Conformalized Tensor Completion with Riemannian Optimization

Hu Sun
Department of Statistics, University of Michigan, Ann Arbor
and
Yang Chen*
Department of Statistics and Michigan Institute for Data Science
University of Michigan, Ann Arbor

September 24, 2025

Abstract

Tensor data, or multi-dimensional arrays, is a data format popular in multiple fields such as social network analysis, recommender systems, and brain imaging. It is not uncommon to observe tensor data containing missing values, and tensor completion aims at estimating the missing values given the partially observed tensor. Sufficient efforts have been spared on devising scalable tensor completion algorithms, but few on quantifying the uncertainty of the estimator. In this paper, we nest the uncertainty quantification (UQ) of tensor completion under a split conformal prediction framework and establish the connection of the UQ problem to a problem of estimating the missing propensity of each tensor entry. We model the data missingness of the tensor with a tensor Ising model parameterized by a low-rank tensor parameter. We propose to estimate the tensor parameter by maximum pseudo-likelihood estimation (MPLE) with a Riemannian gradient descent algorithm. Extensive simulation studies have been conducted to justify the validity of the resulting conformal interval. We apply our method to the regional total electron content (TEC) reconstruction problem. Supplemental materials of the paper are available online.

Keywords: Tensor Completion; Uncertainty Quantification; Conformal Prediction; Riemannian Gradient Descent; Binary Tensor Decomposition.

*Email: ychenang@umich.edu

1 Introduction

Tensor, or multi-dimensional array, has become a popular data format in several applications such as collaborative filtering (Bi et al. 2018), financial time series modeling (Li & Xiao 2021), hypergraph networks analysis (Ke et al. 2019), neuroimaging study (Li et al. 2018), and astrophysics imaging analysis (Sun, Manchester, Jin, Liu & Chen 2023). Tensor gains this popularity due to its efficient representation of structural high-dimensional data. For example, in collaborative filtering (Bi et al. 2018), the rating data is naturally embedded in a 3-way tensor with each entry being the rating by a user on a certain item under a specific context. In neuroimaging analysis (Wei et al. 2023), as another example, each brain voxel in the 3-way tensor is identified by its coordinate in the 3-D Euclidean space.

Tensor completion (Yuan & Zhang 2016, Xia et al. 2021, Cai, Li, Poor & Chen 2022) is a technique that provides an estimator of the tensor when missing values are present. Typically, given only one tensor sample with missingness, tensor completion aims at finding a low-rank tensor that best imputes the missing entries. Various optimization techniques (Kressner et al. 2014, Yuan & Zhang 2016, Wang et al. 2019, Lee & Wang 2020, Cai, Li, Poor & Chen 2022, Qi et al. 2023) have been proposed for computationally efficient tensor completion and the statistical error of tensor completion has also been carefully investigated (Xia et al. 2021).

However, given the progress above, very little work has been done on the uncertainty quantification of tensor completion. Existing work on the uncertainty quantification of matrix completion (Chen, Fan, Ma & Yan 2019) and tensor completion (Cai, Poor & Chen 2022) typically relies on asymptotic analysis of the estimator by a specific completion algorithm and assumes that data is missing uniformly at random. In this paper, we aim to devise a data-driven approach that does not rely on a specific choice of the completion algorithm nor assume the data is missing uniformly at random, which is more adaptive to

real application scenarios.

Conformal prediction (Vovk et al. 2005) is a model-agnostic approach for uncertainty quantification. Recently, Gui et al. (2023) applies the idea of conformal prediction to matrix completion under the assumption that data is missing independently. The method requires one to estimate the missing propensity of each matrix entry and weigh them accordingly to construct well-calibrated confidence regions. In this paper, we generalize this idea to tensor completion. The generalization is non-trivial, as one cannot simply reshape the tensor back to a matrix for the conformal prediction without significantly increasing the dimensionality of the nuisance parameter. We keep the tensor structure and leverage low-rank tensor representations for dimension reduction. Furthermore, we do not assume data is missing independently but allow for locally dependent missingness. We capture such dependency of missingness by a novel low-rank tensor Ising model, which could be of independent interest. Finally, we propose a Riemannian gradient descent algorithm (Kressner et al. 2014) for scalable computation, which is necessary since tensor data is typically high-dimensional.

The key insight of the method is that one puts a higher weight on the tensor entries with a higher probability of missing. Such a weighted conformal prediction approach (Tibshirani et al. 2019) is also seen in spatial conformal prediction (Mao et al. 2022) and localized conformal prediction (Guan 2023), where higher weights are put on neighbors in the Euclidean or feature space. However, our method is significantly different in that we estimate the weights by using the entire tensor and determine the weights of all tensor entries altogether, while other methods determine the weight of each data locally and thus can be slow under the tensor setting.

The remainder of the paper is organized as follows. We outline the notations used in the paper in Section 1.1. Section 2 describes the conformalized tensor completion (CTC) method and the probabilistic model for the data missingness. Section 3 is dedicated to the computational algorithm of the CTC. We validate the performance of our proposed CTC

using extensive simulations in Section 4 and a real data application to a geophysics dataset in Section 5. Section 6 concludes. The supplemental material contains technical proofs and additional details and results of the simulation and data application.

1.1 Notation

Throughout this paper, we use calligraphic boldface letters (e.g. $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}$) for tensors with at least three modes, boldface uppercase letters (e.g. $\mathbf{X}, \mathbf{Y}$) for matrices, boldface lowercase letters (e.g. $\mathbf{u}, \mathbf{v}$) for vectors, and blackboard boldface letters (e.g. $\mathbb{S}, \mathbb{T}$) for sets. To index a tensor/matrix/vector, we use square brackets with subscripts such as $[\boldsymbol{\mathcal{A}}]_{i_1 \ldots i_K}, [\mathbf{X}]_{ij}, [\mathbf{u}]_i$, and will ignore the square brackets when it is clear from the context. For a positive integer n, we denote its index set $\{1, \ldots, n\}$ as $[n]$. For a K-mode tensor with size $d_1 \times \cdots \times d_K$, we use $\mathbb{S}$ to denote $[d_1] \times \cdots \times [d_K]$, namely the indices of all tensor entries, and we often use a single index such as i, j, s instead of a K-tuple to denote elements from $\mathbb{S}$ for notational brevity.

For any tensors $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, we use $\mathbf{vec}(\boldsymbol{\mathcal{X}}), \mathbf{vec}(\boldsymbol{\mathcal{Y}})$ to denote the corresponding vectorized tensors, where all entries are aligned in such an order that the first index changes the fastest. We use $\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle$ to denote tensor inner product and basically $\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle = \mathbf{vec}(\boldsymbol{\mathcal{X}})^\top \mathbf{vec}(\boldsymbol{\mathcal{Y}})$. Tensor Frobenius norm $\|\boldsymbol{\mathcal{X}}\|_{\mathrm{F}}$ is defined as $\sqrt{\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}} \rangle}$ and tensor max-norm $\|\boldsymbol{\mathcal{X}}\|_\infty$ is defined as $\max_{s \in \mathbb{S}} |\boldsymbol{\mathcal{X}}_s|$. For any tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ and any matrix $\mathbf{U} \in \mathbb{R}^{J \times d_k}$, the k-th mode tensor-matrix product, denoted as $\boldsymbol{\mathcal{X}} \times_k \mathbf{U}$, is a tensor of size $d_1 \times \cdots \times d_{k-1} \times J \times d_{k+1} \times \cdots \times d_k$ that satisfies:

$$[\boldsymbol{\mathcal{X}} \times_k \mathbf{U}]_{i_1 \ldots i_{k-1} j i_{k+1} \ldots i_K} = \sum_{i_k=1}^{d_k} [\boldsymbol{\mathcal{X}}]_{i_1 \ldots i_k \ldots d_K} [\mathbf{U}]_{j i_k}.$$

More preliminaries on tensor notations and the related algebra will be covered in later sections, and we refer our readers to Kolda & Bader (2009) for more references on the related tensor algebra. In this paper, when referring to a tensor that is a random variable,

we add a tilde over the top of the tensor, such as $\widetilde{\boldsymbol{\mathcal{W}}}, \widetilde{\boldsymbol{\mathcal{X}}}$, and use the raw version $\boldsymbol{\mathcal{W}}, \boldsymbol{\mathcal{X}}$ to denote concrete samples. We add an asterisk to the superscript, such as $\boldsymbol{\mathcal{X}}^*, \boldsymbol{\mathcal{B}}^*$ to denote the non-random, ground truth parameters.

2 Method

Suppose we have a K-mode random tensor $\widetilde{\boldsymbol{\mathcal{X}}}$ of size $d_1 \times \cdots \times d_K$. Further, suppose that one obtains a sample $\boldsymbol{\mathcal{X}}$ for $\widetilde{\boldsymbol{\mathcal{X}}}$ with part of the entries in $\boldsymbol{\mathcal{X}}$ missing. To encode the missingness in $\boldsymbol{\mathcal{X}}$, we define the binary missingness tensor $\boldsymbol{\mathcal{W}} \in \{-1, 1\}^{d_1 \times \cdots \times d_K}$ and set $\boldsymbol{\mathcal{W}}_s = 1$ when $\boldsymbol{\mathcal{X}}_s$ is observed and $\boldsymbol{\mathcal{W}}_s = -1$ when $\boldsymbol{\mathcal{X}}_s$ is missing. We assume that the missingness $\boldsymbol{\mathcal{W}}$ is a sample of a random binary tensor $\widetilde{\boldsymbol{\mathcal{W}}}$ whose likelihood is $p(\cdot)$.

The tensor completion problem (Yuan & Zhang 2016, Xia et al. 2021, Cai, Li, Poor & Chen 2022) deals with estimating the values in $\boldsymbol{\mathcal{X}}$ where $\boldsymbol{\mathcal{W}}_s = -1$, i.e., where data is missing. Although the main framework of our paper does not rely on a specific choice of the tensor completion algorithm, it is beneficial to provide one example here, which is also the algorithm we will be using in our numerical experiments and data application.

Since one only has one sample $\boldsymbol{\mathcal{X}}$ of $\widetilde{\boldsymbol{\mathcal{X}}}$, estimating the missing values in $\boldsymbol{\mathcal{X}}$ is impossible without imposing additional parsimony over the estimator. Following the literature on tensor completion (Kressner et al. 2014, Xia et al. 2021, Cai, Li & Xia 2022*b*), we assume that the estimator has a low tensor rank and solve for the estimator by the following constrained least-square problem:

$$\min_{\boldsymbol{\mathcal{A}}:\mathrm{rank}(\boldsymbol{\mathcal{A}})\leq \mathbf{r}} \quad \frac{1}{2} \sum_{s:\boldsymbol{\mathcal{W}}_s=1} (\boldsymbol{\mathcal{X}}_s - \boldsymbol{\mathcal{A}}_s)^2 , \tag{1}$$

where the notion of tensor rank will be introduced later. We denote the minimizer of (1) as $\widehat{\boldsymbol{\mathcal{X}}}$. The goal of the paper is to quantify the uncertainty for $\widehat{\boldsymbol{\mathcal{X}}}$ by constructing a confidence interval $C(\widehat{\boldsymbol{\mathcal{X}}})$ around $\widehat{\boldsymbol{\mathcal{X}}}$ to cover $\boldsymbol{\mathcal{X}}$ with a pre-specified level of confidence. The framework, called conformalized tensor completion, will be introduced next.

2.1 Conformalized Tensor Completion (CTC)

Conformal prediction (Vovk et al. 2005) is a model-agnostic, distribution-free approach for predictive uncertainty quantification. To put in the context of the tensor completion problem, we utilize specifically the *split conformal prediction* (Papadopoulos et al. 2002) approach for its simplicity and scalability to complex data structures such as tensor data. We leave the discussion of *full conformal prediction* (Shafer & Vovk 2008) to future work.

Split conformal prediction starts by partitioning all observed entries in $\boldsymbol{\mathcal{X}}$, whose indices are denoted as $\mathbb{S}_{obs}$, randomly into a training set $\mathbb{S}_{tr}$ and a calibration set $\mathbb{S}_{cal}$. One first provides a tensor completion estimator $\widehat{\boldsymbol{\mathcal{X}}}$ using the training set *only*, say by solving for (1) using entries in $\mathbb{S}_{tr}$. Then one calculates the *non-conformity score* over the calibration set by a score function $\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s)$ such as $\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s) = |\boldsymbol{\mathcal{X}}_s - \widehat{\boldsymbol{\mathcal{X}}}_s|$. To quantify the uncertainty of $\widehat{\boldsymbol{\mathcal{X}}}_{s^*}$ at any missing entry $s^* \in \mathbb{S}_{miss}$, where $\mathbb{S}_{miss}$ includes the indices of all missing entries, the canonical conformal interval at $(1-\alpha)$ confidence level is constructed as $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) = \{x \in \mathbb{R} | \mathcal{S}(x, \widehat{\boldsymbol{\mathcal{X}}}_{s^*}) \leq \widehat{q}\}$, with $\widehat{q}$ defined as:

$$\widehat{q} = \mathcal{Q}_{1-\alpha}\left(\frac{1}{|\mathbb{S}_{cal}|+1} \cdot \sum_{s \in \mathbb{S}_{cal}} \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s)} + \frac{1}{|\mathbb{S}_{cal}|+1} \cdot \delta_{+\infty}\right), \tag{2}$$

where δ_a is a point mass at $x = a$ and $\mathcal{Q}_\tau(\cdot)$ extracts the $(100\tau)^{\text{th}}$ quantile of a distribution. The validity of such a conformal interval $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}})$ relies on the assumption of *data exchangeability* (Lei et al. 2018). To put it in the context of tensor completion, we re-label $\mathbb{S}_{cal} \cup \{s^*\}$ as $\{s_1, \ldots, s_{n+1}\}$, with $n = |\mathbb{S}_{cal}|$ and $s_{n+1} = s^*$ and define event $\boldsymbol{\mathcal{E}}_0$ as:

$$\boldsymbol{\mathcal{E}}_0 = \left\{\widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ if and only if } s \in \mathbb{S}_{tr} \cup \mathbb{S}', \mathbb{S}' \subset \{s_1, \ldots, s_{n+1}\} \text{ and } |\mathbb{S}'| = n\right\}. \tag{3}$$

The data exchangeability assumption is equivalent to saying that the probability:

$$\mathrm{P}\left[\widetilde{\boldsymbol{\mathcal{W}}}_{s_k} = -1 \text{ and } \widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ for } s \in \mathbb{S}_k \middle| \boldsymbol{\mathcal{E}}_0\right]$$

is equal for all $k = 1, \ldots, n+1$, where $\mathbb{S}_k = \{s_1, \ldots, s_{n+1}\} \setminus \{s_k\}$. Equivalently, this

states that conditioning on observing data only from $\mathbb{S}_{tr}$ and n out of $n+1$ entries from $\{s_1, \ldots, s_{n+1}\}$, it is equally likely to observe any n entries from $\{s_1, \ldots, s_{n+1}\}$. This assumption will hold when data are missing independently with the same probability, a common assumption made in the literature on matrix/tensor completion uncertainty quantification (Chen, Fan, Ma & Yan 2019, Cai, Poor & Chen 2022). However, this assumption might not hold when the data missingness is dependent or when the missingness is independent but with heterogeneous probabilities. Therefore, it is necessary to account for more general data missing patterns when conducting uncertainty quantification.

We modify the canonical conformal prediction to accommodate more general data missing patterns by re-weighting each calibration entry using the weighted exchangeability framework (Tibshirani et al. 2019). The result is summarized in Proposition 2.1.

Proposition 2.1. *For any testing entry $s^* \in \mathbb{S}_{miss}$, let $s^* = s_{n+1}$ and $\mathbb{S}_{cal} \cup \{s^*\} = \{s_1, \ldots, s_{n+1}\}$ and $\mathbb{S}_k = \{s_1, \ldots, s_{n+1}\} \setminus \{s_k\}$, then define $p_k(s^*)$ as:*

$$p_k(s^*) = \mathrm{P}\left(\widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ if and only if } s \in \mathbb{S}_{tr} \cup \mathbb{S}_k\right), \tag{4}$$

for $k = 1, \ldots, n+1$. Let $\widehat{\boldsymbol{\mathcal{X}}}$ be the output of any tensor completion method using entries only from $\mathbb{S}_{tr}$ and define $\widehat{q}_{s^*}$ as:

$$\widehat{q}_{s^*} = \mathcal{Q}_{1-\alpha}\left(\sum_{i=1}^{n} \omega_i(s^*) \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})} + \omega_{n+1}(s^*) \cdot \delta_{+\infty}\right), \quad \text{where } \omega_k(s^*) = \frac{p_k(s^*)}{\sum_{i=1}^{n+1} p_i(s^*)}, \tag{5}$$

and construct the $(1-\alpha)$-level conformal interval as $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) = \{x \in \mathbb{R} | \mathcal{S}(x, \widehat{\boldsymbol{\mathcal{X}}}_{s^*}) \leq \widehat{q}_{s^*}\}$, then given the definition of $\boldsymbol{\mathcal{E}}_0$ in (3), we have:

$$\mathrm{P}\left(\boldsymbol{\mathcal{X}}_{s^*} \in C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) \middle| \boldsymbol{\mathcal{E}}_0\right) \geq 1 - \alpha. \tag{6}$$

We provide the detailed proof in Appendix A.1. Proposition 2.1 indicates that as long as one can properly weight each calibration entry in proportion to $p_k(s^*)$ as defined in (4), one can obtain the conditional coverage guarantee in (6). A similar result to Proposition 2.1

has been established for conformalized matrix completion (Gui et al. 2023), where the data is assumed to be missing independently. In our paper, we do not assume independent missingness but provide a more general statement that requires one to weight each calibration and testing entry by directly evaluating the likelihood of $\widetilde{\boldsymbol{\mathcal{W}}}$ under $n+1$ different missingness, where each time we set 1 out of $n+1$ entries as missing. In Section 2.2, we will formally introduce the likelihood of the binary tensor $\widetilde{\boldsymbol{\mathcal{W}}}$ that nests the independent missingness as a special case.

2.2 Missing Propensity Model

The key to constructing the conformal interval with coverage guarantee is to properly weight each calibration sample by $p_k(s^*)$ in (4), which requires the knowledge of the likelihood of $\widetilde{\boldsymbol{\mathcal{W}}}$. In practice, one does not have access to such knowledge but needs to estimate the likelihood of $\widetilde{\boldsymbol{\mathcal{W}}}$, given a single sample $\boldsymbol{\mathcal{W}}$, and then plug in (4) to get an estimator $\widehat{p}_k(s^*)$. Previous works (Chen, Fan, Ma & Yan 2019, Cai, Poor & Chen 2022, Gui et al. 2023) assume that all matrix/tensor entries are missing independently, potentially with heterogeneous probabilities. This assumption, however, is not general enough. For example, for spatio-temporal tensors, data might be missing together if located close in space or time.

Accounting for the dependencies of binary random variables turns out to be even more challenging in our context because all the binary random variables in $\widetilde{\boldsymbol{\mathcal{W}}}$ are embedded in a tensor grid with ultra-high dimensionality. In this paper, we do not account for arbitrary data missing patterns but focus on independent missingness and locally dependent missingness. These two types of missingness are common for many data applications, such as recommender systems (Bi et al. 2018), neuro-imaging (Li et al. 2017), and remote sensing (Sun et al. 2022). A more flexible dependency structure for missingness could be modeled; however, for tractability, we limit our focus to these two settings via the lens of the Ising model (Cipra 1987), which provides a way of modeling dependency among binary

random variables.

