LEARNING STRUCTURALLY DISTINCT FACTORS IN MANY-BODY APPROXIMATION FOR TENSORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Decomposing tensors in an interpretable manner is essential in data analysis and knowledge discovery. *Many-body approximation* (MBA) for non-negative tensors is a tensor decomposition method that allows each factor to be interpreted as an interaction among a subset of modes of a given tensor. In this paper, we first uncover a structural dependency in MBA: factors containing common modes necessarily share identical slices, regardless of the model parameters. To remove this structural dependency while retaining the convex optimization procedure of MBA, we propose extended many-body approximation (EMBA), which extends each mode of the input tensor by one dimension before applying MBA. We prove that entries selected from different EMBA factors are not identical as functions of the parameters, thereby eliminating the structural dependency present in MBA. Experiments on reconstruction using reduced set of factors and recursive approximation show the improved explanatory capacity and robustness of EMBA over MBA. In factor quantization experiments, EMBA almost perfectly reconstructs synthetic data generated from Boolean factors even under 1-bit quantization. These results demonstrate that EMBA enables structurally distinct factors whose utility extends beyond full-tensor reconstruction, highlighting the importance of factor structure in tensor decomposition.

1 INTRODUCTION

Tensor decomposition (Kolda & Bader, 2009) approximates a higher-order tensor by combining simpler tensors and is used in applications such as signal processing (Sidiropoulos et al., 2017) and knowledge-graph analysis (Nickel et al., 2015; 2011). Conventional methods, including CP (Harshman et al., 1970) and Tucker decompositions (Tucker, 1966), seek low-rank components. Although such components can sometimes be interpreted (Williams et al., 2018), their latent rank structure offers limited control over explicit interactions among observed modes. Many-body approximation (MBA) for non-negative tensors (Ghalamkari et al., 2023) has been proposed as a new form of tensor decomposition that does not rely on low-rank components. MBA is an interaction-based tensor decomposition derived from Legendre decomposition (Sugiyama et al., 2018). Unlike low-rank methods, MBA introduces neither a latent rank dimension nor a component index; instead, it represents a tensor through *factors* associated with specified subsets of the observed *modes* (e.g., time, space, and features). For example, given a (normalized) fourth-order tensor $\mathbf{T}$ [1] , MBA can construct the following *3-body approximation* of $\mathbf{T}$:

$$T_{i_1,i_2,i_3,i_4} \simeq P_{i_1,i_2,i_3,i_4} = X^{(1,2,3)}_{i_1,i_2,i_3} X^{(1,2,4)}_{i_1,i_2,i_4} X^{(1,3,4)}_{i_1,i_3,i_4} X^{(2,3,4)}_{i_2,i_3,i_4}, \tag{1}$$

where $\mathbf{P}$ denotes the approximation and $\mathbf{X}$ represents a factor; for example, $\mathbf{X}^{(1,2,3)}$ captures the interaction between the first, second and third modes. Here, each indexed expression, such as $X^{(1,2,3)}_{i_1,i_2,i_3}$, denotes a scalar entry of the corresponding factor tensor, and the product in equation 1 is ordinary scalar multiplication, involving neither summation nor tensor contraction. Because each factor is indexed by observable modes, MBA factors admit mode-level interpretation; their utility as reusable factors remains unexplored.

[1] In this paper, tensors are denoted by symbols such as $\mathbf{T}$, and their entries by the corresponding symbols such as T_{i_1,i_2,i_3}.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

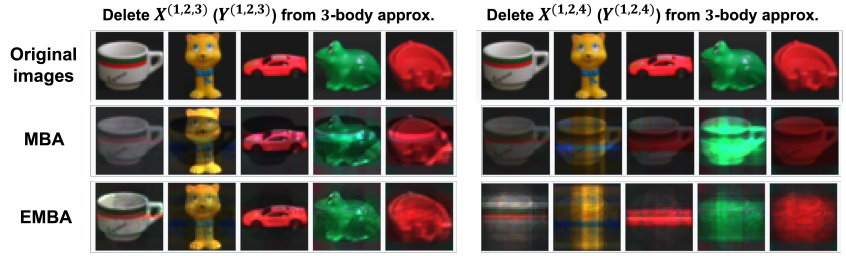


Figure 1: Factor deletions in 3-body approximation of a fourth-order tensor constructed from images, where image ID is the fourth mode. $\mathbf{X}$ and $\mathbf{Y}$ denote factors obtained by MBA and EMBA, respectively. Left: MBA and EMBA reconstructions without $\mathbf{X}^{(1,2,3)}$ and $\mathbf{Y}^{(1,2,3)}$, respectively, which capture horizontal position, vertical position, and color. Right: MBA and EMBA reconstructions without $\mathbf{X}^{(1,2,4)}$ and $\mathbf{Y}^{(1,2,4)}$, respectively, which capture horizontal position, vertical position, and image ID.

The factors obtained by MBA can serve as meaningful building blocks that capture underlying structures and interactions in data, and thus can be potentially useful for downstream tasks such as interpretation, compression, or binary representation learning. More generally, this perspective raises the question of whether the factors produced by a tensor decomposition possess desirable structure in their own right, beyond their ability to reconstruct the input tensor. This motivates us to go beyond full-tensor reconstruction performance and investigate whether factors remain useful when they are used individually, partially retained, recursively approximated, or quantized. In this work, we first analyze the factors learned by MBA and reveal that factors sharing common modes can contain slices that are deterministically identical as functions of the parameters. Based on this observation, we propose *Extended Many-Body Approximation* (*EMBA*), which eliminates structural dependencies between factors by extending an input tensor before applying MBA, and then extracting only the components that contribute to the original tensor. The extended tensor is obtained from the original tensor by adding one extra dimension to each mode. We show that this simple extension-and-extraction procedure enables EMBA to learn factors without the dependencies that arise in MBA.

Our theoretical results show that EMBA removes the forced cross-factor equalities characterized in Section 3. Since EMBA avoids identical slice sharing between factors, removing a factor causes less unintended structural collapse in reconstruction. Here, one such result is shown in Figure 1, which illustrates the effect of factor removal on color images (COIL-100 dataset (Nene et al., 1996)). After removing individual factors, MBA causes structural collapse in the reconstructed images, including shadows of the cup from the first image and unintended homogenization of object shapes toward that image. In contrast, EMBA preserves the original structures more faithfully, indicating that information is distributed more robustly across factors.

To see the advantages of EMBA other than the robustness for factor removal, we will further consider two additional analyses of semantic properties of EMBA. First, we consider recursive applications of (E)MBA, in which higher-order factors are recursively decomposed into lower-order factors. Experimental results show that EMBA maintains reconstruction quality more effectively than MBA even after recursive approximations, suggesting that EMBA learns more meaningful structures whose semantic information remains preserved even when they are further decomposed into lower-order interactions. Second, we investigate the robustness to quantization of the learned factors. EMBA exhibits smaller quantization errors than MBA, which we attribute to the reduced structural redundancy and increased diversity of the learned factors.

2 Related Work

Conventional tensor decomposition methods are primarily based on low-rank structures (Kolda & Bader, 2009). CP decomposition (Harshman et al., 1970) approximates a tensor as a sum of rank-one tensors. Tucker decomposition (Tucker, 1966) can be viewed as a tensor extension of singular value decomposition (SVD) (Eckart & Young, 1936), providing a subspace-based representation of the tensor. Building on these approaches, more recent methods such as tensor train (TT) (Oseledets, 2011) introduce a structured low-rank representation, expressing a tensor as a sequence of low-order

tensors. Tensor networks (Orús, 2014) have been proposed as a more general framework, representing tensors as networks and unifying various low-rank decompositions. Tensor decomposition for structure learning and efforts to improve the interpretability of tensor decomposition have been extensively studied. Non-negative tensor factorization (NTF) (Cichocki et al., 2008), which imposes non-negativity, has been proposed to improve applicability to probabilistic tasks. Boolean tensor factorization (Miettinen, 2011) models tensors using logical operations and binary factors, enabling the extraction of discrete and interpretable patterns. Methods such as RESCAL (Nickel et al., 2011), DistMult (Yang et al., 2014), and ComplEx (Trouillon et al., 2017) have been proposed to capture relational structures and apply knowledge graphs. These methods still fundamentally rely on low-rank structures and represent tensors as combinations of latent components, without explicitly modeling higher-order interactions among modes. Beyond reconstruction, interpreting tensor-decomposition factors as meaningful structures for data understanding is also relevant to interpretable data analysis and explainable AI (Doshi-Velez & Kim, 2017; Lipton, 2018; Rudin, 2019).

Many-body approximation (MBA) for non-negative tensors (Ghalamkari et al., 2023) represents a different paradigm that does not rely on low-rank structure. Instead of learning low-rank structures, MBA decomposes a tensor into factors that explicitly model interactions among subsets of modes. In this framework, the user specifies which mode interactions to capture, rather than selecting a latent decomposition rank. This allows each factor to have a clear and interpretable meaning in terms of mode interactions, analogous to disentangled representation learning (Wang et al., 2024). Although MBA yields a multiplicative form related to direct factorizations of positive tensors (Aswani, 2016), it parameterizes hierarchical natural parameters that induce the factors, thereby defining an interaction-based exponential-family model. Our proposed method, EMBA, removes structural dependencies among MBA factors induced by this cumulative natural-parameter construction, improving their interpretability and explanatory power while preserving the interaction-based KL-projection framework. MBA can be viewed as a special case of Legendre decomposition (Sugiyama et al., 2018). MBA and Legendre decomposition are based on information geometry (Amari, 2016) and enable theoretical guarantees of convex optimality with respect to the KL divergence (Kullback & Leibler, 1951). In addition, efficient optimization can be achieved via natural gradient methods with quadratic convergence. EMBA improves the properties of the factors learned by MBA while preserving its theoretical guarantees, such as convex optimality and quadratic convergence. Recent work extends MBA in complementary directions, including deformed decomposition, which generalizes the divergence and algebra underlying MBA (Ghalamkari et al., 2026), and pseudo-non-linear data augmentation, which uses an information-geometric energy-based representation for controllable augmentation (Hu & Sugiyama, 2026). In contrast, EMBA addresses dependencies among factors in the original MBA parameterization.

3 Probing into Decomposed Factors in MBA

3.1 Many-body approximation for non-negative tensors

Many-body approximation (MBA) for non-negative tensors (Ghalamkari et al., 2023) learns a parameter tensor $\boldsymbol{\theta}$ of the same shape as the input tensor $\mathbf{T}$, and approximates $\mathbf{T}$ by a product of factors, each capturing an interaction among a subset of modes. For example, the 2-body approximation of a third-order tensor is $\mathsf{T}_{i_1,i_2,i_3} \simeq \mathsf{P}_{i_1,i_2,i_3} = \mathsf{X}^{(1,2)}_{i_1,i_2}\mathsf{X}^{(1,3)}_{i_1,i_3}\mathsf{X}^{(2,3)}_{i_2,i_3}$, where

$$\mathsf{X}^{(k,l)}_{i_k,i_l} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2}\sum_{i'_k=2}^{i_k}\boldsymbol{\theta}^{(k)}_{i'_k} + \sum_{i'_k=2}^{i_k}\sum_{i'_l=2}^{i_l}\boldsymbol{\theta}^{(k,l)}_{i'_k,i'_l} + \frac{1}{2}\sum_{i'_l=2}^{i_l}\boldsymbol{\theta}^{(l)}_{i'_l}\right). \tag{2}$$

Here, Z is a normalization constant, and $\boldsymbol{\theta}^{(k)}$ and $\boldsymbol{\theta}^{(k,l)}$ denote the entries of $\boldsymbol{\theta}$ whose indices other than k (and l) are 1. A parameter shared by k factors contributes to each of them with weight $1/k$ (the factors $1/2$ above); the general definition is given in Appendix A. MBA minimizes the KL divergence from $\mathbf{T}$ to $\mathbf{P}$ and inherits the convex optimality of Legendre decomposition (Sugiyama et al., 2018).

In the remaining part of this section, we investigate the factors learned by MBA from two viewpoints: which parts of different factors are forced to coincide, and to what extent they remain distinct as functions of $\boldsymbol{\theta}$. These findings motivate EMBA, which applies MBA to a tensor extended by one dimension per mode and extracts the part corresponding to the original tensor (Section 4).

