Response Letter

We are grateful to the reviewers for their careful reading and thoughtful comments. Please note that the **Response to Major Comments Summarized by Editor** consists of general comments raised by both reviewers. We address these issues upfront **in pages 1-7 of this letter**.

Response to Major Comments Summarized by Editor

1. *Comparison with other related methods should be included as pointed out by a reviewer.*

Response: We thank the editor and both reviewers for pointing out additional related literature. Some references (Chi and Kolda, 2012; Hong et al., 2020; Raskutti et al., 2019) were already included in the first version, but we have added more comparisons in the revision. We have also added other relevant papers suggested by reviewers. The current comparison consists of 12 tensor regression/factorization methods. Moreover, we have revised our numerical experiments substantially to compare our method with more recent work. Specifically,

(a) we have added discussions about unsupervised model in Section 1:

"...The first line is a class of *unsupervised* tensor decomposition such as classical Tucker and CP decomposition (De Lathauwer et al., 2000; Kolda and Bader, 2009; Wang and Song, 2017) and generalized decomposition for non-Gaussian data (Chi and Kolda, 2012; Tarzanagh and Michailidis, 2019; Hong et al., 2020; Li, 2020). Regardless of the implementation, the unsupervised methods aim to find the best low-rank representation of a data tensor alone. In contrast, our model is a *supervised* tensor learning, which aims to identify the association between a data tensor and multiple features. The low-rank factorization is determined jointly by the tensor data and feature matrices in our model."

(b) we have added discussions about tensor-on-tensor models in Section 1:

"...The second line of work studies the tensor-to-tensor regression. This category is further divided into three scenarios, depending on whether tensor is treated as predictors (Zhou et al., 2013; Raskutti et al., 2019; Han et al., 2020), as responses (Li and Zhang, 2017; Sun and Li, 2017; Zhang et al., 2018; Lock and Li, 2018; Luo et al., 2018), or both (Lock, 2018; Gahrooei et al., 2020). As we show in Section 5, our supervised tensor decomposition falls into this general category, and we provide a *provable* solution in new settings that have broader practical significance. Earlier work in this vein (Lock, 2018; Lock and Li, 2018; Gahrooei et al., 2020; Li, 2020) focuses on algorithm development, but not on the statistical accuracy. Li and Zhang (2017) introduces an envelope-based approach to identify sufficient dimension reduction (Adragni and Cook, 2009), but its theory is restricted to Gaussian data with one-sided feature matrix only. Raskutti et al. (2019) establishes the statistical accuracy for convex relaxed maximum likelihood estimator (MLE) of tensor regression. However, convex relaxation for tensor optimizations suffers from computational intractability and statistical sub-optimality. Recent work has demonstrated the success of non-convex approaches in various tensor problems (Sun and Li, 2017; Zhang et al., 2018;

Raskutti et al., 2019; Han et al., 2020); we go step further by allowing multiple feature matrices with either fixed or random designs. In Sections 4.2, we show that incorporating multiple feature matrices substantially improves the statistical accuracy. We provide a detailed comparison in Section 5; see Table 1."

(c) we have added a new Section 5. **Connection to other tensor regression methods**:

"...We compare our supervised tensor decomposition (**STD**) with recent 12 tensor methods in the literature. Table 1 summarizes these methods with their properties from four aspects: i) model specification, ii) number of feature matrices allowed, (iii) capability of addressing non-Gaussian response, and (iv) capability of addressing non-independent noise. The four closet methods to our are **SupCP** (Lock and Li, 2018), **Envelope** (Li and Zhang, 2017), **mRRR** (Luo et al., 2018) and **GLSNet** (Zhang et al., 2018); these methods all relate a data tensor to feature matrices with low-rank structure on the coefficients. As seen from the table, our method is the only one that allows multiple feature matrices among the five. **Envelope** and **SupCP** are developed for Gaussian data, and the Gaussianity facilities flexible extension to non-independent noise. In particular, **Envelope** allows correlation in Kronecker structured form, whereas **SupCP** allows correlation implicitly through decomposing the latent factors into fixed effects (related to features) and random effects (unrelated to features). On the other hand, the other three methods (**mRRR**, **GLSNet** and **STD**) are developed for exponential family distribution with possibly non-additive noises. The generality makes the full modeling of correlation computationally challenging. We will compare the numerical performance of these methods in Section 6."

(d) We have added new numerical experiments in Section 6 comparing the more current and related methods with ours. The section now reads:

"...These four methods are the closest methods to ours, in that they all relate a data tensor to feature matrices with low-rank structure on the coefficients. We consider Gaussian and Bernoulli tensors in experiments. For methods not applicable for Bernoulli data (**SupCP** and **Envelope**), we provide the algorithm $\{-1, 1\}$-valued tensors as inputs. Because **mRRR** allows matrix response only, we provide the algorithm the unfolded matrix of response tensor as inputs. We measure the accuracy using the response error defined as $1 - \mathrm{Cor}(\hat{\mathcal{Y}}, f(\Theta_{\text{true}}))$, where $\hat{\mathcal{Y}}$ is the fitted tensor from each method, and $f(\Theta_{\text{true}})$ is the true conditional mean of the tensor. Note that the response error is a scale-insensitive metric; a smaller error implies a better fit of the model.
The comparison is assessed from three aspects: (i) benefit of incorporating features from multiple modes; (ii) prediction error with respect to sample size; (iii) robustness of model misspecification. We use similar simulation setups as in our first experiment in last section. We consider rank $\boldsymbol{r} = (3, 3, 3)$ (low) vs. $(4, 5, 6)$ (high), signal $\alpha = 3$ (low) vs. 6 (high), dimension d ranging from 20 to 100 for modes with features, and $d = 20$ for modes without features. The method **Envelope** and **mRRR** require the tensor rank as inputs, respectively. For fairness, we provide both algorithms the true rank. The methods **SupCP** and **GLSNet** use different notions of model rank, and **GLSNet** takes sparsity as an input. We use a grid search to set the hyperparameters in **SupCP** and **GLSNet** that give the best performance.

Method	Model	No. of features	non-Gaussianity	Non-independence
STD (Ours)	$\mathbb{E}\mathcal{Y} = f(\mathcal{B} \times \{\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3\}),\ \mathcal{B} = \mathcal{C} \times \{\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{M}_3\}$	3	$\surd$	$\times$
GCP, CP-ARP, CORALS	$\mathbb{E}\mathcal{Y} = f(\llbracket \boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3 \rrbracket)$	0	$\surd$	$\times$
DCOT	$\mathbb{E}\mathcal{Y} = f((\mathcal{C}_1 + \mathcal{C}_2) \times \{\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{M}_3\})$	0	$\surd$	$\times$
LRT, CRT	$y_n = \langle \mathcal{B},\ \mathcal{X}_n \rangle + \epsilon_n$, various structure on $\mathcal{B}$	0	$\times$	$\times$
STAR	$y_n = \sum_m \langle \mathcal{B}_m, \mathcal{F}_m(\mathcal{X}_{ijk}) \rangle + \epsilon_n$, sparse-CP $\mathcal{B}_m$	0	$\times$	$\times$
SupCP	$\mathcal{Y} = \llbracket \boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3 \rrbracket + \mathcal{E},\ \boldsymbol{A}_1 = \boldsymbol{X}\boldsymbol{B} + \mathcal{E}',\ \mathcal{E} \perp \mathcal{E}'$	1	$\times$	$\surd$
mRRR	$\mathbb{E}\boldsymbol{Y} = f(\boldsymbol{X}\boldsymbol{B})$, low-rank $\boldsymbol{B}$	1	$\surd$	$\times$
Envelope	$\mathcal{Y} = \mathcal{B} \times_3 \boldsymbol{X} + \mathcal{E},\ \mathcal{B} = \mathcal{C} \times \{\boldsymbol{M}_1, \boldsymbol{M}_2, \boldsymbol{I}\},\ \mathcal{E} \sim \mathcal{TN}(\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \boldsymbol{I})$	1	$\times$	$\surd$
GLSNet	$\mathbb{E}\mathcal{Y} = f(\boldsymbol{1} \otimes \boldsymbol{\Theta} + \mathcal{B} \times_3 \boldsymbol{X})$, low-rank $\boldsymbol{\Theta}$, sparse $\mathcal{B}$	1	$\surd$	$\times$
STORE	$\mathcal{Y} = \mathcal{B} \times_3 \boldsymbol{X} + \mathcal{E}$, sparse-CP $\mathcal{B}$	1	$\times$	$\times$

Table 1: Comparison of tensor regression/factorization methods. We focus on order-3 tensors for illustration. Calligraphic letters denote tensors, bold capital letters denote matrices, and little letters denote scalars. The dimension of tensors and matrices can be identified from the contexts.

