xi}}^{(n,s)T}\hat{\boldsymbol{\Xi}}^{(n,s)} + \mathrm{diag}\{\frac{c_1^s}{d_1^s},...,\frac{c_L^s}{d_L^s}\}\Big] - \frac{e^s}{f^s}\mathcal{Y}^{(k)}\Big(\overset{N}{\underset{n=1,n\neq k}{\diamond}} \hat{\boldsymbol{\Xi}}^{(n,s)}\Big)$$

and α_t is chosen by the diminishing rule [42, page 227].

Until Convergence

Set $\hat{\boldsymbol{\Xi}}^{(k,s+1)} = \boldsymbol{\Xi}^{(k,t+1)}$.

Update the parameter of $Q(\gamma_l)^{s+1}$

$$c_l^{s+1} = \sum_{n=1}^{N}\frac{J_n}{2} + \epsilon$$

$$d_l^{s+1} = \sum_{n=1}^{N}\frac{1}{2}\hat{\boldsymbol{\Xi}}_{:,l}^{(n,s+1)T}\hat{\boldsymbol{\Xi}}_{:,l}^{(n,s+1)} + \epsilon$$

Update the parameter of $Q(\beta)^{s+1}$

$$e^{s+1} = \epsilon + \frac{\prod_{n=1}^{N} J_n}{2}$$

$$f^{s+1} = \epsilon + \frac{1}{2} \parallel \mathcal{Y} - [\![\hat{\boldsymbol{\Xi}}^{(1,s+1)}, \hat{\boldsymbol{\Xi}}^{(2,s+1)}, \cdots, \hat{\boldsymbol{\Xi}}^{(N,s+1)}]\!] \parallel_F^2$$

Until Convergence

to a single variational pdf $Q(\boldsymbol{\Theta}_k)$ after fixing other variational pdfs $\{Q(\boldsymbol{\Theta}_j), j \neq k\}$ [32, page 138]. Therefore, the iterative algorithm, in which a single variational pdf is optimized in each iteration with other variational pdfs fixed, is essentially a coordinate descent algorithm in the functional space of variational pdfs. Since the subproblem in each iteration is not only convex but also has a unique solution, the limit point generated by the coordinate descent steps over the functional space of variational pdfs is guaranteed to be at least a stationary point of the KL divergence [32, page 163].

2) Automatic rank determination: During the inference, the expectations of some precision parameters $\{\gamma_l\}$, i.e., $\{\frac{c_l^s}{d_l^s}\}$ will go to a very large value. It indicates that the corresponding columns in the factor matrices are close to zero vectors, thus playing no role in data interpretation. As a result, after convergence, those columns can be pruned out and the number of remaining columns in each factor matrix gives the estimate of tensor rank.

In practice, to reduce the computational complexity, the pruning would be executed during the iteration. In particular, in each iteration, after the precision estimates $\{\frac{c_l^s}{d_l^s}\}$ exceed a certain threshold (e.g., 10^6), the associated columns are safely pruned out. After every pruning, it is equivalent to starting minimization of the KL divergence of a new (but smaller) probabilistic model, with the current variational distributions acting as the initialization of the new minimization. Therefore, the pruning steps will not affect the convergence, and are widely used in recent related works [23]-[30], [49]-[51].

Usually, the hyper-parameters $\{c_l^0, d_l^0\}$ of the prior gamma distribution $\mathrm{gamma}(\gamma_l|c_l^0, d_l^0)$ are set to be a very small number $\epsilon = 10^{-6}$ to approach a non-informative prior. Otherwise, their values might affect the behavior of tensor rank estimate. For example, if we prefer a high value of the tensor rank, the initial value d_l^0 can be set to be very large while the initial value c_l^0 can be set to be small, so that the update of $\frac{c_l}{d_l}$ can be steered towards a small value in order to promote a high tensor rank. However, how to set the hyper-parameters to accurately control the degree of low-rank is challenging, and deserves future investigation.

3) Relationship to NALS algorithm: The mean-field variational inference for tensor CPD problem could be interpreted as alternating optimizations over the Riemannian space (in which the Euclidean space is a special case). This insight has been revealed in previous works [25], [45], and can also be found in the proposed algorithm above. For example, for the precision parameters and the noise power parameter, the variational pdfs are with no constraint on the functional form, and thus the corresponding alternating optimization is over the Riemannian space due to the exponentially conjugacy property of the proposed probabilistic model [25], [45]. On the other hand, for unknown factor matrices, since the variational pdfs to be optimized are with a delta functional form, the corresponding alternating optimization is over the Euclidean space, thus is similar the conventional NALS step. However, there is a significant difference. In the proposed algorithm, there is a shrinkage term $\boldsymbol{\Gamma}$ appeared in the Hessian matrix in (24), and $\boldsymbol{\Gamma}$ will be updated together with other parameters in the algorithm. This intricate interaction is due to the employed Bayesian framework, and cannot be revealed by NALS framework. Consequently, the proposed algorithm is a generalization of the NALS algorithm, with the additional novel feature in automatic rank determination achieved via optimization in Riemannian space.

4) Computational complexity: For each iteration, the computational complexity is dominated by computing the gradient of each factor matrix in (25), costing $O(\prod_{n=1}^{N} J_n L)$. From this expression, it is clear that the computational complexity in each iteration is linear with respect to the tensor dimension product $\prod_{n=1}^{N} J_n$. Consequently, the complexity of the algorithm is $O(q(\prod_{n=1}^{N} J_n L))$ where q is the iteration number at convergence.

5) Speeding up the algorithm via acceleration schemes and parallel computations: From the proposed inference algorithm, it is clear that the bottleneck of the algorithm efficiency is the update of factor matrices $\{\boldsymbol{\Xi}^{(n)}\}_{n=1}^{N}$ via solving problem (22). Fortunately, if the problem is well conditioned, in the sense that the condition number of the Hessian matrix $\boldsymbol{H}^{(k)}$ in (24) is smaller than a threshold (e.g., 100), acceleration schemes, including variants of the Nesterov scheme [44, page

81 and page 90], [18], can be utilized to significantly reduce the required number of iteration for solving problem (22), thus speeding up the proposed algorithm. Besides reducing the iteration number for convergence, ideas of leveraging parallel computing architecture like message passing interface (MPI) in [18], [19] and super-computer in [43] deserve future investigation. For instance, it is easy to see that the non-negatively constrained quadratic problem (22) is separable across the rows of the factor matrix $\boldsymbol{\Xi}^{(n)}$, making it possible to optimize multiple rows of the factor matrices in parallel. More sophisticated schemes on parallel computations could be found in [18], [19], [43].