To start with, the Ising model prescribes a Boltzmann distribution for $\widetilde{\boldsymbol{\mathcal{W}}}$: $p(\widetilde{\boldsymbol{\mathcal{W}}}) \propto \exp[-\beta\mathcal{H}(\widetilde{\boldsymbol{\mathcal{W}}})]$, where $\beta > 0$ is the inverse temperature parameter and $\mathcal{H}(\widetilde{\boldsymbol{\mathcal{W}}})$ is the *Hamiltonian* of $\widetilde{\boldsymbol{\mathcal{W}}}$, describing the "energy" of $\widetilde{\boldsymbol{\mathcal{W}}}$. In our paper, we extend the richness of this model by augmenting $p(\widetilde{\boldsymbol{\mathcal{W}}})$ with an unknown tensor parameter $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ such that:

$$p(\widetilde{\boldsymbol{\mathcal{W}}}|\boldsymbol{\mathcal{B}}) \propto \exp\{-\mathcal{H}(\widetilde{\boldsymbol{\mathcal{W}}}|\boldsymbol{\mathcal{B}})\} \tag{7}$$

$$\mathcal{H}(\widetilde{\boldsymbol{\mathcal{W}}}|\boldsymbol{\mathcal{B}}) = -\frac{1}{2}\sum_{i\sim j} g(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j)\widetilde{\boldsymbol{\mathcal{W}}}_i\widetilde{\boldsymbol{\mathcal{W}}}_j - \sum_i h(\boldsymbol{\mathcal{B}}_i)\widetilde{\boldsymbol{\mathcal{W}}}_i, \tag{8}$$

where $i, j \in [d_1]\times\cdots\times[d_K]$, $g(\cdot,\cdot)$ and $h(\cdot)$ being pre-specified functions with β incorporated, and $i \sim j$ means that the two entries indexed by i and j are "neighbors". For brevity, we often denote $g(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j)$ as g_{ij} and $h(\boldsymbol{\mathcal{B}}_i)$ as h_i for any i, j. We call (7) and (8) our missing propensity model.

One can interpret the unknown parameter $\boldsymbol{\mathcal{B}}$ as a 1-dimensional feature of each tensor entry. Each neighboring pair of entries i and j contribute to the Hamiltonian via their "co-missingness" $\widetilde{\boldsymbol{\mathcal{W}}}_i\widetilde{\boldsymbol{\mathcal{W}}}_j$ and the interaction of their features $\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j$ through $g(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j)$. The function $g(\cdot,\cdot)$ describes the tendency of neighboring entries to be observed or missing together. Every entry i also contributes to the Hamiltonian via $h(\boldsymbol{\mathcal{B}}_i)$, where the function $h(\cdot)$ describes the tendency of each entry to be observed or missing. Here we provide two concrete examples of the model.

Example 2.2 (Bernoulli Model). Suppose that $g(\cdot,\cdot) = 0$, and let $h(x) = 0.5 \cdot \log f(x)/[1 - f(x)]$, where $f(\cdot)$ is an inverse link function (e.g., sigmoid function). The missing propensity model indicates that every $s \in [d_1] \times \cdots \times [d_K]$ is missing independently with:

$$\widetilde{\boldsymbol{\mathcal{W}}}_s = \begin{cases} 1, & p = f(\boldsymbol{\mathcal{B}}_s) \\ -1, & p = 1 - f(\boldsymbol{\mathcal{B}}_s). \end{cases} \tag{9}$$

Example 2.3 (Ising Model). Suppose that $h(\cdot) = x/2$, and let $g(x, y) = xy$. Under this

scenario, the conditional distribution of $\widetilde{\boldsymbol{\mathcal{W}}}_s$, given all other entries in $\widetilde{\boldsymbol{\mathcal{W}}}$ as $\widetilde{\boldsymbol{\mathcal{W}}}_{-s}$, is:

$$p(\widetilde{\boldsymbol{\mathcal{W}}}_s = 1|\boldsymbol{\mathcal{B}}, \widetilde{\boldsymbol{\mathcal{W}}}_{-s}) = \frac{\exp\left[2\boldsymbol{\mathcal{B}}_s \sum_{j\in\mathcal{N}(s)} \widetilde{\boldsymbol{\mathcal{W}}}_j \boldsymbol{\mathcal{B}}_j + \boldsymbol{\mathcal{B}}_s\right]}{1 + \exp\left[2\boldsymbol{\mathcal{B}}_s \sum_{j\in\mathcal{N}(s)} \widetilde{\boldsymbol{\mathcal{W}}}_j \boldsymbol{\mathcal{B}}_j + \boldsymbol{\mathcal{B}}_s\right]} = f(\boldsymbol{\mathcal{B}}_s|\sigma_s), \tag{10}$$

where $\mathcal{N}(s) = \{j \in [d_1] \times \cdots \times [d_K] | s \sim j\}$, and $f(x|\sigma) = [1 + \exp(-x/\sigma)]^{-1}$ is the sigmoid function with scale parameter σ. This model is similar to the Bernoulli model in (9) but has an entry-specific scale parameter $\sigma_s = (2\sum_{j\in\mathcal{N}(s)} \widetilde{\boldsymbol{\mathcal{W}}}_j \boldsymbol{\mathcal{B}}_j + 1)^{-1}$ that depends on the missingness and feature of the neighboring entries.

Our missing propensity model shares several similarities with the previous literature on modeling the missingness of matrix/tensor data. Liang et al. (2016), Schnabel et al. (2016), Wang et al. (2018) model the missing probability via a logistic regression model with mode-specific features, which is similar to our setup in Example 2.2 with $\boldsymbol{\mathcal{B}}$ having low CP rank. (Ma & Chen 2019) models the missing probability via denoising the binary missingness mask with a nuclear-norm penalty Davenport et al. (2014), which is similar to our low-rank setting introduced later. Our model is distinct in the sense that it explicitly models the local dependency of the missingness, as characterized by the neighboring structure and the bivariate function $g(\cdot, \cdot)$.

Given the missing propensity model in (7) and (8), we can compute the $p_k(s^*)$ according to (4) and obtain the conformal weight $\omega_k(s^*)$ as:

$$\omega_k(s^*) = \frac{p_k(s^*)}{\sum_{i=1}^{n+1} p_i(s^*)} = \frac{\exp\left[-2\sum_{s_j\in\mathcal{N}(s_k)} g(\boldsymbol{\mathcal{B}}_{s_k}, \boldsymbol{\mathcal{B}}_{s_j}) \widetilde{\boldsymbol{\mathcal{W}}}_{s_j} - 2h(\boldsymbol{\mathcal{B}}_{s_k})\right]}{\sum_{i=1}^{n+1} \exp\left[-2\sum_{s_j\in\mathcal{N}(s_i)} g(\boldsymbol{\mathcal{B}}_{s_i}, \boldsymbol{\mathcal{B}}_{s_j}) \widetilde{\boldsymbol{\mathcal{W}}}_{s_j} - 2h(\boldsymbol{\mathcal{B}}_{s_i})\right]}, \tag{11}$$

with $s_1, \ldots, s_n \in \mathbb{S}_{cal}$, $s^* = s_{n+1}$, and $\widetilde{\boldsymbol{\mathcal{W}}}_s = 1$ only if $s \in \mathbb{S}_{tr} \cup \mathbb{S}_{cal} \cup \{s^*\}$. The dependency of $\omega_k(s^*)$ on s^* makes it computationally inefficient to scale (11) to all $s^* \in \mathbb{S}_{miss}$ since one has to temporarily set $\widetilde{\boldsymbol{\mathcal{W}}}_{s^*} = 1$ to compute all the weights. To speed up the computation, we approximate the weight in (11) by plugging in $\widetilde{\boldsymbol{\mathcal{W}}}_s = \boldsymbol{\mathcal{W}}_s$ for all s, which removes the dependency of ω_k on s^*. Although this simplification means that one cannot obtain the exact conformal weights, in Appendix A.2, we show that the error caused by this

approximation over the distribution of the non-conformity score in (5) is negligible, and we also show this empirically in Section 4.

With this approximation, the conformal weight ω_k is now proportional to $(1-\tilde{p}_{s_k})/\tilde{p}_{s_k}$, where $\tilde{p}_s = p(\widetilde{\boldsymbol{\mathcal{W}}}_s = 1 | [\widetilde{\boldsymbol{\mathcal{W}}}]_{s'} = [\boldsymbol{\mathcal{W}}]_{s'}, \forall s' \neq s)$ is the full conditional probability of entry s being observed given all other entries. Next, we will discuss the estimation of $\boldsymbol{\mathcal{B}}$ given $\boldsymbol{\mathcal{W}}$.

3 Estimating Algorithm

In this section, we discuss the details of estimating $\boldsymbol{\mathcal{B}}$ based on a single binary tensor sample $\boldsymbol{\mathcal{W}}$ drawn from the missing propensity model specified by (7) and (8). More specifically, we attempt to estimate $\boldsymbol{\mathcal{B}}$ using $\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}$, the binary tensor with $\boldsymbol{\mathcal{W}}_s = 1$ if and only if $s \in \mathbb{S}_{tr}$. We describe the estimation framework in Section 3.1 and the algorithm in Section 3.2.

3.1 Low-rank MPLE Framework

Since we only have access to one sample $\boldsymbol{\mathcal{W}}$ and the tensor parameter $\boldsymbol{\mathcal{B}}$ is of the same dimensionality as $\boldsymbol{\mathcal{W}}$, it is infeasible to obtain an estimator $\widehat{\boldsymbol{\mathcal{B}}}$ without imposing additional constraints over $\boldsymbol{\mathcal{B}}$. Similar to previous literature (Wang & Li 2020, Cai, Li & Xia 2022*a*), we assume that the tensor $\boldsymbol{\mathcal{B}}$ has low tensor rank.

In this paper, we assume that the tensor $\boldsymbol{\mathcal{B}}$ has a low Tensor-Train (TT) rank (Oseledets 2011). A low TT-rank tensor $\boldsymbol{\mathcal{A}}$ can be represented by a series of 3-mode TT factor tensors $\boldsymbol{\mathcal{F}}_k \in \mathbb{R}^{r_{k-1} \times d_k \times r_k}, k = 1, \ldots, K, r_0 = r_K = 1$, where for every entry of $\boldsymbol{\mathcal{A}}$, one has:

$$[\boldsymbol{\mathcal{A}}]_{i_1,\ldots,i_K} = [\boldsymbol{\mathcal{F}}_1]_{:i_1:}[\boldsymbol{\mathcal{F}}_2]_{:i_2:} \cdots [\boldsymbol{\mathcal{F}}_K]_{:i_K:}, \tag{12}$$

with the right-hand side being a series of matrix multiplications. We say $\mathbf{r} = (r_1, \ldots, r_{K-1})$ is the TT-rank of $\boldsymbol{\mathcal{A}}$ and compactly, we write $\boldsymbol{\mathcal{A}} = [\boldsymbol{\mathcal{F}}_1, \ldots, \boldsymbol{\mathcal{F}}_K]$ and $\text{rank}^{\text{tt}}(\boldsymbol{\mathcal{A}}) = \mathbf{r}$. As compared to the more commonly used Tucker rank (Kolda & Bader 2009), the Tensor-Train rank ensures that the number of parameters representing a low-rank tensor scales

linearly with K, the number of modes, making the low TT-rank tensors more efficient for representing high-order tensors.

To ensure the identifiability of TT factors $\boldsymbol{\mathcal{F}}_1, \ldots, \boldsymbol{\mathcal{F}}_K$ in (12), it is often required that $\boldsymbol{\mathcal{F}}_1, \ldots, \boldsymbol{\mathcal{F}}_{K-1}$ being *left-orthogonal*. A 3-mode tensor $\boldsymbol{\mathcal{F}} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$ is left-orthogonal if $\mathbf{L}(\boldsymbol{\mathcal{F}})^\top \mathbf{L}(\boldsymbol{\mathcal{F}}) = \mathbf{I}_{d_3 \times d_3}$, where $\mathbf{L}(\cdot) : \mathbb{R}^{d_1 \times d_2 \times d_3} \mapsto \mathbb{R}^{(d_1 d_2) \times d_3}$ is the so-called left-unfolding operator. Finding the representation (12) of a low TT-rank tensor under the left orthogonality constraint can be achieved by the TT-SVD algorithm (Oseledets 2011). For completeness, we restate the TT-SVD algorithm in Algorithm 1 and denote it as $\mathrm{SVD}^{\mathrm{tt}}_{\mathbf{r}}(\cdot)$.

Algorithm 1 Tensor-Train Singular Value Decomposition (TT-SVD)

Input: Tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, tensor-train rank $\mathbf{r} = (r_1, \ldots, r_{K-1})$.
  $\boldsymbol{\mathcal{A}} \leftarrow \boldsymbol{\mathcal{X}}$, $r_0, r_K \leftarrow 1$.
  for $k = 1, \ldots, K-1$ **do**
    $\mathbf{A} \leftarrow \mathrm{reshape}[\boldsymbol{\mathcal{A}}, (r_{k-1} d_k, d_{k+1} \cdots d_K)]$. % reshape$(\cdot,\cdot)$ from MATLAB
    Conduct SVD on $\mathbf{A}$ and truncate at rank r_k: $\mathbf{A} \approx \mathbf{U}\mathbf{S}\mathbf{V}^\top$.
    $\boldsymbol{\mathcal{F}}_k \leftarrow \mathrm{reshape}[\mathbf{U}, (r_{k-1}, d_k, r_k)]$.
    $\boldsymbol{\mathcal{A}} \leftarrow \mathbf{S}\mathbf{V}^\top$.
  end for
  $\boldsymbol{\mathcal{F}}_K \leftarrow \mathrm{reshape}(\boldsymbol{\mathcal{A}}, (r_{K-1}, d_K, 1))$.
Output: Tensor-Train representation $\widehat{\boldsymbol{\mathcal{X}}} = [\boldsymbol{\mathcal{F}}_1, \ldots, \boldsymbol{\mathcal{F}}_K]$ with $\mathrm{rank}^{\mathrm{tt}}(\widehat{\boldsymbol{\mathcal{X}}}) \le \mathbf{r}$.

Given the assumption that the tensor $\boldsymbol{\mathcal{B}}$ has low TT-rank $\mathbf{r} = (r_1, \ldots, r_{K-1})$, we can re-formulate the MLE of $\boldsymbol{\mathcal{B}}$ as the solution of a low-rank tensor learning problem:

$$\widehat{\boldsymbol{\mathcal{B}}} = \underset{\boldsymbol{\mathcal{B}}:\mathrm{rank}^{\mathrm{tt}}(\boldsymbol{\mathcal{B}}) \le \mathbf{r}}{\arg\min} \; - \log p(\widetilde{\boldsymbol{\mathcal{W}}} = \boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}} | \boldsymbol{\mathcal{B}}), \tag{13}$$

where $\mathrm{rank}^{\mathrm{tt}}(\boldsymbol{\mathcal{B}}) = (r'_1, \ldots, r'_{K-1}) \le \mathbf{r}$ means that $r'_k \le r_k$ for any $k = 1, \ldots, K-1$.

However, the likelihood in (13) is incorrect since we did not account for the random splitting of the training set and the calibration set, and it is also difficult to evaluate the normalizing constant of the likelihood. To tackle these issues, we consider estimating $\boldsymbol{\mathcal{B}}$ by the maximum pseudo-likelihood estimator (MPLE), which is a common approach for the estimation and inference of the Ising model (Ravikumar et al. 2010, Barber & Drton 2015,

Bhattacharya & Mukherjee 2018). Formally, for each entry i, define $\tilde{p}_i(\boldsymbol{\mathcal{B}})$ as:

$$\begin{aligned}\tilde{p}_i(\boldsymbol{\mathcal{B}}) &= p\left(\widetilde{\boldsymbol{\mathcal{W}}}_i = 1 | [\widetilde{\boldsymbol{\mathcal{W}}}]_s = [\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_s, \forall s \neq i, \boldsymbol{\mathcal{B}}\right) \\ &= \frac{\exp\left[2\sum_{j\in\mathcal{N}(i)} g(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j)[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_j + 2h(\boldsymbol{\mathcal{B}}_i)\right]}{1 + \exp\left[2\sum_{j\in\mathcal{N}(i)} g(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j)[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_j + 2h(\boldsymbol{\mathcal{B}}_i)\right]}. \end{aligned} \tag{14}$$

and we often write it directly as $\tilde{p}_i$. The low-rank MPLE of $\boldsymbol{\mathcal{B}}$ can now be written as:

$$\widehat{\boldsymbol{\mathcal{B}}} = \underset{\boldsymbol{\mathcal{B}}:\mathrm{rank}^{\mathrm{tt}}(\boldsymbol{\mathcal{B}})\leq \mathbf{r}}{\arg\min}\ \ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}}) = -\sum_{i:[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i=1} \log q\tilde{p}_i - \sum_{i:[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i=-1} \log\left(1 - q\tilde{p}_i\right), \tag{15}$$

where $q \in (0, 1)$ is the probability of selecting an observed entry into the training set. We discuss the optimization algorithm for solving (15) next.

3.2 Riemannian Gradient Descent (RGrad) Algorithm

To solve for (15), a natural idea is to directly estimate the tensor-train factors $\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_K$ for $\widehat{\boldsymbol{\mathcal{B}}}$ one at a time, while keeping the others fixed, and iterate until convergence. Such an alternating minimization algorithm has been applied to low-rank binary tensor decomposition (Wang & Li 2020, Lee & Wang 2020). However, alternating minimization is computationally inefficient here as each step requires fitting a generalized linear model (GLM) with high-dimensional covariates. Another candidate approach for estimating $\widehat{\boldsymbol{\mathcal{B}}}$ is the projected gradient descent (Chen, Raskutti & Yuan 2019), where in each iteration one updates $\boldsymbol{\mathcal{B}}$ along the gradient direction first and then projects it back to the low-rank tensor space with TT-SVD. This is also undesirable since the projection for a high-rank tensor can be very slow.

In this paper, we propose an optimization technique called Riemannian gradient descent (RGrad), motivated by the fact that rank-$\mathbf{r}$ tensor-train tensors lie on a smooth manifold (Holtz et al. 2012), which we denote as $\mathbb{M}_{\mathbf{r}}$. As compared to the aforementioned methods, RGrad is faster because each step updates $\boldsymbol{\mathcal{B}}$ with a gradient along the tangent space of $\boldsymbol{\mathcal{B}}$, avoiding fitting multiple high-dimensional GLMs. Also, the projection from the

tangent space back to the manifold $\mathbb{M}_{\mathbf{r}}$ is faster than the projected gradient descent since the tensors in the tangent space are also low-rank. RGrad has been extensively applied to tensor completion (Kressner et al. 2014, Steinlechner 2016, Cai, Li & Xia 2022*b*), generalized tensor learning (Cai, Li & Xia 2022*a*) and tensor regression (Luo & Zhang 2022). The current work, to the best of our knowledge, is the first to apply RGrad to the low TT-rank binary tensor decomposition. We break down the procedures of RGrad into three steps.