- Data: tensor response $\mathcal{Y}$, feature matrices $\boldsymbol{X}, \boldsymbol{X}_k$, predictor tensor $\mathcal{X}_n$, scalar response y_n, sample index n, tensor mode $k = 1, 2, 3$.
- Parameter: Tuckor factors $\boldsymbol{M}_k$, CP factors $\boldsymbol{A}_k$, CP decomposition $\llbracket \boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}_3 \rrbracket$, coefficient tensor and matrix $\mathcal{B}, \mathcal{B}_m, \boldsymbol{\Theta}, \boldsymbol{B}$.
- Function: a known link function $f(\cdot)$, a known basis function $\mathcal{F}_m(\cdot)$.
- Noise: Gaussian tensor with i.i.d. entries $\mathcal{E}, \mathcal{E}'$, Gaussian tensor with Kronecker covariance $\mathcal{E} \sim \mathcal{TN}(\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \boldsymbol{I})$, meaning $\mathrm{Cov}(\mathrm{vec}(\mathcal{E})) = \boldsymbol{\Sigma}_1 \otimes \boldsymbol{\Sigma}_2 \otimes \boldsymbol{I}$.

- GCP: Generalized canonical polyadic tensor decomposition (Hong et al., 2020);
- CP-APR: CP alternating Poisson regression (Chi and Kolda, 2012);
- CORALS: Generalized co-clustering method (Li, 2020);
- DCOT: Double core tensor decomposition (Tarzanagh and Michailidis, 2019);
- SupCP: Supervised PARAFAC/CANDECOMP factorization (Lock and Li, 2018);
- mRRR: Mixed-response reduced-rank regression (Luo et al., 2018);
- Envelope: Parsimonious tensor response regression (Li and Zhang, 2017);
- GLSNet: Generalized connectivity matrix response regression (Zhang et al., 2018);
- STORE: Sparse tensor response regression (Sun and Li, 2017);
- LTR: Low-rank tensor regression (Han et al., 2020);
- CRT: Convex regularized multi-response tensor regression (Raskutti et al., 2019);
- STAR: Sparse tensor additive regression (Hao et al., 2021).

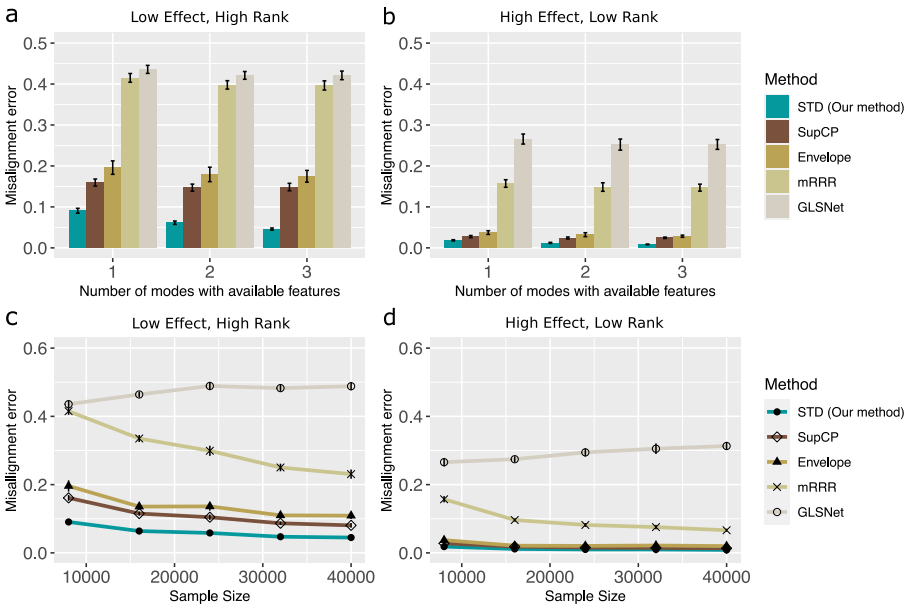

Figure 1: Comparison between tensor methods with Gaussian data. Panels (a) and (b) plot estimation error versus the number of modes with available features. Panels (c) and (d) plot ME versus the effective sample size d^2. We consider rank $\boldsymbol{r} = (3, 3, 3)$ (low), $\boldsymbol{r} = (4, 5, 6)$ (high), and signal $\alpha = 3$ (low), $\alpha = 6$ (high).

Figure 1a-b shows the impact of features to estimation error. We see that our **STD** outperforms others, especially in the low-signal high-rank setting. As the number of informative modes increases, the **STD** exhibits a reduction in error whereas others remain unchanged. The accuracy gain in Figure 1 demonstrates the benefit of incorporating informative features from multiple modes. In addition, we find that the relative performance among the competing methods reveals the benefits of low-rankness. The second best method is **SupCP** which imposes low-rankness on three modes; the next one is **Envelope** which imposes low-rankness on two modes; the less favorable one is **mRRR** which imposes low-rank structure on one mode only; the worst one is **GLSNet** which imposes sparsity but no low-rankness on the feature effects.

Figure 1c-d compares the prediction error with respect to effective sample size d^2. For fair comparison, we consider the setting with feature matrix on one mode only. We find that our **STD** method has similar performance as **Envelope** and **SupCP** in the high-signal low-rank regime, whereas the improvement becomes more pronounced in the low-signal high-rank regime. The latter setting is notably harder, and our **STD** method shows advantage in addressing this challenge. Among other methods, **Envelope**, **SupCP**, and **mRRR** show decreasing errors as d increases, implying the benefits of low-rankness methods. In contrast, **GLSNet** suffers from non-decreasing error and indicates the poor fit of sparsity methods in addressing low-rank data.

We also evaluate the performance comparison with Bernoulli tensors. Figure 2 indicates the necessarity of generalized model in addressing non-Gaussian data. Indeed, methods that assume Gaussiannity (**Envelope** and **SucCP**) perform less favorably in Bernoulli setting (Figure 2c) compared to Gaussian setting (Figure 1c). Our method shows improved accuracy as the number of informative features increases (Figure 2a-b). In the absence of

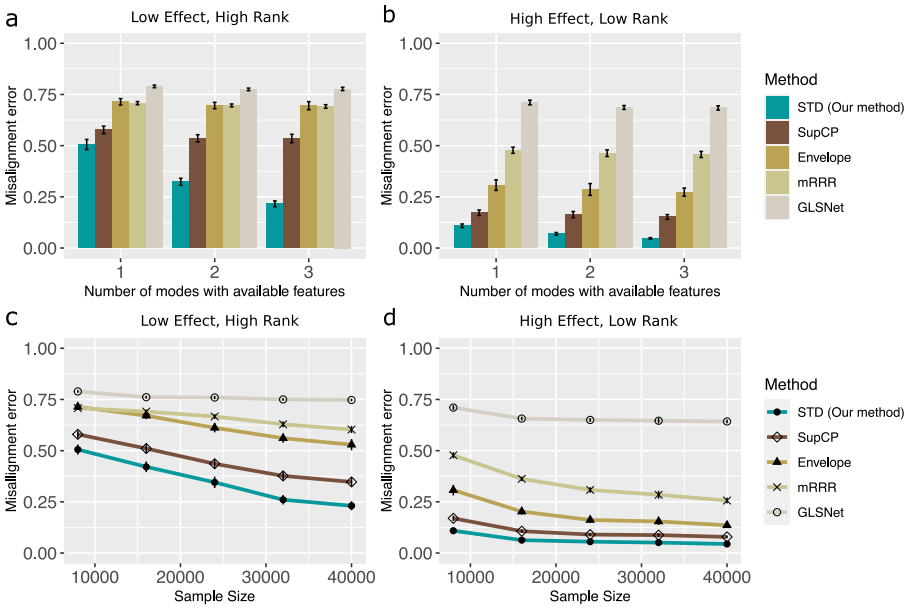

Figure 2: Comparison between tensor methods with Binary data. The panel legends are the same as in Figure 1.

multiple features, our method still performs favorably compared to others (Figure 2c-d), for the same reasons we have argued in Gaussian data. ”

(e) we have added a new experiment in Section 6 to evaluate the robustness of our method with model misspecification:

“Lastly, we assess the performance of our method **STD** under model misspecification. We consider two aspects: (i) non-independent noise, and (ii) sparse feature effects. Note that our method **STD** imposes neither of these two assumptions, so the experiment allows us to assess the robustness. We select competing methods from Table 1 that specifically addresses these two aspects. We use **Envelope** and **SupCP** as benchmark for noise correlation experiment, and **GLSNet** for sparsity experiment.