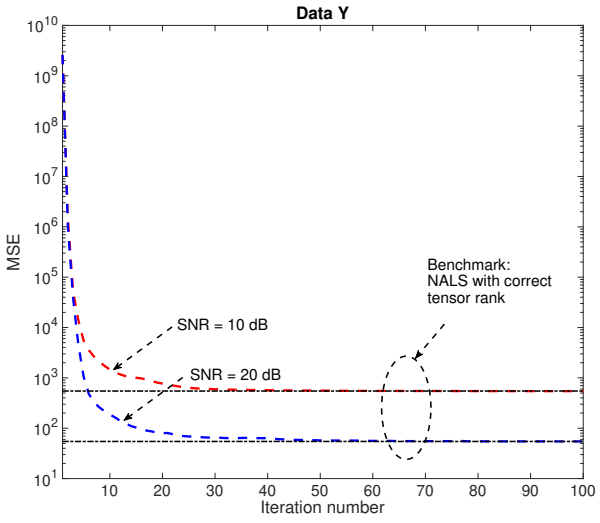
V. Numerical results

In this section, numerical results using synthetic data are firstly presented to assess the performance of the proposed algorithm in terms of convergence property, factor matrix recovery, tensor rank estimation and running time. Next, the proposed algorithm is utilized to analyze two real-world data sets (the amino acids fluorescence data and the ENRON email corpus), for demonstration on blind source separation and social group clustering. For all the simulated algorithms, the initial factor matrix $\hat{\boldsymbol{\Xi}}^{(k,0)}$ is set as the singular value decomposition (SVD) approximation $\boldsymbol{U}_{:,1:L}\left(\boldsymbol{S}_{1:L,1:L}\right)^{\frac{1}{2}}$ where $[\boldsymbol{U},\boldsymbol{S},\boldsymbol{V}] = \text{SVD}[\mathcal{Y}^{(k)}]$ and $L = \min\{J_1, J_2, ..., J_N\}$. The parameter ϵ is set to be 10^{-6}. The algorithms are deemed to be converged when $\| [\![\hat{\boldsymbol{\Xi}}^{(1,s+1)}, \hat{\boldsymbol{\Xi}}^{(2,s+1)}, \cdots, \hat{\boldsymbol{\Xi}}^{(N,s+1)}]\!] - [\![\hat{\boldsymbol{\Xi}}^{(1,s)}, \hat{\boldsymbol{\Xi}}^{(2,s)}, \cdots, \hat{\boldsymbol{\Xi}}^{(N,s)}]\!] \|_F^2 < 10^{-6}$. All experiments were conducted in Matlab R2015b with an Intel Core i7 CPU at 2.2 GHz.

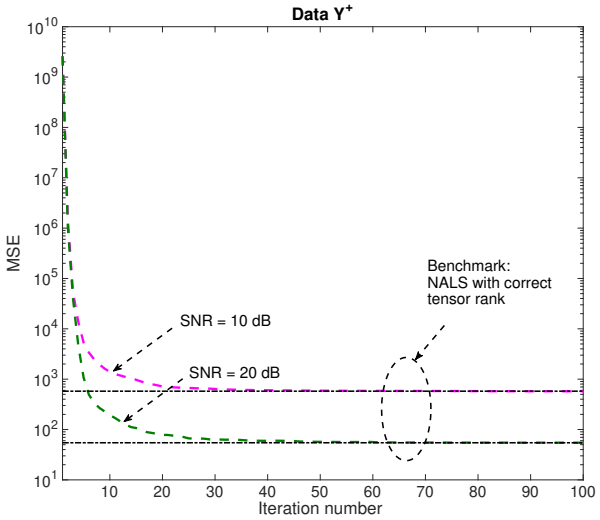
A. Validation on Synthetic Data

A three dimensional tensor $\mathcal{X} = [\![\boldsymbol{M}^{(1)}, \boldsymbol{M}^{(2)}, \boldsymbol{M}^{(3)}]\!] \in \mathbb{R}^{100\times100\times100}$ with rank $R = 10$ is considered as the noise-free data tensor. Each element in factor matrix $\boldsymbol{M}^{(n)}$ is independently drawn from a uniform distribution over $[0, 1]$ and thus is nonnegative. On the other hand, two observation data tensors are considered: 1). the data $\mathcal{X}$ is corrupted by a noise tensor $\mathcal{W} \in \mathbb{R}^{100\times100\times100}$, i.e., $\mathcal{Y} = \mathcal{X} + \mathcal{W}$, with each element of noise tensor $\mathcal{W}$ being independently drawn from a zero-mean Gaussian distribution with variance σ_w^2, and this corresponds to the Gaussian likelihood model (9); 2). the data $\mathcal{Y}^+$ is obtained by setting the negative elements of $\mathcal{Y}$ to zero, i.e., $\mathcal{Y}^+_{i_1,i_2,i_3} = \mathcal{Y}_{i_1,i_2,i_3}\text{U}(\mathcal{Y}_{i_1,i_2,i_3} \geq 0)$, and the truncated Gaussian likelihood model (10) is employed to fit these data. The SNR is defined as $10\log_{10}\left(\| \mathcal{X} \|_F^2 / \mathbb{E}_{p(\mathcal{W})}\left[\| \mathcal{W} \|_F^2\right]\right) = 10\log_{10}\left(\| \mathcal{X} \|_F^2 / (100^3\sigma_w^2)\right)$. For the proposed algorithm, the step size sequence is chosen as $\alpha_t = 10^{-3}/(t+1)$ [42], and the gradient projection update is terminated when $||f(\boldsymbol{\Xi}^{(k,t)}) - f(\boldsymbol{\Xi}^{(k,t-1)})||_F \leq 10^{-3}$. Each result in this subsection is obtained by averaging 100 Monte-Carlo runs.

Figure 4 presents the convergence performances of the proposed algorithm under different SNRs and different test data, where the mean-square-error (MSE) $\| [\![\hat{\boldsymbol{\Xi}}^{(1,s)}, \hat{\boldsymbol{\Xi}}^{(2,s)}, \hat{\boldsymbol{\Xi}}^{(3,s)}]\!] - \mathcal{X} \|_F^2$ is chosen as the assessment criterion. From Figure 4 (a), it can be seen that for test data $\mathcal{Y}$, the MSEs of the proposed algorithm, which assumes



(a)



(b)

Figure 4: Convergence of the proposed algorithm for different test data

no knowledge of tensor rank, decrease significantly in the first few iterations and converge to the MSE of the NALS algorithm [16] (with exact tensor rank) under SNR = 10 dB and SNR = 20 dB. Similar convergence performances can be observed for the test data $\mathcal{Y}^+$. This is of no surprise because approximating (15) by (14) dose not make any changes on the algorithm framework of variational inference, and thus the excellent convergence performance of the proposed algorithm is expected (as discussed in Section IV. C).

The MSE measures the performance of low-rank tensor recovery. However, due to the uniqueness property of tensor CPD [1], each factor matrix can be recovered up to an unknown permutation and scaling ambiguity. To directly assess the accuracies of factor matrices recovery , the best congruence ratio (BCR), which involves computing the MSE between the true factor matrix $\boldsymbol{M}^{(k)}$ and the estimated factor matrix $\hat{\boldsymbol{\Xi}}^{(k)}$,

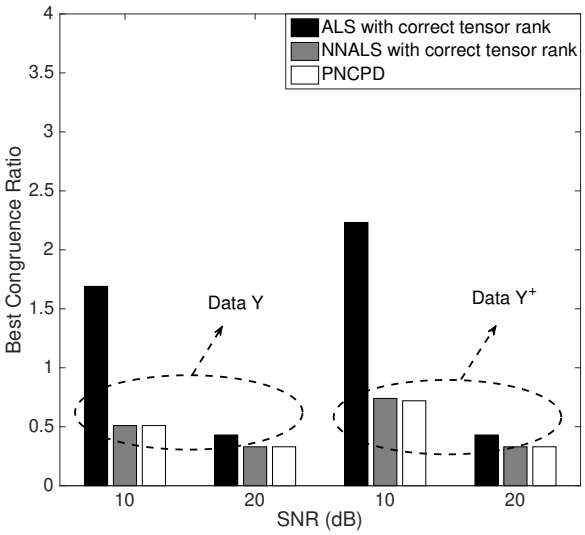


Figure 5: Best congruence ratios of the proposed algorithm for different test data