Step I: Compute Vanilla Gradient. We first compute the vanilla gradient $\nabla \ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}})$ at the current iterative value $\boldsymbol{\mathcal{B}}$. Formally, the vanilla gradient tensor $\boldsymbol{\mathcal{G}}$ satisfies:

$$[\boldsymbol{\mathcal{G}}]_i = 2 \sum_{j \in \mathcal{N}(i)} (\boldsymbol{\mathcal{V}}_i [\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_j + \boldsymbol{\mathcal{V}}_j [\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i)\, g_x(\boldsymbol{\mathcal{B}}_i, \boldsymbol{\mathcal{B}}_j) + 2h'(\boldsymbol{\mathcal{B}}_i)\boldsymbol{\mathcal{V}}_i, \tag{16}$$

where $g_x(\cdot,\cdot) = \partial g(\cdot,\cdot)/\partial x$ and $\boldsymbol{\mathcal{V}}_i = (1-\tilde{p}_i)(1-q\tilde{p}_i)^{-1}(q\tilde{p}_i - \mathbb{1}_{\{[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i=1\}})$, with $\tilde{p}_i$ defined in (14). Typically, we require the neighboring structure $\mathcal{N}(i)$ to be symmetric across all entries i and require function $g(x,y)$ to be a bivariate polynomial of the form $\sum_{\alpha,\beta} c_{\alpha,\beta} x^\alpha y^\beta$, which would then allow one to compute (16) much faster with convolutions.

Step II: Tangent Space Projected Gradient Descent. Suppose that the current iterative value $\boldsymbol{\mathcal{B}}$ has a tensor-train representation $\boldsymbol{\mathcal{B}} = [\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_K]$. Then any tensor $\boldsymbol{\mathcal{A}}$ within the tangent space $\mathbb{T}$ at $\boldsymbol{\mathcal{B}}$ has an explicit form:

$$\boldsymbol{\mathcal{A}} = \sum_{k=1}^{K} \boldsymbol{\mathcal{C}}_k, \quad \boldsymbol{\mathcal{C}}_k = [\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_{k-1}, \boldsymbol{\mathcal{Y}}_k, \boldsymbol{\mathcal{T}}_{k+1}, \ldots, \boldsymbol{\mathcal{T}}_K], \tag{17}$$

with the constraint that $\mathbf{L}(\boldsymbol{\mathcal{Y}}_k)^\top \mathbf{L}(\boldsymbol{\mathcal{T}}_k) = \mathbf{O}_{r_k \times r_k}$ for all $k < K$, where $\mathbf{O}$ is a zero matrix, and $\boldsymbol{\mathcal{C}}_k$ has the property that $\langle \boldsymbol{\mathcal{C}}_i, \boldsymbol{\mathcal{C}}_j \rangle = 0$ for all $i \neq j$. In this step, one projects the vanilla gradient $\boldsymbol{\mathcal{G}}$ from step I onto $\mathbb{T}$ and obtains the projected gradient $\mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{G}})$. Thanks to the orthogonality of different $\boldsymbol{\mathcal{C}}_k$, the projection problem can be solved via:

$$\min_{\boldsymbol{\mathcal{Y}}_k : \mathbf{L}(\boldsymbol{\mathcal{Y}}_k)^\top \mathbf{L}(\boldsymbol{\mathcal{T}}_k) = \mathbf{O}_{r_k \times r_k}} \frac{1}{2} \|\boldsymbol{\mathcal{G}} - \boldsymbol{\mathcal{C}}_k\|_{\mathrm{F}}^2, \quad \text{s.t. } \boldsymbol{\mathcal{C}}_k = [\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_{k-1}, \boldsymbol{\mathcal{Y}}_k, \boldsymbol{\mathcal{T}}_{k+1}, \ldots, \boldsymbol{\mathcal{T}}_K], \tag{18}$$

for any $k \leq K-1$ and $\boldsymbol{\mathcal{Y}}_k$ is unconstrained if $k = K$. Solution to (18) is:

$$\mathbf{L}(\widehat{\boldsymbol{\mathcal{Y}}}_k) = \left[\mathbf{I}_{r_{k-1}d_k} - \mathbf{L}(\boldsymbol{\mathcal{T}}_k)\mathbf{L}(\boldsymbol{\mathcal{T}}_k)^\top\right] \left(\boldsymbol{\mathcal{B}}^{\leq k-1} \otimes \mathbf{I}_{d_k}\right)^\top \boldsymbol{\mathcal{G}}^{<k>} \left(\boldsymbol{\mathcal{B}}^{\geq k+1}\right)^\top \left[\boldsymbol{\mathcal{B}}^{\geq k+1} \left(\boldsymbol{\mathcal{B}}^{\geq k+1}\right)^\top\right]^{-1}, \tag{19}$$

for $k \leq K-1$ and:

$$\mathbf{L}(\widehat{\boldsymbol{\mathcal{Y}}}_K) = \left(\boldsymbol{\mathcal{B}}^{\leq K-1} \otimes \mathbf{I}_{d_K}\right)^\top \boldsymbol{\mathcal{G}}^{<K>}, \tag{20}$$

where $\otimes$ is the matrix Kronecker product. In (19) and (20), $\boldsymbol{\mathcal{G}}^{<k>}$ is the k-mode separation of tensor $\boldsymbol{\mathcal{G}}$, which basically reshapes $\boldsymbol{\mathcal{G}}$ to a matrix of size $\left(\prod_{l \leq k} d_l\right) \times \left(\prod_{l>k} d_l\right)$. Any tensor $\boldsymbol{\mathcal{B}}$ has its k-mode separation as $\boldsymbol{\mathcal{B}}^{<k>} = \boldsymbol{\mathcal{B}}^{\leq k}\boldsymbol{\mathcal{B}}^{\geq k+1}$, where $\boldsymbol{\mathcal{B}}^{\leq k}, \boldsymbol{\mathcal{B}}^{\geq k+1}$ are called the k-th left part and $(k+1)$-th right part. Given that $\boldsymbol{\mathcal{B}} = [\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_K]$, one can recursively compute $\boldsymbol{\mathcal{B}}^{\leq k}$ as $(\boldsymbol{\mathcal{B}}^{\leq k-1} \otimes \mathbf{I}_{d_k})\mathbf{L}(\boldsymbol{\mathcal{T}}_k)$ and $\boldsymbol{\mathcal{B}}^{\geq k+1}$ as $\mathbf{R}(\boldsymbol{\mathcal{T}}_{k+1})(\mathbf{I}_{d_{k+1}} \otimes \boldsymbol{\mathcal{B}}^{\geq k+2})$ following the convention that $\boldsymbol{\mathcal{B}}^{\leq 0} = \boldsymbol{\mathcal{B}}^{\geq K+1} = 1$, where $\mathbf{R}(\cdot) : \mathbb{R}^{d_1 \times d_2 \times d_3} \mapsto \mathbb{R}^{d_1 \times d_2 d_3}$ is the right-unfolding operator.

After computing $\widehat{\boldsymbol{\mathcal{Y}}}_k$ with (19) and (20), one ends up with $\widehat{\boldsymbol{\mathcal{C}}}_k = [\boldsymbol{\mathcal{T}}_1, \ldots, \widehat{\boldsymbol{\mathcal{Y}}}_k, \ldots, \boldsymbol{\mathcal{T}}_K]$ and thus the projected gradient $\mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{G}}) = \sum_k \widehat{\boldsymbol{\mathcal{C}}}_k$. With this projected gradient, one updates $\boldsymbol{\mathcal{B}}$ via $\widetilde{\boldsymbol{\mathcal{B}}} \leftarrow \boldsymbol{\mathcal{B}} - \eta\mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{G}})$, where η is a constant step size.

Step III: Retraction. As a property of low TT-rank tensors, the updated tensor $\widetilde{\boldsymbol{\mathcal{B}}}$ has its TT-rank upper bounded by $2\mathbf{r}$. To enforce the rank constraint, the last step of RGrad is to retract $\widetilde{\boldsymbol{\mathcal{B}}}$ back to the manifold $\mathbb{M}_{\mathbf{r}}$. We do so by applying TT-SVD to $\widetilde{\boldsymbol{\mathcal{B}}}$: $\boldsymbol{\mathcal{B}}' = \mathrm{SVD}^{\mathrm{tt}}_{\mathbf{r}}(\widetilde{\boldsymbol{\mathcal{B}}})$, and $\boldsymbol{\mathcal{B}}'$ will be the value used for the next iteration.

We summarize the RGrad algorithm in Algorithm 2. To provide an initial estimator of $\boldsymbol{\mathcal{B}}$, we apply TT-SVD to a randomly perturbed version of the binary tensor $\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}$, which works quite well empirically. We typically set $\eta = 0.1$ and denote the output of Algorithm 2 as $\mathrm{RGrad}(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}, \mathbf{r})$. The algorithm terminates when $\|\boldsymbol{\mathcal{B}}' - \boldsymbol{\mathcal{B}}\|_{\mathrm{F}}$ falls below a prespecified threshold. By assuming $d_k = O(d), r_k = O(r), \forall k$ and $\max_s |\mathcal{N}(s)| = O(K)$, the computational complexity of RGrad is $O(K(d^K r^2 + dr^3))$ per iteration. See Steinlechner (2016) for more details on the computational complexity of RGrad. In Appendix C.6, we verify the numerical convergence of Algorithm 2, making runtime comparisons with competing meth-

Algorithm 2 MPLE of Low-rank Ising Model with Riemannian Gradient Descent

Input: Binary tensor $\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}$, tensor-train rank $\mathbf{r} = (r_1, \ldots, r_{K-1})$, step size η, train-calibration split probability q.

Initialize: let $\boldsymbol{\mathcal{E}} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$ and $\widehat{\boldsymbol{\mathcal{B}}} \leftarrow \mathrm{SVD}^{\mathrm{tt}}_{\mathbf{r}}(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}} + \boldsymbol{\mathcal{E}}) = [\widehat{\boldsymbol{\mathcal{T}}}_1, \ldots, \widehat{\boldsymbol{\mathcal{T}}}_K]$ by Algorithm 1.

for $l = 1, \ldots, l_{\max}$ **do**

  Compute the vanilla gradient $\boldsymbol{\mathcal{G}}$ using (16).

  for $k = 1, \ldots, K$ **do**

    Compute $\widehat{\boldsymbol{\mathcal{Y}}}_k$ following (19) if $k < K$ and (20) if $k = K$.

    $\widehat{\boldsymbol{\mathcal{C}}}_k \leftarrow [\widehat{\boldsymbol{\mathcal{T}}}_1, \ldots, \widehat{\boldsymbol{\mathcal{T}}}_{k-1}, \widehat{\boldsymbol{\mathcal{Y}}}_k, \widehat{\boldsymbol{\mathcal{T}}}_{k+1}, \ldots, \widehat{\boldsymbol{\mathcal{T}}}_K]$.

  end for

  $\mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{G}}) \leftarrow \sum_{k=1}^{K} \widehat{\boldsymbol{\mathcal{C}}}_k$.

  $\widetilde{\boldsymbol{\mathcal{B}}} \leftarrow \widehat{\boldsymbol{\mathcal{B}}} - \eta \mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{G}})$.

  $\widehat{\boldsymbol{\mathcal{B}}} \leftarrow \mathrm{SVD}^{\mathrm{tt}}_{\mathbf{r}}(\widetilde{\boldsymbol{\mathcal{B}}}) = [\widehat{\boldsymbol{\mathcal{T}}}_1, \ldots, \widehat{\boldsymbol{\mathcal{T}}}_K]$ by Algorithm 1.

end for

Output: Maximum Pseudo-Likelihood Estimator (MPLE) $\widehat{\boldsymbol{\mathcal{B}}}$ with $\mathrm{rank}^{\mathrm{tt}}(\widehat{\boldsymbol{\mathcal{B}}}) \leq \mathbf{r}$.

ods and exploring an adaptive stepsize scheme. We summarize the conformalized tensor completion (CTC) algorithm in Algorithm 3.

Remark 3.1 (Fast Entry-wise Quantile Computation). In the last step of Algorithm 3, we compute the empirical $(1-\alpha)$-quantile of the weighted eCDF of the non-conformity score of all calibration data. The for-loop looks slow, as one needs to evaluate the quantile for each testing entry s^* separately. However, $\widehat{q}_{s^*}$ can be computed faster via:

$$\widehat{q}_{s^*} = \begin{cases} +\infty, & \text{if } \omega_{s^*} \geq \alpha \\ \mathcal{Q}_{\frac{1-\alpha}{1-\omega_{s^*}}} \left(\sum_{s \in \mathbb{S}_{cal}} \frac{\omega_s}{1-\omega_{s^*}} \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s)} \right), & \text{if } \omega_{s^*} < \alpha. \end{cases}$$

which only requires one to evaluate the quantile of a fixed weighted eCDF shared by all testing entries, and can be computed efficiently with sorting-based approaches or quantile sketching (Greenwald & Khanna 2001) if the calibration set is large in size.

Remark 3.2 (Rank Selection). The implementation of the CTC algorithm requires a proper choice of the tensor-train rank $\mathbf{r}$ for the low-rank Ising model. Typically, in low-rank tensor learning literature (Wang & Li 2020, Cai, Li & Xia 2022*a*), either the Akaike Information Criterion (AIC) (Akaike 1973) or the Bayesian Information Criterion (BIC) (Schwarz 1978) is used for the rank selection. Unfortunately, they are not applicable here since we can only

Algorithm 3 Conformalized Tensor Completion (CTC)

Input: Data tensor $\boldsymbol{\mathcal{X}}$, tensor-train rank $\mathbf{r}$, train-calibration split probability $q \in (0,1)$, target mis-coverage $\alpha \in (0,1)$, arbitrary tensor completion algorithm $\boldsymbol{\mathcal{A}}$.

$\mathbb{S} \leftarrow \{s \in [d_1] \times \cdots \times [d_K] | \boldsymbol{\mathcal{X}}_s \neq \text{NaN}\}$. % indices of entries that are observed

$\boldsymbol{\mathcal{W}} \leftarrow 2 \times \mathbb{1}_{\{s \in \mathbb{S}\}} - 1$.

Randomly partition $\mathbb{S}$ independently into $\mathbb{S}_{tr} \cup \mathbb{S}_{cal}$ with probability q and $1-q$.

$\widehat{\boldsymbol{\mathcal{X}}} \leftarrow \boldsymbol{\mathcal{A}}(\boldsymbol{\mathcal{X}}_{\mathbb{S}_{tr}})$. % $[\boldsymbol{\mathcal{X}}_{\mathbb{S}_{tr}}] = \boldsymbol{\mathcal{X}}_s$ if $s \in \mathbb{S}_{tr}$ and NaN otherwise

$\widehat{\boldsymbol{\mathcal{B}}} \leftarrow \text{RGrad}(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}, \mathbf{r})$. % RGrad$(\cdot,\cdot)$ is Algorithm 2

for $s \in \mathbb{S}_{cal} \cup \mathbb{S}^c$ **do**

$\tilde{p}_s \leftarrow \left\{1 + \exp\left[-2\sum_{j \in \mathcal{N}(s)} [\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_j g(\widehat{\boldsymbol{\mathcal{B}}}_s, \widehat{\boldsymbol{\mathcal{B}}}_j) - 2h(\widehat{\boldsymbol{\mathcal{B}}}_s)\right]\right\}^{-1}$.

$\omega_s \leftarrow (1-\tilde{p}_s)\tilde{p}_s^{-1}$.

end for

for $s^* \in \mathbb{S}^c$ **do** % See Remark 3.1

Re-normalize $\omega_s, s \in \mathbb{S}_{cal}$ and ω_{s^*} s.t. $\sum_{s \in \mathbb{S}_{cal}} \omega_s + \omega_{s^*} = 1$.

$\widehat{q}_{s^*} \leftarrow \mathcal{Q}_{1-\alpha}\left(\sum_{s \in \mathbb{S}_{cal}} \omega_s \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s)} + \omega_{s^*} \cdot \delta_{+\infty}\right)$.

end for

Output: $(1-\alpha)$-level conformal interval $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) \leftarrow \{x \in \mathbb{R} | \mathcal{S}(x, \widehat{\boldsymbol{\mathcal{X}}}_{s^*}) \leq \widehat{q}_{s^*}\}, \forall s^* \in \mathbb{S}^c$.

compute the pseudo-likelihood. According to previous literature on the model selection of Markov Random Fields (Ji & Seymour 1996, Csiszár & Talata 2006, Matsuda et al. 2021), one can replace the likelihood in AIC/BIC with pseudo-likelihood and obtain the Pseudo-AIC (P-AIC) and Pseudo-BIC (P-BIC), which are still consistent under some regularity conditions. The P-AIC and P-BIC are defined as:

$$\text{P-AIC}(r') = 2\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}} | \widehat{\boldsymbol{\mathcal{B}}}) + 2\left\{\sum_{k=1}^{K-1}\left[d_k r'_{k-1} r'_k - (r'_k)^2\right] + d_K r'_{K-1}\right\}. \tag{21}$$

$$\text{P-BIC}(r') = 2\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}} | \widehat{\boldsymbol{\mathcal{B}}}) + \left\{\sum_{k=1}^{K-1}\left[d_k r'_{k-1} r'_k - (r'_k)^2\right] + d_K r'_{K-1}\right\} \log\left(\prod_{k=1}^{K} d_k\right). \tag{22}$$

Among all candidate ranks, we select the rank with the smallest P-AIC or P-BIC. In Section C.2 of the supplemental material, we provide empirical evidence on the consistency of P-AIC and the inconsistency of P-BIC.

Remark 3.3 (Estimation and Coverage Error Bound). In Section A.3 of the supplemental material, we derive theoretically the non-asymptotic bound for $\|\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\|_{\text{F}}$ under the special case where $g(x,y) = 0$ (i.e. the Bernoulli model) and with the same assumption we further

derive the coverage probability lower bound of the CTC algorithm in Section A.4. It is a remarkable result that the estimating error, as well as the shortfall of the coverage from the target coverage, increases with $(r^*\bar{d}/d^*)^{1/2}$, where $r^*, \bar{d}, d^*$ are $\prod_k r_k, \sum_k d_k, \prod_k d_k$ for $\boldsymbol{\mathcal{B}}^*$, respectively. If one assumes that $d_k = O(d), r_k = O(r), \forall k$, then the estimation error and coverage shortfall scales with $(r/d)^{(K-1)/2}$. Higher r/d indicates that the data missing pattern is more complex and thus the uncertainty quantification is harder. However, we do not have the results when $g(x, y) \neq 0$ given theoretical challenges, we show empirically in Section 4 that this tendency also holds for the Ising model.

4 Simulation Experiments

In this section, we validate the effectiveness of the proposed conformalized tensor completion algorithm via numerical simulations. We consider an order-3 cubical tensor of size $d \times d \times d$ and summarize our simulation settings below. Additional details about the simulation setups and results are included in Section C of the supplemental material. Our code is available on GitHub.

4.1 Simulation Setup

We simulate the $d \times d \times d$ true tensor parameter $\boldsymbol{\mathcal{B}}^*$ via the Gaussian tensor block model (TBM) (Wang & Zeng 2019), where $\boldsymbol{\mathcal{B}}^* = \boldsymbol{\mathcal{C}} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 + \boldsymbol{\mathcal{E}}_1$ with $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{r \times r \times r}$ being a core tensor with i.i.d. entries from a Gaussian mixture model: $0.5 \cdot \mathcal{N}(1, 0.5) + 0.5 \cdot \mathcal{N}(-1, 0.5)$, and $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3 \in \{0, 1\}^{d \times r}$ with only a single 1 in each row and the noise tensor $\boldsymbol{\mathcal{E}}_1 \overset{i.i.d.}{\sim} \mathcal{N}(0, 0.2)$. We choose this model for generating $\boldsymbol{\mathcal{B}}^*$ to ensure that $\boldsymbol{\mathcal{B}}^*$ has a checkerboard structure, as shown in Figure 1, and note that the noiseless part of $\boldsymbol{\mathcal{B}}^*$ is also of low tensor-train rank. We re-scale the simulated $\boldsymbol{\mathcal{B}}^*$ such that $\|\boldsymbol{\mathcal{B}}^*\|_\infty = 2$. We enforce each column of $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3$ to have 1s in consecutive rows so that the simulated $\boldsymbol{\mathcal{B}}^*$ demonstrates a noisy "checker box" structure, as illustrated in Figure 1(a).