Figure 3a-b assesses the impact of noise correlation to the estimation accuracy. The data is simulated from **Envelope** model with envelope dimensions $r = (3, 3)$ (low) and $(4, 5)$ (high). The noise is generated from a zero-mean Gaussian tensor with Kronecker structured covariance; see Supplementary Notes for details. As expected, **Envelope** shows the best performance in the high correlation setting. Remarkably, we find that our method **STD** has comparable and sometimes better performance when noise correlation is moderate-to-low. In contrast, **SupCP** appears less suitable in this setting. Although **SupCP** allows noise correlation implicitly through latent random factors, the induced correlation may not belong to the Kronecker covariance structure in the simulation.

Figure 3c-d assesses the impact of sparsity to estimation performance. We generate data from **GLSNet** model, except that we modify the coefficient tensor to be joint sparse and low-rank (the original **GLSNet** model assumes full-rankness on the coefficient tensor). The sparsity level (x-axis in Figure 3c-d) quantifies the proportion of zero entries in the coefficient tensor. Since neither our method **STD** nor **GLSNet** follow the simulated model, this setting allows a fair comparison. We find that our method outperforms **GLSNet** in the

low-rank setting, whereas **GLSNet** shows a better performance in the high-rank setting. This observation suggests the robustness of our method to sparsity when the tensor of interest is simultaneously low-rank and sparse. When sparsity is the only salient structure, then methods specifically addressing sparsity would provide a better fit."

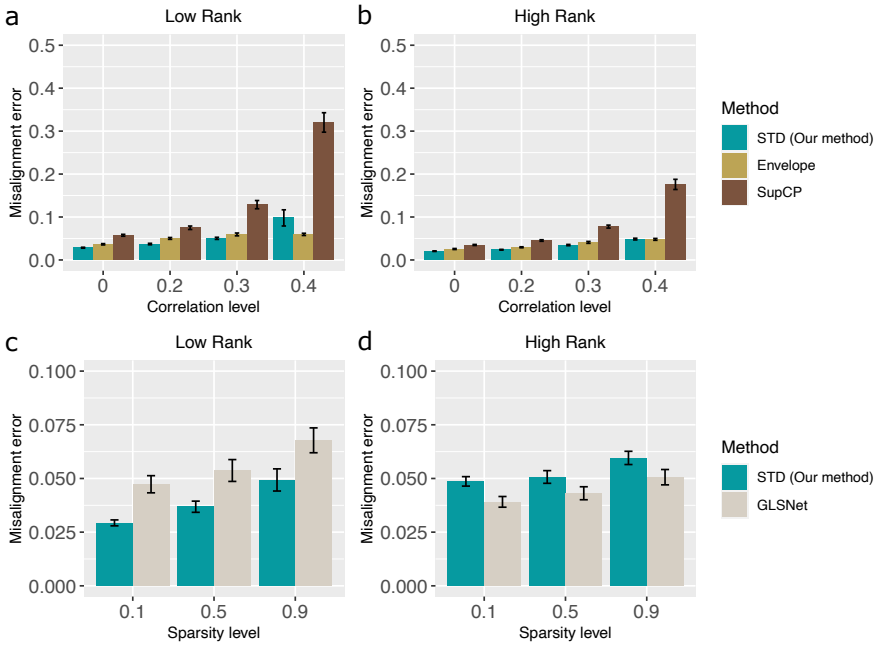

Figure 3: Comparison between tensor methods under model misspecification. Panels (a)-(b) assess the error correlation, and panels (c)-(d) assess the sparsity.

For self-consistency, we quote the detailed simulation setups for the Figure 3 in the Supplementary Notes, Section B:

" **B.1 Detailed simulation setup for Figure 3a-b**

We generate data from **Envelope** model (Li and Zhang, 2017) with slight modification. We simulate response tensor $\mathcal{Y} \in \mathbb{R}^{d\times d\times d}$ from the following model with envelope dimension (u_1, u_2),

$$\begin{aligned} &\mathcal{Y}|\boldsymbol{X} = \mathcal{B}\times_3 \boldsymbol{X} + \mathcal{E} = \mathcal{C}\times\{\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \boldsymbol{X}\} + \mathcal{E}, \\ &\text{with} \quad \mathcal{E} \sim \mathcal{TN}(\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \boldsymbol{I}), \quad \boldsymbol{\Sigma}_k = \boldsymbol{\Gamma}_k\boldsymbol{\Omega}_k\boldsymbol{\Gamma}_k^T + \boldsymbol{\Gamma}_{0k}\boldsymbol{\Omega}_{0k}\boldsymbol{\Gamma}_{0k}^T + \boldsymbol{I}, \quad k = 1, 2, \end{aligned} \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{d\times p}$ is the feature matrix, $\mathcal{B} = \mathcal{C}\times\{\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \boldsymbol{I}\} \in \mathbb{R}^{d\times d\times p}$ is the coefficient tensor, $\mathcal{C} \in \mathbb{R}^{\mu_1\times\mu_2\times p}$ is a full-rank core tensor, $\mathcal{TN}(\cdot,\ \cdot,\ \cdot)$ represents zero-mean tensor normal distribution with Kronecker structured covariance, $\boldsymbol{\Gamma}_k \in \mathbb{O}(d, u_k)$ consists of orthogonal columns, $\boldsymbol{\Gamma}_{0k} \in \mathbb{O}(d, d-u_k)$ is the orthogonal complement of $\boldsymbol{\Gamma}_k$, and $\boldsymbol{\Omega}_k = \boldsymbol{A}_k\boldsymbol{A}_k^T$, $\boldsymbol{\Omega}_{0k} = \boldsymbol{A}_{k0}\boldsymbol{A}_{k0}^T$ with $\boldsymbol{A}_k \in \mathbb{R}^{u_k\times u_k}$, $\boldsymbol{A}_{k0} \in \mathbb{R}^{(d-u_k)\times(d-u_k)}$.
The entries of $\boldsymbol{X}$ are i.i.d. drawn from $\mathcal{N}(0, 1)$, the entries of $\boldsymbol{A}_k$, $\boldsymbol{A}_{k0}$ are i.i.d. drawn from Uniform$[-\gamma, \gamma]$, and the entries of core tensor $\mathcal{C}$ are i.i.d. drawn from Uniform$[-3, 3]$. We call γ the *correlation level*. Note that the only distinction between model (1) and standard **Envelope** model is the additional identity matrix $\boldsymbol{I}$ in the expression of $\boldsymbol{\Sigma}_k$. When $\gamma = 0$,

the model (1) reduces to our **STD** model with rank $\boldsymbol{r} = (u_1, u_2, p)$. We set $d = 20, p = 5$ in our simulation.