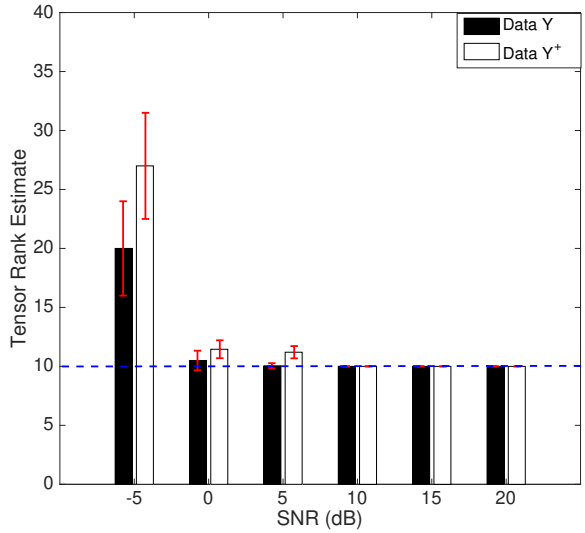


Figure 6: Tensor rank estimates of the proposed algorithm for different test data

is widely used as the assessment criterion. It is defined as

$$\sum_{k=1}^{3} \min_{\boldsymbol{\Delta}^{(k)}, \boldsymbol{P}^{(k)}} \frac{||\boldsymbol{M}^{(k)} - \hat{\boldsymbol{\Xi}}^{(k)} \boldsymbol{P}^{(k)} \boldsymbol{\Delta}^{(k)}||_F}{||\boldsymbol{M}^{(k)}||_F},$$

where the diagonal matrix $\boldsymbol{\Delta}^{(k)}$ and the permutation matrix $\boldsymbol{P}^{(k)}$ are found via the greedy least-squares column matching algorithm [48]. From Figure 5, it is seen that both the proposed algorithm (labeled as PNCPD) and the NALS algorithm (with exact tensor rank) achieve much better factor matrix recovery than the ALS algorithm (with exact tensor rank) [1]. This shows the importance of incorporating the nonnegative constraint into the algorithm design. Furthermore, the factor matrix recovery performances of the proposed algorithm under test data $\mathcal{Y}$ and $\mathcal{Y}^+$ are indistinguishable under SNR = 20 dB. This shows that when SNR is high, equation (14) gives a quite good approximation to equation (15), thus leading to remarkably accurate factor matrices recovery. Although the BCR of the proposed algorithm is higher for the data $\mathcal{Y}^+$ than that for the data $\mathcal{Y}$, it is with nearly the same performance as that of the NALS algorithm (with exact tensor rank).

On the other hand, the tensor rank estimates of the proposed algorithm under different SNRs are presented in Figure 6, with each vertical bar showing the mean and the error bars showing the standard derivation of tensor rank estimates. The blue horizontal dashed line indicates the true tensor rank. From Figure 6, it is seen that the proposed algorithm recovers the true tensor rank with 100% accuracy for a wide range of SNRs, in particular when SNR is larger than 10 dB. Even though the performance is not 100% accurate when SNR is 0 dB and 5 dB, the estimated tensor rank is still close to the true tensor rank with a high probability for test data $\mathcal{Y}$. However, under these two low SNRs, the rank estimation performances of the proposed algorithm for the data $\mathcal{Y}^+$ degrade significantly. This is because equation (15) cannot be well approximated by equation (14) under very low SNRs. Furthermore, the proposed algorithm fails to give correct rank estimates when SNR is lower than -5 dB for both two test data sets, since the noise with very large power masks the low-rank structure of the data.

To assess the tensor rank estimation accuracy when the tensor is with a larger true rank, we apply the the proposed algorithm to the tensor data $\mathcal{Y}$ with the true rank $R = \{10, 30, 50\}$ and SNR = 20 dB. The rank estimation performance is presented in Table I. From Table I, it can be seen that the proposed algorithm recovers the rank accurately when the true rank is 10 and 30. However, when $R = 50$, the proposed algorithm fails to accurately recover the tensor rank. This could be explained by the fact that the Gaussian-gamma prior would lead to the sparsest estimation result [52], and thus fail to work well when the true rank is high. This has also been observed in a recent matrix decomposition work [53], in which a Gaussian-Wishart prior has been employed to tackle the high-rank estimation challenge. However, employing the Gaussian-Wishart prior for tensor decompositions is challenging as Wishart distribution is inherently defined for a matrix [53]. Thus, high rank tensor decomposition, which is both important and interesting, would be left as future work.

The simulation results presented so far are for well-conditioned tensors, i.e., the columns in each of the factor matrices are independently generated. In order to fully assess the rank learning ability of the proposed algorithm, we consider another noise-free three dimensional tensor $\bar{\mathcal{X}} = [\![\bar{\boldsymbol{M}}^{(1)}, \boldsymbol{M}^{(2)}, \boldsymbol{M}^{(3)}]\!] \in \mathbb{R}^{100\times100\times100}$ with rank $R = 10$. The factor matrix is set as $\bar{\boldsymbol{M}}^{(1)} = 0.1\mathbf{1}_{100\times10} + 2^{-s}\boldsymbol{M}^{(1)}$, and each element in factor matrices $\{\boldsymbol{M}^{(n)}\}_{n=1}^{3}$ is independently drawn from a uniform distribution over $[0, 1]$. According to the definition of the tensor condition number [54], [55], when s increases, the correlation among the columns in the factor matrix $\bar{\boldsymbol{M}}^{(1)}$ increases, and the tensor condition number becomes larger. In particular, when s goes to infinity, the condition number of the tensor goes to infinity too. We apply the proposed algorithm to $\bar{\mathcal{X}}$ corrupted with noise:

Table I: Performance of tensor rank estimation versus different true tensor ranks for tensor data $\mathcal{Y}$ when SNR = 20 dB

True tensor rank	Mean of tensor rank estimates	Standard derivation of tensor rank estimates	Percentage of correct tensor rank estimates
10	10	0	100%
30	29.6	1.27	90%
50	28.15	18.29	25%

$\bar{\mathcal{Y}} = \bar{\mathcal{X}} + \bar{\mathcal{W}}$, where each element of noise tensor $\bar{\mathcal{W}}$ is independently drawn from a zero-mean Gaussian distribution with variance $\bar{\sigma}_w^2$. Table II shows the rank estimation accuracy of the proposed algorithm when SNR = 20 dB. It can be seen that the proposed algorithm can correctly estimate the tensor rank when $s < 5$. But as the tensor conditional number increases (i.e., the columns are more correlated in the factor matrix $\bar{\boldsymbol{M}}^{(1)}$), the tensor rank estimation performance decreases significantly.

Next, we consider an extreme case in which the columns in all factor matrices are highly correlated: $\bar{\mathcal{X}} = [\![\bar{\boldsymbol{M}}^{(1)}, \bar{\boldsymbol{M}}^{(2)}, \bar{\boldsymbol{M}}^{(3)}]\!] \in \mathbb{R}^{100\times100\times100}$ with rank $R = 10$, where each factor matrix $\bar{\boldsymbol{M}}^{(n)} = 0.1\mathbf{1}_{100\times10} + 2^{-s}\boldsymbol{M}^{(n)}$, and each element in factor matrices $\{\boldsymbol{M}^{(n)}\}_{n=1}^{3}$ is independently drawn from a uniform distribution over $[0, 1]$. With the same observation data model as $\bar{\mathcal{Y}}$ as before and when SNR = 20 dB, the percentages of correct tensor rank estimate are shown in Table III. It can be seen that it is difficult for the proposed algorithm to accurately estimate the tensor rank even when $s = 1$.