3.2 Structural Restrictions in MBA

As can be seen in the 2-body approximation of equation 2, there exist cases in which two elements (or sets of elements) from different factors become identical as functions of $\boldsymbol{\theta}$. In such cases, these elements are always equal regardless of the specific values of $\boldsymbol{\theta}$. In this sense, these factors exhibit a structural coupling: selected entries are forced to coincide regardless of the learned parameter values. For example, in equation 2, the 1-body parameters $\boldsymbol{\theta}^{(1)}$, $\boldsymbol{\theta}^{(2)}$, and $\boldsymbol{\theta}^{(3)}$ are each shared between two factors. In particular, $\boldsymbol{\theta}^{(1)}$ is shared between $\mathbf{X}^{(1,2)}$ and $\mathbf{X}^{(1,3)}$. It follows from equation 2 that the following holds regardless of $\boldsymbol{\theta}$.

$$X^{(1,2)}_{i_1,1} = X^{(1,3)}_{i_1,1}, \quad X^{(1,2)}_{1,i_2} = X^{(2,3)}_{i_2,1}, \quad X^{(1,3)}_{1,i_3} = X^{(2,3)}_{1,i_3}. \tag{3}$$

A derivation of these relationships is provided in Appendix B.1, while a proof sketch is as follows: The first columns of $X^{(1,2)}$ and $X^{(1,3)}$ coincide, since both are computed solely from the first term inside the parentheses in equation 2. The other equalities can be derived analogously. In addition, we consider the following factorization as another example of this restriction.

$$T_{i_1,i_2,i_3,i_4} \simeq P_{i_1,i_2,i_3,i_4} = X^{(1,2,3)}_{i_1,i_2,i_3} X^{(1,2)}_{i_1,i_2} X^{(4)}_{i_4}.$$

In this case, the following holds regardless of $\boldsymbol{\theta}$.

$$X^{(1,2,3)}_{i_1,i_2,1} = X^{(1,2)}_{i_1,i_2}. \tag{4}$$

Its derivation is provided in Appendix B.2. In this example, the first slice of $\mathbf{X}^{(1,2,3)}$ is identical to $\mathbf{X}^{(1,2)}$ and therefore provides no additional degree of freedom. Generally, the following proposition holds for factors obtained by MBA.

Proposition 3.1 (Structurally identical slices in MBA factors)**.** *Let $\mathbf{X}^{(\boldsymbol{s})}$ and $\mathbf{X}^{(\boldsymbol{t})}$ be factors obtained by MBA and let $\boldsymbol{u} = \boldsymbol{s} \cap \boldsymbol{t}$ be their shared mode set. Let $\boldsymbol{s} \setminus \boldsymbol{u}$ and $\boldsymbol{t} \setminus \boldsymbol{u}$ be non-shared modes in $\mathbf{X}^{(\boldsymbol{s})}$ and $\mathbf{X}^{(\boldsymbol{t})}$, respectively. Then, the slices obtained by fixing the indices of all non-shared modes to 1 coincide, i.e.,*

$$X^{(\boldsymbol{s})}_{\boldsymbol{i}_u,\, \mathbf{1}_{\boldsymbol{s}\setminus\boldsymbol{u}}} = X^{(\boldsymbol{t})}_{\boldsymbol{i}_u,\, \mathbf{1}_{\boldsymbol{t}\setminus\boldsymbol{u}}},$$

for all parameter values $\boldsymbol{\theta}$.

A proof is provided in Appendix B.3. In EMBA, such equalities do not arise as will be discussed in the following section.

Here, we call an entry of a factor an *interior entry* if every coordinate of its index is at least 2. Although MBA factors can contain forced cross-factor equalities at their boundary entries, their interior entries are not identical as functions of $\boldsymbol{\theta}$.

Proposition 3.2 (Functionally distinct interior entries of MBA factors)**.** *Let a and b be entries drawn from two distinct MBA factors, with every coordinate of their respective factor indices at least 2. Then a and b are not identical as functions of $\boldsymbol{\theta}$. Equivalently, there exists a parameter value $\boldsymbol{\theta}$ for which $a \neq b$.*

A proof is given in Appendix B.4. Based on this proposition, we introduce EMBA in the next section.

4 Extended Many-Body Approximation

4.1 Detailed Formulation of EMBA

In EMBA, we first construct an extended tensor $\mathbf{T}^{(\text{extend})}$ by extending each mode of the original tensor $\mathbf{T}^{(\text{original})}$ by one additional dimension and filling the newly introduced entries with zeros. We define the extension operator $\text{Ext} : \mathbb{R}^{I_1 \times \cdots \times I_N}_{\geq 0} \to \mathbb{R}^{(I_1+1) \times \cdots \times (I_N+1)}_{\geq 0}$, which maps a tensor to its zero-padded extension. The extended tensor $\mathbf{T}^{(\text{extend})}$ is then defined as

$$T^{(\text{extend})}_{i_1,i_2,\ldots,i_N} = \left(\text{Ext}(T^{(\text{original})})\right)_{i_1,i_2,\ldots,i_N} = \begin{cases} 0 & \text{if any } i_k = 1, \\ T^{(\text{original})}_{i_1-1,i_2-1,\ldots,i_N-1} & \text{otherwise,} \end{cases} \tag{5}$$

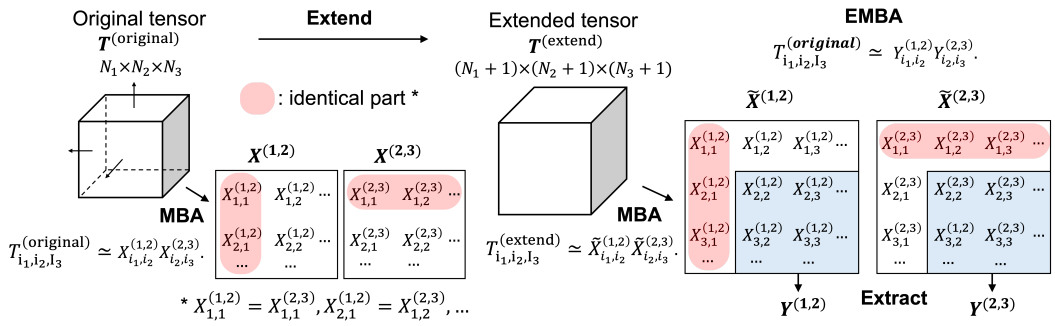


Figure 2: The overview of EMBA and an example of the dependence between MBA factors.

MBA is applied to the extended tensor $\mathbf{T}^{(\text{extend})}$ and the corresponding factors are learned as follows.

$$T^{(\text{extend})}_{i_1,i_2,\ldots,i_N} \simeq P^{(\text{MBA})}_{i_1,i_2,\ldots,i_N} = \widetilde{X}^{(\boldsymbol{s}_1)}_{\boldsymbol{i}_{\boldsymbol{s}_1}} \widetilde{X}^{(\boldsymbol{s}_2)}_{\boldsymbol{i}_{\boldsymbol{s}_2}} \ldots \widetilde{X}^{(\boldsymbol{s}_M)}_{\boldsymbol{i}_{\boldsymbol{s}_M}}, \tag{6}$$

where $\widetilde{\mathbf{X}}^{(\boldsymbol{s}_m)}$ denotes a factor tensor that captures the interaction among the modes specified by the index subset $\boldsymbol{s}_m$. Each subset $\boldsymbol{s}_m \subseteq \{1, \ldots, N\}$ represents a group of modes, and $\widetilde{\mathbf{X}}^{(\boldsymbol{s}_m)}$ models their corresponding interaction. For example, when $\boldsymbol{s}_m = \{1, 2\}$, $\widetilde{\mathbf{X}}^{(1,2)}$ is a matrix that represents the interaction between the modes 1 and 2. M is the number of interactions (i.e., the number of factors). Next, we define the restriction operator $\text{Res} : \mathbb{R}_{\geq 0}^{(I_1+1)\times\cdots\times(I_N+1)} \to \mathbb{R}_{\geq 0}^{I_1\times\cdots\times I_N}$, which extracts the sub-tensor corresponding to the original domain. Using this operator, we get the unnormalized approximation of EMBA as follows:

$$P'^{(\text{EMBA})}_{i_1,i_2,\ldots,i_N} = \left(\text{Res}(P^{(\text{MBA})})\right)_{i_1,i_2,\ldots,i_N} = P'^{(\text{MBA})}_{i_1+1,i_2+1,\ldots,i_N+1} = Y^{(\boldsymbol{s}_1)}_{\boldsymbol{i}_{\boldsymbol{s}_1}} Y^{(\boldsymbol{s}_2)}_{\boldsymbol{i}_{\boldsymbol{s}_2}} \ldots Y^{(\boldsymbol{s}_M)}_{\boldsymbol{i}_{\boldsymbol{s}_M}},$$

where $\mathbf{Y} = \text{Res}(\widetilde{\mathbf{X}})$ is the restriction of $\widetilde{\mathbf{X}}$ to the original domain. Since $P^{(\text{EMBA})}$ is not normalized, we obtain the EMBA approximation by applying a normalization operator:

$$\mathbf{T}^{(\text{original})} \simeq \mathbf{P}^{(\text{EMBA})} = \text{Norm}\left(\mathbf{P}'^{(\text{EMBA})}\right), \tag{7}$$

where $\text{Norm}(\cdot)$ rescales a tensor so that its entries sum to one. Figure 2 describes the overview of EMBA and the algorithm is provided in Appendix C.

Throughout this paper, we distinguish between the factors obtained by MBA and EMBA, and refer to $\mathbf{X}$ as the factors learned by MBA and $\mathbf{Y}$ as those derived from EMBA.

4.2 Structural Advantages and Optimization Properties of EMBA

4.2.1 Absence of forced cross-factor equalities

In EMBA, only the part of the factor learned in MBA with indices greater than or equal to 2 is extracted as an EMBA factor $\mathbf{Y}$. Therefore, the distinctness stated in Proposition 3.2 holds for all elements in EMBA factors. This observation can be formalized as the following theorem.

Theorem 4.1 (Functionally distinct entries of EMBA factors)**.** *Any two entries drawn from distinct EMBA factors are not identical as functions of $\boldsymbol{\theta}$. Equivalently, for every such pair, there exists a parameter value $\boldsymbol{\theta}$ for which the two entries differ.*

Proof. By Proposition 3.2, any two elements of $\widetilde{\mathbf{X}}^{(\boldsymbol{s})}$ and $\widetilde{\mathbf{X}}^{(\boldsymbol{t})}$ whose indices are all greater than or equal to 2 are not identical as functions of $\boldsymbol{\theta}$. The EMBA factors $\mathbf{Y}^{(\boldsymbol{s})}$ and $\mathbf{Y}^{(\boldsymbol{t})}$ are obtained by extracting only these parts, namely the entries whose indices are all greater than or equal to 2, from $\widetilde{\mathbf{X}}^{(\boldsymbol{s})}$ and $\widetilde{\mathbf{X}}^{(\boldsymbol{t})}$, respectively. Therefore, any two elements drawn from different EMBA factors are not identical as functions of $\boldsymbol{\theta}$. □

By Theorem 4.1, after restriction to the original domain, no pair of entries from distinct EMBA factors is identical as a function of $\boldsymbol{\theta}$. Thus, the equality pattern in equation 3 does not arise in EMBA.

The following proposition shows that each EMBA factor entry can be associated with a distinct n-body parameter.

Proposition 4.2 (Injective parameter assignment in EMBA)**.** *Each entry of the EMBA factors can be injectively assigned to an n-body parameter optimized by EMBA. Consequently, the number of parameters optimized by EMBA is at least the total number of entries in the EMBA factors.*

A proof is given in Appendix B.5. In contrast, MBA does not admit such an injective assignment in general, because lower-order parameters can be shared across factors.

4.2.2 SCALABILITY

The computational complexity of MBA is polynomial in the number of parameters (Ghalamkari et al., 2023), which is upper bounded by the size of the input tensor. For an original tensor of size $I_1 \times \cdots \times I_N$, EMBA uses an extended tensor of size $\prod_{k=1}^{N}(I_k + 1)$, giving a size ratio of $\prod_{k=1}^{N}(1 + 1/I_k)$. For a fixed number of modes N, this ratio approaches 1 as the mode sizes increase; when N grows, it can instead increase multiplicatively. For a factor with scope $\boldsymbol{s}$, the corresponding overhead is $\prod_{k\in\boldsymbol{s}}(1+1/I_k)$. Thus, for fixed mode order and interaction order, EMBA preserves MBA's computational structure with a small overhead for large mode sizes. Appendix F.1 supports this prediction: MBA and EMBA have closely matched empirical scaling exponents over both sweeps and comparable per-Newton-step costs. EMBA requires more Newton iterations in our implementation, increasing total runtime by at most $6.3\times$ in the tested settings.

4.2.3 UPPER BOUND ON KL DIVERGENCE

The following proposition holds for the approximation obtained by EMBA.

Proposition 4.3 (KL bound under EMBA extraction)**.** *Let $\mathbf{T}^{(original)}$ be a given tensor, and let $\mathbf{P}^{(EMBA)}$ be the approximation obtained by EMBA as defined in equation 7. Let $\mathbf{T}^{(extend)}$ be the tensor constructed by equation 5, and let $\mathbf{P}^{(MBA)}$ be the approximation obtained by applying MBA to $\mathbf{T}^{(extend)}$ as in equation 6. Then, the following property holds.*

$$D_{\mathrm{KL}}\left(\mathbf{T}^{(extend)} \parallel \mathbf{P}^{(MBA)}\right) \geq D_{\mathrm{KL}}\left(\mathbf{T}^{(original)} \parallel \mathbf{P}^{(EMBA)}\right)$$

Proposition 4.3 separates the two optimization-related properties of EMBA. EMBA consists of three steps: extending the input tensor, applying MBA to the extended tensor, and extracting and normalizing the resulting approximation. The second step is exactly the standard MBA optimization applied to $\mathbf{T}^{(\mathrm{extend})}$ and therefore inherits MBA's convex optimality with respect to the extended-domain KL divergence. Proposition 4.3 concerns the third step: it shows that restricting and normalizing $\mathbf{P}^{(\mathrm{MBA})}$ cannot increase the KL divergence relative to the MBA approximation on the extended tensor. Thus, EMBA retains the convex MBA optimization in the extended domain while its additional extraction step is guaranteed not to worsen the KL divergence in the sense of Proposition 4.3. A detailed proof is provided in Appendix B.6.