Given the simulated $\boldsymbol{\mathcal{B}}^*$, we then simulate the binary data missingness tensor $\boldsymbol{\mathcal{W}}$ from the Ising model. Throughout this section, we suppose that two tensor entries i and j are neighbors, i.e., $i \sim j$, if and only if their indices differ by 1 in just one mode. Consequently, for 3-way tensors, each non-boundary entry has six neighbors. We simulate $\boldsymbol{\mathcal{W}}$ from the missing propensity model specified by (7) and (8) with a block-Gibbs sampler and generate samples from a Monte Carlo Markov Chain (MCMC). The MCMC has 4×10^4 iterations with the first 10^4 samples burnt in, and we take one sample every other 10^3 iterations to end up with $n = 30$ samples. In Figure 1(b), we visualize one simulated $\boldsymbol{\mathcal{W}}$.

Lastly, the data tensor $\boldsymbol{\mathcal{X}}$ is generated from an additive noise model: $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}^* + \boldsymbol{\mathcal{E}}$, which is similar to $\boldsymbol{\mathcal{B}}^*$, with $\boldsymbol{\mathcal{X}}^*$ having a Tucker rank $(3, 3, 3)$. The noiseless tensor $\boldsymbol{\mathcal{X}}^*$ also possesses a "checker box" structure and is contaminated by the noise tensor $\boldsymbol{\mathcal{E}}$, whose distribution depends on the specific simulation setting described later. We re-scale $\boldsymbol{\mathcal{X}}^*$ to have $\|\boldsymbol{\mathcal{X}}^*\|_\infty = 2$ and define the signal-to-noise ratio (SNR) of $\boldsymbol{\mathcal{X}}$ as $\|\boldsymbol{\mathcal{X}}^*\|_\infty / \|\boldsymbol{\mathcal{E}}\|_\infty$ and re-scale $\boldsymbol{\mathcal{E}}$ such that SNR$= 2$. The data tensor $\boldsymbol{\mathcal{X}}$ is then masked by $\boldsymbol{\mathcal{W}}$, as plotted in Figure 1(c).

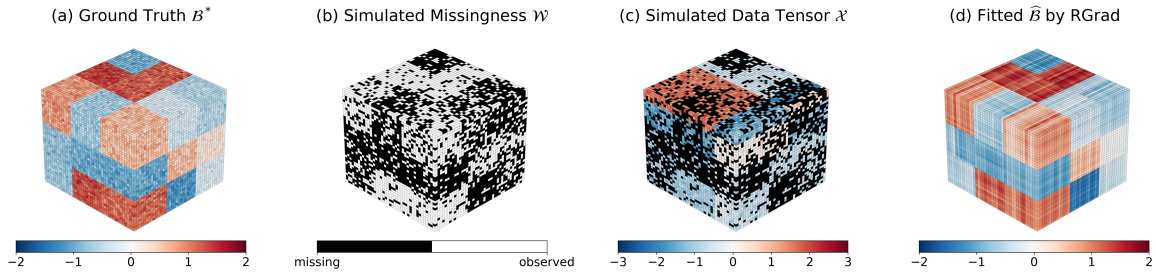

Figure 1: Visualizations of key tensors in the simulation setup. (a) Ising model parameter tensor $\boldsymbol{\mathcal{B}}^*$ with $d = 40, r = 3$. (b) Simulated binary tensor $\boldsymbol{\mathcal{W}}$ with $g(x, y) = xy/15, h(x) = x/2$. (c) Simulated data tensor $\boldsymbol{\mathcal{X}}$ masked by $\boldsymbol{\mathcal{W}}$ with $r_0 = 3, \text{SNR} = 2.0$ and $\boldsymbol{\mathcal{E}}$ having i.i.d. $\mathcal{N}(0, 1)$ entries. (d) Estimated parameter $\widehat{\boldsymbol{\mathcal{B}}}$ from RGrad based on a 70% training set.

4.2 Conformal Prediction Validation

To validate the efficacy of the proposed conformalized tensor completion (CTC) algorithm, we consider the simulation setting with $d \in \{40, 60, 80, 100\}$, $r \in \{3, 5, 7, 9\}$, $g(x, y) \in$

$\{0, xy/15\}$. The noise tensor $\boldsymbol{\mathcal{E}}$ is simulated based on two different uncertainty regimes: 1) constant noise: $[\boldsymbol{\mathcal{E}}]_s \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$; 2) adversarial noise: $[\boldsymbol{\mathcal{E}}]_s$ follows independent Gaussian distribution $\mathcal{N}(0, \sigma_s^2)$, with $\sigma_s = [2\exp(\boldsymbol{\mathcal{B}}_s^*)/[1+\exp(\boldsymbol{\mathcal{B}}_s^*)]^{-1}$. The adversarial noise simulates cases where the missing entries have higher uncertainty than the observed entries.

For each simulation scenario, we apply the correctly specified CTC algorithm with P-AIC selected rank and call it **RGrad**. As a benchmark, we also consider two other versions of conformal inference: 1) **unweighted**: the unweighted conformal prediction; 2) **oracle**: the weighted conformal prediction with the true tensor parameter $\boldsymbol{\mathcal{B}}^*$. We conduct a simulation over $n = 30$ repetitions, and for each repetition, we randomly split the observed entries into a training and a calibration set with $q = 0.7$ and evaluate the constructed conformal intervals on the missing entries, denoted as $\mathbb{S}_{miss}$. For the tensor completion algorithm, we choose low Tucker rank tensor completion coupled with Riemannian gradient descent (Wang, Chen & Wei 2023). We use the absolute residual $\mathcal{S}(y, \widehat{y}) = |y - \widehat{y}|$ as the non-conformity score. To evaluate the conformal intervals, we define the average mis-coverage metric as:

$$\text{Average Mis-coverage } \% = \frac{100}{|\mathbb{Q}|} \sum_{\tau \in \mathbb{Q}} \left| \tau - \frac{1}{|\mathbb{S}_{miss}|} \sum_{s \in \mathbb{S}_{miss}} \mathbb{1}_{\{\boldsymbol{\mathcal{X}}_s \in \widehat{C}_{\tau,s}(\widehat{\boldsymbol{\mathcal{X}}})\}} \right|, \tag{23}$$

with $\mathbb{Q} = \{0.80, 0.81, \ldots, 0.98, 0.99\}$. We plot the average mis-coverage with $r = 3$ in Figure 2. We also plot the results with $r = 9$ in Section C.3 of the supplemental material.

According to the results, we find that with constant entry-wise uncertainty, even the unweighted conformal intervals perform decently, but still have more mis-coverage than the oracle case. Using our CTC algorithm significantly shrinks the mis-coverage and matches the performance of the oracle case. Under the adversarial noise regime, we observed significant mis-coverage ($> 10\%$) of the unweighted conformal prediction, and using the CTC algorithm provides conformal intervals with $< 1\%$ of mis-coverage, indicating that our method helps in constructing well-calibrated confidence intervals.

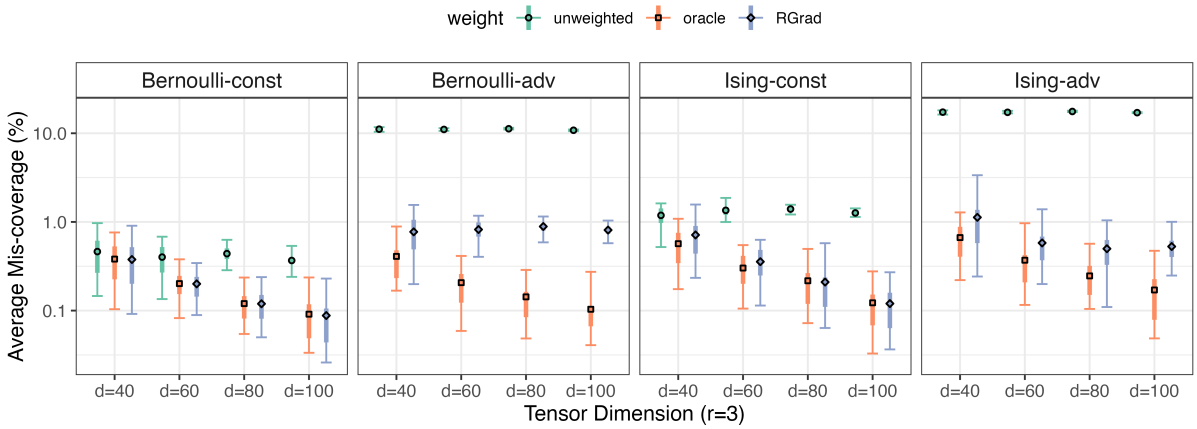

Figure 2: The average mis-coverage of three conformal prediction methods with $d \in \{40, 60, 80, 100\}$, $r = 3$ under the Bernoulli and Ising model. Two uncertainty regimes: constant noise (const) and adversarial noise (adv) are considered. Results are based on 30 repetitions, error bars show the 2.5%, 97.5% quantiles, and the thicker lines show the range of 25% to 75% quantiles. The y-axis is plotted in log10-scale.

The mis-coverage is even worse for the unweighted conformal prediction when missingness is locally dependent based on the Ising model, and the CTC algorithm still provides conformal intervals at the target coverage. In Figure S.2 of Section C.3 of the supplemental material, we further show that the mis-coverage of the unweighted conformal prediction is mainly under-coverage, as it cannot account for the increase of uncertainty in the testing set under adversarial noise.

To provide a full landscape on how the conformal intervals based on our CTC algorithm perform under different tensor rank r and tensor dimension d of the underlying parameter $\boldsymbol{\mathcal{B}}^*$, we visualize in Figure 3 the empirical coverage of 90% and 95% conformal intervals under different missingness and uncertainty regimes by r/d, i.e. the rank-over-dimension of the tensor $\boldsymbol{\mathcal{B}}^*$, based on our RGrad method. Generally speaking, the higher r/d is, the more difficult it is to estimate the missing propensity of the tensor data and thus the worse the coverage of the conformal intervals, which echoes our theoretical result in Section A.4 of the supplemental material. Therefore, we conclude that our proposed method would provide well-calibrated conformal intervals when the underlying missingness model has a low tensor rank relative to the tensor size (i.e., $r << d$).

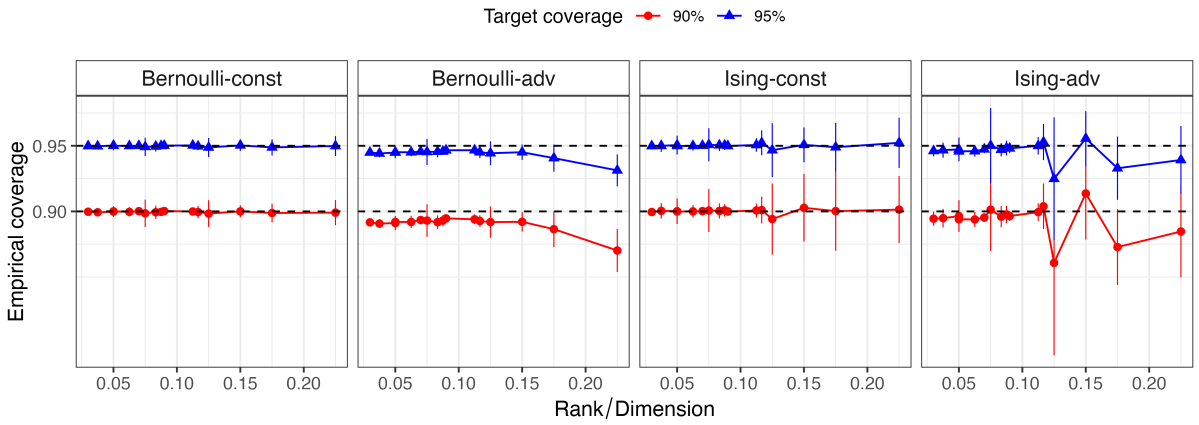

Figure 3: RGrad empirical coverage of the 90% and 95% conformal intervals under the Bernoulli and Ising model with two noise regimes. x-axis is the r/d of the tensor parameter $\boldsymbol{\mathcal{B}}^*$. Results are based on $n = 30$ repetitions, and error bars are ± 1.96 standard deviations.

In Section C.3 of the supplemental material, we also compare our RGrad approach with other binary tensor decomposition approaches, such as CP and Tucker decomposition, for estimating the missing propensity and conducting conformal prediction. We find our method performs consistently well under all kinds of dependency and uncertainty regimes. In Section C.5 of the supplemental material, we further explore other choices of the non-conformity score, including two-sided and normalized scores, and verify the performance of conformal prediction under these settings.

5 Data Application to TEC Reconstruction

Our proposed method can account for the locally dependent data missingness, which is a common data missing pattern for spatial data; therefore, we apply our method to a spatio-temporal tensor completion problem in this section as an application. Specifically, we consider the total electron content (TEC) reconstruction problem over the territory of the USA and Canada. The TEC data have severe missing data problems since they can be measured only if the corresponding spatial location has a ground-based receiver. An accurate prediction of the TEC can foretell the impact of space weather on the positioning,

navigation, and timing (PNT) service (Wang et al. 2021, Younas et al. 2022). Existing literature (Pan et al. 2021, Sun et al. 2022, Wang, Zou, Sun & Chen 2023) focuses on imputation and prediction of the global and regional TEC and lacks data-driven approaches for quantifying the uncertainty of the imputation, and we aim at filling in this gap.

In Figure 4(a), we plot the TEC distribution over the USA and Canada from the VISTA TEC database (Sun, Chen, Zou, Ren, Chang, Wang & Coster 2023). The VISTA TEC is a pre-imputed version of the Madrigal TEC (MIT Haystack Observatory 2012), which has $> 80\%$ of the data missing globally. We use the VISTA TEC as the ground truth and the Madrigal TEC data missingness to mask out entries in the VISTA TEC to simulate data missingness close to what scientists observe in practice.

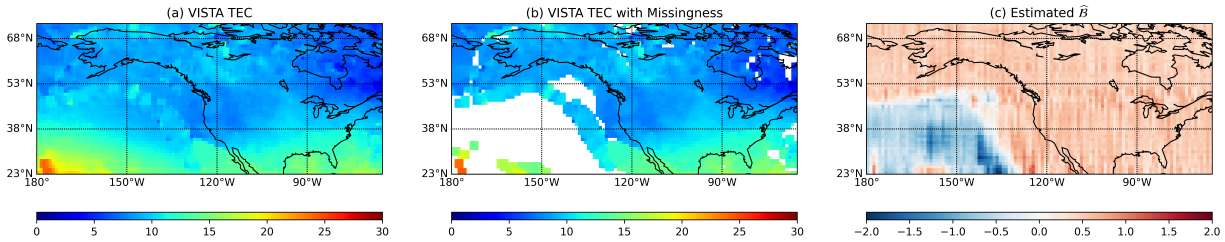

Figure 4: (a) The VISTA TEC at 00:02:30 UT, September 1, 2017. (b) The VISTA TEC in (a) with data missingness from the Madrigal TEC. (c) Fitted $\widehat{\boldsymbol{\mathcal{B}}}$ based on the Ising model.

To set up the experiment, we use the first 20 days of data in September 2017, and each day consists of a tensor of size $50 \times 115 \times 96$. We use the first 5 days as a validation set to search for the best $g(\cdot, \cdot)$ function for the Ising model. For each day, we fit the CTC algorithm with a simple tensor completion algorithm based on (1) with a Tucker rank at $(3, 3, 3)$ and pick the tensor-train rank $\mathbf{r} = (r, r)$ by P-AIC. Based on Figure 3, we know that the Ising model exhibits under-coverage as r/d increases over 0.15; therefore, we select the rank r from $2 \leq r \leq 7$ only. For each day, we consider the Ising model with $g(x, y) = 5xy/4, h(x) = x/2$, the Bernoulli model with $g(x, y) = 0, h(x) = x/2$, and the unweighted conformal prediction for comparison. In Table 1, we report the results on the average mis-coverage % and the empirical coverage of 90% and 95% CI.

method	mis-coverage %	90% CI coverage %	95% CI coverage %
unweighted	42.1(6.49)	46.3(6.58)	52.3(7.23)
Bernoulli	23.1(5.34)	64.6(5.97)	76.8(5.03)
Ising	6.01(2.45)	90.0(6.06)	94.2(3.74)

Table 1: Mis-coverage % and empirical coverage of CI at 90% and 95% level for the unweighted conformal prediction and weighted conformal prediction with Bernoulli and Ising model for data during Sept 6 to Sept 20, 2017.

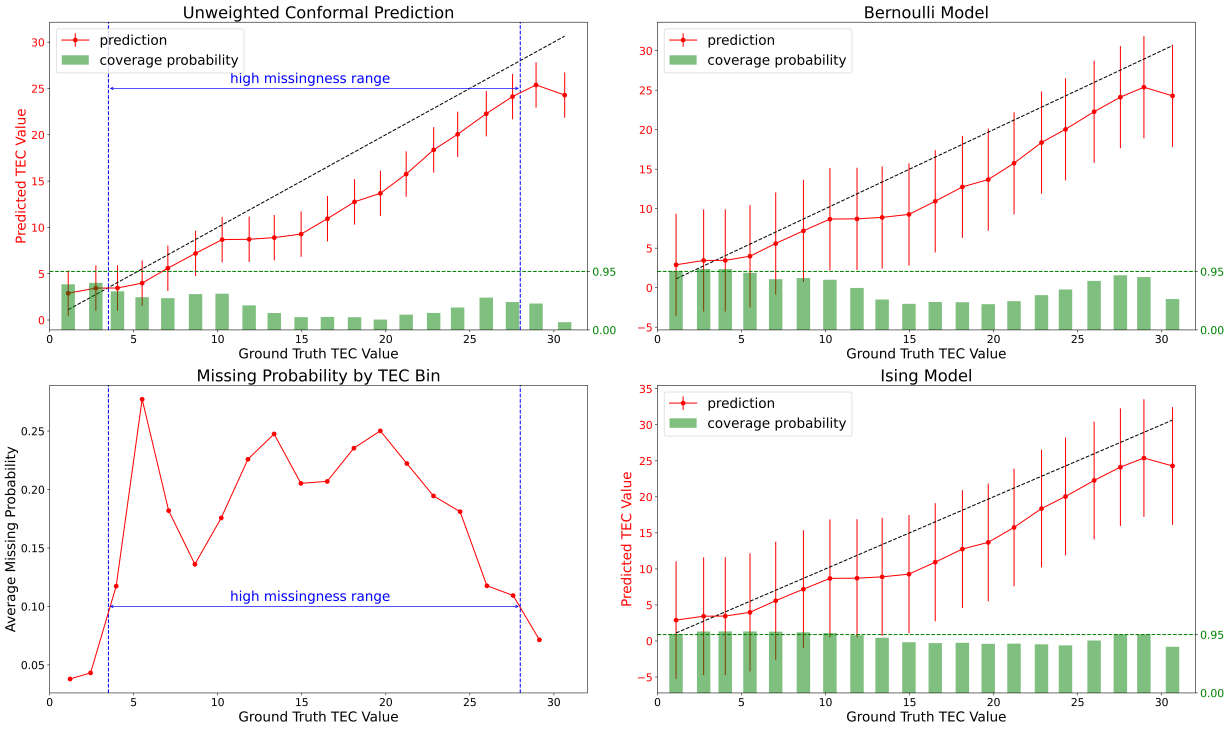

Figure 5: All except the lower-left panels show the average 95% conformal intervals and the empirical coverage for 20 different bins of TEC values on Sept 6, 2017. Each bin spans 1.5 TEC units. The lower-left panel shows the missing probability of different bins. A bin is termed "high missingness" if $> 10\%$ of the data is missing.

