Bayesian Latent Factor Model for Higher-order Data: an Extended Abstract

Zerui Tao[1,2], Xuyang Zhao[1,2], Toshihisa Tanaka[1,2], and Qibin Zhao[2]

[1]Tokyo University of Agriculture and Technology
[2]RIKEN AIP
{zerui.tao, qibin.zhao}@riken.jp

Abstract

Latent factor models are canonical tools to learn low-dimensional and linear embedding of original data. Traditional latent factor models are based on low-rank matrix factorization of covariance matrices. However, for higher-order data with multiple modes, i.e., tensors, this simple treatment fails to take into account the mode-specific relations. This ignorance leads to inefficiency in analysis of complex structures as well as poor data compression ability. In this paper, unlike covariance matrices, we investigate high-order covariance tensor directly by exploiting tensor ring (TR) format and propose the Bayesian TR latent factor model, which can represent complex multi-linear correlations and achieves efficient data compression. To overcome the difficulty of finding the optimal TR-ranks and simultaneously imposing sparsity on loading coefficients, a multiplicative Gamma process (MGP) prior is adopted to automatically infer the ranks and obtain sparsity. Then, we establish efficient parameter-expanded EM algorithm to learn the maximum a posteriori (MAP) estimate of model parameters.

1 Introduction

Latent factor models provide promising tools for inferring latent structures and dimension reduction [2, 1, 4, 11]. Traditional latent factor models aim to tackle with vector features and seek for low-dimensional linear embedding of original data. Specifically, supposing the data have $P \times P$ covariance matrix $\boldsymbol{V}$, latent factor models find a low-rank representation $\boldsymbol{V} = \boldsymbol{W}\boldsymbol{W}^\intercal + \boldsymbol{\Sigma}$, where $\boldsymbol{W} \in \mathbb{R}^{P \times K}$ is the loading matrix with $K \ll P$ and $\boldsymbol{\Sigma}$ is diagonal. Adopting this approximation, we can use K latent factors to represent the original data for downstreaming learning tasks, such as clustering and classification. Despite the achievements of traditional latent factor models, they are not designed to model higher-order data, i.e., tensors. The naïve vectorization may suffer from the curse of dimensionality and fail to take into account the mode-specific relations.

To overcome these drawbacks, we try to leverage the merits of tensor networks (TNs) to factor models. In this work, instead of finding low-rank approximation of *covariance matrices*, our motivation is to directly investigate tensor decomposition for *covariance tensors*. For higher-order data, we reckon that the covariances can be naturally represented by tensors. For example, for a matrix data $\boldsymbol{Y}^{(n)} \in \mathbb{R}^{I \times J}$, the covariance $\boldsymbol{\mathcal{V}}$ is an order-4 tensor of shape $I \times J \times I \times J$, where $\boldsymbol{\mathcal{V}}_{ijmn} = \text{var}(\boldsymbol{Y}_{ij}, \boldsymbol{Y}_{mn})$. To model the covariance tensor, we suppose that it admits the TR format [10].

Since the TR-ranks is a vector, it is hard to tune the TR-ranks as well as the factor numbers, which are shown to be essential to the performance. Moreover, it is desirable to obtain sparse loading core tensors for learning interpretable latent factors. To address these issues, we extend the multiplicative

Second Workshop on Quantum Tensor Networks in Machine Learning, 35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Gamma process (MGP) prior [1] to TR format, for both global and local sparsity. Then, we establish efficient Parameter-eXpanded EM (PX-EM) algorithm for maximum a posteriori (MAP) estimate.

2 Preliminaries

2.1 Bayesian Latent Factor Model

Considering N observed data $\{\boldsymbol{y}^{(n)}\}_{n=1}^{N} \in \mathbb{R}^P$, the generic form of a latent factor model is [1]

$$\boldsymbol{y}^{(n)} = \boldsymbol{W}\boldsymbol{\eta}^{(n)} + \boldsymbol{\epsilon}^{(n)}, \quad \forall n = 1, \ldots, N, \tag{1}$$

where $\boldsymbol{W} \in \mathbb{R}^{P\times K}$ is called the loading matrix, $\boldsymbol{\eta}^{(n)} \in \mathbb{R}^K$ are latent factors and $\boldsymbol{\epsilon}^{(n)} \in \mathbb{R}^P$ are noises. The latent factors are supposed to follow standard Gaussian distribution, i.e., $\boldsymbol{\eta}^{(n)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_K)$. Moreover, suppose $\boldsymbol{\epsilon}^{(n)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_P^2)$. Under such conditions, we have $\boldsymbol{y}^{(n)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{V})$, where $\boldsymbol{V} = \boldsymbol{W}\boldsymbol{W}^\intercal + \boldsymbol{\Sigma}$. Hence, the main task of latent factor models is to find a low-rank representation of the covariance matrix.