Table II: Performance of tensor rank estimation when the columns in one factor matrix are correlated and SNR = 20 dB

s	0	1	3	5	100
Percentage of correct tensor rank estimates	100%	100%	100%	25%	5%

Table III: Performance of tensor rank estimation when the columns in all factor matrices are correlated and SNR = 20 dB

s	0	1	3	5	100
Percentage of correct tensor rank estimates	100%	40%	0%	0%	0%

Finally, as discussed in the Section IV. C, acceleration schemes could be incorporated to speed up the proposed algorithm. As discussed in Section IV. A, in the first few iterations, since some precision parameters are learnt to be very large while some of them are with very small values, the average condition number of the Hessian matrix of problem (22), i.e., $\frac{1}{3}\sum_{k=1}^{3}\text{condition\_number}\left(\boldsymbol{H}^{(k)}\right)$, is very large. After several iterations, the proposed algorithm has gradually recovered the tensor rank, and then the remaining precision parameters are with comparable values, leading to a well-conditioned Hession matrix $\boldsymbol{H}^{(k)}$ of problem (22). These results can be observed in Figure 7. Inspired by the pioneering work [18], the Nesterov scheme [44, page 90] is utilized for the problem (22) when the condition number of the Hessian matrix is smaller than 100. Consequently, even with the same MSE and BCR performances, the accelerated algorithm is much faster than the default version of the proposed algorithm[3] as presented in Table IV. Besides the Nesterov scheme, other advances in first-order optimizations could be incorporated to further improve the scalability of the algorithm. However, this is not the main focus of this paper, and thus in the following, we only examine the performances of the default version of the proposed algorithm for real-world applications.

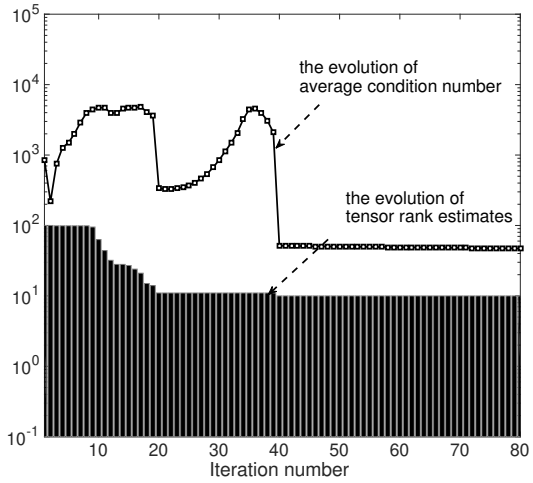


Figure 7: The average condition number of the Hessian matrix (24) and the tensor rank estimate versus number of iterations for test data $\mathcal{Y}$ when SNR = 20 dB.

Table IV: Average running time in seconds of the proposed algorithm for different test data. The accelerated algorithm is labeled as PNCPD-A.

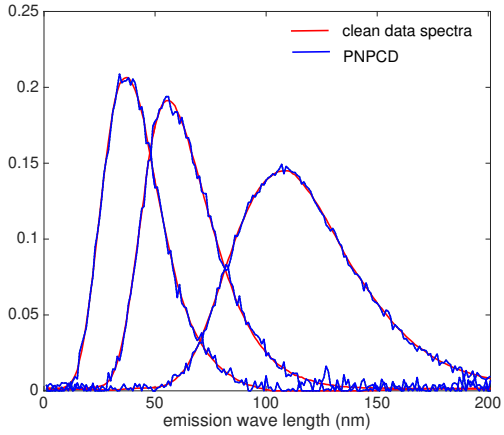
Data	SNR = 10 dB		SNR = 20 dB	
	PNCPD	PNCPD-A	PNCPD	PNCPD-A
$\mathcal{Y}$	113.61	**63.66**	62.52	**52.08**
$\mathcal{Y}^+$	99.48	**58.02**	59.61	**49.86**

B. Fluorescence Data Analysis

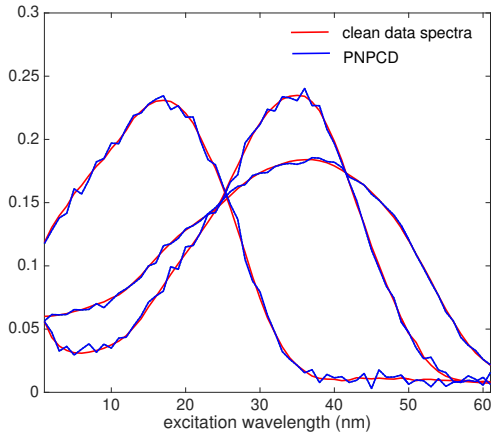
In this subsection, the proposed algorithm is utilized to analyze the amino acids fluorescence data[4] [44]. This data set consists of five laboratory-made samples. Each sample contains different amounts of tyrosine, tryptophan and phenylalanine dissolved in phosphate buffered water. The samples were measured by fluorescence and were corrupted by Gaussian noise with power 0.1, resulting in SNR = 0.16 dB. The fluorescence excitation-emission measured (EEM) data collected is with size $5\times201\times61$, and should be representable

[3]The presented time for the accelerated scheme includes the time for computing the condition numbers.

[4]http://www.models.life.ku.dk



(a)



(b)

Figure 8: The estimates of (a) emission spectra and (b) excitation spectra using the proposed algorithms, with the clean data spectra serving as the benchmark.

with a CPD model with rank 3, since there are three different types of amino acid and each individual amino acid gives a rank-one CPD component.

The proposed PNCPD algorithm was run to decompose the EEM tensor data with initial rank $L = 5$. For the proposed algorithm, the step size sequence is chosen as $\alpha_t = 10^{-2}/(t+1)$ [42], and the gradient projection update is terminated when the gradient norm is smaller than 10^{-3}. At convergence, the proposed algorithm identified the correct tensor rank[5] $R = 3$. Furthermore, the emission spectra and the excited spectra of three amino acids, which are obtained from the decomposed factor matrices [44], are shown in Figure 8, with the clean data spectra[6] serving as the benchmark. From Figure 8, it can be seen that the recovered spectra from the proposed algorithm are very close to the clean data spectra, with the MSE of the emission spectra estimation equals 1.51×10^{-4} per wavelength and the MSE of the excitation spectra estimation equals 1.08×10^{-4} per wavelength.

[5]Notice that even if the initial tensor rank L is set as 20, which is much larger than the true tensor rank 3, the proposed algorithm can still recover the true tensor rank. This shows that the proposed method is not sensitive to the initial tensor rank value.

[6]The clean data spectra is obtained by decomposing the clean data [44] using the NALS algorithm with correct tensor rank $R = 3$.

C. ENRON E-mail Data Mining

In this subsection, the ENRON Email corpus[7] [3] was analyzed. This data set is with size $184 \times 184 \times 44$, and contains e-mail communication records between 184 people within 44 months, in which each entry denotes the number of e-mail exchanged between two particular people within a particular month. Before fitting the data to the proposed algorithms, the same pre-processing as in [2], [3] is applied to the non-zero data to compress the dynamic range. Then, the proposed algorithm is utilized to fit the data into the proposed nonnegative CPD model, with the initial rank set as $L = 44$, the step size sequence being $\alpha_t = 1/(t+1)$ [42], and the gradient projection update terminated when the gradient norm is smaller than 10^{-3}. As introduced in the Appendix A, the estimated tensor rank has the physical meaning of the number of underlying social groups, and each element in the first factor matrix can be interpreted as the score that a particular person belongs to a particular email sending group.

During the inference, the tensor rank estimate gradually reduces to the value 4, indicating that there are four underlying social groups. This is consistent with the results from [2], [3], which are obtained via trail-and-error experiments. After sorting the scores of each column in the first factor matrix, the people with top 10 scores in each social group is shown in Table II. From the information of each person presented in Table II, the clustering results can be clearly interpreted. For instance, the people in the first group are either in legal department or lawyers, thus being clustered together. The clustering results of the proposed algorithms are also consistent with the results from [2], [3], which are obtained via nonlinear programming methods assuming the knowledge of tensor rank. Finally, interesting patterns can be observed from the temporal cluster profiles, which are obtained from the third factor matrix [2], [3], as illustrated in Figure 9. It is clear that when the company has important events, such as the change of CEO, crisis breaks and bankruptcy, distinct peaks appear. Notice that in this example, all the data entries are nonnegative and the proposed algorithm still works well. This indicates that the tensor CPD with nonnegative factors is a model that matches this social clustering task.