In Figure 5, we visualize the average 95% CI and its empirical coverage for 20 different bins of TEC values on Sept 6, 2017. It is shown that the data missingness is not uniform across different bins of TEC values, and different bins have different distributions of the imputation errors (see how the prediction deviates from the truth), making the unweighted conformal prediction less favorable, especially when data missingness is high. These empirical results reveal that by accounting for the heterogeneity and the spatial dependency of data missingness, one can construct well-calibrated confidence intervals using our method.

6 Conclusion

In this paper, we propose a data-driven approach for quantifying the uncertainty of tensor completion. Our method consists of two major steps. We first estimate the missing propensity of each tensor entry using a parameterized Ising model with a low tensor-train rank parameter, and then plug in the missing propensity estimator to weight each tensor entry, and then construct the confidence region with split conformal prediction. We implement the estimation of missing propensity with a computationally efficient Riemannian gradient descent algorithm and validate the resulting conformal intervals with extensive simulation studies and an application to regional TEC reconstruction. We focus on tensor-train rank in our implementation, but our method can be easily extended to other tensor ranks as long as the low-rank tensors lie on a smooth manifold.

There are two limitations of our method. Firstly, we do not have a systematic approach to determine the best specification of the Ising model, e.g., the hyperparameters for the function $g(x, y)$ and $h(x)$. We recommend using a joint gradient descent approach, where one uses RGrad for $\boldsymbol{\mathcal{B}}$ and gradient descent for the hyperparameters. Alternatively, one can also try grid search on a held-out validation set.

Secondly, our Ising model can only account for locally dependent missingness, but not arbitrary missingness. In our simulation experiment, we consider adversarial noise, which is somewhat similar to the missing not-at-random (MNAR) scenario where the uncertainty of the entry is related to the missing propensity. Our model can be extended to handle more flexible missingness, such as those in Tabouy et al. (2020), Sportisse et al. (2020). We leave these topics to future research.

Acknowledgement

We thank Shasha Zou for helpful discussions on the TEC data. YC is supported by NSF DMS 2113397, NSF PHY 2027555, NASA Federal Award No. 80NSSC23M0192 and No. 80NSSC23M0191.

References

Akaike, H. (1973), Information Theory and an Extension of the Maximum Likelihood Principle, *in* 'Proc. 2nd International Symposium of Information Theory', Akademiai Kiado, pp. 267–281.

Barber, R. F. & Drton, M. (2015), 'High-dimensional Ising Model Selection with Bayesian Information Criteria', *Electronic Journal of Statistics* **9**, 567–607.

Bhattacharya, B. B. & Mukherjee, S. (2018), 'Inference in Ising Models', *Bernoulli* **24**(1), 493–525.

Bi, X., Qu, A. & Shen, X. (2018), 'Multilayer Tensor Factorization with Applications to Recommender Systems', *The Annals of Statistics* **46**(6B), 3308 – 3333.
URL: *https://doi.org/10.1214/17-AOS1659*

Cai, C., Li, G., Poor, H. V. & Chen, Y. (2022), 'Nonconvex Low-rank Tensor Completion from Noisy Data', *Operations Research* **70**(2), 1219–1237.

Cai, C., Poor, H. V. & Chen, Y. (2022), 'Uncertainty Quantification for Nonconvex Tensor Completion: Confidence intervals, Heteroscedasticity and Optimality', *IEEE Transactions on Information Theory* **69**(1), 407–452.

Cai, J.-F., Li, J. & Xia, D. (2022*a*), 'Generalized Low-rank plus Sparse Tensor Estima-

tion by Fast Riemannian Optimization', *Journal of the American Statistical Association* pp. 1–17.

Cai, J.-F., Li, J. & Xia, D. (2022*b*), 'Provable Tensor-Train Format Tensor Completion by Riemannian Optimization', *Journal of Machine Learning Research* **23**(1), 5365–5441.

Chen, H., Raskutti, G. & Yuan, M. (2019), 'Non-convex Projected Gradient Descent for Generalized Low-rank Tensor Regression', *The Journal of Machine Learning Research* **20**(1), 172–208.

Chen, Y., Fan, J., Ma, C. & Yan, Y. (2019), 'Inference and Uncertainty Quantification for Noisy Matrix Completion', *Proceedings of the National Academy of Sciences* **116**(46), 22931–22937.

Cipra, B. A. (1987), 'An Introduction to the Ising Model', *The American Mathematical Monthly* **94**(10), 937–959.

Csiszár, I. & Talata, Z. (2006), 'Consistent Estimation of the Basic Neighborhood of Markov Random Fields', *The Annals of Statistics* **34**(1), 123–145.

Davenport, M. A., Plan, Y., Van Den Berg, E. & Wootters, M. (2014), '1-bit Matrix Completion', *Information and Inference: A Journal of the IMA* **3**(3), 189–223.

Farias, V., Li, A. A. & Peng, T. (2022), Uncertainty quantification for low-rank matrix completion with heterogeneous and sub-exponential noise, *in* 'International Conference on Artificial Intelligence and Statistics', PMLR, pp. 1179–1189.

Greenwald, M. & Khanna, S. (2001), 'Space-Efficient Online Computation of Quantile Summaries', *ACM SIGMOD Record* **30**(2), 58–66.

Guan, L. (2023), 'Localized Conformal Prediction: A Generalized Inference Framework for Conformal Prediction', *Biometrika* **110**(1), 33–50.

Gui, Y., Barber, R. & Ma, C. (2023), 'Conformalized Matrix Completion', *Advances in Neural Information Processing Systems* **36**, 4820–4844.

Holtz, S., Rohwedder, T. & Schneider, R. (2012), 'On Manifolds of Tensors of Fixed TT-rank', *Numerische Mathematik* **120**(4), 701–731.

Hong, D., Kolda, T. G. & Duersch, J. A. (2020), 'Generalized Canonical Polyadic Tensor Decomposition', *SIAM Review* **62**(1), 133–163.

Ji, C. & Seymour, L. (1996), 'A Consistent Model Selection Procedure for Markov Random Fields based on Penalized Pseudolikelihood', *The Annals of Applied Probability* **6**(2), 423–443.

Ke, Z. T., Shi, F. & Xia, D. (2019), 'Community Detection for Hypergraph Networks via Regularized Tensor Power Iteration', *arXiv preprint arXiv:1909.06503* .

Kolda, T. G. & Bader, B. W. (2009), 'Tensor Decompositions and Applications', *SIAM review* **51**(3), 455–500.

Kressner, D., Steinlechner, M. & Vandereycken, B. (2014), 'Low-rank Tensor Completion by Riemannian Optimization', *BIT Numerical Mathematics* **54**, 447–468.

Lee, C. & Wang, M. (2020), Tensor Denoising and Completion based on Ordinal Observations, *in* 'International conference on machine learning', PMLR, pp. 5778–5788.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J. & Wasserman, L. (2018), 'Distribution-free Predictive Inference for Regression', *Journal of the American Statistical Association* **113**(523), 1094–1111.

Li, X., Xu, D., Zhou, H. & Li, L. (2018), 'Tucker Tensor Regression and Neuroimaging Analysis', *Statistics in Biosciences* **10**, 520–545.

Li, X., Ye, Y. & Xu, X. (2017), Low-rank Tensor Completion with Total Variation for Visual Data Inpainting, *in* 'Proceedings of the AAAI Conference on Artificial Intelligence', Vol. 31.

Li, Z. & Xiao, H. (2021), 'Multi-Linear Tensor Autoregressive Models', *arXiv preprint arXiv:2110.00928* .

Liang, D., Charlin, L., McInerney, J. & Blei, D. M. (2016), Modeling User Exposure in Recommendation, *in* 'Proceedings of the 25th international conference on World Wide Web', pp. 951–961.

Luo, Y. & Zhang, A. R. (2022), 'Tensor-on-tensor Regression: Riemannian Optimization, Over-parameterization, Statistical-computational gap, and their Interplay', *arXiv preprint arXiv:2206.08756* .

Ma, W. & Chen, G. H. (2019), 'Missing Not at Random in Matrix Completion: The Effectiveness of Estimating Missingness Probabilities Under a Low Nuclear Norm Assumption', *Advances in Neural Information Processing Systems* **32**.

Ma, W. & Xia, D. (2024), 'Statistical Inference in Tensor Completion: Optimal Uncertainty Quantification and Statistical-to-Computational Gaps', *arXiv preprint arXiv:2410.11225* .

Mao, H., Martin, R. & Reich, B. J. (2022), 'Valid Model-Free Spatial Prediction', *Journal of the American Statistical Association* pp. 1–11.

Matsuda, T., Uehara, M. & Hyvarinen, A. (2021), 'Information Criteria for Non-normalized Models', *Journal of Machine Learning Research* **22**(158), 1–33.

MIT Haystack Observatory (2012), 'Madrigal database'. `http://millstonehill.haystack.mit.edu/`.

Oseledets, I. V. (2011), 'Tensor-Train Decomposition', *SIAM Journal on Scientific Computing* **33**(5), 2295–2317.

Pan, Y., Jin, M., Zhang, S. & Deng, Y. (2021), 'Tec Map Completion through a Deep Learning Model: SNP-GAN', *Space Weather* **19**(11), e2021SW002810.

Papadopoulos, H., Proedrou, K., Vovk, V. & Gammerman, A. (2002), Inductive Confidence Machines for Regression, *in* 'Machine Learning: ECML 2002: 13th European Conference on Machine Learning Helsinki, Finland, August 19–23, 2002 Proceedings 13', Springer, pp. 345–356.

Qi, J., Yang, C.-H. H., Chen, P.-Y. & Tejedor, J. (2023), 'Exploiting Low-rank Tensor-Train Deep Neural Networks based on Riemannian Gradient Descent with Illustrations of Speech Processing', *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **31**, 633–642.

Ravikumar, P., Wainwright, M. J. & Lafferty, J. D. (2010), 'High-dimensional Ising Model Selection using ℓ_1-Regularized Logistic Regression', *The Annals of Statistics* pp. 1287–1319.

Schnabel, T., Swaminathan, A., Singh, A., Chandak, N. & Joachims, T. (2016), Recommendations as Treatments: Debiasing Learning and Evaluation, *in* 'International Conference on Machine Learning', PMLR, pp. 1670–1679.

Schwarz, G. (1978), 'Estimating the Dimension of a Model', *Annals of Statistics* **6**(2), 461–464.

Shafer, G. & Vovk, V. (2008), 'A Tutorial on Conformal Prediction.', *Journal of Machine Learning Research* **9**(3).

Sportisse, A., Boyer, C. & Josse, J. (2020), 'Imputation and Low-rank Estimation with Missing not at Random Data', *Statistics and Computing* **30**(6), 1629–1643.

Steinlechner, M. (2016), 'Riemannian Optimization for High-Dimensional Tensor Completion', *SIAM Journal on Scientific Computing* **38**(5), S461–S484.

Sun, H., Chen, Y., Zou, S., Ren, J., Chang, Y., Wang, Z. & Coster, A. (2023), 'Complete Global Total Electron Content Map Dataset based on a Video Imputation Algorithm VISTA', *Scientific Data* **10**(1), 236.

Sun, H., Hua, Z., Ren, J., Zou, S., Sun, Y. & Chen, Y. (2022), 'Matrix completion methods for the total electron content video reconstruction', *The Annals of Applied Statistics* **16**(3), 1333–1358.

Sun, H., Manchester, W., Jin, M., Liu, Y. & Chen, Y. (2023), Tensor Gaussian Process with Contraction for Multi-Channel Imaging Analysis, *in* 'Proceedings of the 40th International Conference on Machine Learning', PMLR, pp. 32913–32935.

Tabouy, T., Barbillon, P. & Chiquet, J. (2020), 'Variational Inference for Stochastic Block Models from Sampled Data', *Journal of the American Statistical Association* **115**(529), 455–466.

Tibshirani, R. J., Foygel Barber, R., Candes, E. & Ramdas, A. (2019), 'Conformal Prediction under Covariate Shift', *Advances in Neural Information Processing Systems (Neurips)* **32**.

Tomioka, R. & Suzuki, T. (2014), 'Spectral Norm of Random Tensors', *arXiv preprint arXiv:1407.1870* .

Vovk, V., Gammerman, A. & Shafer, G. (2005), *Algorithmic Learning in a Random World*, Vol. 29, Springer.

Wang, H., Chen, J. & Wei, K. (2023), 'Implicit Regularization and Entrywise Convergence of Riemannian Optimization for Low Tucker-rank Tensor Completion', *Journal of Machine Learning Research* **24**(347), 1–84.

Wang, J., Zhao, G., Wang, D. & Li, G. (2019), Tensor Completion using Low-rank Tensor Train Decomposition by Riemannian Optimization, *in* '2019 Chinese Automation Congress (CAC)', IEEE, pp. 3380–3384.

Wang, M. & Li, L. (2020), 'Learning from Binary Multiway Data: Probabilistic Tensor Decomposition and its Statistical Optimality', *Journal of Machine Learning Research* **21**(1), 6146–6183.

Wang, M. & Zeng, Y. (2019), 'Multiway Clustering via Tensor Block Models', *Advances in neural information processing systems* **32**.

Wang, M., Zheng, X., Yang, Y. & Zhang, K. (2018), Collaborative Filtering with Social Exposure: A Modular Approach to Social Recommendation, *in* 'Proceedings of the AAAI Conference on Artificial Intelligence', Vol. 32.

Wang, Z., Zou, S., Liu, L., Ren, J. & Aa, E. (2021), 'Hemispheric Asymmetries in the Mid-latitude Ionosphere During the September 7–8, 2017 Storm: Multi-instrument Observations', *Journal of Geophysical Research: Space Physics* **126**, e2020JA028829.

Wang, Z., Zou, S., Sun, H. & Chen, Y. (2023), 'Forecast Global Ionospheric TEC: Apply Modified U-Net on VISTA TEC Data Set', *Space Weather* **21**(8), e2023SW003494.

Wei, B., Peng, L., Guo, Y., Manatunga, A. & Stevens, J. (2023), 'Tensor Response Quantile Regression with Neuroimaging Data', *Biometrics* **79**(3), 1947–1958.

Xia, D., Yuan, M. & Zhang, C.-H. (2021), 'Statistically Optimal and Computationally Efficient Low Rank Tensor Completion from Noisy Entries', *The Annals of Statistics* **49**(1).

Younas, W., Khan, M., Amory-Mazaudier, C., Amaechi, P. O. & Fleury, R. (2022), 'Middle and Low Latitudes Hemispheric Asymmetries in $\Sigma O/N2$ and TEC during intense magnetic storms of solar cycle 24', *Advances in Space Research* **69**, 220–235.

Yuan, M. & Zhang, C.-H. (2016), 'On Tensor Completion via Nuclear Norm Minimization', *Foundations of Computational Mathematics* **16**(4), 1031–1068.

SUPPLEMENTARY MATERIAL

This supplemental material contains three sections. Section A contains the proofs for proposition 2.1 and additional theoretical results for the estimation error and coverage guarantee for the Bernoulli model. Section B describes the technical lemmas used in Section A. Section C contains additional details and results of the simulation experiments. All figures, tables, and equations in the supplemental material are numbered with a prefix "S", which distinguishes them from the main paper numbering.

Contents

A Proofs of Theorems and Propositions

Throughout this section, for any tensor $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, we use $\bar{d}, d^*$ to denote $\sum_k d_k$ and $\prod_k d_k$, respectively. For any tensor-train rank $\mathbf{r} = (r_1, \ldots, r_{K-1})$, we use r^* to denote $\prod_k r_k$. We use $c, c', C, C_0, C_1, \ldots$ to denote positive absolute constants and $c_K, c'_K, C_K, C_{K,0}, C_{K,1}, \ldots$ to denote positive constants that only relate to K. For two sequences $\{a_n\}_{n=1}^{\infty}$ and $\{b_n\}_{n=1}^{\infty}$, we use $a_n \asymp b_n$ to represent $\lim_{n\to\infty} a_n/b_n = C > 0$, with C being finite.

A.1 Proof of Proposition 2.1

Proof. Given any testing entry $s^* \in \mathbb{S}_{miss}$, we relabel all elements in $\mathbb{S}_{cal} \cup \{s^*\}$ as $\{s_1, \ldots, s_{n+1}\}$. Now recall the definition of $\boldsymbol{\mathcal{E}}_0$ as:

$$\boldsymbol{\mathcal{E}}_0 = \left\{ \widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ for } s \in \mathbb{S}_{tr} \cup \mathbb{S}_{cal}, \mathbb{S}_{cal} \cup \{s^*\} = \{s_1, \ldots, s_{n+1}\} \text{ and } \widetilde{\boldsymbol{\mathcal{W}}}_s = -1 \text{ o.w.} \right\},$$

namely one observes data only at $\mathbb{S}_{tr}$ and n out of $n+1$ entries from $\{s_1, \ldots, s_{n+1}\}$.

Let V denote the non-conformity score of the testing entry, then the weighted exchangeability framework in Tibshirani et al. (2019) states that one can treat V as a weighted draw from $\{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_1}, \widehat{\boldsymbol{\mathcal{X}}}_{s_1}), \ldots, \mathcal{S}(\boldsymbol{\mathcal{X}}_{s_{n+1}}, \widehat{\boldsymbol{\mathcal{X}}}_{s_{n+1}})\}$, with weight being:

$$\mathrm{P}\left[V = \mathcal{S}(\boldsymbol{\mathcal{X}}_{s_k}, \widehat{\boldsymbol{\mathcal{X}}}_{s_k}) \middle| \boldsymbol{\mathcal{E}}_0 \right] = \frac{\mathrm{P}\left[\widetilde{\boldsymbol{\mathcal{W}}}_{s_k} = -1, \widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ for } s \in \mathbb{S}_k \middle| \boldsymbol{\mathcal{E}}_0 \right]}{\sum_{l=1}^{n+1} \mathrm{P}\left[\widetilde{\boldsymbol{\mathcal{W}}}_{s_l} = -1, \widetilde{\boldsymbol{\mathcal{W}}}_s = 1 \text{ for } s \in \mathbb{S}_l \middle| \boldsymbol{\mathcal{E}}_0 \right]},$$

where $\mathbb{S}_k = \{s_1, \ldots, s_{n+1}\} \setminus \{s_k\}$, for $k = 1, \ldots, n+1$. Multiplying both the numerator and the denominator by $\mathrm{P}(\boldsymbol{\mathcal{E}}_0)$ leads to the weight in the form of $p_k / \sum_{l=1}^{n+1} p_l$, with p_k defined as (4). The coverage guarantee in (6) is then a direct result of Theorem 2 of Tibshirani et al. (2019). □

A.2 Approximation Error of Conformal Weights

In this subsection, we estimate the additional errors introduced by the approximation we made when computing the conformal weight in (11). Basically, to eliminate the dependency of $\omega_k(s^*)$ on the specific $s^* \in \mathbb{S}_{miss}$, we modify (11) as:

$$\omega_k = \frac{p_k}{\sum_{i=1}^{n+1} p_i} = \frac{\exp\left[-2\sum_{s_j \in \mathcal{N}(s_k)} g(\boldsymbol{\mathcal{B}}_{s_k}, \boldsymbol{\mathcal{B}}_{s_j}) \boldsymbol{\mathcal{W}}_{s_j} - 2h(\boldsymbol{\mathcal{B}}_{s_k})\right]}{\sum_{i=1}^{n+1} \exp\left[-2\sum_{s_j \in \mathcal{N}(s_i)} g(\boldsymbol{\mathcal{B}}_{s_i}, \boldsymbol{\mathcal{B}}_{s_j}) \boldsymbol{\mathcal{W}}_{s_j} - 2h(\boldsymbol{\mathcal{B}}_{s_i})\right]}, \tag{S.1}$$

i.e. setting $\widetilde{\boldsymbol{\mathcal{W}}}_{s_{n+1}} = -1$ instead of 1. We summarize the result in the proposition below.