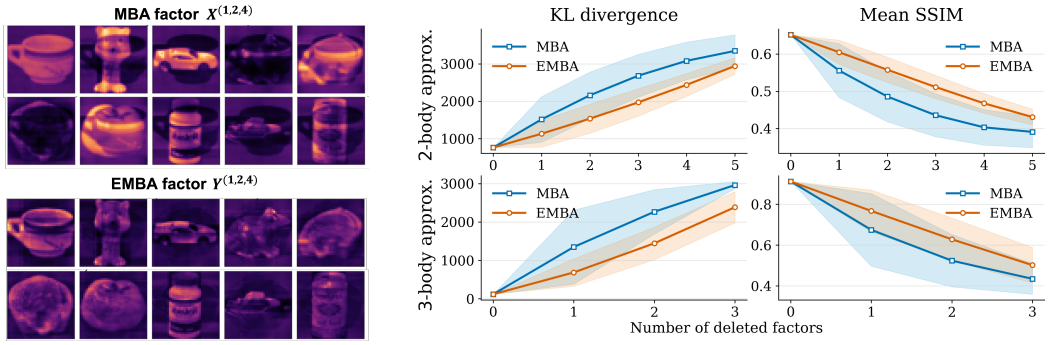
From an information-geometric perspective (Amari, 2016), EMBA can be viewed as lifting the MBA optimization problem to a higher-dimensional manifold with additional degrees of freedom, followed by restriction to the original domain. The proposition does not compare EMBA with MBA fitted directly to the original tensor; this comparison is evaluated empirically in Section 5.

5 EXPERIMENTS

In this section, we compare EMBA with MBA to assess the explanatory capacity of individual factors and to examine whether the learned factors preserve semantic structure under recursive approximation and quantization. Unless otherwise stated, MBA and EMBA are optimized using the common optimization settings described in Appendix D.3.

Table 1: Results when only one factor is retained. Values are mean ± standard deviation over factors.

Approximation	KL divergence ↓		SSIM ↑	
	MBA	EMBA	MBA	EMBA
1-body approx.	**3203.0** ± 200.9	**3203.0** ± 200.9	**0.408** ± 0.010	**0.408** ± 0.010
2-body approx.	3346.4 ± 431.5	**2942.3** ± 222.0	0.391 ± 0.042	**0.431** ± 0.021
3-body approx.	2963.0 ± 94.2	**2384.9** ± 406.4	0.434 ± 0.076	**0.502** ± 0.087



(a) Factor visualization

(b) Number of deleted factors and approximation results.

Figure 3: Performance comparison on the COIL-100 dataset.

5.1 Explanatory Capacity of Individual Factors

In this experiment, we evaluate reconstruction quality after removing factors from the learned tensor decomposition. This measures how well factors can be reused in isolation or in partial products after training. We regard this as an operational measure of factor-level explanatory utility, rather than as a direct measure of semantic structure. Following the setup used in (Ghalamkari et al., 2023), we select 10 images from the COIL-100 dataset (Nene et al., 1996) and construct a tensor of size $40 \times 40 \times 3 \times 10$. In addition to the KL divergence, we employ the Structural Similarity Index Measure (SSIM) (Wang et al., 2004). SSIM is computed by averaging local similarity values over spatial locations using an 11×11 Gaussian sliding window with standard deviation $\sigma = 1.5$, following the standard setting in (Wang et al., 2004): $\mathrm{SSIM}(\mathbf{d}, \hat{\mathbf{d}}) = \frac{1}{|\Omega|} \sum_{w \in \Omega} \mathrm{SSIM}_w(\mathbf{d}, \hat{\mathbf{d}})$, where Ω denotes the set of local windows. The details of SSIM_w are provided in Appendix E.1. SSIM is computed independently for each RGB channel, and the resulting values are averaged to obtain an image-level score. The final score is obtained by averaging all images.

Table 1 reports the approximation performance when reconstructing the data using each factor individually, obtained from n-body approximation ($n = 1, 2, 3$). In the case of the 1-body approximation, there is no sharing of modes between factors, and thus no difference is observed between MBA and EMBA. In contrast, for the 2-body and 3-body approximations on this COIL-100 construction, EMBA factors achieve better reconstruction performance when used individually.

Fig. 3(a) visualizes the 3-body factors $\mathbf{X}^{(1,2,4)}$ and $\mathbf{Y}^{(1,2,4)}$, which correspond to interactions among spatial coordinates and image IDs, excluding the color mode, learned by MBA and EMBA, respectively. Each slice corresponds to an individual image. In the factors learned by MBA, it can be observed that the first image appears as a shadow across all slices, indicating the presence of shared artifacts. In contrast, this phenomenon is no longer observed in the factors learned by EMBA. Fig. 3 (b) shows the results as a function of the number of retained factors. Even when multiple factors are retained instead of a single one, EMBA consistently performs better than MBA. We present some visualizations in Appendix F.4. This advantage is not limited to image tensors: on the real-world METR-LA traffic (Li et al., 2017) and EEG (Schalk, 2009) datasets, EMBA achieves lower mean KL divergence and higher mean cosine similarity than MBA for every number of deleted factors (Appendix F.2).

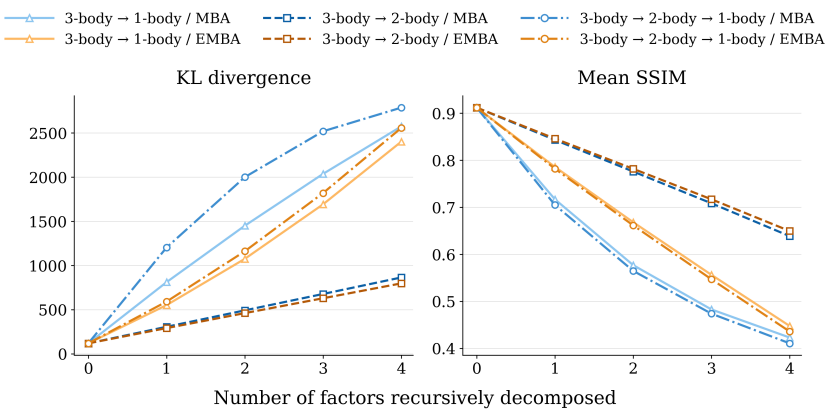


Figure 4: KL divergence and SSIM results of different ways of recursive approximation.

5.2 Robustness to Recursive Approximation

In this experiment, we investigate whether the learned factors preserve meaningful semantic information even after being recursively decomposed into lower-order factors. If a factor admits a compact lower-order approximation, reconstruction quality should remain high after the factor is recursively decomposed and substituted back into the original product. Therefore, robustness to recursive approximation can be interpreted as an indicator of how well each factor preserves meaningful structures of the underlying data.

We apply the 3-body approximation using MBA and EMBA to a tensor constructed from the COIL-100 dataset. Subsequently, depending on the experimental setting, we further apply MBA or EMBA to the obtained 3-body factors. In the setting of (3-body $\rightarrow$ 1-body), a subset of the 3-body factors is approximated using the 1-body approximation, and the resulting approximations are used in place of the original 3-body factors for data reconstruction. For example, if the method is MBA and all 3-body factors are decomposed recursively, we perform a multi-stage approximation as follows:

$$T_{i_1,i_2,i_3,i_4} \simeq P_{i_1,i_2,i_3,i_4} = X^{(1,2,3)}_{i_1,i_2,i_3} X^{(1,2,4)}_{i_1,i_2,i_4} X^{(1,3,4)}_{i_1,i_3,i_4} X^{(2,3,4)}_{i_2,i_3,i_4} \simeq \widehat{X}^{(1,2,3)}_{i_1,i_2,i_3} \widehat{X}^{(1,2,4)}_{i_1,i_2,i_4} \widehat{X}^{(1,3,4)}_{i_1,i_3,i_4} \widehat{X}^{(2,3,4)}_{i_2,i_3,i_4},$$

where $\widehat{\mathbf{X}}^{(k,l,m)}$ is the approximation of $\mathbf{X}^{(k,l,m)}$ by 1-body approximation. In addition to the setting of (3-body $\rightarrow$ 1-body), we also consider the setting of (3-body $\rightarrow$ 2-body), as well as a two-stage setting of (3-body $\rightarrow$ 2-body $\rightarrow$ 1-body). As shown in Fig. 4, EMBA consistently outperforms MBA in all settings. These results show that, under the recursive approximation procedures considered here, EMBA factors retain reconstruction quality more effectively than MBA factors. This property is practically desirable, as it suggests that the learned representation admits a hierarchical and compositional structure, enabling further decomposition for model compression, facilitating multi-stage modeling, and allowing individual factors to be reused or transferred across different tasks.

5.3 Robustness to Quantization

To evaluate the robustness to quantization, we quantized the factors obtained by decomposing the COIL-100 dataset and synthetic boolean tensors. Quantization was performed using 1-bit and 2-bit. For each factor, we optimized quantization thresholds by coordinate descent (Tseng, 2001) to minimize the KL divergence of the reconstructed tensor. The representative value assigned to each quantization interval was the mean of the factor entries assigned to that interval. In the experiment with COIL-100 dataset, we include non-negative Tucker decomposition (NNTucker) (Kim & Choi, 2007), non-negative tensor-train factorization (NTTF) (Shcherbakova & Tyrtyshnikov, 2019), and non-negative tensor ring decomposition (NTR) (Yu et al., 2021) for comparison. For fairness, the rank of each of these methods is set to the smallest uniform rank whose storage in bytes exceeds that of EMBA. A threshold is optimized for each core tensor or factor tensor. Under the quantization procedure used here, EMBA yields lower reconstruction KL divergence than MBA at both 1-bit and 2-bit precision. We further evaluate the storage–accuracy trade-off against these methods across a sweep of model ranks (Appendix F.3). At the storage budget of EMBA, EMBA achieves the lowest

Table 2: KL divergence after factor/core quantization for NNTucker, NTTF, NTR, MBA, and EMBA. Baselines: the smallest uniform rank whose storage exceeds that of EMBA; KL divergence and ratio are the mean over 5 random seeds for NNTucker, NTTF, and NTR (MBA and EMBA are deterministic). Bytes: storage of the quantized model.

			KL divergence ↓						
				(2) Quantized		Ratio: (2)/(1)		Bytes	
Dataset	Method	Form/Rank	(1) Unquantized	2bit	1bit	2bit	1bit	2bit	1bit
COIL-100	NNTucker	(13,13,13,13)	180.83	744.25	1811.69	4.12	10.02	7,722.5	3,841.25
	NTTF	(24,24,24)	75.65	822.50	1573.18	10.87	20.79	6,716	3,342
	NTR	(17,17,17,17)	**29.83**	988.70	1454.78	33.15	48.77	6,943.25	3,455.62
	MBA	3-body approx.	119.41	551.19	1316.88	4.62	11.03	6,584	3,236
	EMBA	3-body approx.	119.41	**434.33**	**932.70**	**3.64**	**7.81**	6,584	3,236

Table 3: Accuracy after quantization for MBA and EMBA on a synthetic boolean tensor generated as $\mathcal{T}_{i_1,i_2,i_3} = \mathcal{A}_{i_1,i_2}\mathcal{B}_{i_2,i_3}$ with $\mathcal{A}_{i_1,i_2}, \mathcal{B}_{i_2,i_3} \sim \mathrm{Bernoulli}(p_{\mathrm{one}})$, averaged over $p_{\mathrm{one}} \in \{0.3, 0.5, 0.7\}$.

				Quantized	
Dataset	Form	Method	Unquantized	2bit	1bit
Synthetic boolean tensor	2-body connected: $\mathbf{X}^{(1,2)}\mathbf{X}^{(2,3)}$	MBA	**1.000** $\pm$ 0.000	0.975 $\pm$ 0.011	0.834 $\pm$ 0.090
		EMBA	**1.000** $\pm$ 0.000	**0.995** $\pm$ 0.003	**0.993** $\pm$ 0.007
	2-body approx.: $\mathbf{X}^{(1,2)}\mathbf{X}^{(1,3)}\mathbf{X}^{(2,3)}$	MBA	**1.000** $\pm$ 0.000	0.878 $\pm$ 0.085	0.789 $\pm$ 0.109
		EMBA	**1.000** $\pm$ 0.000	**0.994** $\pm$ 0.002	**0.993** $\pm$ 0.006

quantized KL divergence among the compared decompositions (Table 2), and no swept configuration, even with larger storage, reaches a lower KL divergence (shown in Appendix F.3). These results show that the benefit of EMBA is not only robustness relative to MBA, but also improved compression quality at the storage budget of EMBA.