B.2 Detailed simulation setup for Figure 3c-d

We generate the data from **GLSNet** model (Zhang et al., 2018) with slight modification. We simulate the binary response tensor $\mathcal{Y} \in \{0,1\}^{d\times d\times d}$ from the following model

$$\mathbb{E}[\mathcal{Y}|\boldsymbol{X}] = f(\mathbf{1} \otimes \boldsymbol{\Theta} + \mathcal{B} \times_3 \boldsymbol{X}),$$

where $f(\cdot)$ is the logistic link, $\boldsymbol{X} \in \mathbb{O}(d,p)$ is the feature matrix with orthonormal columns, $\boldsymbol{\Theta} = \boldsymbol{A}\boldsymbol{A}^T \in \mathbb{R}^{d\times d}$ is a rank-R intercept matrix, where the entries of $\boldsymbol{A} \in \mathbb{R}^{d\times R}$ are simulated from i.i.d. standard normal. Unlike original **GLSNet** model, we generate joint sparse and low-rank structure to the coefficient tensor $\mathcal{B}$ as follows.
To generate $\mathcal{B}$, we firstly generate a low-rank tensor $\mathcal{B}_0$ as

$$\mathcal{B}_0 = \mathcal{C} \times \boldsymbol{M}_1 \times \boldsymbol{M}_2 \times \boldsymbol{M}_3,$$

where $\mathcal{C} \in \mathbb{R}^{R\times R\times R}$ is a full-rank core tensor, $\boldsymbol{M}_1, \boldsymbol{M}_2 \in \mathbb{R}^{d\times R}$ and $\boldsymbol{M}_3 \in \mathbb{R}^{p\times R}$ are the factor matrices with orthonormal columns. We simulate i.i.d. uniform entries in $\mathcal{C}$ and rescale the tensor $\mathcal{B}_0$ such that $\|\mathcal{B}_0\|_{\max} = 2$. Last, we obtain a sparse $\mathcal{B}$ by randomly setting sd^2p entries in $\mathcal{B}_0$ to zero. We call s the *sparsity level* which quantifies the proportion of zero's in $\mathcal{B}$. Hence, the generated tensor $\mathcal{B}$ is of sparsity level s and of low-rank (R, R, R). We set $d = 20$, $p = 5$ and consider the combination of rank $R = 2$ (low), 4 (high) and sparsity level $s = \{0, 0.3, 0.5\}$ in the simulation. "

2. *Additional discussion on side information should be included that involves i) how do we know in practice if side information is appropriate; ii) dimensionality of the side information.*

Response: See our response to **# 5, Reviewer 2**.

3. *The authors should discuss initialization and how to obtain the initial error condition. Can something be said about the proposed random initialization in Algorithm 1?*

Response: We have added theoretical justifications to two initializations. See our response **# Initialization, Reviewer 1.**

4. *There is a disconnect between the proposed algorithm and established theory. Can the authors establish statistical properties for the iterative algorithm? If not, what are the technical challenges that are not present in the related literature?*

Response: We have added new theory to establish the accuracy for both (i) global MLE and (ii) local optimizer. See our response to **# Theoretical Studies, Reviewer 1.**

5. *Some discussion on the computationally tractable selection of ranks should be included. See comment from Reviewer 2.*

Response: We have added more detailed discussions and reported the running time for rank selection. See our response to **# 8, Reviewer 2**.

Point-by-point Response to Reviewer 1

We greatly appreciate your valuable comments and suggestions. We have carefully addressed all the questions. Note that general comments raised by both reviewers were addressed upfront in **pages 1-7 of this response letter, Response to Major Comments Summarized by Editor**. We will make explicit reference to earlier section when needed.

This paper introduces a novel supervised tensor decomposition model when there are side information on multiple modes. This model can be viewed as an extension to tensor response regression. As compared to existing tensor response regression models (Rabusseau and Kadri, 2016; Li and Zhang, 2017; Sun and Li, 2017), the proposed method has three advantages. First, it allows covariate information on multiple modes instead of one mode. Second, it allows high-dimensionality on both the tensor size d_k and covariate size p_k. Third, it allows a exible data type of the tensor, including continuous, count, and binary observations. Data satisfying these properties are not rare, and the paper has provided three data examples. Therefore, I believe this is a useful and well-motivated work. The construction of the estimator leads to a non-convex optimization. Both algorithm and theoretical study are constructed. This work can be regarded as a non-trivial extension of these tensor response regression models as well as Wang and Li (2020) on probabilistic tensor decomposition which decomposes a tensor with continuous, count, or binary observations. Overall, I think this is an interesting work that has potential usefulness and will contribute to the literature of tensor regression. I enjoy reading the paper and feel it would be suitable for JCGS if the following comments are addressed. I recommend a major revision.

Initialization:

1. *How to initialize to achieve the initial error condition, as stated in Proposition 4.1 (Global convergence)?*

Response: We recommend two schemes for initialization: random initialization (cold start) and spectral initialization (warm start). In the high signal-to-noise regime, we have added theoretical guarantees for the statistical accuracy of warm start within polynomial running time. In the low signal-to-noise regime, the statistical-computational gap arises in which case random initialization often provides better empirical results. **We have added a new section for provable initialization and its implementation.** See our detailed response to **# Theoretical Studies**.

We have added the following paragraph in Section 4.1:

> "...We provide two initialization schemes, one with QR-adjusted spectral initialization (warm initialization), and the other with random initialization (cold initialization). The warm initialization is an extension of unsupervised spectral initialization (Zhang and Xia, 2018) to supervised setting with multiple feature matrices. ... The initialization scheme is described in Algorithm 1.
>
> The warm initialization enjoys provable accuracy guarantees at a cost of extra technical assumptions (see Section 4.2). The cold initialization, on the other hand, shows robust in practice but its theoretical guarantee remains an open challenge (Luo and Zhang, 2021). We incorporate both options in our software package to provide flexibility to practitioners..."

Algorithm 1 QR-adjusted spectral initialization

Input: Response tensor $\mathcal{Y} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, feature matrices $\boldsymbol{X}_k \in \mathbb{R}^{d_k \times p_k}$, Tucker rank $\boldsymbol{r}$.

1: Normalize date tensor $\bar{\mathcal{Y}} \leftarrow \mathcal{Y}$ for Gaussian model, $\bar{\mathcal{Y}} \leftarrow 2\mathcal{Y} - 1$ for Bernoulli model, and $\bar{\mathcal{Y}} \leftarrow \log(\mathcal{Y} + 0.5)$ for Poisson model.

2: Normalize feature matrices via QR factorization $\boldsymbol{X}_k = \boldsymbol{Q}_k \boldsymbol{R}_k$ for all $k \in [K]$.

3: Obtain $\bar{\mathcal{B}} \leftarrow \bar{\mathcal{Y}} \times \{\boldsymbol{Q}_1^T, \ldots, \boldsymbol{Q}_K^T\}$ by projecting $\bar{\mathcal{Y}}$ to the multilinear feature space.

4: Obtain $\hat{\mathcal{B}}^{(0)} \leftarrow \mathrm{HOSVD}(\bar{\mathcal{B}}, \boldsymbol{r})$.

5: Normalize representation $\{\hat{\mathcal{C}}^{(0)}, \hat{\boldsymbol{M}}_1^{(0)}, \ldots, \hat{\boldsymbol{M}}_K^{(0)}\}$ such that $\hat{\mathcal{C}}^{(0)} \times \{\hat{\boldsymbol{M}}_1^{(0)}, \ldots, \hat{\boldsymbol{M}}_K^{(1)}\} = \hat{\mathcal{B}}^{(0)} \times \{\boldsymbol{R}_1^{-1}, \ldots, \boldsymbol{R}_K^{-1}\}$ and $\hat{\boldsymbol{M}}_k^{(0)} \in \mathbb{O}(p, r)$ for all $k \in [K]$.

Output: Core tensor $\hat{\mathcal{C}}^{(0)}$ and factors $\hat{\boldsymbol{M}}_k^{(0)}$ for all $k \in [K]$.