VI. Conclusions

In this paper, probabilistic tensor CPD with nonnegative factors has been investigated under unknown tensor rank. In particular, the nonnegative Gaussian-gamma prior, which was shown to have both sparsity-promoting and exponentially conjugacy properties, was introduced as the building block of the proposed probabilistic model. Then, an efficient inference algorithm was derived with an integrated feature of automatic rank determination. Extensive numerical results of both synthetic data and real-world data were presented to show the remarkable performance of the proposed algorithm.

[7]The original source of the data is from [3], and we greatly appreciate Prof. Vagelis Papalexakis for sharing the data with us.

Table V: Social groups with people in top 10 scores in each group for the ENRON e-mail data using the proposed algorithms

Legal	Government Affair Executive
'Tana Jones (tana.jones) Employee Financial Trading Group ENA Legal', 'Sara Shackleton (sara.shackleton) Employee ENA Legal', 'Mark Taylor (mark.taylor) Manager Financial Trading Group ENA Legal', 'Stephanie Panus (stephanie.panus) Senior Legal Specialist ENA Legal', 'Marie Heard (marie.heard) Senior Legal Specialist ENA Legal', 'Mark Haedicke (mark.haedicke) Managing Director ENA Legal', 'Susan Bailey (susan.bailey) Legal Assistant ENA Legal', 'Louise Kitchen (louise.kitchen) President Enron Online', 'Kay Mann (kay.mann) Lawyer', 'Debra Perlingiere (debra.perlingiere) Legal Specialist ENA Legal'	'Jeff Dasovich (jeff.dasovich) Employee Government Relationship Executive', 'James Steffes (james.steffes) VP Government Affairs', 'Steven Kean (steven.kean) VP Chief of Staff', 'Richard Shapiro (richard.shapiro) VP Regulatory Affairs', 'David Delainey (david.delainey) CEO ENA and Enron Energy Services', 'Richard Sanders (richard.sanders) VP Enron Wholesale Services', 'Shelley Corman (shelley.corman) VP Regulatory Affairs', 'Margaret Carson (margaret.carson) Employee Corporate and Environmental Policy', 'Mark Haedicke (mark.haedicke) Managing Director ENA Legal', 'Vince Kaminski (vince.kaminski) Manager Risk Management Head'
Trading / Top Executive	**Pipeline**
'Michael Grigsby (mike.grigsby) Director West Desk Gas Trading', 'Kevin Presto (m..presto) VP East Power Trading', 'Mike McConnell (mike.mcconnell) Executive VP Global Markets', 'John Arnold (john.arnold) VP Financial Enron Online', 'Louise Kitchen (louise.kitchen) President Enron Online', 'David Delainey (david.delainey) CEO ENA and Enron Energy Services', 'John Lavorato (john.lavorato) CEO Enron America', 'Sally Beck (sally.beck) COO ', 'Joannie Williamson (joannie.williamson) Executive Assistant ', 'Liz Taylor (liz.taylor) Executive Assistant to Greg Whalley '	'Michelle Lokay (michelle.lokay) Admin. Asst. Transwestern Pipeline Company (ETS)', 'Kimberly Watson (kimberly.watson) Employee Transwestern Pipeline Company (ETS)', 'Lynn Blair (lynn.blair) Employee Northern Natural Gas Pipeline (ETS)', 'Shelley Corman (shelley.corman) VP Regulatory Affairs', 'Drew Fossum (drew.fossum) VP Transwestern Pipeline Company (ETS)', 'Lindy Donoho (lindy.donoho) Employee Transwestern Pipeline Company (ETS)', 'Kevin Hyatt (kevin.hyatt) Director Asset Development TW Pipeline Business (ETS)', 'Darrell Schoolcraft (darrell.schoolcraft) Employee Gas Control (ETS)', 'Rod Hayslett (rod.hayslett) VP Also CFO and Treasurer', 'Susan Scott (susan.scott) Employee Transwestern Pipeline Company (ETS)'

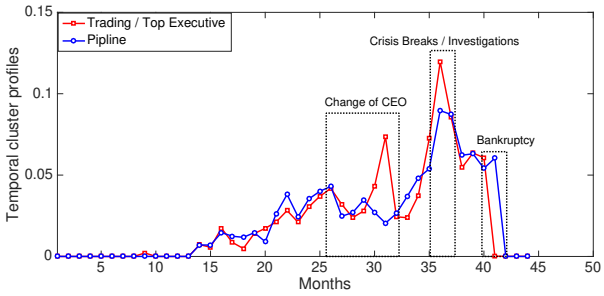


Figure 9: Temporal cluster profiles (from the third factor matrix) for the ENRON e-mail dataset

APPENDIX

A. Motivation examples

In this appendix, two motivating examples for probabilistic tensor CPD with nonnegative factors are presented.

1) Motivating example 1 (fluorescence spectroscopy): Fluorescence spectroscopy is a fast, simple and inexpensive method to determine the concentration of any solubilized sample based on its fluorescent properties, and is widely used in chemical, pharmaceutical and biomedical fields [34], [35]. In fluorescence spectroscopy, an excitation beam with a certain wavelength λ_i passes through a solution in a cuvette. The excited chemical species in the sample will change their electronic states and then emit a beam of light, of which its spectrum is measured at the detector. Mathematically, let the concentration of the r^{th} specie in the sample be c_r, and the excitation value at wavelength λ_i be $a_r(\lambda_i)$. Then, the noise-free measured spectrum intensity at the wavelength λ_j is $a_r(\lambda_i)b_r(\lambda_j)c_r$, where $b_r(\lambda_j)$ is the emission value of the r^{th} species at the wavelength λ_j. If there are R different species in the sample, the noise-free fluorescence excitation-emission measured (EEM) data at λ_j is

$$p_{i,j} = \sum_{r=1}^{R} a_r(\lambda_i)b_r(\lambda_j)c_r. \tag{27}$$

Assume the excitation beam contains I wavelengths, and the noise-free EEM data is collected at J different wavelengths, an $I \times J$ data matrix is obtained as

$$\boldsymbol{P} = \sum_{r=1}^{R} \boldsymbol{A}_{:,r} \circ \boldsymbol{B}_{:,r} c_r, \tag{28}$$

where symbol $\circ$ denotes vector outer product, $\boldsymbol{A}_{:,r} \in \mathbb{R}^{I\times 1}$ is a vector with the i^{th} element being $a_r(\lambda_i)$ and $\boldsymbol{B}_{:,r} \in \mathbb{R}^{J\times 1}$ is a vector with the j^{th} element being $b_r(\lambda_j)$. Assume $K > 1$ samples with the same chemical species but with different concentration of each specie are measured. Let the concentration of the r^{th} specie in the k^{th} sample be $c_{k,r}$, then after stacking the noise-free EEM data for each sample along a third dimension, a three dimensional (3D) tensor data $\mathcal{P} \in \mathbb{R}^{I\times J\times K}$ can be obtained as

$$\mathcal{P} = \sum_{r=1}^{R} \boldsymbol{A}_{:,r} \circ \boldsymbol{B}_{:,r} \circ \boldsymbol{C}_{:,r} \triangleq [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!], \tag{29}$$

where $\boldsymbol{C}_{:,r} \in \mathbb{R}^{K\times 1}$ is a vector with the k^{th} element being $c_{k,r}$; matrices $\boldsymbol{A} \in \mathbb{R}^{I\times R}$, $\boldsymbol{B} \in \mathbb{R}^{J\times R}$ and $\boldsymbol{C} \in \mathbb{R}^{K\times R}$ are matrices with their r^{th} columns being $\boldsymbol{A}_{:,r}$, $\boldsymbol{B}_{:,r}$ and $\boldsymbol{C}_{:,r}$, respectively.