Next, we generate a third-order boolean tensor from randomly generated boolean matrices $\mathbf{A}$ and $\mathbf{B}$ as follows. Assuming that the underlying data is governed by 2-body interactions, we evaluate how well MBA and EMBA recover this structure, binarizing the approximated tensors with a threshold of 0.5 and reporting accuracy instead of KL divergence. As shown in Table 3, EMBA almost perfectly reconstructs the original data in both the 1 and 2-bit settings, whereas MBA exhibits relatively large errors. Moreover, when a redundant factor $\mathbf{X}^{(1,3)}$ is introduced (2-body approximation), the performance of MBA further degrades, while that of EMBA remains almost unchanged. These results show that EMBA preserves the generated Boolean tensor more accurately after factor quantization in the tested synthetic settings. Visualizations of the factors learned in this experiment are provided in Appendix F.5.

6 Conclusion

In this paper, we proposed EMBA, an extension of MBA that incorporates an extended input tensor. We theoretically showed that EMBA retains the convex MBA optimization on the extended tensor while eliminating the forced cross-factor equalities inherent in MBA. Empirically, EMBA preserves reconstruction quality when factors are used individually or partially removed, and this factor-deletion advantage extends beyond images to traffic and EEG tensors (Secion 5.1, Appendix F.2). EMBA also remains robust under the recursive approximation (Secion 5.2) and achieves improved quantized reconstruction at the storage budget of EMBA relative to MBA and the evaluated low-rank baselines (Secion 5.3, Appendix F.3), while reconstructing the synthetically generated Boolean tensors with near-perfect accuracy after quantization. Finally, scalability experiments support the predicted preservation of MBA's asymptotic computational behavior (Appendix F.1). By removing deterministic cross-factor equalities, EMBA improves factor utility under deletion, recursive approximation, and quantization, providing evidence for improved explanatory capacity and robustness in the evaluated settings. In future work, it would be of interest to develop an information-geometric interpretation of the proposed tensor augmentation, as well as to explore applications of EMBA to boolean tensor factorization and discrete relational data.

AI USE STATEMENT

In this work, we used generative AI tools for editing the paper to improve readability, editing software code, and retrieving related work. We have **not** used generative AI tools for generating synthetic data, developing the core ideas of the proposed method, formulating the mathematical claims and definitions, devising the fundamental ideas of the proofs, interpreting results or designing the structure of the paper, and the other required-disclosure tasks are not applicable to this work.

We have reviewed all AI-assisted work. All AI-edited or AI-drafted text was read and revised by the authors, and all proofs were checked line by line by the authors. All AI-assisted code was tested by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work proposes a general tensor decomposition method. Our experiments use publicly available benchmark data (COIL-100, METR-LA, and the PhysioNet EEG Motor Movement/Imagery Database (EEGMMIDB)) and synthetic data generated by the authors, in accordance with the terms of use of each dataset. We did not collect any data ourselves. EEGMMIDB consists of EEG recordings from human volunteers, which are released by PhysioNet in de-identified form under the public license of the database; we use it only through the anonymous subject numbers it provides, and make no attempt to identify individuals. COIL-100 (object images) and METR-LA (speed readings from highway loop detectors) contain no human subjects or personal information. We do not foresee any direct ethical concerns.

REPRODUCIBILITY STATEMENT

The supplementary material contains the complete source code used for data preprocessing, MBA and EMBA optimization, baseline comparisons, quantization, and evaluation. The paper and appendix specify the model formulation, datasets, hardware and software environments, and optimization settings required to reproduce the experiments. To preserve anonymity during review, the code is provided only as supplementary material. If accepted, we will release the complete implementation as an open-source library.

REFERENCES

Shunichi Amari. *Information geometry and its applications*, volume 194. Springer, 2016.

Anil Aswani. Low-rank approximation and completion of positive tensors. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1337–1364, 2016.

Andrzej Cichocki, Rafal Zdunek, and Shun-ichi Amari. Nonnegative matrix and tensor factorization [lecture notes]. *IEEE signal processing magazine*, 25(1):142–145, 2008.

Fengyu Cong, Qiu-Hua Lin, Li-Dan Kuang, Xiao-Feng Gong, Piia Astikainen, and Tapani Ristaniemi. Tensor decomposition of eeg signals: A brief review. *Journal of neuroscience methods*, 248:59–69, 2015.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

K. Ghalamkari, M. Sugiyama, and Y. Kawahara. Many-body approximation for non-negative tensors. In *Advances in Neural Information Processing Systems*, volume 36, pp. 74077–74102, New Orleans, US, December 2023.

Kazu Ghalamkari, Petr Taborsky, and Morten Mørup. Deformed decomposition for non-negative tensors. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026.

Richard A Harshman et al. Foundations of the parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA working papers in phonetics*, 16(1):84, 1970.

Pingbang Hu and Mahito Sugiyama. Pseudo-non-linear data augmentation: A constrained energy minimization viewpoint. In *International Conference on Learning Representations*, volume 2026, pp. 114081–114103, 2026.

Yong-Deok Kim and Seungjin Choi. Nonnegative tucker decomposition. In *2007 IEEE conference on computer vision and pattern recognition*, pp. 1–8. IEEE, 2007.

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

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

Pauli Miettinen. Boolean tensor factorizations. In *2011 IEEE 11th International Conference on Data Mining*, pp. 447–456. IEEE, 2011.

Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. Columbia object image library (coil-100). Technical report, Technical report CUCS-006-96, 1996.

Maximilian Nickel, Volker Tresp, Hans-Peter Kriegel, et al. A three-way model for collective learning on multi-relational data. In *ICML*, volume 11, pp. 3104482–3104584, 2011.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

Román Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Annals of physics*, 349:117–158, 2014.

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

Tom Pollard, Benjamin E. Moody, Li-wei H. Lehman, Brian J. Gow, Chrystinne Fernandes, Chen Xie, Alistair Johnson, Roger G. Mark, and Thomas Heldt. PhysioNet as a global platform for biomedical research. *Nature Health*, 1(8):792–795, August 2026. ISSN 3005-0693. doi: 10.1038/s44360-026-00096-z. URL https://doi.org/10.1038/s44360-026-00096-z.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

Gerwin Schalk. EEG Motor Movement/Imagery Dataset. *PhysioNet*, September 2009. doi: 10.13026/C28G6P. URL https://doi.org/10.13026/C28G6P. Version 1.0.0.

Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. Bci2000: a general-purpose brain-computer interface (bci) system. *IEEE Transactions on biomedical engineering*, 51(6):1034–1043, 2004.

Elena Shcherbakova and Eugene Tyrtyshnikov. Nonnegative tensor train factorizations and some applications. In *International Conference on Large-Scale Scientific Computing*, pp. 156–164. Springer, 2019.

Nicholas D Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E Papalexakis, and Christos Faloutsos. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on signal processing*, 65(13):3551–3582, 2017.

M. Sugiyama, H. Nakahara, and K. Tsuda. Legendre decomposition for tensors. In *Advances in Neural Information Processing Systems 31*, pp. 8825–8835, Montréal, Canada, December 2018.

Théo Trouillon, Christopher R Dance, Éric Gaussier, Johannes Welbl, Sebastian Riedel, and Guillaume Bouchard. Knowledge graph completion via complex tensor factorization. *Journal of Machine Learning Research*, 18(130):1–38, 2017.

Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

Xin Wang, Hong Chen, Si'ao Tang, Zihao Wu, and Wenwu Zhu. Disentangled representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9677–9696, 2024.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Alex H Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I Ryu, Krishna V Shenoy, Mark Schnitzer, Tamara G Kolda, and Surya Ganguli. Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 98(6):1099–1115, 2018.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

YuYuan Yu, Kan Xie, Jinshi Yu, Qi Jiang, and Shengli Xie. Fast nonnegative tensor ring decomposition based on the modulus method and low-rank approximation. *Science China Technological Sciences*, 64(9):1843–1853, 2021.

A Many-Body Approximation for Non-Negative Tensors

This appendix gives the general MBA factor definition referenced in Section 3. For a non-empty subset of modes $\boldsymbol{s}$, the MBA factor $\mathbf{X}^{(\boldsymbol{s})}$ is defined from the natural parameter tensor $\boldsymbol{\theta}$ by

$$X^{(\boldsymbol{s})}_{\boldsymbol{i_s}} = \frac{1}{\sqrt[M]{Z}} \exp\left[\sum_{\emptyset \neq \boldsymbol{v} \subseteq \boldsymbol{s}} w^{(\boldsymbol{v})} \left(\sum_{\substack{\boldsymbol{i'_v} \in \downarrow \boldsymbol{i_v} \\ i'_{v_k} \geq 2,\ \forall k}} \boldsymbol{\theta}^{(\boldsymbol{v})}_{\boldsymbol{i'_v}} \right) \right] \tag{8}$$

where Z is the normalization constant and M is the number of factors. Here, $\boldsymbol{\theta}^{(\boldsymbol{v})}$ denotes the parameter tensor whose non-1 indices correspond to the modes in $\boldsymbol{v}$. For example, $\boldsymbol{\theta}^{(k)}_{i_k}$ is a 1-body parameter and $\boldsymbol{\theta}^{(k,l)}_{i_k,i_l}$ is a 2-body parameter.

The set $\downarrow \boldsymbol{i_v}$ contains all index vectors $\boldsymbol{i'_v}$ satisfying $i'_d \leq i_d$ for every $d \in \boldsymbol{v}$ (Sugiyama et al., 2018). The weight $w^{(\boldsymbol{v})}$ distributes the contribution of a parameter shared across multiple factors. If $\boldsymbol{\theta}^{(\boldsymbol{v})}$ is included in $q(\boldsymbol{v})$ factors, we set $w^{(\boldsymbol{v})} = 1/q(\boldsymbol{v})$. An n-body approximation includes one factor for every subset of n modes. For the 2-body approximation of a third-order tensor discussed in Section 3, equation 8 reduces to equation 2. Because MBA parameterizes the approximation through an exponentiated energy, it can also be viewed from an energy-based-model perspective (LeCun et al., 2006).

B Proofs and Derivations

B.1 Derivation of equation 3

We restate equation 3 for completeness:

$$X^{(1,2)}_{i_1,1} = X^{(1,3)}_{i_1,1},\ X^{(1,2)}_{1,i_2} = X^{(2,3)}_{i_2,1},\ X^{(1,3)}_{1,i_3} = X^{(2,3)}_{1,i_3}.$$

In 2-body approximation, MBA learns the following approximation $\mathbf{P}$ for the given tensor $\mathbf{T}$.

$$T_{i_1,i_2,i_3} \simeq P_{i_1,i_2,i_3} = X^{(1,2)}_{i_1,i_2} X^{(1,3)}_{i_1,i_3} X^{(2,3)}_{i_2,i_3},$$

where each $\mathbf{X}^{(k,l)}$ is computed from the parameter $\boldsymbol{\theta}$ as follows:

$$X^{(k,l)}_{i_k,i_l} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_k=2}^{i_k} \boldsymbol{\theta}^{(k)}_{i'_k} + \sum_{i'_k=2}^{i_k} \sum_{i'_l=2}^{i_l} \boldsymbol{\theta}^{(k,l)}_{i'_k,i'_l} + \frac{1}{2} \sum_{i'_l=2}^{i_l} \boldsymbol{\theta}^{(l)}_{i'_l} \right), \tag{9}$$

When $i_l = 1$, only the first term within parentheses in equation 9 remains, and thus the following holds:

$$X^{(k,l)}_{i_k,1} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_k=2}^{i_k} \boldsymbol{\theta}^{(k)}_{i'_k} \right). \tag{10}$$

On the other hand, when $i_k = 1$, only the third term within parentheses in equation 9 remains, and thus the following holds:

$$X^{(k,l)}_{1,i_l} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_l=2}^{i_l} \boldsymbol{\theta}^{(l)}_{i'_l} \right). \tag{11}$$

Therefore, we obtain the following:

$$X^{(1,2)}_{i_1,1} = X^{(1,3)}_{i_1,1} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_1=2}^{i_1} \boldsymbol{\theta}^{(1)}_{i'_1} \right),$$

$$X^{(1,2)}_{1,i_2} = X^{(2,3)}_{i_2,1} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_2=2}^{i_2} \boldsymbol{\theta}^{(2)}_{i'_2} \right),$$

$$X^{(1,3)}_{1,i_3} = X^{(2,3)}_{1,i_3} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2} \sum_{i'_3=2}^{i_3} \boldsymbol{\theta}^{(3)}_{i'_3} \right),$$

which establishes equation 3.