2. *As the proposed estimator is obtained from a nonconvex optimization, the possibility of having multiple local optimizers is a problem. In Algorithm 1, a random initialization is used. Can the authors construct an empirical study of how serious the local optimal affects the statistical performance of the randomly initialized estimator? Does the random initialization always lead to the global optimum?*

Response: The empirical performance of random initialization was investigated in Figure 2 in the original paper and the remarks thereof. We have shown in Theorem 4.1 that the same statistical rate holds, not only for the *global optimizer*, but also for every *local optimizer* with sufficiently large objective values. In this sense, local optimality is not necessarily a severe concern in our context. We have added the following sentences for clarification in Sections 4.2.1 and Section 4.2.2:

> "...Inspection of our proof (Supplementary Notes) shows that the desired convergence rate holds not only for the MLE, but also for all local optimizers in the set $\mathcal{L}_{\mathcal{Y}}(\mathcal{C}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K) \geq \mathcal{L}_{\mathcal{Y}}(\mathcal{C}_{\text{true}}, \boldsymbol{M}_{1,\text{true}}, \ldots, \boldsymbol{M}_{K,\text{true}})$. The observation indicates the global optimality is not necessarily a serious concern in our context, as long as the convergent objective is large enough. In next section, we will provide the statistical accuracy for *local* optimizer with provable convergence guarantee, at a cost of extra signal requirement..."

> "...We supply the theory by providing an alternative scheme – random initialization – and investigate its empirical performance. Figure 2 shows the trajectories of objective function ... In the experiment we conduct, we find random initialization appears good enough for Algorithm 1 to find a convergent point with desired statistical guarantees. In practice, we recommend to run both warm and cold initializations, and choose the one with better convergent objective values..."

On a related note, characterizing the optimization landscape and impact of initialization are problems of interest. Inspired by the reviewer's comment, **we have added new theory for provable initialization and per-iteration accuracy.** See our response to **# Theoretical Studies**.

Theoretical Studies:

1. *The statistical properties established in Theorem 4.1 is for the M-estimator in (8). However, due to non-convexity of this optimization, the maximizer of (8) may not be achievable by practical*

algorithms. Therefore, it would be more convincing to prove statistical properties for the iterative estimator from the alternating optimization in Algorithm 1. Such theorem is pretty standard in recent tensor regression models with low-rank structure (Sun and Li, 2017; Raskutti et al., 2019; Hao et al., 2021; Han et al., 2020). If this proof is not possible, it would be helpful if the authors could discuss the technical challenges.

Response: Thank you for the suggestion. Inspired by your comments, we have now proved the statistical properties for the iterative estimator from the alternating optimization in Algorithm 1. We summarize the new results below; the detailed revisions are in Section 4.2.

> "...This section presents the accuracy guarantees for both global and local optimizers of (6). We first provide the statistical accuracy for the global MLE (6). Then, we provide the convergence rate for the local optimizer from Algorithm 1 with warm initialization. The rate reveals an interesting interplay between statistical and computational efficiency. We show that a polynomial number of iterations suffices to reach the desired accuracy under certain assumptions..."

> "...Under mild conditions, our warm initialization enjoys stable performance, and the subsequent iterations further improve the accuracy via linear convergence; i.e. sequence of iterates generated by Algorithm 1 converges to optimal solutions at a linear rate.

Proposition 1 (Polynomial-time angle estimation)**.** *Consider Gaussian tensor models with $b(\theta) = \theta^2/2$ in the objective function (5). Suppose the signal-to-noise ratio $\lambda^2/\phi \geq Cp^{K/2}d^{-K}$ for some sufficiently large universal constant $C > 0$. Under Assumption A1 with scaled feature matrices $\bar{\boldsymbol{X}}_k = \sqrt{d}\boldsymbol{X}_k$, or Assumption A1' with original feature matrices, the outputs from initialization Algorithm 1 and iteration Algorithm 1 satisfy the following two properties.*

(a) With probability at least $1 - \exp(-p)$.

$$\max_{k\in[K]} \sin^2 \Theta(\boldsymbol{M}_{k,\text{true}}, \hat{\boldsymbol{M}}_k^{(0)}) \leq \frac{1}{4}. \tag{2}$$

(b) Let $t = 1, 2, 3, \ldots,$ denote the iteration. There exists a contraction parameter $\rho \in (0, 1)$, such that, with probability at least $1 - \exp(-p)$,

$$\max_{k\in[K]} \sin^2\Theta(\boldsymbol{M}_{k,\text{true}}, \hat{\boldsymbol{M}}_k^{(t)}) \lesssim \underbrace{\frac{\phi p}{\lambda^2 d^K}}_{\textit{statistical error}} + \underbrace{\rho^t \max_{k\in[K]} \sin\Theta^2(\boldsymbol{M}_{k,\text{true}}, \hat{\boldsymbol{M}}_k^{(0)})}_{\textit{algorithmic error}}. \tag{3}$$

Proposition 1 provides the estimation errors for algorithm outputs at initialization and at each of the subsequent iterations. The initialization bound (2) demonstrates the stability of QR-adjusted spectral initialization under a mild SNR requirement $\lambda^2/\phi \gtrsim p^{K/2}d^{-K}$. We can think of d as the sample size while p the number of parameters at mode K. The condition confirms that a higher sample size mitigates the required signal level. The iteration bound (3) consists of two terms: the first term is the statistical error, and the second is the algorithmic error. The algorithmic error decays exponentially with the number of iterations, whereas the statistical error remains the same as t grows. The statistical error is unavoidable and also appears in the global MLE; see Theorem 4.1.

As a direct consequence, we find the optimal iteration t after which the algorithmic error is negligible compared to statistical error.

Theorem 0.1 (Statistical rate for local optimizers)**.** *Consider the same condition as in Proposition 1 and the outputs by combining algorithms 1 and 2. There exists a constant $C > 0$, such that, after $t \gtrsim K \log_{1/\rho} p$ iterations, our algorithm outputs satisfies*

$$\max_{k\in[K]} \sin^2\Theta(\boldsymbol{M}_{k,\text{true}}, \hat{\boldsymbol{M}}_k^{(t)}) \lesssim \frac{\phi p}{\lambda^2 d^K}, \quad \|\mathcal{B}_{\text{true}} - \hat{\mathcal{B}}^{(t)}\|_F^2 \lesssim \frac{\phi(r^K + Kpr)}{d^K}.$$

In practice, the signal level λ is unknown, so the assumption in Theorem 0.1 is challenging to verify in practice. We supply the theory by providing an alternative scheme – random initialization – and investigate its empirical performance..."

2. *As shown in Theorem 4.1, the rate of convergence is in the order $\sum_k p_k / \prod_k d_k$ when other terms fixed. Here p_k is the covariate size along k-th mode. I am curious if there is any insight on the optimality of this scaling. Since handling multiple covariates is one of the key contributions of this work, I think more discussion of the order of p_k would be useful. For instance, it would be helpful to compare this rate with the optimal rate established in tensor regression (Han et al., 2020) and probabilistic tensor decomposition (Wang and Li, 2020).*

Response: We have explained the intuition about our optimality and the benefit of incorporating multiple feature matrices. We added discussions to the scaling factors in Section 4.2:

"...The result in (8) implies that the estimation has a convergence rate $\mathcal{O}(Kp/d^K)$...This rate agrees with intuition, since in our setting, the number of parameters with K feature matrices is of order $\mathcal{O}(Kp)$, whereas the number of tensor entries $\mathcal{O}(d^K)$ corresponds to the total sample size. Because $p \leq d$, our rate is faster than $\mathcal{O}(d^{-(K-1)})$ obtained by tensor decomposition without features (Wang and Li, 2020).."

"...The initialization bound (2) demonstrates the stability of warm initialization under a mild SNR requirement $\lambda^2/\phi \gtrsim p^{K/2} d^{-K}$. We can think of d as the sample size while p the number of parameters at mode K. This threshold is less stringent than $d^{K/2}$ required for tensor decomposition without multiple features (Han et al., 2020; Zhang and Xia, 2018)..."

Numerical comparison:

1. *In Section 5.2 (simulation) and Section 6.1 (HCP data), only the multiple-response GLM method is compared. This setting seems to be a special case of the network response regression (Zhang et al., 2018), therefore it would be interesting to add it in the comparison. If they are not directly comparable, it would be helpful to discuss it.*

   Response: We have added comparison with 12 methods, including GLSNet (Zhang et al., 2018), in the new Section 5. For alternative methods in numerical analysis, we have also replaced the earlier old methods by more recent methods, including Zhang et al. (2018), in Section 6. See our response to **# 1, Major Comments Summarized by Editor** for full details.