It is easy to see that the noise-free data model in (29) yields exactly the tensor CPD model [1], and that is why CPD algorithms work very well for EEM data analysis [9]. More specifically, accounting for the possible Gaussian noise, the EEM data analysis aims to solve the following problem:

$$\begin{aligned} &\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} \| \mathcal{P} - [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!] \|_F^2 \\ &\text{s.t.} \quad \boldsymbol{A} \geq \boldsymbol{0}_{I\times R}, \boldsymbol{B} \geq \boldsymbol{0}_{J\times R}, \boldsymbol{C} \geq \boldsymbol{0}_{K\times R}, \end{aligned} \tag{30}$$

where the nonnegative constraints are enforced due to the physical nature of elements in matrices $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{C}$ as introduced above.

2) Motivating example 2 (social group clustering): Social group clustering could be benefited by tensor data analysis, by which multiple views of social network are provided [2]–[4]. For example, consider a 3D email data set $\mathcal{Y} \in \mathbb{R}^{I\times J\times K}$ with each element $\mathcal{Y}(i, j, k)$ denoting the number of emails sent from person i to person j at the k^{th} day. Each frontal slice $\mathcal{Y}(:, :, k)$ represents the connection intensity among different pair of peoples in the k^{th} day, while each slice $\mathcal{Y}(:, j, :)$ shows the temporal evolution of the number of received mails for the person j from each of the other person in the data set. Consequently, decomposing the tensor $\mathcal{Y}$ into latent CPD factors $\{\boldsymbol{A} \in \mathbb{R}^{I\times R}, \boldsymbol{B} \in \mathbb{R}^{J\times R}, \boldsymbol{C} \in \mathbb{R}^{K\times R}\}$ reveals different clustering groups from different views (i.e., different tensor dimensions). In particular, using the unfolding property of tensor CPD [1], we have

$$\mathcal{Y}^{(1)} = (\boldsymbol{C} \diamond \boldsymbol{B})\boldsymbol{A}^T, \tag{31}$$

$$\mathcal{Y}^{(3)} = (\boldsymbol{B} \diamond \boldsymbol{A})\boldsymbol{C}^T, \tag{32}$$

where $\mathcal{Y}^{(k)}$ is a matrix obtained by unfolding the tensor $\mathcal{Y}$ along its k^{th} dimension [1], and symbol $\diamond$ denotes the Khatri-Rao product (i.e., column-wise Kronecker product). From (31), each column vector $\mathcal{Y}^{(1)}(:,i) \in \mathbb{R}^{JK\times 1}$ can be written as $\mathcal{Y}^{(1)}(:,i) = \sum_{r=1}^{R}(\boldsymbol{C}\diamond\boldsymbol{B})_{:,r}\boldsymbol{A}_{i,r}$, which is a linear combination of column vectors in matrix $(\boldsymbol{C}\diamond\boldsymbol{B}) \in \mathbb{R}^{JK\times R}$ with coefficients $\{\boldsymbol{A}_{i,r}\}_{r=1}^{R}$, and it represents the email sending pattern of person i. Thus, each column vector in matrix $\boldsymbol{C}\diamond\boldsymbol{B}$ can be interpreted as one of the R underlying email sending patterns, and $\boldsymbol{A}_{i,r}$ is the linear combining coefficient to generate the person i's email pattern. Similarly, from (32), each column in $\boldsymbol{B}\diamond\boldsymbol{A}$ can be interpreted as a temporal pattern and $\boldsymbol{C}_{k,r}$ is the coefficient of the r^{th} temporal pattern for generating the k^{th} day's pattern. Obviously, in contrast to the matrix-based model such as k-means or Gaussian mixture model, the tensor CPD model succeeds in mining clustering structures in multidimensional data. To find the latent factor matrices from the social network data $\mathcal{Y}$, the following problem is usually solved:

$$\begin{aligned} &\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} \parallel \mathcal{Y} - [\![\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}]\!] \parallel_F^2 \\ &\text{s.t.} \quad \boldsymbol{A} \geq \boldsymbol{0}_{I\times R}, \boldsymbol{B} \geq \boldsymbol{0}_{J\times R}, \boldsymbol{C} \geq \boldsymbol{0}_{K\times R}, \end{aligned} \tag{33}$$

where the nonnegative constraints are added to allow only additions among the R latent rank-1 components. This leads to a parts-based representation of the data, in the sense that each rank-1 component is a part of the data, thus further enhancing the model interpretability.

B. Properties on nonnegative Gaussian-gamma prior

Firstly, it is shown that generally a gamma distribution is not conjugate to a truncated Gaussian distribution. Without loss of generality, consider a truncated Gaussian-gamma distribution pair with the following form:

$$\begin{aligned} p(\boldsymbol{w},\{\alpha_s\}_{s=1}^S) &= \prod_{s=1}^{S} p(\boldsymbol{w}_s|\alpha_s)p(\{\alpha_s\}_{s=1}^S) \\ &= \prod_{s=1}^{S} \frac{\mathcal{N}(\boldsymbol{w}_s|\boldsymbol{\mu}_s,\alpha_s^{-1}\boldsymbol{I}_{M_s})}{\int_{\underline{\boldsymbol{w}_s}}^{\overline{\boldsymbol{w}_s}} \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{\mu}_s,\alpha_s^{-1}\boldsymbol{I}_{M_s})} \mathrm{U}\left(\underline{\boldsymbol{w}_s},\overline{\boldsymbol{w}_s}\right) \\ &\quad \times \frac{b_s^{a_s}}{\Gamma(a_s)}\alpha_s^{a_s-1}\exp(-b_s\alpha_s), \end{aligned} \tag{34}$$

where the truncated Gaussian pdf is with support $[\underline{\boldsymbol{w}_s},\overline{\boldsymbol{w}_s}]$; parameter $\boldsymbol{\mu}_s,\alpha_s\boldsymbol{I}_{M_s}$ are the mean vector and the precision matrix of the un-rectified Gaussian pdf. In order to compute the the posterior pdf $p(\alpha_s|\boldsymbol{w})$, we only keep the terms relevant to α_s and then the following expression is obtained:

$$p(\alpha_s|\boldsymbol{w}) \propto \frac{\exp\left(-\alpha_s\left(\frac{||\boldsymbol{w}_s-\boldsymbol{\mu}_s||_F^2}{2}+b\right)\right)\alpha_s^{a_s-1+\frac{1}{2}}}{\int_{\underline{\boldsymbol{w}_s}}^{\overline{\boldsymbol{w}_s}} \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{\mu}_s,\alpha_s^{-1}\boldsymbol{I}_{M_s})}. \tag{35}$$

Obviously, expression (35) does not take the same functional form as that of the gamma distribution, and thus violates the definition of the conjugacy.