B.2 Derivation of equation 4

We restate equation 4 for completeness:

$$\mathsf{X}^{(1,2,3)}_{i_1,i_2,1} = \mathsf{X}^{(1,2)}_{i_1,i_2}.$$

In the example, MBA learns the following approximation P for the given tensor T.

$$\mathsf{T}_{i_1,i_2,i_3,i_4} \simeq \mathsf{P}_{i_1,i_2,i_3,i_4} = \mathsf{X}^{(1,2,3)}_{i_1,i_2,i_3}\mathsf{X}^{(1,2)}_{i_1,i_2}\mathsf{X}^{(4)}_{i_4},$$

where $\mathsf{X}^{(1,2)}$ and $\mathsf{X}^{(1,2,3)}$ are computed from the parameter $\boldsymbol{\theta}$ as follows:

$$\mathsf{X}^{(1,2)}_{i_1,i_2} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2}\sum_{i'_1=2}^{i_1}\boldsymbol{\theta}^{(1)}_{i'_1} + \frac{1}{2}\sum_{i'_1=2}^{i_1}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(1,2)}_{i'_1,i'_2} + \frac{1}{2}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(2)}_{i'_2}\right),$$

$$\mathsf{X}^{(1,2,3)}_{i_1,i_2,i_3} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2}\sum_{i'_1=2}^{i_1}\boldsymbol{\theta}^{(1)}_{i'_1} + \frac{1}{2}\sum_{i'_1=2}^{i_1}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(1,2)}_{i'_1,i'_2} + \frac{1}{2}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(2)}_{i'_2} + \sum_{i'_3=2}^{i_3}\boldsymbol{\theta}^{(3)}_{i'_3}\sum_{i'_1=2}^{i_1}\sum_{i'_3=2}^{i_3}\boldsymbol{\theta}^{(1,3)}_{i'_1,i'_3} + \sum_{i'_2=2}^{i_2}\sum_{i'_3=2}^{i_3}\boldsymbol{\theta}^{(2,3)}_{i'_2,i'_3} + \sum_{i'_1=2}^{i_1}\sum_{i'_2=2}^{i_2}\sum_{i'_3=2}^{i_3}\boldsymbol{\theta}^{(1,2,3)}_{i'_1,i'_2,i'_3}\right).$$

When $i_3 = 1$,

$$\mathsf{X}^{(1,2,3)}_{i_1,i_2,1} = \frac{1}{\sqrt[3]{Z}} \exp\left(\frac{1}{2}\sum_{i'_1=2}^{i_1}\boldsymbol{\theta}^{(1)}_{i'_1} + \frac{1}{2}\sum_{i'_1=2}^{i_1}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(1,2)}_{i'_1,i'_2} + \frac{1}{2}\sum_{i'_2=2}^{i_2}\boldsymbol{\theta}^{(2)}_{i'_2}\right) = \mathsf{X}^{(1,2)}_{i_1,i_2},$$

which establishes equation 4.

B.3 Proof of Proposition. 3.1

We restate the proposition for completeness:

Proposition 3.1. (Structurally identical slices in MBA factors). Let $\mathsf{X}^{(\boldsymbol{s})}$ and $\mathsf{X}^{(\boldsymbol{t})}$ be factors obtained by MBA and let $\boldsymbol{u} = \boldsymbol{s} \cap \boldsymbol{t}$ be their shared mode set. Let $\boldsymbol{s} \setminus \boldsymbol{u}$ and $\boldsymbol{t} \setminus \boldsymbol{u}$ be non-shared modes in $\mathsf{X}^{(\boldsymbol{s})}$ and $\mathsf{X}^{(\boldsymbol{t})}$, respectively. Then, the slices obtained by fixing the indices of all non-shared modes to 1 coincide, i.e.,

$$\mathsf{X}^{(\boldsymbol{s})}_{\boldsymbol{i}_u,\,\mathbf{1}_{\boldsymbol{s}\setminus\boldsymbol{u}}} = \mathsf{X}^{(\boldsymbol{t})}_{\boldsymbol{i}_u,\,\mathbf{1}_{\boldsymbol{t}\setminus\boldsymbol{u}}},$$

for all parameter values $\boldsymbol{\theta}$.

Proof. From the definition of MBA in equation 8, each factor $\mathsf{X}^{(\boldsymbol{s})}$ is given by

$$\mathsf{X}^{(\boldsymbol{s})}_{\boldsymbol{i}_{\boldsymbol{s}}} = \frac{1}{\sqrt[M]{Z}} \exp\left[\sum_{\emptyset\neq\boldsymbol{v}\subseteq\boldsymbol{s}} w^{(\boldsymbol{v})}\left(\sum_{\substack{\boldsymbol{i}'_{\boldsymbol{v}}\in\downarrow\boldsymbol{i}_{\boldsymbol{v}}\\ i'_{v_k}\geq 2,\ \forall k}} \boldsymbol{\theta}^{(\boldsymbol{v})}_{\boldsymbol{i}'_{\boldsymbol{v}}}\right)\right].$$

Consider the slice $\mathsf{X}^{(\boldsymbol{s})}_{\boldsymbol{i}_u,\,\mathbf{1}_{\boldsymbol{s}\setminus\boldsymbol{u}}}$, where all indices in $\boldsymbol{s} \setminus \boldsymbol{u}$ are fixed to 1. For any subset $\boldsymbol{v} \subseteq \boldsymbol{s}$ that contains at least one mode in $\boldsymbol{s} \setminus \boldsymbol{u}$, the corresponding index vector $\boldsymbol{i}_{\boldsymbol{v}}$ has an entry equal to 1. Therefore, there are no $\boldsymbol{i}'_{\boldsymbol{v}} \in\downarrow \boldsymbol{i}_{\boldsymbol{v}}$ that meets $i'_{v_k} \geq 2$ for all k. Hence, the inner summation over such $\boldsymbol{v}$ is empty and does not contribute anything. Consequently, only the terms corresponding to the subsets $\boldsymbol{v} \subseteq \boldsymbol{u}$ remain, and we obtain

$$\mathsf{X}^{(\boldsymbol{s})}_{\boldsymbol{i}_u,\,\mathbf{1}_{\boldsymbol{s}\setminus\boldsymbol{u}}} = \frac{1}{\sqrt[M]{Z}} \exp\left[\sum_{\emptyset\neq\boldsymbol{v}\subseteq\boldsymbol{u}} w^{(\boldsymbol{v})}\left(\sum_{\substack{\boldsymbol{i}'_{\boldsymbol{v}}\in\downarrow\boldsymbol{i}_{\boldsymbol{v}}\\ i'_{v_k}\geq 2,\ \forall k}} \boldsymbol{\theta}^{(\boldsymbol{v})}_{\boldsymbol{i}'_{\boldsymbol{v}}}\right)\right].$$

Applying the same argument to $\mathbf{X}^{(t)}$, we obtain

$$X^{(t)}_{i_u,\,\mathbf{1}_{t\setminus u}} = \frac{1}{\sqrt[M]{Z}} \exp\left[\sum_{\emptyset\neq v\subseteq u} w^{(v)} \left(\sum_{\substack{i'_v \in \downarrow i_v \\ i'_{v_k}\geq 2,\ \forall k}} \theta^{(v)}_{i'_v} \right)\right].$$

Since the remaining expressions are identical, we conclude that

$$X^{(s)}_{i_u,\,\mathbf{1}_{s\setminus u}} = X^{(t)}_{i_u,\,\mathbf{1}_{t\setminus u}},$$

which completes the proof. □

B.4 Proof of Proposition. 3.2

We restate the proposition for completeness:

Proposition 3.2. (Functionally distinct interior entries of MBA factors). Let a and b be entries drawn from two distinct MBA factors, with every coordinate of their respective factor indices at least 2. Then a and b are not identical as functions of $\boldsymbol{\theta}$. Equivalently, there exists a parameter value $\boldsymbol{\theta}$ for which $a \neq b$.

Proof. Let $a = \mathbf{X}^{(s)}_{i_s}$ and $b = \mathbf{X}^{(t)}_{i_t}$ be two interior entries of distinct MBA factors. Without loss of generality, assume that $|s| \geq |t|$. Since $s \neq t$, we have $s \not\subseteq t$.

From equation 8, the parameter tensor $\boldsymbol{\theta}^{(s)}$ appears in the exponent of $\mathbf{X}^{(s)}$ and in factors whose scopes are super-sets of s. Therefore, $\boldsymbol{\theta}^{(s)}$ does not appear in the exponent of $\mathbf{X}^{(t)}$. On the other hand, because every coordinate of i_s is at least 2, at least one parameter of $\boldsymbol{\theta}^{(s)}$ is involved in the exponent of a. Hence, a depends on a parameter in $\boldsymbol{\theta}^{(s)}$ that b does not depend on. Therefore, a and b are not identical as functions of $\boldsymbol{\theta}$. □

B.5 Proof of Proposition 4.2

We restate the proposition for completeness:

Proposition 4.2.(Injective parameter assignment in EMBA). Each entry of the EMBA factors can be injectively assigned to an n-body parameter optimized by EMBA. Consequently, the number of parameters optimized by EMBA is at least the total number of entries in the EMBA factors.

Proof. For each factor $\mathbf{Y}^{(s)}$, define an assignment from its entry indexed by i_s to the entry of $\boldsymbol{\theta}^{(s)}$ indexed by $i_s + \mathbf{1}$. Here, $\mathbf{1}$ denotes the all-one index vector of dimension $|s|$.

The EMBA factor $\mathbf{Y}^{(s)}$ is the restriction of the MBA factor $\widetilde{\mathbf{X}}^{(s)}$ to indices whose coordinates are all at least 2. Equivalently, $\mathbf{Y}^{(s)}_{i_s}$ is obtained from $\widetilde{\mathbf{X}}^{(s)}_{i_s+\mathbf{1}}$. Equation 8 shows that the computation of $\widetilde{\mathbf{X}}^{(s)}_{i_s+\mathbf{1}}$ includes the corresponding $|s|$-body parameter entry $\boldsymbol{\theta}^{(s)}_{i_s+\mathbf{1}}$. Thus, each entry of $\mathbf{Y}^{(s)}$ is assigned to an $|s|$-body parameter entry.

For a fixed scope s, this assignment is injective because distinct indices i_s correspond to distinct shifted indices $i_s + \mathbf{1}$. For distinct scopes $s \neq s'$, the parameter tensors $\boldsymbol{\theta}^{(s)}$ and $\boldsymbol{\theta}^{(s')}$ are disjoint. Combining these assignments over all factors yields an injective assignment from EMBA factor entries to optimized n-body parameter entries. Therefore, the number of optimized parameters is at least the total number of entries in the EMBA factors. □

B.6 Proof of Proposition. 4.3

We restate the proposition for completeness:

Proposition 4.3. Let $\mathbf{T}^{(\text{original})}$ be a given tensor, and let $\mathbf{P}^{(\text{EMBA})}$ be the approximation obtained by EMBA as defined in equation 7. Let $\mathbf{T}^{(\text{extend})}$ be the tensor constructed by equation 5), and let

$\mathbf{P}^{(\text{MBA})}$ be the approximation obtained by applying MBA to $\mathbf{T}^{(\text{extend})}$ as in equation 6. Then, the following property holds.

$$D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{extend})} \,\|\, \mathbf{P}^{(\text{MBA})}\Big) \geq D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{original})} \,\|\, \mathbf{P}^{(\text{EMBA})}\Big)$$

Proof. By the definition of the KL divergence,

$$D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{extend})} \,\|\, \mathbf{P}^{(\text{MBA})}\Big) = \sum_{i_1,\ldots,i_N} T^{(\text{extend})}_{i_1,\ldots,i_N} \log \frac{T^{(\text{extend})}_{i_1,\ldots,i_N}}{P^{(\text{MBA})}_{i_1,\ldots,i_N}}.$$

From the definition of $\mathbf{T}^{(\text{extend})}$ (equation 5), we have

$$T^{(\text{extend})}_{i_1,\ldots,i_N} = 0 \quad \text{if any } i_k = 1.$$

Therefore, the summation reduces to the indices satisfying $i_k \geq 2$ for all k, and we obtain

$$\sum_{i_1,\ldots,i_N} T^{(\text{extend})}_{i_1,\ldots,i_N} \log \frac{T^{(\text{extend})}_{i_1,\ldots,i_N}}{P^{(\text{MBA})}_{i_1,\ldots,i_N}} = \sum_{i_1,\ldots,i_N \geq 2} T^{(\text{extend})}_{i_1,\ldots,i_N} \log \frac{T^{(\text{extend})}_{i_1,\ldots,i_N}}{P^{(\text{MBA})}_{i_1,\ldots,i_N}}.$$

By $P^{(\text{MBA})}_{i_1+1,i_2+1,\ldots,i_N+1} = \tilde{P}^{(\text{EMBA})}_{i_1,i_2,\ldots,i_N}$ and $T^{(\text{extend})}_{i_1+1,i_2+1,\ldots,i_N+1} = \tilde{T}^{(\text{original})}_{i_1,i_2,\ldots,i_N}$,

$$\sum_{i_1\geq 2,\ldots,i_N\geq 2} T^{(\text{extend})}_{i_1,\ldots,i_N} \log \frac{T^{(\text{extend})}_{i_1,\ldots,i_N}}{P^{(\text{MBA})}_{i_1,\ldots,i_N}} = \sum_{i'_1,\ldots,i'_N} T^{(\text{original})}_{i'_1,\ldots,i'_N} \log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{\tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N}}.$$

By $\sum_{i'_1,\ldots,i'_N} \tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N} \leq 1$ and $P^{(\text{EMBA})}_{i_1,i_2,\ldots,i_N} = \mathrm{Norm}\Big(\tilde{P}^{(\text{EMBA})}_{i_1,i_2,\ldots,i_N}\Big)$,

$$\tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N} \leq P^{(\text{EMBA})}_{i'_1,\ldots,i'_N}.$$