2.2 Tensor Ring Decomposition

Now we introduce some basics of the Tensor Ring (TR) decomposition [10], which is also known as matrix product states in TN. For an order-D tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1\times\cdots\times I_D}$, the TR format denoted as

$$\boldsymbol{\mathcal{X}} = \ll \boldsymbol{\mathcal{Q}}^{(1)}, \ldots, \boldsymbol{\mathcal{Q}}^{(D)} \gg, \tag{2}$$

where $\boldsymbol{\mathcal{Q}}^{(d)} \in \mathbb{R}^{R_d\times I_d\times R_{d+1}}, \forall d = 1, \ldots, D$ are core tensors and $R_{D+1} = R_1$. The sequence $\{R_d\}_{d=1}^{D}$ is called TR-rank. Each element of the full tensor $\boldsymbol{\mathcal{X}}$ can be expressed as matrix product of the core tensors, namely, $\boldsymbol{\mathcal{X}}_{\mathbf{i}} = \mathrm{tr}\left(\boldsymbol{Q}^{(1)}[i_d]\cdots\boldsymbol{Q}^{(D)}[i_D]\right)$, where $\boldsymbol{Q}^{(d)}[i_d] \in \mathbb{R}^{R_d\times R_{d+1}}$ is the i_d-th lateral slice of the d-th core tensor.

The subchains of TR is defined as tensor contractions among a subsequence of cores tensors. For example, the left subchain $\boldsymbol{\mathcal{Q}}^{<d} \in \mathbb{R}^{R_1\times\prod_{j=1}^{d-1} I_j\times R_d}$, right subchain $\boldsymbol{\mathcal{Q}}^{>d} \in \mathbb{R}^{R_{d+1}\times\prod_{j=d+1}^{D} I_j\times R_1}$ are defined as $\boldsymbol{Q}^{<d}[\overline{i_1\cdots i_{d-1}}] = \prod_{j=1}^{d-1}\boldsymbol{Q}^{(j)}[i_j]$ and $\boldsymbol{Q}^{>d}[\overline{i_{d+1}\cdots i_D}] = \prod_{j=d+1}^{D}\boldsymbol{Q}^{(j)}[i_j]$. Similarly, we can define $\boldsymbol{\mathcal{Q}}^{\neq d} \in \mathbb{R}^{R_{d+1}\times\prod_{j=1,j\neq d}^{D} I_j\times R_d}$. If $\boldsymbol{\mathcal{X}}$ admits the TR format (2), then

$$\boldsymbol{X}_{[d]} = \boldsymbol{Q}_{(2)}^{(d)}(\boldsymbol{Q}_{[2]}^{\neq d})^\intercal, \quad \forall d = 1, \ldots, D, \tag{3}$$

where $\boldsymbol{Q}_{(2)}^{(d)}$ and $\boldsymbol{Q}_{[2]}^{\neq d}$ are classical mode-2 unfolding and mode-2 unfolding respectively [10].

3 Bayesian Tensor Ring Latent Factor Model

In this section, we introduce the proposed Bayesian tensor ring latent factor (TRLF) model. By using the TR format, TRLF is suitable to model high dimensional data. Moreover, by using MGP prior, our model can obtain low-rank and sparse factors.

Model Formulation In stead of finding low-rank matrix factorization of the covariance matrix, i.e., Eq. (1), we generalize the latent factor model to higher-order data. Now suppose the observed data is an order-D tensor $\boldsymbol{\mathcal{Y}}^{(n)} \in \mathbb{R}^{P_1\times\cdots\times P_D}$. For N observations, we stack them into an order-$(D+1)$ tensor, denoted as $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{P_1\times\cdots\times P_D\times N}$. Then we extend Eq. (1) using the TR format, namely,

$$\boldsymbol{\mathcal{Y}} = \ll \boldsymbol{\mathcal{Q}}^{(1)}, \ldots, \boldsymbol{\mathcal{Q}}^{(D)}, \boldsymbol{\eta} \gg + \boldsymbol{\mathcal{E}}, \tag{4}$$

where $\boldsymbol{\mathcal{Q}}^{(d)} \in \mathbb{R}^{R_d\times P_d\times R_{d+1}}$ are loading core tensors, $\boldsymbol{\eta} \in \mathbb{R}^{R_D\times N\times R_1}$ is the latent factor and $\boldsymbol{\mathcal{E}}$ is the noise tensor with the same size of the data. Since the latent factors are matrices here, we assume that they follow the standard matrix Normal distribution, namely,

$$\boldsymbol{\eta}^{(n)} \sim \mathcal{MN}(\boldsymbol{0}, \boldsymbol{I}_{R_{D+1}}, \boldsymbol{I}_{R_1}), \quad \forall n = 1, \ldots, N. \tag{5}$$