2. *There are other recent tensor response regression models (Li and Zhang, 2017; Sun and Li, 2017; Raskutti et al., 2019). Some of them are missing in the paper. It would be interesting to see the comparison with more recent tensor response regression models in Section 5.3. If they are not directly comparable, it would be helpful to discuss it.*

Response: We have made changes as suggested. Comparisons are summarized in our response to **# 1, Major Comments Summarized by Editor**.

Minor issues and typos:

1. *The notation α is used for both the maximum norm constraint on page 13, and the signal level on page 19. Are they same? If not, it would be helpful to use different notations.*

   Response: They are the same. The sentence now reads:

   > "... Here $\alpha > 0$ controls the magnitude of the effect size, which is also the maximum norm of coefficient tensor as in (7)...."

2. *In the optimization (8), the authors add a maximum norm constraint on the linear predictor to avoid the divergence in the non-Gaussian variance. Can authors provide more explanations on this constraint?*

   Response: We have added the following explanation in Section 4.1:

   > "...The maximum norm constraint on the linear predictor Θ is a technical condition to ensures the existence (boundedness) of MLE. The condition precludes the ill-defined MLE when the optimizer of (6) diverges to $\pm\infty$; this phenomenon may happen in logistic regression when the Bernoulli responses $\{0, 1\}$ are perfectly separable by covariates (Wang and Li, 2020). For Gaussian models, no maximum norm constraint is needed. In Section 4.2, we show that setting α to an extremely large constant does not compromise the statistical rate in quantities of interest. In practice, the unbounded search is often indistinguishable from the bounded search, since the boundary constraint $\|\Theta\|_\infty \leq \alpha$ would likely never be active. Similar techniques are commonly used in high-dimensional non-Gaussian problems (Wang and Li, 2020; Han et al., 2020)..."

3. *What's the tuning range for the ranks $r_1, ..., r_K$ in the experiments?*

   Response: For simulation, we have added the following details in Section 6:

   > "...For each combination, we minimize BIC using a grid search from $(r_1 - 3, r_2 - 3, r_3 - 3)$ to $(r_1 + 3, r_2 + 3, r_3 + 3)$. We remove invalid rank combinations such as $r_{\max}^2 \geq \prod_{k=1}^3 r_k$ and use parallel search to reduce the computational cost...."

   For data application, we have added the following clarification in Section 7. Detailed rank selection is provided in a new section **Rank selection for Nations data, Supplementary Notes**.

   > "... We use BIC as guidance to select the rank of coefficient tensor $\mathcal{B}$. Since several rank configurations give similar BIC values, we present here the most interpretable results with $\boldsymbol{r} = (4, 4, 4)$. Detailed rank selection procedure is in Supplementary Notes..."

4. *Typos: 1) line -3 on page 18, $\boldsymbol{M}_2\boldsymbol{X}_3$ should be $\boldsymbol{M}_2\boldsymbol{X}_2$; 2) Table 2, the column names $d = 20$ and $d = 40$ are missing.*

   Response: We have made the correction as suggested.

Point-by-point Response to Reviewer 2

We greatly appreciate your valuable comments and suggestions. We have carefully addressed all the questions. Note that general comments raised by both reviewers were addressed upfront in **pages 1-7 of this response letter, Response to Major Comments Summarized by Editor**. We will make explicit reference to earlier section when needed.

The paper develops a Tucker decomposition technique for tensor data, which can consider side information in each mode. In particular, the model is built upon the exponential family to handle non-Gaussian data, and the method is formulated as a generalized tensor regression problem. An alternating algorithm is proposed to estimate the model parameters. The statistical convergence is derived for the proposed method. The applications to a brain imaging example and an international relation example reveal some interesting structures in data. Overall, the paper is well written and easy to read. The numerical results demonstrate the potential utilities of the proposed method. There are a few places where the paper can be further improved.

1. *The term "interactive" is used in the title and throughout the paper. However, it is not very clear from the paper what it means to be interactive. It seems the side information is always assumed given and static. In that case, I would recommend removing the term.*

Response: We have removed the term as suggested.

2. *There are quite a few non-Gaussian tensor models for various analyses in the literature (Tarzanagh and Michailidis, 2019; Li, 2020; Chi and Kolda, 2012; Hong et al., 2020). A more thorough literature review is warranted.*

Response: Thank you for pointing out the related literature. The methods (Chi and Kolda, 2012; Hong et al., 2020) were reviewed and compared in the first version. We have added detailed literature discussions with 12 more methods and the numerical comparisons. See our response to **# 1, Major Comments Summarized by Editor** for the full comparison.

3. *The proposed method is closely related to the envelope model (especially for Gaussian data)(Li and Zhang, 2017). It would be of interest to discuss the similarity and difference between the two.*

Response: Per reviewer's suggestion, we have added more literature reviews and updated the numerical comparisons. See our response to, **# 1, Major Comments Summarized by Editor**.

4. *The dimension requirements seem unnecessarily strong. In particular, the number of features in the side information (p_k) is required to be smaller than the dimension of the corresponding mode of the tensor data (d_k); the rank of the core tensor (r_k) needs to be smaller than p_k. In practice, how to handle a large number of features (i.e., $p > d$)? If the number of features is small (e.g., in the HCP example, there are only 2 individual features where age is further discretized as dummy variables), can the rank of the core tensor exceed the feature dimension? More discussions are needed.*

Response: There are two questions here, one on $r \leq p$, and the other on $p \leq d$. We discuss them separately.

Regarding the condition $r \leq p$. This is not a requirement but rather a natural fact based on definition of tensor rank. Notice that p is the dimension of parameter tensor of interest, and r is the rank of the parameter tensor. Numerically, one must have $r \leq p$; i.e, the rank of core tensor cannot exceed the feature dimension.

From the modeling perspective, the fact $r \leq p$ comes from the goal of the problem: we aim to identify tensor factors that are most attributable to features. In the HCP example, our goal is to find the relationship between data tensor $\mathcal{Y} \in \{0,1\}^{68\times68\times136}$ and the feature $\boldsymbol{X} \in \mathbb{R}^{4\times136}$. The fullest linear model has coefficient $\mathcal{B} \in \mathbb{R}^{68\times68\times4}$, with rank upper bounded by 4 on the last mode.

Regarding the condition $p \leq d$. This condition requires the feature to have full column rank. We view it as a mild condition, because same condition is also imposed in classical linear regression. In the presence of rank deficiency, we recommend to remove redundant features before applying our method.

We have added more discussions in Sections 3 and 4.2.

> "..the identifiability of $\mathcal{B}$ requires the feature matrices $\boldsymbol{X}_k$ are of full column rank with $p_k \leq d_k$. We impose this rank non-deficiency assumption to $\boldsymbol{X}_k$; this is a mild condition common in literature (Lock and Li, 2018; Li and Zhang, 2017; Li, 2020). In the presence of rank deficiency, we recommend to remove redundant features from $\boldsymbol{X}_k$ before applying our method. ..."

> "...the requirement $p \leq d$ is necessary to ensure rank non-deficiency of feature matrices $\boldsymbol{X}_k$.."

5. *A related question to the above comment: if the side information is little (small p_k) or irrelevant to the underlying structure of the tensor data, would the proposed model structure (3) be overly restrictive rather than helpful? In other words, the "supervised tensor factor" $\boldsymbol{X}_k\boldsymbol{M}_k$ must reside in the column space of the features $\boldsymbol{X}_k$. If $\boldsymbol{X}_k$ is not well selected, would that lead to a biased decomposition of the original data? In practice, how do we know if a set of features are appropriate to use? The envelope model (Li and Zhang, 2017; Lock and Li, 2018) may offer an alternative perspective by separating the data variation into a material part (related to $\boldsymbol{X}_k$) and an immaterial part (irrelevant to $\boldsymbol{X}_k$). It's not clear whether the idea directly applies to the proposed method in this paper since the low-rank model is on the deterministic mean tensor. Some discussions would be helpful.*

Response: We agree with the basic premise of the reviewer that the our method relies on the choice of feature matrices. In fact, this does not undermine our method's utility. A major point we want to make in response is that the goal of our problem is to estimate the *conditional pattern* of $\mathcal{Y}|\boldsymbol{X}$, not the *overall pattern* of $\mathcal{Y}$. This motivation has guaranteed the validness (unbiasedness) of our factors, even when the $\boldsymbol{X}$ is poorly related to $\mathcal{Y}$. The tensor factors we learned are not intended to explain the most variation in the $\mathcal{Y}$; instead, they are intended to capture the variation that is most attributable to features.