Since $\tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N} \leq P^{(\text{EMBA})}_{i'_1,\ldots,i'_N}$ and the logarithm is monotonic,

$$\log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{\tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N}} \geq \log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{P^{(\text{EMBA})}_{i'_1,\ldots,i'_N}}.$$

Multiplying both sides by the non-negative weights $T^{(\text{original})}_{i'_1,\ldots,i'_N}$ and summing over all indices, we obtain

$$\sum_{i'_1,\ldots,i'_N} T^{(\text{original})}_{i'_1,\ldots,i'_N} \log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{\tilde{P}^{(\text{EMBA})}_{i'_1,\ldots,i'_N}} \geq \sum_{i'_1,\ldots,i'_N} T^{(\text{original})}_{i'_1,\ldots,i'_N} \log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{P^{(\text{EMBA})}_{i'_1,\ldots,i'_N}}.$$

By the definition of the KL divergence,

$$\sum_{i'_1,\ldots,i'_N} T^{(\text{original})}_{i'_1,\ldots,i'_N} \log \frac{T^{(\text{original})}_{i'_1,\ldots,i'_N}}{P^{(\text{EMBA})}_{i'_1,\ldots,i'_N}} = D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{original})} \,\|\, \mathbf{P}^{(\text{EMBA})}\Big).$$

Therefore, the following inequality holds:

$$D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{extend})} \,\|\, \mathbf{P}^{(\text{MBA})}\Big) \geq D_{\mathrm{KL}}\Big(\mathbf{T}^{(\text{original})} \,\|\, \mathbf{P}^{(\text{EMBA})}\Big).$$

□

C ALGORITHM

Algorithm 1 Extended Many-Body Approximation (EMBA)

Input: Non-negative, normalized tensor $\mathsf{T}^{(\text{original})} \in \mathbb{R}_{\geq 0}^{I_1 \times \cdots \times I_N}$, a set of interactions between modes $\mathcal{S} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_M\}$

Output: EMBA factors $\{\mathbf{Y}^{(\boldsymbol{s}_m)}\}_{m=1}^{M}$ and approximation $\mathsf{P}^{(\text{EMBA})}$

1: Construct the extended tensor $\mathsf{T}^{(\text{extend})}$ as follows:

$$T^{(\text{extend})}_{i_1,\ldots,i_N} = \begin{cases} 0, & \text{if any } i_k = 1, \\ T^{(\text{original})}_{i_1-1,\ldots,i_N-1}, & \text{otherwise.} \end{cases}$$

2: Apply MBA to $\mathsf{T}^{(\text{extend})}$:

$$P^{(\text{MBA})}_{i_1,i_2,\ldots,i_N} = \widetilde{X}^{(\boldsymbol{s}_1)}_{\boldsymbol{i}_{\boldsymbol{s}_1}} \widetilde{X}^{(\boldsymbol{s}_2)}_{\boldsymbol{i}_{\boldsymbol{s}_2}} \cdots \widetilde{X}^{(\boldsymbol{s}_M)}_{\boldsymbol{i}_{\boldsymbol{s}_M}},$$

3: **for** $m = 1, \ldots, M$ **do**

4: Extract the original-domain factor:

$$Y^{(\boldsymbol{s}_m)}_{\boldsymbol{i}_{\boldsymbol{s}_m}} = \widetilde{X}^{(\boldsymbol{s}_m)}_{\boldsymbol{i}_{\boldsymbol{s}_m}+\mathbf{1}},$$

where $\mathbf{1}$ denotes the all-one vector of dimension $|\boldsymbol{s}_m|$.

5: **end for**

6: Construct the unnormalized approximation:

$$P'^{(\text{EMBA})}_{i_1,i_2,\ldots,i_N} = Y^{(\boldsymbol{s}_1)}_{\boldsymbol{i}_{\boldsymbol{s}_1}} Y^{(\boldsymbol{s}_2)}_{\boldsymbol{i}_{\boldsymbol{s}_2}} \cdots Y^{(\boldsymbol{s}_M)}_{\boldsymbol{i}_{\boldsymbol{s}_M}}.$$

7: Normalize:

$$\mathsf{P}^{(\text{EMBA})} = \text{Norm}\left(\mathsf{P}'^{(\text{EMBA})}\right).$$

return $\{\mathbf{Y}^{(\boldsymbol{s}_m)}\}_{m=1}^{M}$ and $\mathsf{P}^{(\text{EMBA})}$

D Experimental Setup

D.1 Hardware Environment

The experiments were conducted in a Docker container with the following environment:

- **Operating system**: Debian GNU/Linux 12 (bookworm)
- **Host machine CPU**: AMD Ryzen Threadripper PRO 3975WX
- **CPU Cores / Threads**: 32 cores / 64 threads
- **CPU Clock speed**: 2.2–4.37 GHz
- **L3 cache**: 128 MB
- **Kernel version**: 5.15.0-127-generic (Ubuntu)

D.2 Software Environment

We used the following tools and libraries that represent the core components of our experimental framework.

- **IgTensors**[2]: for Many-Body Approximation (MBA) for non-negative tensors

D.3 Optimization Settings for MBA and EMBA

For every MBA and EMBA experiment, the initial $\boldsymbol{\theta}$ is obtained in closed form by taking the logarithm of the given tensor and then differencing it along every mode (a discrete alternating-sum transform), rather than by random sampling. Newton's method runs for at most 150 iterations or until the residual falls below 10^{-5}. We enable step halving (halving a Newton step that would not reduce the residual) in every experiment. We distinguish the number of optimization-time parameters from the number of factor entries retained after restriction.

E Detailed Information of Experiments

E.1 Structural Similarity Index Measure

The local SSIM within a window w is defined as

$$\mathrm{SSIM}_w(\mathbf{d}, \hat{\mathbf{d}}) = \frac{(2\mu_w\hat{\mu}_w + C_1)(2\sigma_w^{d\hat{d}} + C_2)}{(\mu_w^2 + \hat{\mu}_w^2 + C_1)(\sigma_w^2 + \hat{\sigma}_w^2 + C_2)},$$

where μ_w and $\hat{\mu}_w$ denote the local means of $\mathbf{d}$ and $\hat{\mathbf{d}}$, σ_w^2 and $\hat{\sigma}_w^2$ denote the corresponding variances, and $\sigma_w^{d\hat{d}}$ denotes the covariance. These statistics are computed using a Gaussian weighting function. Specifically,

$$\mu_w = \sum_{p\in w} G(p)\, d(p), \quad \hat{\mu}_w = \sum_{p\in w} G(p)\, \hat{d}(p),$$
$$\sigma_w^2 = \sum_{p\in w} G(p)\, (d(p) - \mu_w)^2, \quad \hat{\sigma}_w^2 = \sum_{p\in w} G(p)\, (\hat{d}(p) - \hat{\mu}_w)^2,$$
$$\sigma_w^{d\hat{d}} = \sum_{p\in w} G(p)\, (d(p) - \mu_w)(\hat{d}(p) - \hat{\mu}_w),$$

where $G(p)$ denotes the Gaussian weight at position p within the window, normalized such that $\sum_{p\in w} G(p) = 1$. In our experiments, we use a Gaussian kernel with standard deviation $\sigma = 1.5$, following (Wang et al., 2004). The constants C_1 and C_2 are used for numerical stability and are defined as

$$C_1 = (K_1 L)^2, \quad C_2 = (K_2 L)^2,$$

where L is the dynamic range of the pixel values and set as 1, and $K_1 = 0.01$, $K_2 = 0.03$, following (Wang et al., 2004). For color images, the SSIM is computed independently for each RGB channel, and the final value is obtained by averaging over all channels.

[2] https://github.com/gkazunii/IgTensors

Table 4: Scalability with the mode size N ($D = 3$). FIM size: params$^2 \times 8$ bytes. Alloc./step (allocated memory per step) and Time/step: whole run divided by the number of Newton iterations.

	Params		FIM size (MiB)		Alloc./step (MiB)		Time/step (s)		Time (s)		Newton iter.		
N	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	KLD ↓
10	270	330	0.556	0.831	0.943	1.10	0.00220	0.00214	0.007	0.026	3	12	0.1438
20	1,140	1,260	9.92	12.1	14.1	13.9	0.0377	0.0438	0.113	0.526	3	12	0.1681
30	2,610	2,790	52.0	59.4	71.6	66.2	0.257	0.263	0.771	3.157	3	12	0.1779
40	4,680	4,920	167	185	227	204	0.820	0.882	2.459	10.579	3	12	0.1809
50	7,350	7,650	412	446	557	489	2.19	2.38	6.563	28.556	3	12	0.1834

Table 5: Scalability with the number of modes D ($N = 10$). Columns as in Table 4.

	Params		FIM size (MiB)		Alloc./step (MiB)		Time/step (s)		Time (s)		Newton iter.		
D	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	MBA	EMBA	KLD ↓
4	522	640	2.08	3.12	3.37	3.92	0.00795	0.0117	0.024	0.140	3	12	0.1828
5	855	1,050	5.58	8.41	10.5	11.5	0.0509	0.0594	0.153	0.772	3	13	0.1921
6	1,269	1,560	12.3	18.6	42.7	41.7	0.396	0.493	1.187	6.404	3	13	0.1928
7	1,764	2,170	23.7	35.9	287	263	4.38	5.90	13.154	82.584	3	14	0.1931

F Additional Results

F.1 Additional Scalability Results

Setup. We measure the time and memory of MBA and EMBA with a 2-body approximation on random tensors, in two sweeps: the mode size N grows with the number of modes fixed to $D = 3$ ($N = 10, \ldots, 50$), and D grows with $N = 10$ fixed ($D = 4, \ldots, 7$). "Params" is the number of estimated parameters θ; "FIM size" is params$^2 \times 8$ bytes, the size of the Fisher information matrix built by the Newton method; "Alloc./step" (allocated memory per step) and "Time/step" are the memory allocated and the time of the whole run divided by the number of Newton iterations. All runs use the implementation of the main experiments (zero padding, exact Newton method).

Results. Tables 4 and 5 list the results, and Figure 5 plots time and memory against the number of tensor entries. MBA and EMBA reach the same KL divergence in all settings. The cost of one Newton step of EMBA is comparable to that of MBA (time ratio between 0.97 and 1.47). EMBA needs 12–14 Newton iterations, against 3 for MBA, so its total time is larger, but it remains below $10\times$ that of MBA in all settings (at most $6.3\times$). A log-log fit of the time against the number of tensor entries gives similar exponents for MBA and EMBA in both sweeps (D sweep: 0.91 and 0.92; N sweep: 1.44 and 1.45; $R^2 \geq 0.99$).

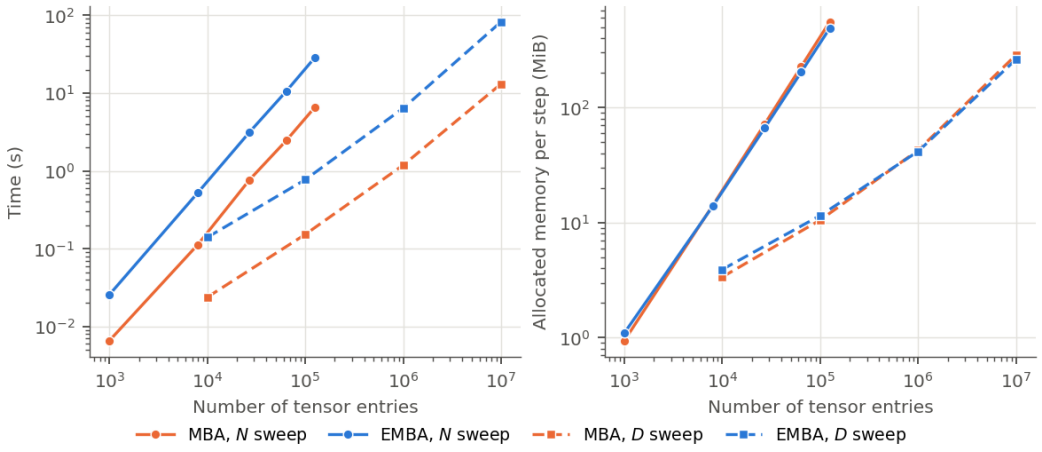


Figure 5: Time (left) and allocated memory per Newton step (right) of MBA and EMBA against the number of tensor entries, for the sweep over N ($D = 3$) and the sweep over D ($N = 10$).

Table 6: Factor deletion on METR-LA (3-body approximation, 4 factors). KL divergence in units of 10^6; mean $\pm$ standard deviation over the deletion patterns; bold: better of MBA and EMBA. Last column: number of patterns in which EMBA has the lower KL divergence.

		KL divergence ↓		Cosine similarity ↑		
k	Patterns	MBA	EMBA	MBA	EMBA	EMBA lower KLD
no deletion	1	0.076	0.076	0.9952	0.9952	–
1	4	12.218 ± 5.036	$\mathbf{3.025 \pm 4.591}$	0.2638 ± 0.1178	$\mathbf{0.8069 \pm 0.3034}$	4/4
2	6	13.477 ± 5.849	$\mathbf{3.256 \pm 3.367}$	0.3029 ± 0.1862	$\mathbf{0.7482 \pm 0.2335}$	6/6
3	4	9.108 ± 8.412	$\mathbf{2.182 \pm 0.675}$	0.4517 ± 0.2562	$\mathbf{0.7934 \pm 0.1382}$	4/4

Table 7: Factor deletion on EEG (2-body approximation, 10 factors). KL divergence in units of 10^3; mean $\pm$ standard deviation over the deletion patterns; bold: better of MBA and EMBA. Last column: number of patterns in which EMBA has the lower KL divergence.