Moreover, we suppose all the noises independently follow Gaussian distribution,

$$\boldsymbol{\mathcal{E}}_{p_1\cdots p_D}^{(n)} \sim \mathcal{N}(0, \tau^{-1}), \quad \forall n = 1, \ldots, N. \tag{6}$$

Note that instead of one single loading matrix $\boldsymbol{W}$ in Eq. (1), our model represents the loading matrices by D loading core tensors $\{\boldsymbol{\mathcal{Q}}^{(d)}\}_{d=1}^{D}$. Such construction has two main advantages. First, the core tensors have a much more compact form. If we vectorize the data $\boldsymbol{\mathcal{Y}}$, the size of the loading matrix grows exponentially with tensor order D. However, by directly tackle with the tensors, the parameter numbers grow linearly with D. Second, in our model, the core tensors are stacked in a deep and hierarchical manner, which can capture complex multi-linear relations.

Despite the enormous size of parameters, the naïve vectorization dismisses the mode-specific relations of tensor data. To this end, we introduce the covariance tensor for tensor data $\boldsymbol{\mathcal{Y}}^{(n)}$, i.e., $\boldsymbol{\mathcal{V}}_{p_1\cdots p_D p'_1\cdots p'_D} = \mathrm{var}(\boldsymbol{\mathcal{Y}}^{(n)}_{p_1\cdots p_D}, \boldsymbol{\mathcal{Y}}^{(n)}_{p'_1\cdots p'_D})$, where $\boldsymbol{\mathcal{V}}$ is order-$(2D)$. By adopting (4), we have

$$\boldsymbol{\mathcal{V}}_{p_1\cdots p_D p'_1\cdots p'_D} = \tau^{-1} + \mathrm{tr}\left(\boldsymbol{Q}^{(1)}[p_1]\cdots\boldsymbol{Q}^{(D)}[p_D]\cdot(\boldsymbol{Q}^{(D)}[p'_D])^{\intercal}\cdots(\boldsymbol{Q}^{(D)}[p'_D])^{\intercal}\right).$$

The low-rank covariance tensor follows a symmetric TR format. If we reshape $\boldsymbol{\mathcal{V}}$ to matrix form, this is a matrix-TR format, which is a much more expressive extension of the Kronecker structure [7]. Indeed, given Eq. (3), (4), (5) and (6), by proper permutations and reshapings, we have

$$\mathrm{vec}(\boldsymbol{\mathcal{Y}}^{(n)}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{V}), \tag{7}$$

where $\boldsymbol{V} = \boldsymbol{Q}^{\leq D}_{[2]}(\boldsymbol{Q}^{\leq D}_{[2]})^{\intercal} + \mathrm{diag}(\tau^{-1})$.

Prior Distributions To get sparse loading core tensors, we extend the Multiplicative Gamma Process (MGP) [1] to multi-way scenario. For each elements of the core tensors, we assume

$$\boldsymbol{Q}^{(d)}_{jh}[i] \mid \phi^{(d)}_{jih}, u^{(d)}_{j}, u^{(d+1)}_{h} \sim \mathcal{N}\left(0, (\phi^{(d)}_{jih})^{-1}(u^{(d)}_{j})^{-1}(u^{(d+1)}_{h})^{-1}\right),$$

for $i = 1,\ldots,P_d$, $j = 1,\ldots,R_d$, $h = 1,\ldots,R_{d+1}$ and $d = 1,\ldots,D$, where $\{u^{(d)}\}_{d=1}^{D}$ are global shrinkage prior to induce sparse and low-rank estimators and $\{\phi^{(d)}\}_{d=1}^{D}$ are local shrinkage prior to prevent the model from over shrinkage. Then, we put the MGP on the global shrinkage parameters $\boldsymbol{u}$,

$$u^{(d)}_{h} = \prod_{l=1}^{h}\delta^{(d)}_{l}, \quad \delta^{(d)}_{l} \sim Ga(\alpha_\delta, 1),$$

where α_δ is set larger than 1 to encourage sparsity. Furthermore, the local shrinkage follows Gamma distribution $\phi^{(d)}_{jih} \sim Ga(\nu,\nu)$. Finally, we assume the noise precision follows $\tau \sim Ga(\alpha_\tau, \beta_\tau)$.