But fortunately, when $\boldsymbol{\mu}_s = \boldsymbol{0}_{M_s\times 1}$, $\underline{\boldsymbol{w}_s} = \boldsymbol{0}_{M_s\times 1}$ and $\overline{\boldsymbol{w}_s} = \boldsymbol{\infty}$, it is easy to show that $\int_{\boldsymbol{0}_{M_s\times 1}}^{\boldsymbol{\infty}} \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{0}_{M_s\times 1},\alpha_s^{-1}\boldsymbol{I}_{M_s}) = \frac{1}{2^{M_s}}$ due to the symmetric property of the Gaussian pdf. Consequently, under this specific setting, equation (35) becomes

$$p(\alpha_s|\boldsymbol{w}) \propto \exp\left(-\alpha_s\left(\frac{||\boldsymbol{w}_s-\boldsymbol{\mu}_s||_F^2}{2}+b\right)\right)\alpha_s^{a_s-1+\frac{1}{2}}, \tag{36}$$

which takes exactly the same functional form as that of the gamma distribution. As a result, **Property 1** is proved, and this important finding motivates the use of the nonnegative Gaussian-gamma prior in the probabilistic modeling. Furthermore, under this parameter setting, the marginal distribution for parameter $\boldsymbol{w}$ can be computed as follows:

$$\begin{aligned} p^+(\boldsymbol{w}) &= \int \prod_{s=1}^{S} p(\boldsymbol{w}_s|\alpha_s)p(\{\alpha_s\}_{s=1}^S)d\{\alpha_s\} \\ &= \prod_{s=1}^{S} 2^{M_s} \int \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{\mu}_s,\alpha_s^{-1}\boldsymbol{I}_{M_s})\mathrm{gamma}(\alpha_s|a_s,b_s)d\alpha_s \\ &\quad \times \mathrm{U}(\boldsymbol{w}_s \geq \boldsymbol{0}_{M_s\times 1}). \end{aligned} \tag{37}$$

Further using the fact $\int \mathcal{N}(\boldsymbol{w}_s|\boldsymbol{\mu}_s,\alpha_s^{-1}\boldsymbol{I}_{M_s})\mathrm{gamma}(\alpha_s|a_s,b_s)$ $d\alpha_s = (\frac{1}{\pi})^{\frac{M_s}{2}}\frac{\Gamma(a_s+M_s/2)}{(2b_s)^{-a_s}\Gamma(a_s)}(2b_s + \boldsymbol{w}_s^T\boldsymbol{w}_s)^{-a_s-M_s/2}$ [37], **Property 2** can be proved.

REFERENCES

[1] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455-500, Aug. 2009.

[2] E. E. Papalexakis, N. D. Sidiropoulos, and R. Bro, "From K-means to higher-way co-clustering: multilinear decomposition with sparse latent factors," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 493-506, Jan. 2013.

[3] B. W. Bader, R. A. Harshman, and T. G. Kolda, "Temporal analysis of social network using three-way DEDICOM," Sandia National Labs, TR SAND2006-2161, 2006.

[4] W. Peng and T. Li, "Temporal relating co-clustering on directional social network and author-topic evolution," *Knowledge and Information System*, pp. 1-20, Mar. 2010.

[5] P. Rai, C. Hu, M. Harding, and L. Carin, "Scalable probabilistic tensor factorization for binary and count data," in *Proceedings of the International Joint Conferences on Artificial Intelligence(IJCAI)*, pp. 3770-3776, Jul. 2015.

[6] C. Hu, P. Rai, C. Chen, M. Harding, L. Carin, "Scalable bayesian non-negative tensor factorization for massive count data," in *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML)*, pp. 53-70, Sep. 2015.

[7] H. Chen, and J. Li, "DrugCom: synergistic discovery of drug combinations using tensor decomposition," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pp. 899-904, Dec. 2018.

[8] M. M. Alamdari, N. L. Dang Khoa, Y. Wang, B. Samali, and X. Zhu, "A multi-way data analysis approach for structural health monitoring of a cable-stayed bridge," *Structural Health Monitoring*, vol. 18, no. 1, pp. 35-48, Nov. 2019.

[9] R. Bro, "Review on multiway analysis in chemistry 2000-2005," *Critical Reviews in Analytical Chemistry*, vol. 35, pp. 279-293, Jan. 2006.

[10] L. Cheng, Y.-C. Wu, J. Zhang, and L. Liu, "Subspace identification for DOA estimation in massive / full-dimension MIMO system: Bad data mitigation and automatic source enumeration," *IEEE Transactions on Signal Processing*, vol. 63, no. 22, pp. 5897-5909, Nov. 2015.

[11] L. Cheng, Y.-C. Wu, S. Ma, J. Zhang and L. Liu, "Channel estimation in full-dimensional massive MIMO system using one training symbol," in *Proceedings of the IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 341-346, Jul. 2017.

[12] F. Roemer, M. Haardt, and G. D. Galdo, "Analytical performance assessment of multi-dimensional matrix and tensor-based ESPRIT-Type algorithms," *IEEE Transactions on Signal Processing*, vol. 62, no. 10, pp. 2611-2625, Apr. 2014.

[13] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa and H. A. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 145-163, Feb. 2015.

[14] A. P. Liavas and N. D. Sidiropoulos, "Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers," *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5450-5463, Jul. 2015.

[15] S. Smith, A. Beri and G. Karypis, "Constrained tensor factorization with accelerated AO-ADMM," in *Proceedings of the International Conference on Parallel Processing (ICPP)*, pp. 111-120, Jul. 2017.

[16] A. Cichocki, R. Zdunek, A. H. Phan and S. I. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*, John Wiley & Sons, 2009.

[17] A. Cichocki, M. Mørup, P. Smaragdis, W. Wang and R. Zdunek, *Advances in Nonnegative Matrix and Tensor Factorization*, Computational Intelligence and Neuroscience, 2008.

[18] A. P. Liavas, G. Kostoulas, G. Lourakis, K. Huang, and N. D. Sidiropoulos, "Nesterov-based alternating optimization for nonnegative tensor factorization: algorithm and parallel implementations," *IEEE Transactions on Signal Processing*, vol. 66, no. 4, Feb. 2018.

[19] K. Huang, N. D. Sidiropoulos, and A. P. Liavas, "A flexible and efficient algorithmic framework for constrained matrix and tensor factorization," *IEEE Transactions on Signal Processing* vol. 64, no.19, pp. 5052-5065, Oct. 2016.

[20] E. Acar, D. M. Dunlavy, and T. G. Kolda, "A scalable optimization approach for fitting canonical tensor decomposition," *Journal of Chemometrics*, vol. 25, no. 2, pp. 67-86, Jan. 2011.

[21] A. P. Liavas, and N.D. Sidiropoulos, "Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers," *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5450-5463, Jul. 2015.

[22] B. Ermis and A. T. Cemgil, "A Bayesian tensor factorization model via variational inference for link prediction," arXiv preprint arXiv:1409.8276 (2014).

[23] Q. Zhao, L. Zhang, and A. Cichocki, "Bayesian CP factorization of incomplete tensors with automatic rank determination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp.1751-1763, Sep. 2015.

[24] L. Cheng, Y-C. Wu, and H. V. Poor, "Probabilistic tensor canonical polyadic decomposition with orthogonal factors," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 663-676, Feb. 2017.

[25] L. Cheng, Y.-C. Wu, and H. V. Poor, "Scaling probabilistic tensor canonical polyadic decomposition to massive data," *IEEE Transactions on Signal Processing*, vol. 66, no. 21, pp. 5534-5548, Nov. 2018.

[26] W. Guo and W. Yu, "Variational Bayesian PARAFAC decomposition for multidimensional harmonic retrieval," in *Proceedings of 2011 IEEE CIE International Conference on Radar*, vol. 2, pp.1864-1867, 2011.

[27] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211-244, Jun. 200.

[28] S. Ji, Y. Xue and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp.2346-2356, Jun. 2008.

[29] S. Nakajima, R. Tomioka, M. Sugiyama and S.D. Babacan, "Perfect dimensionality recovery by variational Bayesian PCA," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 971-979, Jul. 2012.