		KL divergence ↓		Cosine similarity ↑		
k	Patterns	MBA	EMBA	MBA	EMBA	EMBA lower KLD
no deletion	1	0.554	0.554	0.9591	0.9591	–
1	10	0.948 ± 0.400	$\mathbf{0.818 \pm 0.267}$	0.9363 ± 0.0248	$\mathbf{0.9410 \pm 0.0212}$	10/10
2	45	1.415 ± 0.573	$\mathbf{1.173 \pm 0.454}$	0.9077 ± 0.0366	$\mathbf{0.9185 \pm 0.0321}$	43/45
3	120	1.951 ± 0.712	$\mathbf{1.620 \pm 0.634}$	0.8747 ± 0.0450	$\mathbf{0.8917 \pm 0.0416}$	110/120
4	210	2.552 ± 0.816	$\mathbf{2.160 \pm 0.797}$	0.8385 ± 0.0499	$\mathbf{0.8606 \pm 0.0492}$	185/210
5	252	3.216 ± 0.882	$\mathbf{2.794 \pm 0.934}$	0.8002 ± 0.0513	$\mathbf{0.8254 \pm 0.0547}$	214/252
6	210	3.938 ± 0.907	$\mathbf{3.522 \pm 1.033}$	0.7606 ± 0.0494	$\mathbf{0.7863 \pm 0.0575}$	167/210
7	120	4.718 ± 0.885	$\mathbf{4.344 \pm 1.078}$	0.7205 ± 0.0447	$\mathbf{0.7439 \pm 0.0569}$	89/120
8	45	5.553 ± 0.805	$\mathbf{5.262 \pm 1.044}$	0.6804 ± 0.0373	$\mathbf{0.6986 \pm 0.0522}$	31/45
9	10	6.441 ± 0.648	$\mathbf{6.274 \pm 0.891}$	0.6407 ± 0.0274	$\mathbf{0.6510 \pm 0.0421}$	5/10

F.2 Additional Factor-Deletion Results on Different Domain Datasets

Data. *METR-LA* is a traffic-speed dataset (Li et al., 2017). The tensor has 28 days (2012-03-01 to 2012-03-28) $\times$ 24 hours $\times$ 12 five-minute slots $\times$ 20 sensors (selected from 207); entries are speeds in mph, and missing values are recorded as zero (8.9% of the entries). The 28-day window is the earliest of 92 candidates with no missing values; the 20 sensors form a connected cluster of the sensor adjacency graph, grown to maximize internal edge weight among sensors with a missing rate below 5%. *EEG* is the PhysioNet (Pollard et al., 2026) EEG Motor Movement/Imagery dataset (Schalk, 2009; Schalk et al., 2004), represented as channel/frequency/time tensors following (Cong et al., 2015). The tensor has 64 channels $\times$ 8 frequency bands (equal width, 1–49 Hz) $\times$ 20 time windows $\times$ 12 runs (R03–R14) $\times$ 20 subjects (S001–S020); each entry is the amplitude, i.e. the square root of the band power (FFT power summed within the band), of a channel in a window. The square root is used because Newton's method diverges on the raw power (dynamic range $\sim$9000$\times$); it reduces the range to $\sim$95$\times$ and converges cleanly.

Setup. MBA and EMBA are fitted with a 3-body approximation on METR-LA (4 factors) and a 2-body approximation on EEG (10 factors), using Newton's method. Deleting a set of factors removes them from the product; the tensor is reconstructed from the remaining factors, normalized to the total mass of the tensor, and compared with the original by the generalized KL divergence and the cosine similarity. We evaluate all $\binom{n}{k}$ subsets of k deleted factors ($k < n$) and report the mean $\pm$ standard deviation over the subsets.

Results. Without deletion, MBA and EMBA reach practically the same KL divergence and cosine similarity (METR-LA: 0.076×10^6 and 0.9952; EEG: 0.554×10^3 and 0.9591). For every k on both datasets, EMBA has the lower mean KL divergence and the higher mean cosine similarity (Tables 6 and 7). On METR-LA, EMBA has the lower KL divergence in 14 of 14 deletion patterns, and the ratio of the mean KL divergences (MBA/EMBA) is between 4.04 and 4.17; the smallest gap among single-factor deletions occurs for the factor (day, five-minute slot, sensor) (14.10 versus 9.88×10^6). On EEG, EMBA has the lower KL divergence in 854 of 1022 patterns and the higher cosine similarity in 764; the ratio decreases from 1.16 at $k = 1$ to 1.03 at $k = 9$.

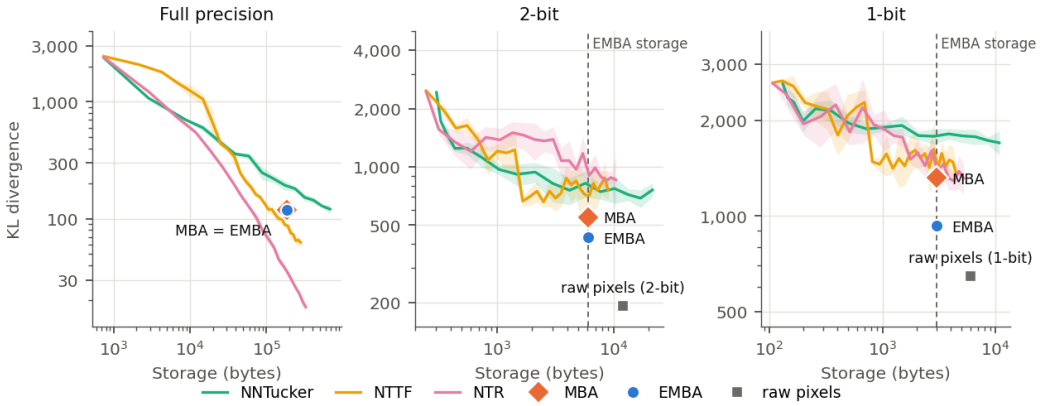


Figure 6: KL divergence versus storage on COIL-100 at full precision (left), 2-bit (center) and 1-bit (right) quantization. Lines: NNTucker, NTTF and NTR over the swept uniform ranks (mean over 5 seeds, shaded band: one standard deviation); MBA and EMBA are single points (identical at full precision). Dashed lines: storage of EMBA. Gray squares: raw pixels quantized directly.

Table 8: Rank sweep on COIL-100 (mean ± standard deviation over 5 seeds; MBA and EMBA are deterministic). For each baseline, the rank with the lowest mean KL divergence in the sweep.

Bits	Method	Rank	Bytes	KL	Cosine
2	NNTucker	16	17,036	688.94 ± 71.19	0.9433 ± 0.0085
	NTTF	14	2,506	659.83 ± 41.15	0.9471 ± 0.0052
	NTR	19	8,617.25	847.77 ± 103.93	0.9258 ± 0.0091
	MBA	–	6,584	551.19	0.9541
	EMBA	–	6,584	**434.33**	**0.9600**
1	NNTucker	17	10,757.75	1698.50 ± 146.69	0.8699 ± 0.0129
	NTTF	28	4,485	1362.69 ± 87.28	0.8985 ± 0.0076
	NTR	19	4,292.62	1288.32 ± 84.85	0.9003 ± 0.0091
	MBA	–	3,236	1316.88	0.8985
	EMBA	–	3,236	**932.70**	**0.9260**

F.3 Additional Results on Section 5.3

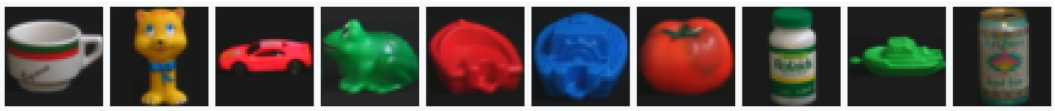
This section supplements the quantization experiment on COIL-100 (Section 5.3) with a rank sweep of the baselines and their performance as compression. The images are the 10 COIL-100 images of the main experiments (40×40 pixels, 3 color channels; 48,000 entries), the thresholds are optimized by coordinate descent (5 sweeps, 25 candidates per threshold), and we report the generalized KL divergence $\sum_i \big(t_i \log(t_i/p_i) - t_i + p_i\big)$ and the cosine similarity of the vectorized tensors. A quantized model with N stored entries in F factors takes $\big(N\,b + 64\,F\,(2^{b+1} - 1)\big)/8$ bytes, since each factor stores its 2^b representative values and $2^b - 1$ thresholds in 64-bit floating point. MBA and EMBA both store $N = 23{,}200$ entries (EMBA's padding is trimmed) and provision a codebook for each of the 14 possible factor shapes of the 3-body approximation ($\binom{4}{1} + \binom{4}{2} + \binom{4}{3}$): 6,584 bytes at 2 bits and 3,236 bytes at 1 bit, although EMBA optimizes more parameters (25,963 versus 20,621).

In addition to the smallest rank whose storage exceeds that of EMBA (Section 5.3), we sweep the uniform rank of each baseline ($r \le 17$ for NNTucker, $r \le 28$ for NTTF, $r \le 21$ for NTR; 5 seeds per rank). Figure 6 plots the KL divergence against the storage, and Table 8 lists the rank with the lowest mean KL divergence for each baseline: 659.83 (NTTF, $r = 14$) at 2 bits and 1,288.32 (NTR, $r = 19$) at 1 bit, compared with 434.33 and 932.70 for EMBA (MBA: 551.19 and 1,316.88). At full precision, NTR and NTTF reach lower KL divergence than MBA and EMBA at comparable storage. Quantizing the 48,000 raw pixels directly with a single codebook gives a KL divergence of 191.45 at 2 bits (12,056 bytes) and 647.96 at 1 bit (6,024 bytes); EMBA at 2 bits (6,584 bytes) is below the 1-bit raw pixels despite using more storage.

F.4 Visualization of Reconstructed Images by Reduced Set of Factors

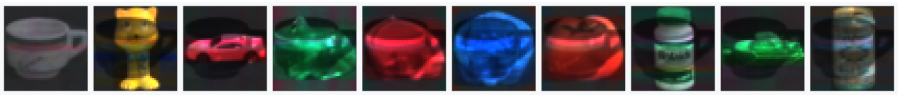
We visualize the images reconstructed in the factor-deletion experiment (Section 5.1), where some of the four factors $\mathbf{X}^{(1,2,3)}$, $\mathbf{X}^{(1,2,4)}$, $\mathbf{X}^{(1,3,4)}$, and $\mathbf{X}^{(2,3,4)}$ of the 3-body approximation (modes: width, height, color channel, and image index) are deleted. Each case reconstructs the images from the remaining factors; MBA is shown above and EMBA below. Cases 1 and 2 delete a single factor. Cases 3 and 4 keep only one factor: $\mathbf{X}^{(1,2,3)}$, which does not depend on the image index, or $\mathbf{X}^{(1,2,4)}$, which links the pixel position and the image index.

Original Images

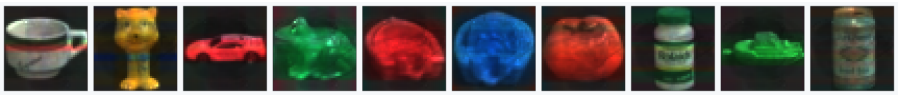


Case 1 (Delete: $\mathbf{X}^{(1,2,3)}$, Remain: $\mathbf{X}^{(1,2,4)}, \mathbf{X}^{(1,3,4)}, \mathbf{X}^{(2,3,4)}$)

MBA

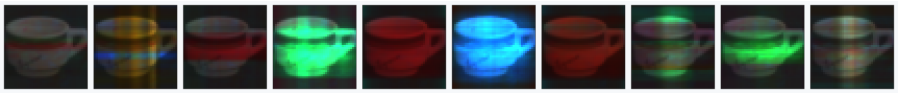


EMBA

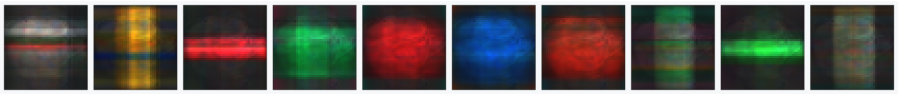


Case 2 (Delete: $\mathbf{X}^{(1,2,4)}$, Remain: $\mathbf{X}^{(1,2,3)}, \mathbf{X}^{(1,3,4)}, \mathbf{X}^{(2,3,4)}$)

MBA

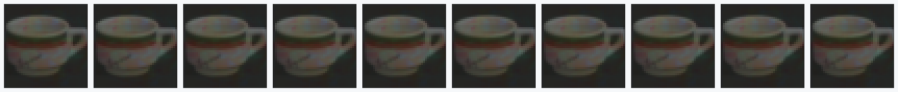


EMBA

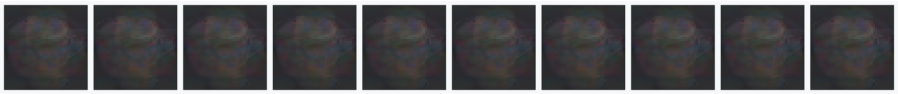


Case 3 (Delete: $\mathbf{X}^{(1,2,4)}, \mathbf{X}^{(1,3,4)}, \mathbf{X}^{(2,3,4)}$, Remain: $\mathbf{X}^{(1,2,3)}$)