There are methods that jointly impose data variation in conditional component $\mathcal{Y}|\boldsymbol{X}$ and residual component $\mathcal{Y}-\mathcal{Y}|\boldsymbol{X}$. These methods have the benefits of joint learning the conditional variation of $\mathbb{E}(\mathcal{Y}|\boldsymbol{X})$ and the variation of residuals. The referred methods, Envelope (Li and Zhang, 2017) and SupCP (Lock and Li, 2018), fall into this category. We have added connection and comparison with these two methods in Section 5 and in Section 6. See our response to **# 1, Major Comments Summarized by Editor**.

For reviewer's interest, we discuss the practical guidance about "which method should we use."

(a) In general, the answer depends on the goal and problem at hand. If the goal is to identify the relationship between tensor data $\mathcal{Y}$ and features $\boldsymbol{X}$, then we would recommend our method. In the three examples illustrated in Section 3.2 and two data applications in Section 7, we

are interested in identifying the relationship between data tensor (brain connectivities, social community patterns) and available features (age, gender). In these cases, direct modeling $\mathcal{Y}|\boldsymbol{X}$ yields a higher accuracy, especially when variation in $\mathcal{Y}|\boldsymbol{X}$ is weak compared to the overall variation in $\mathcal{Y}$.

(b) If the goal is to identify unknown variation in the data tensor $\mathcal{Y}$, then we recommend Envelope (Li and Zhang, 2017) and SupCP (Lock and Li, 2018). These two methods have the benefits of separating the variations of $\mathcal{Y}$ into explained part by $\boldsymbol{X}$ and unexplained part due to unmeasured latent factors.

(c) If the goal is to identify the relationship between $\mathcal{Y}$ and $\boldsymbol{X}$ while adjusting for latent factors, then we would recommend generalized mixed-effect tensor decomposition (MSTD). The MSTD combines the advantages of our STD method and the envelope method. We have described MSTD in the original version of Discussion, and we now elaborate the discussion in light of aforementioned work.

> "(our method)...can be extended by introducing a more general mixed-effect tensor model. For example, in the special case of Gaussian model, we can model the first two moments of data tensor using
>
> $$\mathbb{E}(\mathcal{Y}|\boldsymbol{X}_1,\ldots,\boldsymbol{X}_K) = \mathcal{C}\times\boldsymbol{M}_1\times\cdots\times\boldsymbol{M}_K,$$
> $$\mathrm{Var}(\mathcal{Y}|\boldsymbol{X}_1,\ldots,\boldsymbol{X}_K) = \boldsymbol{\Phi}_1\otimes\cdots\otimes\boldsymbol{\Phi}_K,$$
>
> where $\boldsymbol{\Phi}_k\in\mathbb{R}^{d_k\times d_k}$ is the unknown covariance matrix on the mode $k\in[K]$. For general exponential family, an additional mean-variance relationship should also be considered. The joint estimation of mean model Θ and variance model $\boldsymbol{\Phi}_k$ will lead to more efficient estimation in the presence of unmeasured confounding effects....Suitable regularization such as .... specially-structured covariance (Li and Zhang, 2017; Lock and Li, 2018) should be considered..."

6. *Identifiability. Is the proposed model (3) identifiable? It seems not from Definition 1 (Equivalence class). If it's not identifiable, what benefit does the orthogonality constraint on $\boldsymbol{M}_k$ induce? How to interpret the estimated parameters?*

Response: Our method is to estimate the low-rank tensor $\mathcal{B}$, or equivalently, the core tensor $\mathcal{C}$ and factors $\boldsymbol{M}_k$ that forms $\mathcal{B}=\mathcal{C}\times\{\boldsymbol{M}_1,...,\boldsymbol{M}_K\}$. The tensor $\mathcal{B}$ is identifiable while the factors $\{\mathcal{C},\boldsymbol{M}_1,...,\boldsymbol{M}_K\}$ are not. The non-unique factors incur no concern to the interpretation, because all our parameter estimation and interpretation are about $\mathcal{B}$.

The orthonormality of $\boldsymbol{M}_k$ is imposed purely for technical convenience. This normalization incurs no impacts in our statistical inference, but may help with numerical stability in empirical optimization (De Lathauwer et al., 2000; Kolda and Bader, 2009).

We have added the following discussions in Section 3.1 and Section 4.2.1:

> "...We make several remarks about model identifiability. First, the identifiability of $\mathcal{B}$ requires the feature matrices $\boldsymbol{X}_k$ are of full column rank....Second, the decomposition $\mathcal{B}=\mathcal{C}\times\{\boldsymbol{M}_1,\ldots,\boldsymbol{M}_K\}$ are non-unique, as in standard tensor decomposition (Kolda and Bader, 2009). For any invertible matrices $\boldsymbol{O}_k\in\mathbb{R}^{r_k\times r_k}$, $\mathcal{B}=\mathcal{C}\times\{\boldsymbol{M}_1,\ldots,\boldsymbol{M}_K\}=\mathcal{C}'\times\{\boldsymbol{M}_1\boldsymbol{O}_1,\ldots,\boldsymbol{M}_K\boldsymbol{O}_K\}$ are two equivalent parameterizations with $\mathcal{C}'=\mathcal{C}\times\{\boldsymbol{O}_1^{-1},\ldots,\boldsymbol{O}_K^{-1}\}$. To resolve this ambiguity, we impose orthonormality to $\boldsymbol{M}_k\in\mathbb{O}(p_k,r_k)$ and assess the estimation error of $\boldsymbol{M}_k$ using angle distance. The angle distance is invariant to orthogonal rotations due to its geometric definition. See Section 4.2 for more details. The

orthonormality of $\boldsymbol{M}_k$ is imposed purely for technical convenience. This normalization incurs no impacts in our statistical inference, but may help with numerical stability in empirical optimization (De Lathauwer et al., 2000; Kolda and Bader, 2009)."

"...Recall that the factor matrices $\boldsymbol{M}_k$ are identifiable only up to orthogonal rotations. Therefore, we choose to use angle distance to assess the estimation accuracy of $\boldsymbol{M}_k$..."

7. *The maximum norm constraint on the linear predictor Theta is a bit ad hoc. How to choose the upper bound α in practice? How does the value of alpha affect the final results?*

Response: The maximum norm constraint is a technical condition purely for theoretical analysis. In practice, no choice of α is needed.

We have added the following explanations in Section 4.1:

"...The maximum norm constraint on the linear predictor Θ is a technical condition to ensures the existence (boundedness) of MLE. The condition precludes the ill-defined MLE when the optimizer of (6) diverges to $\pm\infty$; this phenomenon may happen in logistic regression when the Bernoulli responses $\{0, 1\}$ are perfectly separable by covariates (Wang and Li, 2020). For Gaussian models, no maximum norm constraint is needed. In Section 4.2, we show that setting α to an extremely large constant does not compromise the statistical rate in quantities of interest. In practice, the unbounded search is often indistinguishable from the bounded search, since the boundary constraint $\|\Theta\|_\infty \leq \alpha$ would likely never be active. Similar techniques are commonly used in high-dimensional non-Gaussian problems (Wang and Li, 2020; Han et al., 2020)....

8. *The authors propose to use BIC to select ranks via grid search. However, for a high-order tensor, one may have to search over a multi-dimensional grid to find the best rank combination. It seems computationally prohibitive (even for a 3-way tensor). Is there any practical guidance about how to more efficiently select the ranks? What are the computational costs for rank selection in the numerical examples?*

Response: We utilize parallel computing in BIC search to reduce the computational cost. In the absence of parallel computation, one may speed up the rank search using uni-modal properties of marginal BIC. Specifically, we suggest alternating the search between modes and bracket the possible rank range in one mode while holding others fixed. Once the search region is reduced, a small-scale grid search is performed to identify the minimal BIC. We find such practice gives reasonable rank estimates in the two applications considered.