Proposition A.1. *Let $s_1, \ldots, s_n \in \mathbb{S}_{cal}$, $s_{n+1} = s^* \in \mathbb{S}_{miss}$, and let $F^*(\cdot)$ be the CDF of the distribution:*

$$\sum_{i=1}^{n} \omega_i(s^*) \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})} + \omega_{n+1}(s^*) \cdot \delta_{+\infty},$$

with $\omega_i(s^*)$ defined in (11)*. Similarly, let $F(\cdot)$ be the CDF of the distribution:*

$$\sum_{i=1}^{n} \omega_i \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})} + \omega_{n+1} \cdot \delta_{+\infty},$$

with ω_i defined in (S.1)*. Let $\mathbb{N}_0 = \{i \in [n] | s_i \in \mathcal{N}(s_{n+1})\}$, and $\gamma(\boldsymbol{\mathcal{B}}) = \min_{j \in \mathbb{N}_0} g(\boldsymbol{\mathcal{B}}_{s_j}, \boldsymbol{\mathcal{B}}_{s_{n+1}})$. Then we have the following universal bound over $|F^*(x) - F(x)|$:*

$$\sup_{x \in \mathbb{R}} |F^*(x) - F(x)| \leq 3 \cdot \max\left\{1, e^{4\gamma(\boldsymbol{\mathcal{B}})} - 1\right\} \cdot \sum_{k \in \mathbb{N}_0} \omega_k(s^*). \tag{S.2}$$

Proof. Define $\tilde{a}_k$ as $\exp\left[-2\sum_{s_j \in \mathcal{N}(s_k)} g(\boldsymbol{\mathcal{B}}_{s_k}, \boldsymbol{\mathcal{B}}_{s_j}) \widetilde{\boldsymbol{\mathcal{W}}}_{s_j} - 2h(\boldsymbol{\mathcal{B}}_{s_k})\right]$, and define a_k similar to $\tilde{a}_k$ but replace $\widetilde{\boldsymbol{\mathcal{W}}}_{s_j}$ with $\boldsymbol{\mathcal{W}}_{s_j}$. Then we have $\omega_k = a_k / \sum_i a_i$ and $\omega_k(s^*) = \tilde{a}_k / \sum_i \tilde{a}_i$. By

definition, $\tilde{a}_i = a_i$ if and only if $i \notin \mathbb{N}_0$. Then we have for any $l = 1, \ldots, n+1$:

$$
\begin{aligned}
|\omega_l(s^*) - \omega_l| &= \left| \frac{\tilde{a}_l}{\sum_{j\in\mathbb{N}_0} \tilde{a}_j + \sum_{j\notin\mathbb{N}_0} a_j} - \frac{a_l}{\sum_{j\in\mathbb{N}_0} a_j + \sum_{j\notin\mathbb{N}_0} a_j} \right| \\
&= \left| \frac{(\tilde{a}_l - a_l) \cdot \sum_{j\notin\mathbb{N}_0} a_j + (\tilde{a}_l - a_l) \cdot \sum_{j\in\mathbb{N}_0} a_j + a_l \cdot \sum_{j\in\mathbb{N}_0} (a_j - \tilde{a}_j)}{\left(\sum_{j\in\mathbb{N}_0} \tilde{a}_j + \sum_{j\notin\mathbb{N}_0} a_j\right)\left(\sum_{j\in\mathbb{N}_0} a_j + \sum_{j\notin\mathbb{N}_0} a_j\right)} \right| \\
&\leq 2 \cdot \frac{|\tilde{a}_l - a_l|}{\sum_{j\in\mathbb{N}_0} \tilde{a}_j + \sum_{j\notin\mathbb{N}_0} a_j} + \omega_l \cdot \sum_{j\in\mathbb{N}_0} \frac{|\tilde{a}_j - a_j|}{\sum_{j\in\mathbb{N}_0} \tilde{a}_j + \sum_{j\notin\mathbb{N}_0} a_j} \\
&= 2 \cdot \omega_l(s^*) \cdot \left(1 - \exp\left[-4 \cdot \mathbb{1}_{\{l\in\mathbb{N}_0\}} \cdot g(\boldsymbol{\mathcal{B}}_{s_l}, \boldsymbol{\mathcal{B}}_{s_{n+1}})\right]\right) \\
&+ \omega_l \cdot \sum_{j\in\mathbb{N}_0} \omega_j(s^*) \cdot \left(1 - \exp\left[-4 \cdot g(\boldsymbol{\mathcal{B}}_{s_j}, \boldsymbol{\mathcal{B}}_{s_{n+1}})\right]\right) \\
&\leq \max\left\{1, e^{4\gamma(\boldsymbol{\mathcal{B}})-1}\right\} \cdot \left[2 \cdot \omega_l(s^*) \cdot \mathbb{1}_{\{l\in\mathbb{N}_0\}} + \omega_l \cdot \sum_{j\in\mathbb{N}_0} w_j(s^*)\right]. \qquad \text{(S.3)}
\end{aligned}
$$

Then for any $x \in \mathbb{R}$, we have:

$$
\begin{aligned}
|F^*(x) - F(x)| &= \left| \sum_{i:\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})\leq x} (\omega_i(s^*) - \omega_i) \right| \\
&\leq \sum_{i:\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})\leq x} |\omega_i(s^*) - \omega_i| \\
&\leq \sum_{i:\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})\leq x} \max\left\{1, e^{4\gamma(\boldsymbol{\mathcal{B}})-1}\right\} \cdot \left[2 \cdot \omega_i(s^*) \cdot \mathbb{1}_{\{i\in\mathbb{N}_0\}} + \omega_i \cdot \sum_{j\in\mathbb{N}_0} w_j(s^*)\right] \\
&\leq \max\left\{1, e^{4\gamma(\boldsymbol{\mathcal{B}})-1}\right\} \cdot \left[2 \cdot \sum_{i\in\mathbb{N}_0} \omega_i(s^*) + \left(\sum_{i:\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})\leq x} \omega_i\right) \cdot \sum_{j\in\mathbb{N}_0} \omega_j(s^*)\right] \\
&\leq 3 \cdot \max\left\{1, e^{4\gamma(\boldsymbol{\mathcal{B}})} - 1\right\} \cdot \sum_{k\in\mathbb{N}_0} \omega_k(s^*).
\end{aligned}
$$

□

Proposition A.1 provides a universal upper bound over the empirical CDF of the distribution of non-conformity scores under the exact and approximated weights. The upper bound showed that the deviation is determined by the sum of the weights at all entries that are neighbors of s^*. Typically, we specify the neighbors of each tensor entry as all

entries whose indices differ on only one dimension. So for an order-3 tensor, each entry has at most 6 neighbors, but the size of the calibration set is much larger, leading to generally a very small deviation of the empirical CDFs. The bound in (S.2) can be further refined given specific $\boldsymbol{\mathcal{B}}$, but we leave this general conclusion here to show the minimal impact of the approximation step.

A.3 Bernoulli Model Estimation Error Bound

In this subsection, we derive the error bound of the MPLE estimator $\widehat{\boldsymbol{\mathcal{B}}}$ under the assumption that $g(x, y) = 0$, i.e., all entries of $\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}$ are observed independently with probability $q(\exp[-2h(\boldsymbol{\mathcal{B}}^*_s)] + 1)^{-1}$. Evidently, under this assumption, the MPLE is identical to MLE since the pseudo-likelihood is also the true Bernoulli likelihood. Our main result is in Theorem A.6. To establish the theoretical result, we make several additional assumptions:

Assumption A.2. $h(\cdot) : \mathbb{R} \mapsto \mathbb{R}$ is a non-decreasing, non-constant twice continuously differentiable function with $h''(\cdot) \geq 0$.

Assumption A.3. The MPLE estimator $\widehat{\boldsymbol{\mathcal{B}}}$ and the true tensor parameter $\boldsymbol{\mathcal{B}}^*$ have bounded max-norm: $\|\widehat{\boldsymbol{\mathcal{B}}}\|_\infty, \|\boldsymbol{\mathcal{B}}^*\|_\infty \leq \xi$.

We define $f(x) = \exp[2h(x)]/(1 + \exp[2h(x)])$ and the following two constants:

$$\alpha_\xi = \sup_{|x|\leq\xi} |2h'(x)|, \quad \gamma_\xi = \inf_{|x|\leq\xi} \min \left\{ \left[\frac{f'(x)}{f(x)}\right]^2 - \frac{f''(x)}{f(x)}, \frac{qf''(x)}{1 - qf(x)} + \left[\frac{qf'(x)}{1 - qf(x)}\right]^2 \right\}.$$

To see what these two constants represent, recall that the negative log-likelihood for $\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}$ given $\boldsymbol{\mathcal{B}}$ can be written as the sum of each entry's negative log-likelihood $\ell_i([\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i|\boldsymbol{\mathcal{B}})$, which is defined as:

$$\ell_i([\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i|\boldsymbol{\mathcal{B}}) = -\left[\left(\frac{[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i + 1}{2}\right) \log qf(\boldsymbol{\mathcal{B}}_i) + \left(\frac{1 - [\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_i}{2}\right) \log(1 - qf(\boldsymbol{\mathcal{B}}_i))\right].$$

It is not difficult to verify that α_ξ upper bounds $|\partial\ell_i(\cdot|\boldsymbol{\mathcal{B}})/\partial\boldsymbol{\mathcal{B}}_i|$ and γ_ξ lower bounds $\partial^2\ell_i(\cdot|\boldsymbol{\mathcal{B}})/\partial\boldsymbol{\mathcal{B}}_i^2$ for all i as long as $\max_s |\boldsymbol{\mathcal{B}}_s| \leq \xi$. By excluding the trivial case where $h(\cdot)$ is a

constant function, α_ξ is strictly positive. If for all $|x| \le \xi$, we have $1-(1-q)f(x)-f^2(x) > 0$, then we can verify that $\gamma_\xi > 0$ for common choices of $h(\cdot)$, such as the logit model $h(x) = x/2$ or the probit model $h(x) = 2^{-1}\log[\Phi(x)/(1-\Phi(x))]$. For the remainder of the appendix, we will assume generally that $\gamma_\xi > 0$, which is simply saying that the function $\ell_i(\cdot|\boldsymbol{\mathcal{B}})$ is γ_ξ-strongly convex.

Finally, it is useful to define the tensor spectral norm and the tensor nuclear norm here:

Definition A.4. For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{d_1\times\cdots\times d_K}$, its spectral norm, denoted as $\|\boldsymbol{\mathcal{A}}\|_\sigma$, is defined as:

$$\|\boldsymbol{\mathcal{A}}\|_\sigma = \sup_{\mathbf{u}_1,\ldots,\mathbf{u}_K} \langle \boldsymbol{\mathcal{A}}, \mathbf{u}_1 \circ \cdots \circ \mathbf{u}_K\rangle, \quad \mathbf{u}_k \in \mathbb{S}^{d_k-1}, \forall k,$$

where $\circ$ denotes vector outer product and $\mathbb{S}^{d_k-1}$ is a unit sphere in $\mathbb{R}^{d_k}$.

Definition A.5. For a tensor $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{d_1\times\cdots\times d_K}$, its nuclear norm $\|\boldsymbol{\mathcal{C}}\|_*$ is defined as:

$$\|\boldsymbol{\mathcal{C}}\|_* = \inf\left\{\sum_r \lambda_r \middle| \boldsymbol{\mathcal{C}} = \sum_r \lambda_r \mathbf{u}_1 \circ \cdots \circ \mathbf{u}_K, \mathbf{u}_k \in \mathbb{S}^{d_k-1}, \forall k\right\}.$$

With the aforementioned assumptions and notations, we have the following non-asymptotic bound on $\|\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}$:

Theorem A.6. *Assume that $g(x,y) = 0$ and assumption A.2 and A.3 hold, and further assume that $\widehat{\boldsymbol{\mathcal{B}}}$ reaches the global minimum of the negative log-likelihood $\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}})$ and the entry-wise negative log-likelihood is γ_ξ-strongly convex with $\gamma_\xi > 0$, then:*

$$\mathrm{P}\left(\frac{1}{\sqrt{d^*}}\|\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}} \le 2C_{K,1}\frac{\alpha_\xi}{\gamma_\xi}\sqrt{\frac{r^*\bar{d}}{d^*}}\right) \ge 1 - \exp\left(-C_1\bar{d}\log K\right), \tag{S.4}$$

where $C_1, C_{K,1}$ are some positive constants.

Proof. Using Taylor expansion upon $\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\widehat{\boldsymbol{\mathcal{B}}})$ at $\boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{B}}^*$ yields:

$$\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\widehat{\boldsymbol{\mathcal{B}}}) = \ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}}^*) + \left\langle \nabla\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}}^*), \widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\right\rangle + \frac{1}{2}\mathbf{vec}(\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*)^\top \mathbf{H}(\check{\boldsymbol{\mathcal{B}}})\mathbf{vec}(\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*), \tag{S.5}$$

where $\check{\boldsymbol{\mathcal{B}}}$ is a convex combination of $\widehat{\boldsymbol{\mathcal{B}}}$ and $\boldsymbol{\mathcal{B}}^*$. Since, by assumption, $\widehat{\boldsymbol{\mathcal{B}}}$ reaches the global minimum of $\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}})$, or $\ell(\boldsymbol{\mathcal{B}})$ in short, we have $\ell(\widehat{\boldsymbol{\mathcal{B}}}) \leq \ell(\boldsymbol{\mathcal{B}}^*)$, and thus the sum of the last two terms in (S.5) are no greater than zero.

For the first term, let $\boldsymbol{\mathcal{G}}^* = \nabla\ell(\boldsymbol{\mathcal{B}}^*)$ and $\boldsymbol{\mathcal{G}}^*$ satisfies:

$$[\boldsymbol{\mathcal{G}}^*]_s = -[1-f(\boldsymbol{\mathcal{B}}_s^*)]\cdot 2h'(x)\cdot \mathbb{1}_{\{[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_s=1\}} + \frac{qf(\boldsymbol{\mathcal{B}}_s^*)[1-f(\boldsymbol{\mathcal{B}}_s^*)]}{1-qf(\boldsymbol{\mathcal{B}}_s^*)}\cdot 2h'(x)\cdot \mathbb{1}_{\{[\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}]_s=-1\}}, \quad \text{(S.6)}$$

and it is easy to verify that $\mathrm{E}[[\boldsymbol{\mathcal{G}}^*]_s] = 0$ and $\|\boldsymbol{\mathcal{G}}^*\|_\infty \leq \alpha_\xi$. By Lemma B.1, we can lower bound the first term as:

$$\left\langle \nabla\ell(\boldsymbol{\mathcal{W}}_{\mathbb{S}_{tr}}|\boldsymbol{\mathcal{B}}^*), \widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^* \right\rangle \geq -\|\boldsymbol{\mathcal{G}}^*\|_\sigma \|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_*. \quad \text{(S.7)}$$

By Lemma B.2, we have $\mathrm{rank}^{\mathrm{tt}}(\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*) \leq 2\mathbf{r}$, and then by Lemma B.3, we have $\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_* \leq \sqrt{(2r_1)\cdots(2r_{K-1})}\cdot\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}$. Therefore, to lower bound the RHS of (S.7), we only need to upper bound the spectral norm of $\boldsymbol{\mathcal{G}}^*$. Since entry-wisely, $\boldsymbol{\mathcal{G}}^*$ is mean-zero and bounded by α_ξ (therefore the sub-Gaussian norm is α_ξ), we can apply Lemma B.4 and get:

$$\mathrm{P}\left(\|\boldsymbol{\mathcal{G}}^*\|_\sigma \leq \sqrt{8\alpha_\xi^2\left[\bar{d}\log 5K + \log\frac{2}{\delta}\right]}\right) \geq 1-\delta. \quad \text{(S.8)}$$

By setting $\delta = \exp\left(-C_1\bar{d}\log K\right)$, with C_1 be some absolute constant, we can simplify (S.8) as:

$$\mathrm{P}\left(\|\boldsymbol{\mathcal{G}}^*\|_\sigma \leq C_K\alpha_\xi\sqrt{\bar{d}}\right) \geq 1-\exp\left(-C_1\bar{d}\log K\right), \quad \text{(S.9)}$$

with $C_K = \sqrt{8\left(\log 5K + C_1\log K + 1\right)}$.

Combining these results, we can lower bound the RHS of (S.7) by:

$$-\|\boldsymbol{\mathcal{G}}^*\|_\sigma\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_* \geq -C_{K,1}\alpha_\xi\sqrt{\bar{d}r^*}\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}, \quad \text{(S.10)}$$

with probability at least $1-\exp\left(-C_1\bar{d}\log K\right)$, where $C_{K,1} = 2^{(K-1)/2}C_K$.

For the quadratic form in (S.5), we have:

$$\frac{1}{2}\mathbf{vec}(\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*)^\top\mathbf{H}(\check{\boldsymbol{\mathcal{B}}})\mathbf{vec}(\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*) \geq \frac{\gamma_\xi}{2}\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}^2 > 0. \quad \text{(S.11)}$$

Combining (S.10) and (S.11), we obtain:

$$\mathrm{P}\left(\frac{1}{\sqrt{d^*}}\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}} \leq 2C_{K,1}\frac{\alpha_\xi}{\gamma_\xi}\sqrt{\frac{r^*\bar{d}}{d^*}}\right) \geq 1-\exp\left(-C_1\bar{d}\log K\right),$$

which completes the proof. □

Remark A.7. Under the scenario where $d_1 \asymp \cdots \asymp d_K \asymp O(d)$ and $r_1 \asymp \cdots r_{K-1} \asymp O(r)$, the result in (S.4) can be reduced to:

$$\mathrm{P}\left(\frac{1}{\sqrt{d^*}}\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}} \leq 2C_{K}\frac{\alpha_\xi}{\gamma_\xi}\sqrt{\left(\frac{r}{\bar{d}}\right)^{K-1}}\right) \geq 1-\exp\left(-C_1\bar{d}\log K\right).$$

So the estimating error can scale with r/d, where a lower r/d generally poses an easier binary tensor decomposition problem with lower root mean-squared error.

Deriving an error bound similar to Theorem A.6 is quite infeasible for the Ising model. There are two major theoretical challenges:

1. showing the concentration inequality of the spectral norm of the gradient tensor $\boldsymbol{\mathcal{G}}$ at the true parameter $\boldsymbol{\mathcal{B}}^*$. It is easier to derive it under the assumption that all entries are independent (Lemma B.4), but much harder to do so under local dependency.

2. proving the strong convexity of the negative pseudo-likelihood near the true value $\boldsymbol{\mathcal{B}}^*$. Again, when the missingness is completely independent, this is feasible, as we have proved in (S.11). However, there is no such guarantee under the locally dependent missingness since each entry's missingness depends on the parameters within its neighborhood.

Proving these facts requires a significant amount of extra research, and we leave this for future theoretical works.