[30] S. D. Babacan, L. Martin, M. Rafael, and K. K. Aggelos, "Sparse Bayesian methods for low-rank matrix estimation," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 3964-3977, Feb. 2012.

[31] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Rank regularization and Bayesian inference for tensor completion and extrapolation," *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5689-5703, Nov. 2013.

[32] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Foundations and Trends in Machine Learning*, vol. 1, no. 102, pp. 1-305, Jan. 2008.

[33] C. Zhang, J. Butepage, H. Kjellstrom and S Mandt, "Advances in variational inference," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8. pp. 2008-2026, Aug. 2019.

[34] A. Shahzad, G. Kohler, M. Knapp, E. Gaubitzer, M. Puchinger and M. Edetsberger, "Emerging applications of fluorescence spectroscopy in medical microbiology field," *Journal of Translational Medicine*, vol. 7, no. 1, pp. 99, Nov. 2009.

[35] J. R. Albani, *Principles and Applications of Fluorescence Spectroscopy*, John Wiley & Sons, 2008.

[36] A. Cichocki, Z. Rafal, and A. Shunichi, "New algorithms for non-negative matrix factorization in applications to blind source separation," in *Proceeding of 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings(ICASSP)*, vol. 5, pp. 621-624. May. 2006.

[37] S. D. Babacan, S. Nakajima, and M. N. Do, "Bayesian group-sparse modeling and variational inference," *IEEE Transactions on Signal Processing*, vol. 62, no. 11, pp. 2906-2921, Jun. 2014.

[38] T. Brouwer, J. Frellsen, and P. Lio, "Comparative study of inference methods for Bayesian nonnegative matrix factorisation," in *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, pp. 513-529, Sep. 2017.

[39] M. N. Schmidt, and S. Mohamed, "Probabilistic nonnegative tensor factorization using Markov chain Monte Carlo," in *Proceedings of 17th European Signal Processing Conference*, pp. 1918-1922, Aug. 2009.

[40] T. Brouwer, and P. Lio, "Prior and likelihood choices for Bayesian matrix factorisation on small datasets," arXiv preprint arXiv:1712.00288.

[41] Y. Nesterov, *Introductory Lectures on Convex Optimization*, Kluwer Academic Publishers, 2004.

[42] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 1999.

[43] G. Ballard, K. Hayashi, and R. Kannan, "Parallel nonnegative CP decomposition of dense tensors," in *25th IEEE International Conference on High Performance Computing(HiPC) 2018*, 2018.

[44] H. A. L. Kiers, "A three-step algorithm for Candecomp/Parafac analysis of large data sets with multicollinearity," *Journal of Chemometrics*, vol. 12, pp. 155-171, Jun. 1998.

[45] M. Hoffman, D. Blei, J. Paisley, and C. Wang, "Stochastic variational inference," *Journal of Machine Learning Research*, vol. 14, pp. 1303-1347, 2013.

[46] J. C. Arismendi, "Multivariate truncated moments," *Journal of Multivariate Analysis*, vol. 117, pp. 41-75, 2013.

[47] M. Sorensen, L. D. Lathauwer, P. Comon, S. Icart, and L. Deneire, "Canonical polyadic decomposition with a columnwise orthonormal factor matrix," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 4, pp. 1190-1213, 2012.

[48] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Transaction on Signal Processing*, vol. 48, no. 3, pp. 810-823, Mar. 2000.

[49] M. E. Tipping, and C. Faul. Anita, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proceeding of the Ninth International Workshop on Artificial Intelligence and Statistics (AISTAT)*, C. M. Bishop and B. J. Frey, Eds., 2003.

[50] J. Lin, N. Marcel, and L. E. Brian, "Impulsive noise mitigation in powerline communications using sparse Bayesian learning," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1172-1183, Jun. 2013.

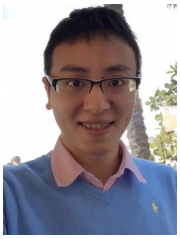
[51] J. Ma, S. Zhang, H. Li, F. Gao, and S. Jin, "Sparse bayesian learning for the time-varying massive MIMO channels: acquisition and tracking," *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 1925-1938, Jul. 2018.

[52] D. P. Wipf and B. D. Rao, "Sparse bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153-2164, Aug. 2004.

[53] L. Yang, J. Fang, H. Duan, H. Li, and B. Zeng, "Fast low-rank Bayesian matrix completion with hierarchical gaussian prior models," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2804-2817, Jun. 2018.

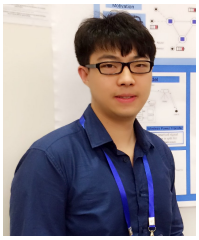
[54] P. Breiding and N. Vannieuwenhoven, "The condition number of joint decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 39, no. 1, pp. 287-309, 2018.

[55] N. Vannieuwenhoven, "Condition numbers for the tensor rank decomposition," *Linear Algebra Appl.*, vol. 535, pp. 35-86, 2017.



Lei Cheng received the B.Eng. degree from Zhejiang University in 2013, and the Ph.D. degree from the University of Hong Kong in 2018. Currently, he is a research scientist in Shenzhen Research Institute of Big Data (SRIBD). His research interests are in tensor data analytics, statistical inference and large-scale optimization.

Xueke Tong received the Master Degree from South China University of Technology in 2015. Currently, she is a Ph.D. student in the department of Electrical and Electronic Engineering, the University of Hong Kong. Her research interests include tensor data analytics, probabilistic model analysis and statistical machine learning.



Shuai Wang (S'16-M'19) received the Ph.D. degree in Electrical and Electronic Engineering from the University of Hong Kong in 2018. From 2018 to 2019, he is a Postdoc and Lecturer at the University of Hong Kong. Currently, he is a research assistant professor at the Southern University of Science and Technology (SUSTech). His research interests include wireless communications, unmanned systems, and machine learning.

Yik-Chung Wu (S'99-M'05-SM'14) received the B.Eng. (EEE) degree and the M.Phil. degree from The University of Hong Kong (HKU), in 1998 and 2001, respectively, and the Ph.D. degree from Texas A&M University, College Station, TX, USA, in 2005. From 2005 to 2006, he was with the Thomson Corporate Research, Princeton, NJ, USA, as a Member of Technical Staff. Since 2006, he has been with HKU, where he is currently an Associate Professor. He was a Visiting Scholar at Princeton University in 2017. His research interests are in general areas of signal processing, machine learning, and communication systems. He served as an Editor for the IEEE COMMUNICATIONS LETTERS and the IEEE TRANSACTIONS ON COMMUNICATIONS. He is currently an Editor of the Journal of Communications and Networks.

H. Vincent Poor (S'72, M'77, SM'82, F'87) received the Ph.D. degree in EECS from Princeton University in 1977. From 1977 until 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. Since 1990 he has been on the faculty at Princeton, where he is the Michael Henry Strater University Professor of Electrical Engineering. From 2006 until 2016, he served as Dean of Princeton's School of Engineering and Applied Science. He has also held visiting appointments at several other institutions, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory, signal processing and machine learning, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the forthcoming book *Advanced Data Analytics for Power Systems* (Cambridge University Press, 2020).

Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences, and is a foreign member of the Chinese Academy of Sciences, the Royal Society and other national and international academies. He received the Technical Achievement and Society Awards of the IEEE Signal Processing Society in 2007 and 2011, respectively. Recent recognition of his work includes the 2017 IEEE Alexander Graham Bell Medal, the 2019 ASEE Benjamin Garver Lamme Award, a D.Sc. *honoris causa* from Syracuse University, awarded in 2017, and a D.Eng. *honoris causa* from the University of Waterloo, awarded in 2019.