Identifiability As most latent factor models, the proposed model is not identifiable. To be specific, if we apply some orthogonal transformations on the neighborhood core tensor, for instance, let $\tilde{\boldsymbol{Q}}^{(d)}[i] = \boldsymbol{Q}^{(d)}[i]\boldsymbol{P}^{\intercal}$ and $\tilde{\boldsymbol{Q}}^{(d+1)}[j] = \boldsymbol{P}\boldsymbol{Q}^{(d)}[j]$, where $\boldsymbol{P}^{\intercal}\boldsymbol{P} = \boldsymbol{I}$, we have $\tilde{\boldsymbol{Q}}^{(d)}[i]\cdot\tilde{\boldsymbol{Q}}^{(d+1)}[j] = \boldsymbol{Q}^{(d)}[i]\cdot\boldsymbol{Q}^{(d+1)}[j]$ and the covariance estimation does not change. This unidentifiability sometimes makes the posterior hard to be optimized. To this end, we adopt the parameter-expansion technique to optimize the transformation and establish a PX-EM algorithm [8, 11].

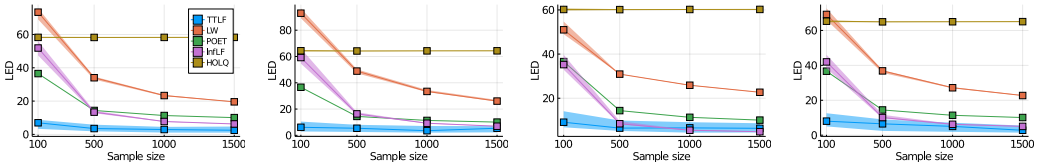
4 Experiments

4.1 Covariance Estimation

We investigate the covariance estimation ability of our model on both synthetic and real data. We compare our model with the following baselines: 1). **LW**, a James-Stein type shrinkage model[6]. 2). **POET** [3], which is a low rank matrix model. 3). **InfLF** [1], the Infinite Latent Factor model, which can be regarded as a vector form of our model. 4). **HOLQ** [5], the Higher-Oder LQ decomposition, which is a generalization of Tucker decomposition. To evaluate the performance, we use the Log-Euclidean Distance (LED) [9], which is a distance for symmetric positive definite matrices.

We consider data of feature length 1000 and different sample sizes. We pick 4 kind of loading matrices, 1). EXP, which is generated using Exponential functions, e.g., $\boldsymbol{W}_{exp}(i,j) = a\exp(-(i-j)^2/b)$, and then the covariance matrix is computed by $\boldsymbol{V}_{exp} = \boldsymbol{W}_{exp}\boldsymbol{W}'_{exp} + \tau\boldsymbol{I}$. 2). PED, which is a Periodic function $\boldsymbol{W}_{ped}(i,j) = a\exp(-\sin^2(\pi|i-j|)/b)$ and $\boldsymbol{V}_{ped} = \boldsymbol{W}_{ped}\boldsymbol{W}'_{ped} + \tau\boldsymbol{I}$. 3). LIN

$\otimes$ EXP, which is computed by $\boldsymbol{V}_{lin\otimes ped} = \boldsymbol{W}_{lin} \otimes \boldsymbol{W}_{ped}\boldsymbol{W}'_{ped} + \tau\boldsymbol{I}$, where $\boldsymbol{W}_{lin}(i,j) = ai \cdot j$. 4). LIN $\otimes$ EXP, which is computed by $\boldsymbol{V}_{lin\otimes ped} = \boldsymbol{W}_{lin} \otimes \boldsymbol{W}_{ped}\boldsymbol{W}'_{ped} + \tau\boldsymbol{I}$. We set $\tau = 1\mathrm{e}{-3}$ for all the cases. Finally we sample the data from Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{V})$. We choose sample size of $N = \{100, 500, 1000, 1500\}$ and repeat every experiment for 50 times. The results are shown in Figure 1. We plot the median value, $1/4$ and $3/4$ quantiles of all the experiments. For the simulation data, our model outperforms the baseline models, especially when the sample size is small.



(a) EXP (b) PED (c) LIN $\otimes$ EXP (d) LIN $\otimes$ PED

Figure 1: Results of covariance estimation in synthetic data analysis. Each subfigure shows results of different shapes. The x-axis is sample sizes and the y-axis is the LED.