We have added the following sentence in Section 6:

"(In simulation)...For each combination, we minimize BIC using a grid search from $(r_1 - 3, r_2 - 3, r_3 - 3)$ to $(r_1 + 3, r_2 + 3, r_3 + 3)$. We removed invalid rank combinations such as $r_{\max}^2 \geq \prod_k r_k$ and used parallel search to reduce the computational cost..."

The BIC selection on real data is detailed in Supplementary Notes, Section C:

"(In real data)...The running time for the rank selection via grid search is 95 secs in total, on an iMac macOS High Sierra 10.13.6 with Intel Core i5 3.8 GHz CPU and 8 GB RAM. This indicates the BIC is feasible in the considered setting. "

9. *In numerical studies, the authors compare the proposed method with a multiple-response GLM approach and comment that the latter is suboptimal because it cannot account for the correlation*

among responses. There are methods in the literature that can address the issue (Yee and Hastie, 2003; Luo et al., 2018). It would be interesting to see some updated comparisons.

Response: We have added new numerical comparisons using aforementioned references in the current version. See our response to **# 1, Major Comments Summarized by Editor**.

10. *In the HCP example, how are the networks constructed in Fig 6? Will the connectivity change with rotations in the equivalence class? In the Nations data example, what would the clustering results be without accounting for the national features (i.e., directly applying Tucker decomposition to the data)? More clarifications are warranted.*

Response: Figure 7 plots top entries in $\mathcal{B}_{\cdot\cdot i}$ for $i = 1, 2, 3, 4$. The connectivity is invariant to the rotation of factor matrices. See our response to **# 6, Reviewer 1**.

As suggested by reviewer, we have added comparison between supervised vs. unsupervised decomposition for Nations data analysis. The full results are in Section C.2-C.3 Supplementary Notes:

> "...We compare the supervised vs. unsupervised decomposition in the *Nations* data analysis. Table 2 shows the clustering results based on classical unsupervised Tucker decomposition without the feature matrices. Table 3 shows the clustering results based on supervised tensor decomposition (**STD**). Compared with supervised decomposition, the unsupervised clustering loses some interpretation. Similar relations *exports* and *relexports*, *ngo* and *relngo* are separated into different clusters..."

Cluster	Relations
I	economicaid, releconomicaid, exportbooks, relexportbooks, weightedunvote, unweightedunvote, tourism, reltourism, tourism3, exports, intergovorgs, ngo, militaryalliance
II	warning, violentactions, militaryactions, duration, severdiplomatic, expeldiplomats, boycottembargo, aidenemy, negativecomm, accusation, protests, unoffialacts, attackembassy, relemigrants, timesincewar, lostterritory, dependent
III	timesinceally, independence, commonbloc0, blockpositionindex
IV	treaties, reltreaties, officialvisits, conferences, booktranslations, relbooktranslations negativebehavior, nonviolentbehavior, emigrants, emigrants3, students, relstudents, relexports, exports3 relintergovorgs, relngo, intergovorgs3, ngoorgs3, embassy, reldiplomacy, commonbloc1, commonbloc2

Economics Military Organization Territory

Table 2: Clustering of relations based on unsupervised tensor decomposition.

Category	Relations
I	warning, violentactions, militaryactions, duration, negativebehavior, protests, severdiplomatic timesincewar, commonbloc0, commonbloc1, blockpositionindex, expeldiplomats
II	emigrants, emigrants3, relemigrants, accusation, nonviolentbehavior, ngoorgs3, commonbloc2, intergovorgs3 releconomicaid, relintergovorgs, relngo, students, relstudents, economicaid, negativecomm, militaryalliance
III	treaties, reltreaties, officialvisits, exportbooks, relexportbooks, booktranslations, relbooktranslations boycottembargo, weightedunvote, unweightedunvote, reltourism, tourism, tourism3, exports, exports3 relexports, intergovorgs, ngo, embassy, reldiplomacy, timesinceally, independence, conferences, dependent
IV	aidenemy, lostterritory, unoffialacts, attackembassy

Economics Military Organization Territory

Table 3: Clustering of relations based on supervised tensor decomposition.

We comment that, although both methods return clustering outputs, the unsupervised and supervised decomposition tackle different problems. The unsupervised decomposition identifies factors that explain most variation in the tensor, whereas the supervised decomposition identifies factors that are most attributable to side features. There is in general no particular pattern one could expect between these two methods. We provide a simple example here for illustration.

Example 1 (Complementary information in supervised vs. unsupervised factors)**.** Consider the following data tensor $\mathcal{Y}$ and one-sided feature matrix $\boldsymbol{X}$,

$$\mathcal{Y} = \boldsymbol{e}_1 \otimes \boldsymbol{e}_1 \otimes \boldsymbol{e}_1 + 10\boldsymbol{e}_2 \otimes \boldsymbol{e}_2 \otimes \boldsymbol{e}_2, \quad \boldsymbol{X} = \boldsymbol{e}_1,$$

where $\boldsymbol{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0)^T$ is the ith canonical basis vector in $\mathbb{R}^d$ for $i = 1, 2$. Now, consider the unsupervised vs. supervised decomposition of $\mathcal{Y}$ with rank $\boldsymbol{r} = (1, 1, 1)$. Then, the top supervised and unsupervised factors are perpendicular to each other,

$$\boldsymbol{M}_{\text{sup},k} \perp \boldsymbol{M}_{\text{unsup},k}, \quad \text{for all } k = 1, 2, 3,$$

where $\boldsymbol{M}_{\text{sup},k}$, $\boldsymbol{M}_{\text{unsup},k}$ denote the mode-k factors from supervised and unsupervised decompositions, respectively.

Remark 1. This example shows complementary information between factors from supervised vs. unsupervised decompositions. In general, one could construct examples such that these two methods return **arbitrarily different** factors.

References

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

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

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

Gahrooei, M. R., Yan, H., Paynabar, K., and Shi, J. (2020). Multiple tensor-on-tensor regression: An approach for modeling processes with heterogeneous sources of data. *Technometrics*, 63(2):1–23.

Han, R., Willett, R., and Zhang, A. (2020). An optimal statistical and computational framework for generalized tensor estimation. *The Annals of Statistics, In press. arXiv preprint arXiv:2002.11255*.

Hao, B., Wang, B., Wang, P., Zhang, J., Yang, J., and Sun, W. W. (2021). Sparse tensor additive regression. *Journal of Machine Learning Research*, 22(64):1–43.

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

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

Li, G. (2020). Generalized co-clustering analysis via regularized alternating least squares. *Computational Statistics & Data Analysis*, 150:106989.

Li, L. and Zhang, X. (2017). Parsimonious tensor response regression. *Journal of the American Statistical Association*, 112(519):1131–1146.

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

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

Luo, C., Liang, J., Li, G., Wang, F., Zhang, C., Dey, D. K., and Chen, K. (2018). Leveraging mixed and incomplete outcomes via reduced-rank modeling. *Journal of Multivariate Analysis*, 167:378–394.

Luo, Y. and Zhang, A. R. (2021). Low-rank tensor estimation via riemannian gauss-newton: Statistical optimality and second-order convergence. *arXiv preprint arXiv:2104.12031*.

Rabusseau, G. and Kadri, H. (2016). Low-rank regression with tensor responses. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 29:1875–1883.

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

Sun, W. W. and Li, L. (2017). STORE: sparse tensor response regression and neuroimaging analysis. *The Journal of Machine Learning Research*, 18(1):4908–4944.

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

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

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

Yee, T. W. and Hastie, T. J. (2003). Reduced-rank vector generalized linear models. *Statistical modelling*, 3(1):15–41.

Zhang, A. and Xia, D. (2018). Tensor SVD: Statistical and computational limits. *IEEE Transactions on Information Theory*, 64(11):7311–7338.

Zhang, J., Sun, W. W., and Li, L. (2018). Network response regression for modeling population of networks with covariates. *arXiv preprint arXiv:1810.03192*.

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