A.4 Bernoulli Model Conformal Inference Coverage Guarantee

In this subsection, we utilize the theoretical result in Theorem A.6 and derive the coverage probability lower bound of the CTC algorithm under the Bernoulli model. The result will reveal how the estimating error of $\boldsymbol{\mathcal{B}}^*$ propagates into the mis-coverage rate. To begin with, we state an essential lemma, which is a trivial extension of Theorem 3.2 of Gui et al. (2023) under the conformalized matrix completion context:

Lemma A.8 (Theorem 3.2 of Gui et al. (2023))**.** *Let $\widehat{\boldsymbol{\mathcal{X}}}$ be the output of any tensor completion algorithm, and $\widehat{\boldsymbol{\mathcal{B}}}$ be the output of the RGrad algorithm and both $\widehat{\boldsymbol{\mathcal{X}}}, \widehat{\boldsymbol{\mathcal{B}}}$ are based on $\mathbb{S}_{tr}$ only, then given that $g(x, y) = 0$, we have:*

$$\mathrm{E}\left[\frac{1}{|\mathbb{S}_{miss}|}\sum_{s\in\mathbb{S}_{miss}} \mathbb{1}_{\{\boldsymbol{\mathcal{X}}_s \in \widehat{C}_{1-\alpha,s}(\widehat{\boldsymbol{\mathcal{X}}})\}}\right] \geq 1 - \alpha - \mathrm{E}[\Delta], \tag{S.12}$$

where $\widehat{C}_{1-\alpha,s}(\widehat{\boldsymbol{\mathcal{X}}})$ is the conformal interval for testing entry s at $(1-\alpha)$ level by the CTC algorithm and Δ is defined as:

$$\Delta = \frac{1}{2}\sum_{s\in\mathbb{S}_{cal}\cup\{s^*\}}\left|\frac{\exp[-2h(\widehat{\boldsymbol{\mathcal{B}}}_s)]}{\sum_{s\in\mathbb{S}_{cal}\cup\{s^*\}}\exp[-2h(\widehat{\boldsymbol{\mathcal{B}}}_s)]} - \frac{\exp[-2h(\boldsymbol{\mathcal{B}}^*_s)]}{\sum_{s\in\mathbb{S}_{cal}\cup\{s^*\}}\exp[-2h(\boldsymbol{\mathcal{B}}^*_s)]}\right|. \tag{S.13}$$

We neglect the proof here since the generalization from matrix to tensor setting is trivial, as one can matricize the tensor into a matrix, and the result holds automatically. By Lemma A.1 in Gui et al. (2023), one can further upper bound Δ by:

$$\Delta \leq \frac{\|\exp[-2h(\widehat{\boldsymbol{\mathcal{B}}})] - \exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1}{\sum_{s\in\mathbb{S}_{cal}}\exp[-2h(\widehat{\boldsymbol{\mathcal{B}}}_s)]}, \tag{S.14}$$

where the $h(\cdot)$ is applied to tensors element-wisely and $\|\cdot\|_1$ is the element-wise tensor ℓ_1 norm. The quantity Δ is trivially bounded by 1 as it is the total-variation (TV) distance between two CDFs of discrete random variables. With this lemma, we now formally state our main result:

Theorem A.9. *Assume that the same assumptions hold as Theorem A.6 and further denote $l_\xi = \inf_{|x|\leq\xi}\exp[-2h(x)], u_\xi = \sup_{|x|\leq\xi}\exp[-2h(x)]$. The $(1-\alpha)$-level conformal interval*

$\widehat{C}_{1-\alpha,s}(\widehat{\boldsymbol{\mathcal{X}}})$ *satisfies:*

$$\mathrm{E}\left[\frac{1}{|\mathbb{S}_{miss}|}\sum_{s\in\mathbb{S}_{miss}}\mathbb{1}_{\{\boldsymbol{\mathcal{X}}_s\in\widehat{C}_{1-\alpha,s}(\widehat{\boldsymbol{\mathcal{X}}})\}}\right]\geq 1-\alpha-\frac{2C_{K,1}c_\xi}{(1-c)(1-q)}\sqrt{\frac{r^*\bar{d}}{d^*}} - \exp[-C_1\bar{d}\log K]-\exp\left[-\frac{c^2(1-q)d^*l_\xi}{2}\right], \quad \text{(S.15)}$$

for any $0<c<1$*, where* q *is the train-calibration split probability in the CTC algorithm and* $c_\xi = u_\xi\alpha_\xi^2/(\gamma_\xi l_\xi^2)$.

Proof. Given Lemma A.8, the coverage guarantee can be derived if one can characterize an upper bound for $\mathrm{E}[\Delta]$. To upper bound Δ, we start from (S.14) and bound the numerator on the RHS of (S.14) as:

$$\begin{aligned}\|\exp[-2h(\widehat{\boldsymbol{\mathcal{B}}})]-\exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1 &\leq \sup_{|x|\leq\xi}|\exp[-2h(x)]\cdot 2h'(x)|\cdot\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_1\\ &\leq u_\xi\alpha_\xi\cdot\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_1\leq u_\xi\alpha_\xi\cdot\sqrt{d^*}\|\widehat{\boldsymbol{\mathcal{B}}}-\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}. \quad \text{(S.16)}\end{aligned}$$

Then we can apply the result of Theorem A.6 to further bound (S.16) with high probability.

For the denominator of the RHS of (S.14), we can lower bound it first as $n_{cal}l_\xi$, and for n_{cal}, since each tensor entry can become a calibration point independently with probability $\exp[-2h(\boldsymbol{\mathcal{B}}_s)](1-q)$, where $0<q<1$ is the train-calibration set split probability, we can then apply the Chernoff bound and obtain:

$$\mathrm{P}\left(n_{cal}\leq(1-c)(1-q)\|\exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1\right)\leq\exp\left[-\frac{c^2(1-q)\|\exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1}{2}\right], \quad \text{(S.17)}$$

for any $0<c<1$. By denoting the event $\{n_{cal}\geq(1-c)(1-q)\|\exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1\}$ as $\mathcal{E}_0$ and the event in (S.4) as $\mathcal{E}_1$ and noticing that $\|\exp[-2h(\boldsymbol{\mathcal{B}}^*)]\|_1\geq d^*l_\xi$, then we have:

$$\mathrm{P}\left(\Delta\leq\frac{2C_{K,1}}{(1-c)(1-q)}\cdot\frac{u_\xi\alpha_\xi^2}{\gamma_\xi l_\xi^2}\cdot\sqrt{\frac{r^*\bar{d}}{d^*}}\right)\geq 1-\exp[-C_1\bar{d}\log K]-\exp\left[-\frac{c^2(1-q)d^*l_\xi}{2}\right],$$

where the probability is the lower bound of the probability of the event $\mathcal{E}_0\cap\mathcal{E}_1$. With this

tail bound on Δ, one can upper bound $\mathrm{E}[\Delta]$ as:

$$\mathrm{E}[\Delta] \leq \frac{2C_{K,1}}{(1-c)(1-q)} \cdot \frac{u_\xi \alpha_\xi^2}{\gamma_\xi l_\xi^2} \cdot \sqrt{\frac{r^* \bar{d}}{d^*}} + \exp[-C_1 \bar{d} \log K] + \exp\left[-\frac{c^2(1-q)d^* l_\xi}{2}\right], \quad \text{(S.18)}$$

and thereby completes the proof. □

Remark A.10. Under the scenario where $d_1 \asymp \cdots \asymp d_K \asymp O(d)$ and $r_1 \asymp \cdots r_{K-1} \asymp O(r)$, the coverage shortfall in (S.15), i.e. the difference between the lower bound in (S.15) and $(1-\alpha)$, can be simplified into:

$$\frac{c_{K,\xi}}{(1-c)(1-q)} \cdot \sqrt{\left(\frac{r}{d}\right)^{K-1}} + \exp\left[-c_K d\right] + \exp\left[-c'_{K,\xi} c^2 (1-q) d^K\right],$$

where $c_{K,\xi}, c'_{K,\xi}$ are positive constants that only relate to K and ξ. The first term is of polynomial order with respect to r/d while the other two terms are of exponential order with respect to d, therefore the first term is the dominating term and the under-coverage of the conformal intervals scale primarily with r/d.

B Technical Lemmas

All technical lemmas listed in this section are cited from existing works. Therefore, we omit the proof here and refer our readers to the corresponding papers cited.

Lemma B.1 (Lemma 1 of Wang & Li (2020)). *For two tensors $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, their inner product $\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle$ can be bounded as:*

$$|\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle| \leq \|\boldsymbol{\mathcal{A}}\|_\sigma \|\boldsymbol{\mathcal{B}}\|_*,$$

where $\|\cdot\|_\sigma, \|\cdot\|_*$ are the tensor spectral norm and the tensor nuclear norm, respectively.

Lemma B.2 (Lemma 24 of Cai, Li & Xia (2022*b*)). *Let $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ be two low tensor-train rank tensors with $\mathrm{rank}^{tt}(\boldsymbol{\mathcal{A}}) = \mathbf{r}_1, \mathrm{rank}^{tt}(\boldsymbol{\mathcal{B}}) \leq \mathbf{r}_2$, respectively. Then one has:*

$$\mathrm{rank}^{tt}(\boldsymbol{\mathcal{A}} + \boldsymbol{\mathcal{B}}) \leq \mathbf{r}_1 + \mathbf{r}_2.$$

Lemma B.3 (Lemma 25 of Cai, Li & Xia (2022*b*)). *Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ be a low tensor-train rank tensor with $\mathrm{rank}^{tt}(\boldsymbol{\mathcal{A}}) = \mathbf{r} = (r_1, \ldots, r_{K-1})$ and has a left-orthogonal representation $\boldsymbol{\mathcal{A}} = [\boldsymbol{\mathcal{T}}_1, \ldots, \boldsymbol{\mathcal{T}}_K]$, then:*

$$\|\boldsymbol{\mathcal{A}}\|_* \leq \sqrt{r_1 \cdots r_{K-1}} \cdot \|\boldsymbol{\mathcal{A}}\|_{\mathrm{F}}.$$

Lemma B.4 (Theorem 1 of Tomioka & Suzuki (2014)). *For a random tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ with mean-zero and independent sub-Gaussian entries with sub-Gaussian norm σ, its spectral norm satisfies:*

$$\|\boldsymbol{\mathcal{A}}\|_\sigma \leq \sqrt{8\sigma^2 \left[\bar{d} \log 5K + \log \frac{2}{\delta}\right]},$$

with probability at least $1 - \delta$.

C Appendix for Simulation

C.1 Details of Simulation Setup

We summarize the data-generating model of all essential tensors involved in the simulation experiment in Table S.1.

C.2 Results on the Missing Propensity Estimation Error

We examine here the effectiveness of the RGrad algorithm for recovering the tensor parameter $\boldsymbol{\mathcal{B}}^*$ from a single observation $\boldsymbol{\mathcal{W}}$. We consider $d \in \{40, 60, 80, 100\}$ and $r \in \{3, 5, 7, 9\}$ when simulating $\boldsymbol{\mathcal{B}}^*$. For simulating $\boldsymbol{\mathcal{W}}$ using the Ising model, we fix $h(x) = x/2$ and consider either $g(x, y) \in \{0, xy/15\}$, where we term the case with $g = 0$ as the (independent) Bernoulli model and the case with $g(x, y) = xy/15$ as the (product) Ising model. We split the training and calibration set randomly based on a $70\% - 30\%$ ratio.

Under each combination of the choices of (d, r, g), we generate $n = 30$ repetitions from a single chain of MCMC and fit RGrad to each repetition with the correctly specified

Tensor	Generating Model	Additional Details
$\boldsymbol{\mathcal{B}}^*$	$\boldsymbol{\mathcal{B}}^* = \boldsymbol{\mathcal{C}} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$ $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{r\times r\times r}, \mathbf{U}_i \in \mathbb{R}^{d_i \times r_i}$	$\boldsymbol{\mathcal{C}} \overset{i.i.d.}{\sim} 0.5 \cdot \mathcal{N}(-1, 0.5) + 0.5 \cdot \mathcal{N}(1, 0.5)$ $\mathbf{U}_i = \begin{bmatrix} 1 & \cdots & 1 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 1 & \cdots & 1 \end{bmatrix}^\top$ and each row of $\mathbf{U}_i$ has $\lceil d_i/r_i \rceil$ ones.
$\boldsymbol{\mathcal{W}}$	$p(\boldsymbol{\mathcal{W}}) \propto \exp\left[-\mathcal{H}(\boldsymbol{\mathcal{W}}\vert\boldsymbol{\mathcal{B}}^*)\right]$ based on (7) and (8)	simulate by block-Gibbs MCMC, where in each proposal we first sample $\mathbb{I}_1 = \{(i_1, \ldots, i_K) \vert \sum_k i_k \text{ is odd}\}$ then $\mathbb{I}_1^c$. Each block is a Bernoulli model.
$\boldsymbol{\mathcal{X}}$	$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}^* + \boldsymbol{\mathcal{E}}$	$\boldsymbol{\mathcal{X}}$ is then masked by $\boldsymbol{\mathcal{W}}$.
$\boldsymbol{\mathcal{X}}^*$	$\boldsymbol{\mathcal{X}}^* = \boldsymbol{\mathcal{C}}^* \times_1 \mathbf{U}_1^* \times_2 \mathbf{U}_2^* \times_3 \mathbf{U}_3^*$ $\boldsymbol{\mathcal{C}}^* \in \mathbb{R}^{3\times 3\times 3}, \mathbf{U}_i^* \in \mathbb{R}^{d_i \times 3}$	$\boldsymbol{\mathcal{C}}^*, \mathbf{U}_1^*, \mathbf{U}_2^*, \mathbf{U}_3^* \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$
$\boldsymbol{\mathcal{E}}$	$[\boldsymbol{\mathcal{E}}]_s \overset{\text{independent}}{\sim} \mathcal{N}(0, \sigma_s^2)$	$\sigma_s = \begin{cases} 1 & \text{constant noise} \\ 0.5\left[1 + \exp(-\boldsymbol{\mathcal{B}}_s^*)\right] & \text{adversarial noise} \end{cases}$

Table S.1: Details of the tensors generated in the simulation experiment.

$g(\cdot, \cdot)$ and a working rank $r' \in \{2, 3, \ldots, 15\}$. In Table S.2, we present the average rank selected by the P-AIC and P-BIC under the Bernoulli and Ising models with various (d, r) combinations.

Bernoulli Model ($g(x, y) = 0$)								
	P-AIC				P-BIC			
rank	$d = 40$	$d = 60$	$d = 80$	$d = 100$	$d = 40$	$d = 60$	$d = 80$	$d = 100$
$r = 3$	**3.0**	**3.0**	**3.0**	**3.0**	2.0	2.0	**3.0**	**3.0**
$r = 5$	**5.0**	**5.0**	**5.0**	**5.0**	2.0	2.1(0.3)	4.0	**5.0**
$r = 7$	6.2(0.4)	**7.0**	**7.0**	**7.0**	2.0	2.0	2.0	2.3(0.4)
$r = 9$	6.0(0.8)	**8.8(0.4)**	**9.0**	**9.0**	2.0	2.0	2.0	2.0
Ising Model ($g(x, y) = xy/15$)								
	P-AIC				P-BIC			
rank	$d = 40$	$d = 60$	$d = 80$	$d = 100$	$d = 40$	$d = 60$	$d = 80$	$d = 100$
$r = 3$	**3.4(2.0)**	**3.0**	**3.0**	**3.0**	2.0	**3.0**	**3.0**	**3.0**
$r = 5$	**7.7(4.1)**	**5.0**	**5.0**	**5.0**	2.0	4.0	**5.0**	**5.0**
$r = 7$	13.9(0.2)	**7.0**	**7.0**	**7.0**	2.0	2.1(0.2)	4.0(0.2)	4.7(0.4)
$r = 9$	13.9(0.2)	**9.0**	**9.0**	**9.0**	2.0	2.0	2.0	3.9(0.3)

Table S.2: Model selection result of the Bernoulli model and Ising model. Each number is the mean rank selected by P-AIC/P-BIC with $n = 30$ repetitions followed by its standard deviations, if non-zero. Boldface is the cases where the true rank is within 1.96 standard deviations of the average rank.

Based on these numerical results, we find that the consistency of P-AIC and P-BIC depends on r/d, or the "low-rankness" $\boldsymbol{\mathcal{B}}^*$. For tensors with high d and low r, both P-AIC and P-BIC are consistent, and the inconsistency emerges as r/d becomes larger. Generally speaking, P-AIC is more robust than P-BIC and is consistent across most of the simulation scenarios except for two cases with small tensor sizes. We therefore suggest using P-AIC for rank selection.

We then evaluate the fitted $\widehat{\boldsymbol{\mathcal{B}}}$ with relative squared error (RSE) defined as: $\|\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}/\|\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}$. The results, as plotted in Figure S.1, exhibit a tendency that as r/d becomes larger, so does the RSE, which echoes the results of the model selection. Additionally, the estimation error is lower for the Ising model, as compared to the Bernoulli model, given the same r and d. We interpret this result as the Ising model estimator can leverage the additional information from neighbors to infer the missing propensity of each tensor entry. In Figure 1(d) of the main paper, we plot the estimator for $\boldsymbol{\mathcal{B}}^*$ shown in 1(a) by RGrad based on a randomly chosen 70% training set, and it is clear that $\widehat{\boldsymbol{\mathcal{B}}}$ reconstructs $\boldsymbol{\mathcal{B}}^*$ very well.

C.3 Results on Conformal Prediction Validation

As a companion result of Figure 2, we plot the empirical coverage and half of the average confidence interval width of three conformal prediction methods under different simulation scenarios in Figure S.2. The mis-coverage of the unweighted conformal prediction comes from under-coverage and is associated with shorter confidence intervals. The reason why unweighted conformal prediction has under-coverage is that under the adversarial noise setting, entries with higher missing propensity also have higher uncertainty, and using a uniform weight underestimates the uncertainty of a missing entry. As one can tell from Figure S.2, our CTC algorithm matches the oracle case quite well and provides well-calibrated confidence intervals.

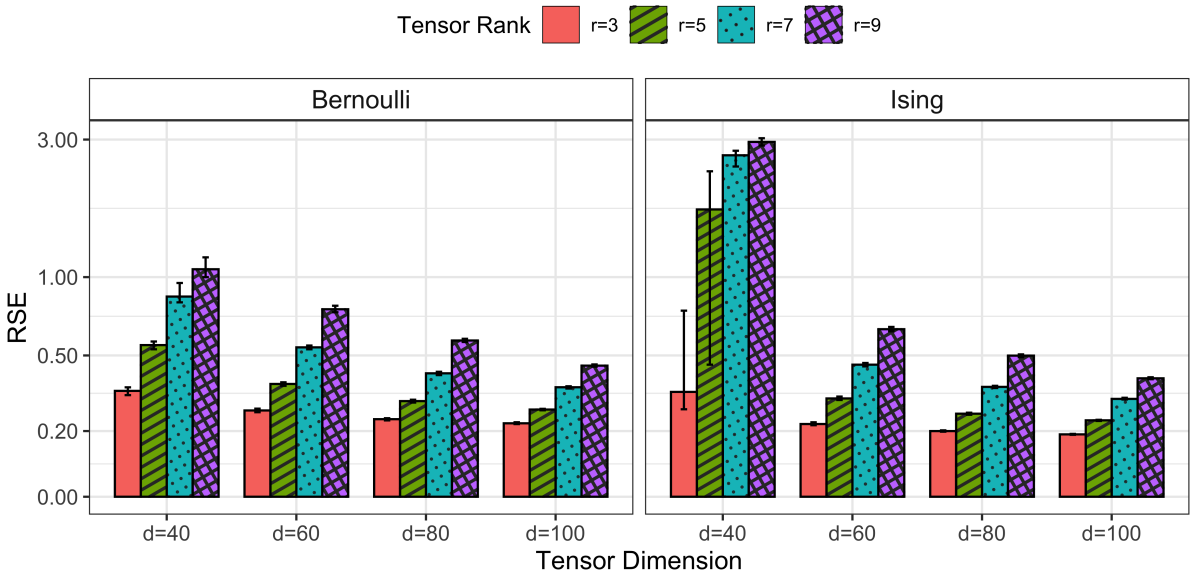

Figure S.1: Relative square error of the MPLE $\widehat{\boldsymbol{\mathcal{B}}}$ under the Bernoulli (left) and Ising model (right). The results are based on $n = 30$ repetitions with the working rank of each sample determined by P-AIC, and each model is fitted by a randomly chosen 70% training set. Error bars show the 2.5% and 97.5% quantiles.