4.2 Supervised Learning with Extracted Factors

In this subsection, we show that the TR/TTLF model actually learn some meaningful latent factors of the original data. We use the U.S. Postal Service (USPS) data to illustrate the experiments[1]. This dataset totally consists of 9298 grayscale images of the handwritten digits from 0 to 9. For each of the images, the shape is 16×16 and the value ranges from -1.0 to 1.0. The whole dataset is split into a training set of size 7291 and a test set of size 2007. We compare our model with the InfLF and tensorize each image to $4 \times 4 \times 4 \times 4$ for our model.

We firstly use TR/TTLF model to extract the latent factors without using the information about the labels. Then we feed the learned latent factors to train a SVM classifier. The results are shown in Figure 2. We can see that the classification accuracy increases as the factor number growing. All the latent factor models improve the performance of the vanilla SVM and the TTLF has the best results. The classification experiment reveals that our model is potential as an unsupervised data preprocessing method. It reduces the feature dimension significantly while increasing the classification accuracy.

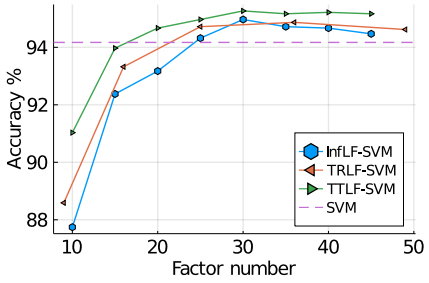


Figure 2: Classification accuracy.

5 Conclusion and Future Work

In this paper, we try to combine the Bayesian latent factor model with TNs. By assuming the covariances are highly structured and can be approximated by TT/TR format, we design the TRLF model to extract latent factors of the original data. We adopt the MGP prior to impose low-rank and sparse latent factors simultaneously and designed efficient PX-EM algorithm to find the MAP estimate. Results show that our model outperforms in several high-dimensional modeling problems. For future research, we are interested in several directions: 1) Non-linear extensions of our model using neural networks; 2) Scalable inference algorithms for large datasets, such as amortized inference; 3) Exploring the TN representations of covariance matrices in other fields.

Acknowledgements

Zerui Tao was supported by the RIKEN Junior Research Associate (JRA) Program. Qibin Zhao was supported by JSPS KAKENHI (Grant No. 20H04249, 20H04208).

[1] `https://web.stanford.edu/~hastie/StatLearnSparsity_files/DATA/zipcode.html`

References

[1] A. Bhattacharya and D. B. Dunson. Sparse bayesian infinite factor models. *Biometrika*, 98(2):291–306, 2011.

[2] Carlos Marinho Carvalho, Jeffrey Chang, Joseph E. Lucas, Joseph R. Nevins, Quanli Wang, and Mike West. High-dimensional sparse factor modeling: Applications in gene expression genomics. *Journal of the American Statistical Association*, 103(484):1438–1456, 2008.

[3] Jianqing Fan, Yuan Liao, and Martina Mincheva. Large covariance estimation by thresholding principal orthogonal complements. *Journal of The Royal Statistical Society Series B-statistical Methodology*, 75(4):603–680, 2013.

[4] Emily B Fox and David B Dunson. Bayesian nonparametric covariance regression. *The Journal of Machine Learning Research*, 16(1):2501–2542, 2015.

[5] David Gerard and Peter Hoff. A higher-order lq decomposition for separable covariance models. *Linear Algebra and its Applications*, 505:57–84, 2016.

[6] Olivier Ledoit and Michael Wolf. Honey, i shrunk the sample covariance matrix. *The Journal of Portfolio Management*, 30(4):110–119, 2004.

[7] Ivan V Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[8] Veronika Ročková and Edward I George. Fast bayesian factor analysis via automatic rotations to sparsity. *Journal of the American Statistical Association*, 111(516):1608–1622, 2016.

[9] Raviteja Vemulapalli and David W. Jacobs. Riemannian metric learning for symmetric positive definite matrices. *arXiv preprint arXiv:1501.02393*, 2015.

[10] Qibin Zhao, Guoxu Zhou, Shengli Xie, Liqing Zhang, and Andrzej Cichocki. Tensor ring decomposition. *arXiv preprint arXiv:1606.05535*, 2016.

[11] Shiwen Zhao, Chuan Gao, Sayan Mukherjee, and Barbara E. Engelhardt. Bayesian group factor analysis with structured sparsity. *Journal of Machine Learning Research*, 17(1):6868–6914, 2016.