MBA

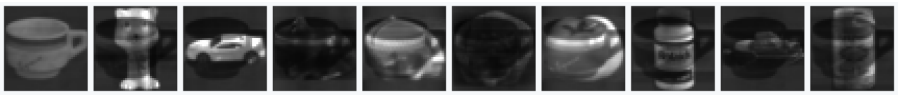


EMBA

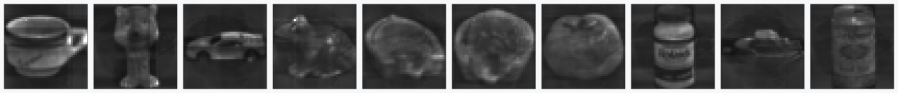


Case 4 (Delete: $\mathbf{X}^{(1,2,3)}, \mathbf{X}^{(1,3,4)}, \mathbf{X}^{(2,3,4)}$, Remain: $\mathbf{X}^{(1,2,4)}$)

MBA



EMBA

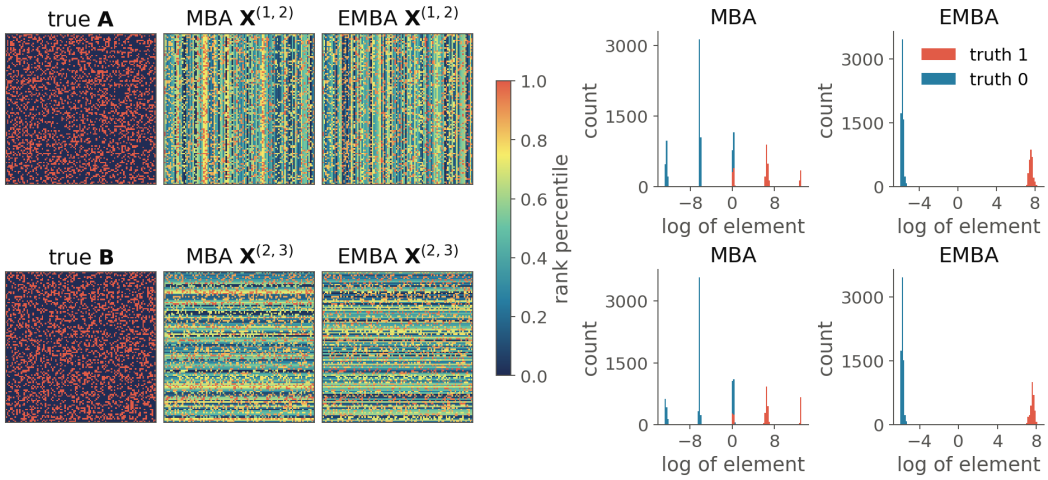


F.5 Visualization of Learned Factors from Synthetic Boolean Tensors

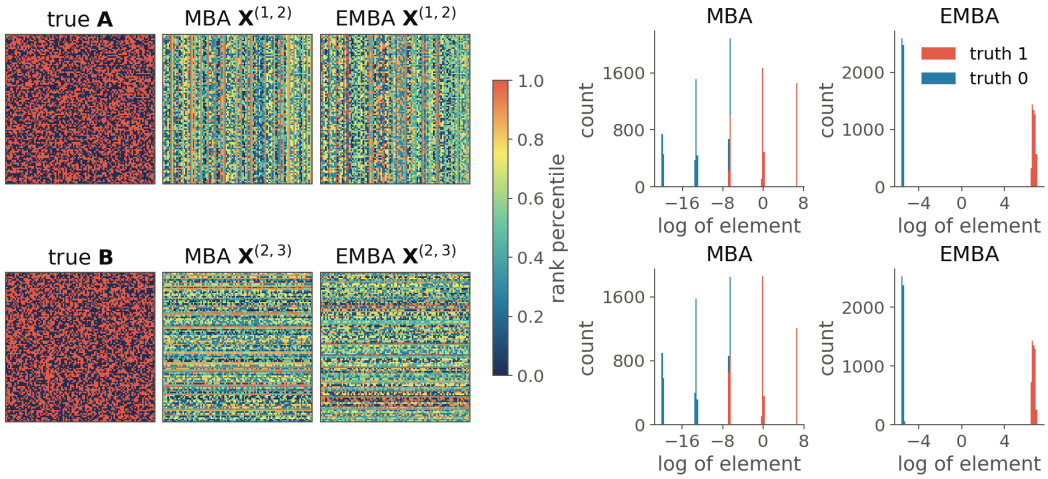
We visualize the factors learned in the synthetic Boolean tensor experiment (Section 5.3), where the tensor $T_{i,j,k} = A_{i,j} \wedge B_{j,k}$ is generated from random Boolean matrices $\mathbf{A}$ and $\mathbf{B}$ with density p_{one}, so that the true factors correspond to $\mathbf{X}^{(1,2)}$ and $\mathbf{X}^{(2,3)}$. Each row shows the true matrix, the factors of MBA and EMBA as rank-percentile heatmaps (red: large, blue: small), and the histograms of the logarithm of the factor elements, stacked by the true value (red: 1, blue: 0). The heatmaps show only the order of the elements; the histograms show their magnitude.

In Cases 1–3, only $\mathbf{X}^{(1,2)}$ and $\mathbf{X}^{(2,3)}$ are used: the elements of the EMBA factors are separated by the true value, whereas those of MBA are not and show row and column patterns.

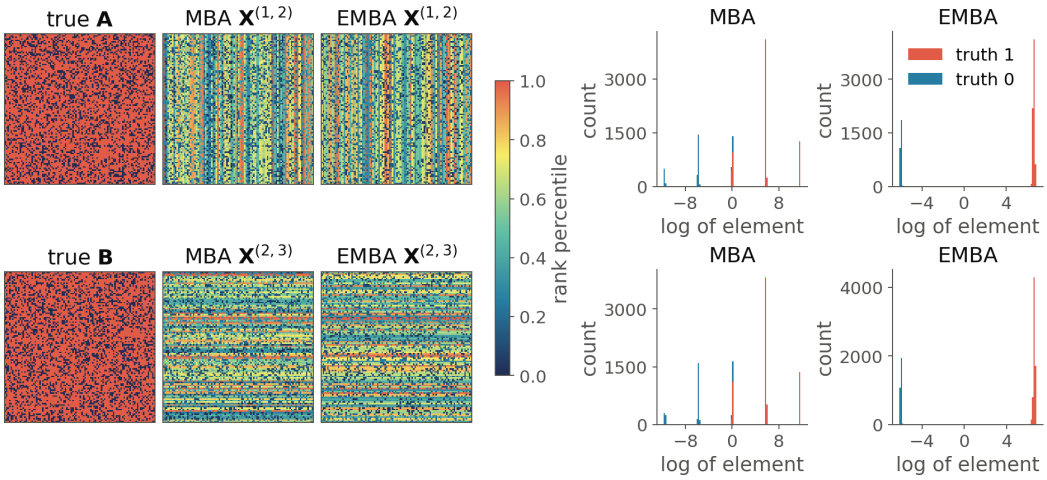
Case 1 (Factors: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}\}$, $p_{\text{one}} = 0.3$.)



Case 2 (Factors: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}\}$, $p_{\text{one}} = 0.5$.)

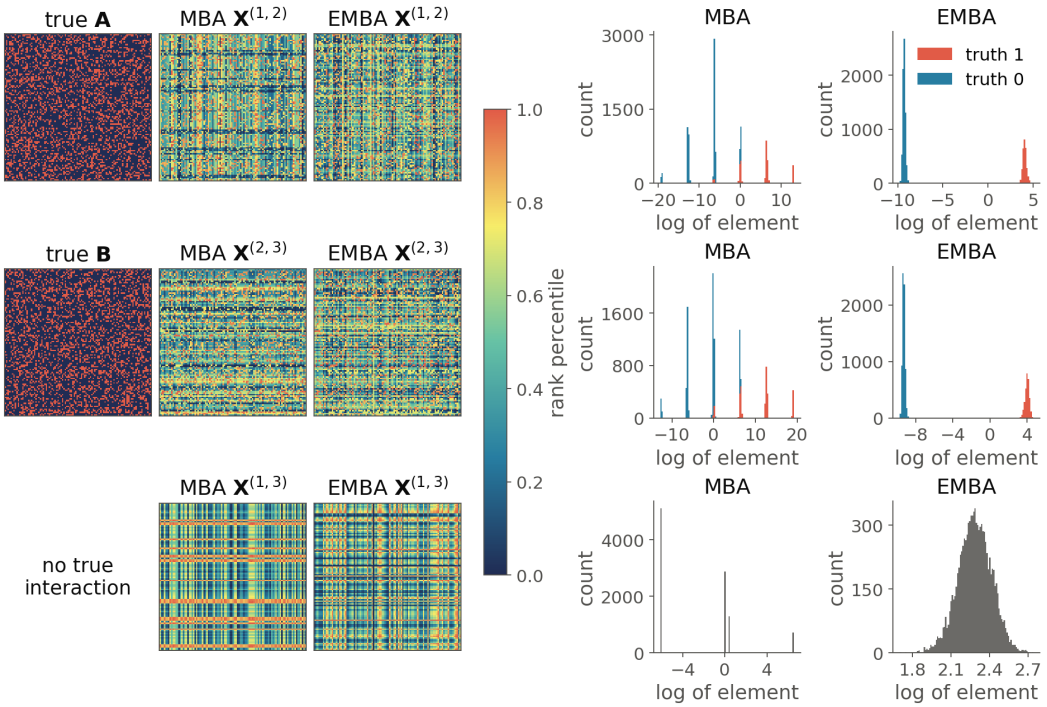


CASE 3 (FACTORS: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}\}$, $p_{\text{one}} = 0.7$.)

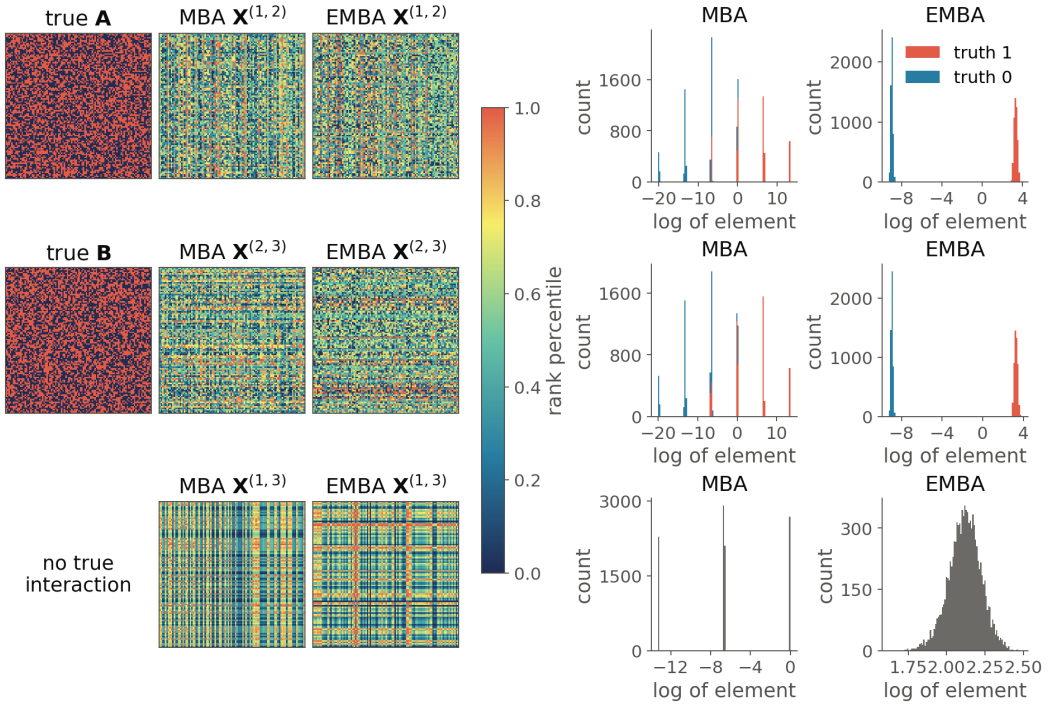


In Cases 4–6, the redundant factor $\mathbf{X}^{(1,3)}$, which has no counterpart in the data, is added: the EMBA factor stays almost constant, whereas the MBA factor shows large row and column patterns.

CASE 4 (FACTORS: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}, \mathbf{X}^{(1,3)}\}$, $p_{\text{one}} = 0.3$.)



CASE 5 (FACTORS: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}, \mathbf{X}^{(1,3)}\}$, $p_{\text{one}} = 0.5$.)



CASE 6 (FACTORS: $\{\mathbf{X}^{(1,2)}, \mathbf{X}^{(2,3)}, \mathbf{X}^{(1,3)}\}$, $p_{\text{one}} = 0.7$.)