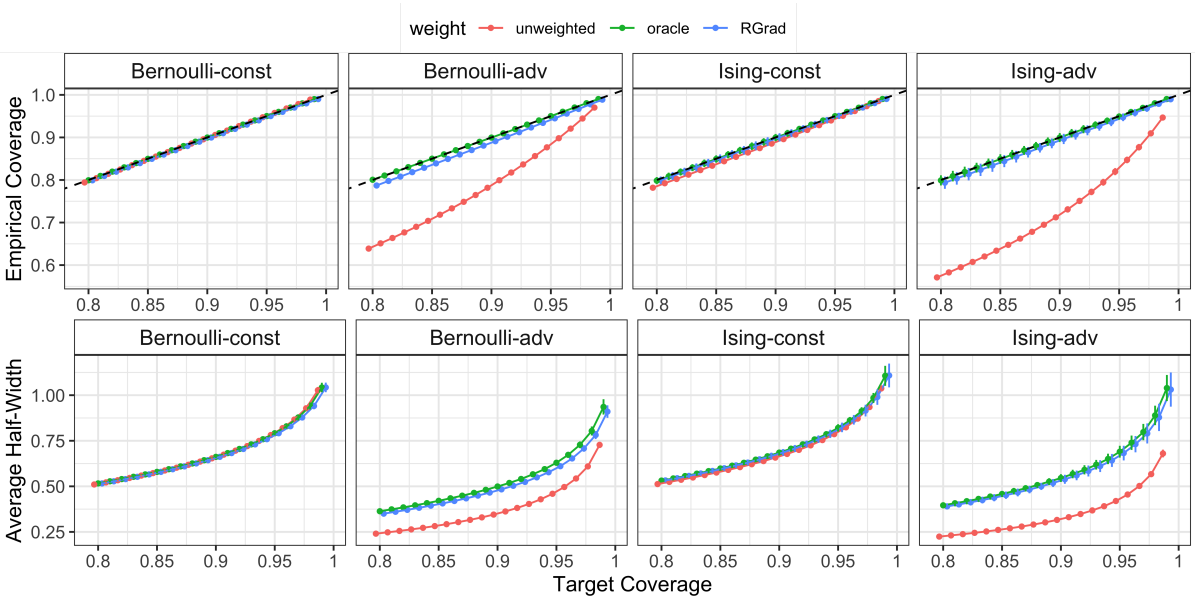

Figure S.2: Empirical coverage and average confidence interval half-width of the three conformal prediction methods across the Bernoulli and Ising model with constant (const) or adversarial (adv) noise. Results are based on $n = 30$ repetitions, and error bars are ± 1.96 standard deviations.

Apart from these results, we also compared other binary tensor decomposition methods for estimating the missing propensity and conducting the weighted conformal prediction

with our method. We mainly consider two competing methods other than the unweighted and oracle conformal prediction: 1) **GCP**: binary tensor decomposition with generalized CP-decomposition (Wang & Li 2020, Hong et al. 2020); 2) **Tucker**: binary tensor decomposition with generalized Tucker-decomposition (Lee & Wang 2020, Cai, Li & Xia 2022*a*). Different from our approach, these two methods assume independence among all the binary entries and thus they are misspecified under the Ising model. We conduct **GCP** with gradient descent following Hong et al. (2020) and **Tucker** with Riemannian gradient descent following (Cai, Li & Xia 2022*a*) and select the corresponding ranks using the BIC criterion, as suggested by the literature. We consider $r = 3, d \in \{40, 60\}$ and list the average mis-coverage % under the constant and adversarial noise regimes as well as the RSE of the estimated $\widehat{\boldsymbol{\mathcal{B}}}$ in Table S.3.

Our finding from Table S.3 is that our method consistently provides well-calibrated confidence intervals close to the oracle case and performs, on average, better than the GCP and Tucker method. Our mis-coverage % is statistically significantly better (p-value < 0.005) than the Tucker method under the adversarial noise regimes across different tensor dimensions and missingness generating models. The GCP method, surprisingly, provides confidence intervals close to our method but has significantly larger RSE for the estimator $\widehat{\boldsymbol{\mathcal{B}}}$. We found that CP-decomposition tends to underestimate the weights of the calibration data; therefore, it has more testing data points with infinitely wide confidence intervals, making it less favorable.

C.4 Results with Misspecified Tensor Completion Model

In this section, we demonstrate the property of our CTC algorithm under a misspecified tensor completion model. We fixate on the scenario where $\boldsymbol{\mathcal{B}}^*$ has rank $r = 3$, and $g(x, y) \in \{0, xy/15\}$. We make a claim in Proposition 2.1 that our conformal inference has valid coverage under any *arbitrary* choice of the tensor completion algorithm that generated $\widehat{\boldsymbol{\mathcal{X}}}$.

(d, Model)	Method	const. mis-coverage %	adv. mis-coverage %	RSE
(40, Bern)	unweighted	0.463(0.244)	11.1(0.389)	/
	oracle	0.381(0.205)	0.409(0.232)	/
	GCP	0.373(0.183)	1.66(0.965)	0.522(0.069)
	Tucker	0.380(0.219)	0.841(0.431)	0.295(0.008)
	RGrad	0.377(0.235)	0.773(0.404)	0.345(0.010)
(60, Bern)	unweighted	0.401(0.165)	11.0(0.241)	/
	oracle	0.202(0.082)	0.207(0.105)	/
	GCP	0.203(0.092)	0.380(0.298)	0.281(0.036)
	Tucker	0.199(0.079)	0.842(0.231)	0.244(0.005)
	RGrad	0.200(0.078)	0.821(0.226)	0.271(0.004)
(40, Ising)	unweighted	1.19(0.298)	17.3(0.528)	/
	oracle	0.568(0.278)	0.666(0.331)	/
	GCP	0.870(0.597)	1.24(0.840)	1.81(0.621)
	Tucker	0.504(0.241)	1.80(0.653)	0.444(0.010)
	RGrad	0.713(0.377)	1.13(1.21)	**0.341**(**0.304**)
(60, Ising)	unweighted	1.35(0.243)	17.2(0.310)	/
	oracle	0.302(0.136)	0.370(0.242)	/
	GCP	0.349(0.181)	0.638(0.506)	1.59(1.16)
	Tucker	0.329(0.216)	2.03(0.368)	0.404(0.007)
	RGrad	0.356(0.154)	0.580(0.339)	**0.224**(**0.003**)

Table S.3: Method comparisons of different conformal prediction methods with $r = 3$. The results include the average mis-coverage % defined in (23) under the constant (const.) and adversarial (adv.) noise regimes as well as the relatively squared error (RSE) of the estimator $\widehat{\boldsymbol{\mathcal{B}}}$.

To exhibit the impact of model misspecification on the conformal inference, we generate the data tensor following $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}^* + \boldsymbol{\mathcal{E}}$, with $\boldsymbol{\mathcal{X}}^*$ having a Tucker rank at $(5, 5, 5)$ and $\boldsymbol{\mathcal{E}}$ following either constant or adversarial noise (see Section 4.2 for detailed definition). Then we run a tensor completion algorithm with a working rank of $(k, k, k), k \in [1, 15]$, and we compute the average coverage and confidence interval width for the 90% CI, where the CI is generated with our CTC algorithm with $\widehat{\boldsymbol{\mathcal{B}}}$ estimated using RGrad. We report the results under tensor dimension $d \in \{40, 60, 80, 100\}$ in Figure S.3.

In line with our expectations, we observe that the coverage probability remains well-calibrated at 90% across all simulation scenarios, regardless of the tensor completion rank. An exception is when missingness is correlated $g(x, y) = xy/15$ and when the tensor dimension is low at $d = 40$, which is not surprising given that the estimation error for $\widehat{\boldsymbol{\mathcal{B}}}$ is

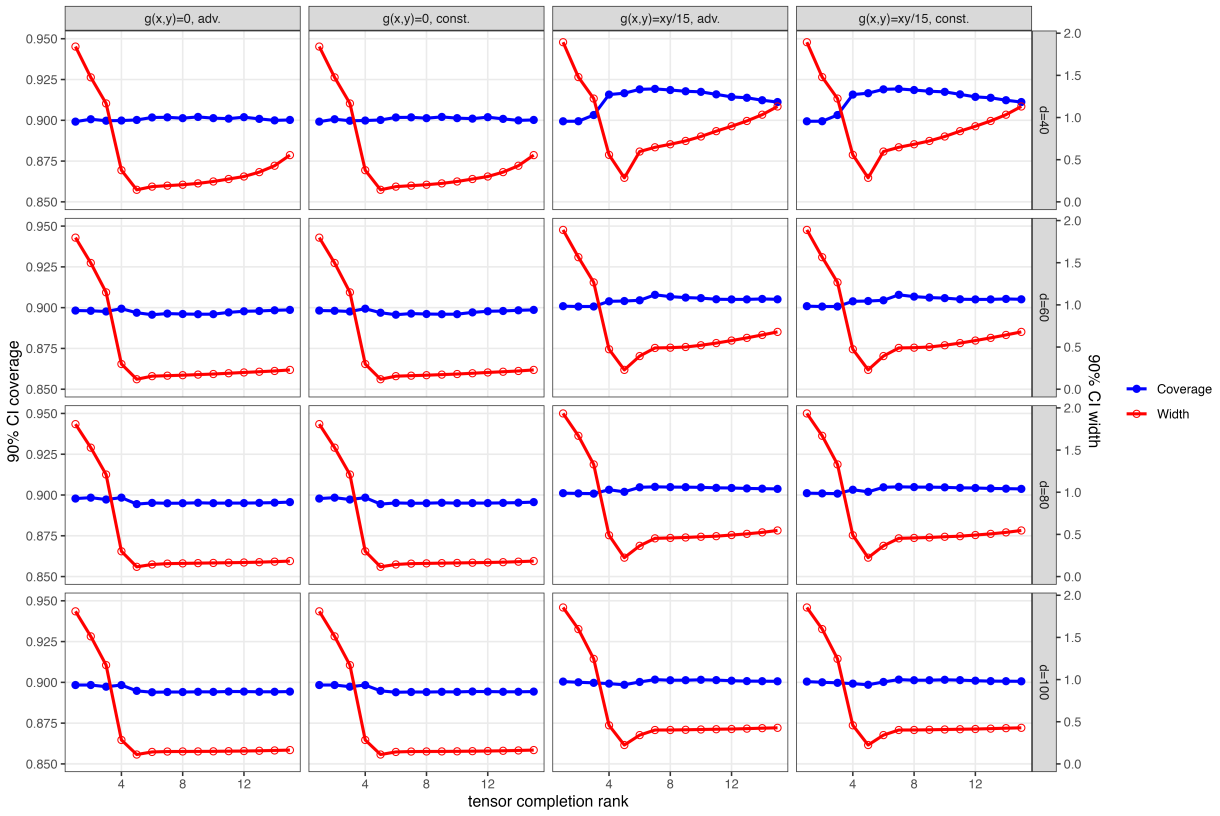

Figure S.3: CTC 90% confidence interval coverage and width against different tensor completion model ranks, under different tensor dimensions d, additive noise distribution (adv. = adversarial, const. = constant) and missingness correlations (as specified by $g(\cdot, \cdot)$). Results based on 30 repetitions, and $\boldsymbol{\mathcal{B}}$ has a fixed rank of $r = 3$ and $\widehat{\boldsymbol{\mathcal{B}}}$ estimated by Algorithm 2.

higher. Also, we notice that the CI width is optimal under the correct rank ($k = 5$), and becomes slightly worse when one overspecifies the model and significantly worse when one underspecifies the model. The conclusion is that our CTC algorithm has well-calibrated coverage probability and will have sub-optimal CI width when one misspecifies the tensor completion model. Given that the tensor completion model can properly select the rank using model selection criteria like AIC/BIC, we consider this a minor issue for our CTC algorithm.

C.5 Results on Other Non-conformity Scores

In this section, we expand the experiments conducted in Section 4.2 by considering other choices of the non-conformity scores. We primarily consider two additional types of scores. Firstly, the two-sided non-conformity score, which is basically $\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s) = \boldsymbol{\mathcal{X}}_s - \widehat{\boldsymbol{\mathcal{X}}}_s$. For any $s^* \in \mathbb{S}_{miss}$, similar to (5), we define $\widehat{q}_{s^*,l}$ and $\widehat{q}_{s^*,r}$ as:

$$\widehat{q}_{s^*,l} = \mathcal{Q}_{(1-\alpha)/2}\left(\sum_{i=1}^{n} \omega_i(s^*) \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})} + \omega_{n+1}(s^*) \cdot \delta_{+\infty}\right), \quad \text{where } \omega_k(s^*) = \frac{p_k(s^*)}{\sum_{i=1}^{n+1} p_i(s^*)},$$

$$\widehat{q}_{s^*,r} = \mathcal{Q}_{(1+\alpha)/2}\left(\sum_{i=1}^{n} \omega_i(s^*) \cdot \delta_{\mathcal{S}(\boldsymbol{\mathcal{X}}_{s_i}, \widehat{\boldsymbol{\mathcal{X}}}_{s_i})} + \omega_{n+1}(s^*) \cdot \delta_{+\infty}\right), \quad \text{where } \omega_k(s^*) = \frac{p_k(s^*)}{\sum_{i=1}^{n+1} p_i(s^*)},$$

and construct the $(1-\alpha)$-level conformal interval as $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) = \{x \in \mathbb{R} | \widehat{q}_{s^*,l} \leq \mathcal{S}(x, \widehat{\boldsymbol{\mathcal{X}}}_{s^*}) \leq \widehat{q}_{s^*,r}\}$.

The second type of score is the normalized non-conformity score, which is defined as $\mathcal{S}(\boldsymbol{\mathcal{X}}_s, \widehat{\boldsymbol{\mathcal{X}}}_s) = |\boldsymbol{\mathcal{X}}_s - \widehat{\boldsymbol{\mathcal{X}}}_s|/\widehat{u}_s$, where $\widehat{u}_s$ is a prior estimate of the uncertainty at s. The $\widehat{u}_s$ can be based on the uncertainty quantification of tensor completion estimators. Typically for matrix/tensor completion work with uncertainty quantification (Chen, Fan, Ma & Yan 2019, Farias et al. 2022, Gui et al. 2023, Ma & Xia 2024), all entries are assumed to be missing independently with the same probability, which makes it harder to directly utilize the asymptotic normality of the estimator for quantifying $\widehat{u}_s$ under our context. In this section, we use the entrywise confidence interval for tensor completion with low Tucker rank, which is also the completion algorithm we use, from Ma & Xia (2024) (see Corollary 1). For each $\widehat{u}_s$, it is proportional to $\|\mathcal{P}_{\mathbb{T}}(\boldsymbol{\mathcal{I}}_s)\|_{\mathrm{F}}$, where $\mathbb{T}$ is the tangent space of $\widehat{\boldsymbol{\mathcal{X}}}$ and $\boldsymbol{\mathcal{I}}_s$ is a binary tensor with all but one entry being 0, and entry s being 1. This is essentially a misspecified model since our data are not missing uniformly at random. The $(1-\alpha)$-level conformal interval is constructed via $C_{1-\alpha,s^*}(\widehat{\boldsymbol{\mathcal{X}}}) = \{x \in \mathbb{R} | \left|\boldsymbol{\mathcal{X}}_{s^*} - \widehat{\boldsymbol{\mathcal{X}}}_{s^*}\right| \leq \widehat{q}_{s^*} \cdot \widehat{u}_{s^*}\}$, where $\widehat{q}_{s^*}$ is defined similarly as (5), with the non-conformity score being the normalized score.

We re-run the experiment in Section 4.2 and summarize the average mis-coverage%,

as defined in (23), in Table S.4. From Table S.4, we saw generally no significant gain in using either score. Given that the prior uncertainty estimates $\widehat{u}_s$ are under a misspecified model, we even see some worse coverage when using the normalized score. In Figure S.4, we also plotted the average width of the 90% confidence interval under un-normalized, normalized, and two-sided non-conformity scores. The impact of misspecifying the prior uncertainty estimate is making the confidence interval wider. Overall, our results suggest that the miscoverage is mild under any of these non-conformity score definitions, which also showcases the robustness of our approach.

Model	d	Unweighted	Oracle	RGrad	RGrad Normalized	RGrad Two-sided
Constant Noise						
Bernoulli	d=40	0.46 (0.24)	0.38 (0.21)	0.38 (0.23)	0.44 (0.25)	0.38 (0.23)
	d=60	0.4 (0.17)	0.2 (0.08)	0.2 (0.08)	0.21 (0.11)	0.2 (0.08)
	d=80	0.44 (0.1)	0.12 (0.06)	0.12 (0.06)	0.12 (0.06)	0.12 (0.06)
	d=100	0.37 (0.08)	0.09 (0.06)	0.09 (0.06)	0.09 (0.04)	0.09 (0.06)
Ising	d=40	1.19 (0.3)	0.57 (0.28)	0.71 (0.38)	0.82 (0.41)	0.71 (0.38)
	d=60	1.35 (0.24)	0.3 (0.14)	0.36 (0.15)	0.36 (0.17)	0.36 (0.15)
	d=80	1.4 (0.11)	0.22 (0.12)	0.21 (0.15)	0.23 (0.14)	0.21 (0.15)
	d=100	1.26 (0.09)	0.12 (0.07)	0.12 (0.07)	0.16 (0.11)	0.12 (0.07)
Adversarial Noise						
Bernoulli	d=40	11.12 (0.39)	0.41 (0.23)	0.77 (0.4)	0.73 (0.46)	0.79 (0.41)
	d=60	11.05 (0.24)	0.21 (0.1)	0.82 (0.23)	0.72 (0.24)	0.83 (0.23)
	d=80	11.24 (0.14)	0.14 (0.07)	0.89 (0.15)	0.82 (0.16)	0.89 (0.15)
	d=100	10.81 (0.12)	0.1 (0.07)	0.81 (0.12)	0.74 (0.1)	0.81 (0.12)
Ising	d=40	17.31 (0.53)	0.67 (0.33)	1.13 (1.21)	1.17 (0.83)	1.06 (1.21)
	d=60	17.23 (0.31)	0.37 (0.24)	0.58 (0.34)	0.51 (0.28)	0.59 (0.35)
	d=80	17.61 (0.2)	0.25 (0.14)	0.5 (0.26)	0.43 (0.27)	0.51 (0.25)
	d=100	17.08 (0.12)	0.17 (0.12)	0.53 (0.2)	0.49 (0.19)	0.53 (0.19)

Table S.4: Average mis-coverage% (and standard deviation, also in percentage) for unweighted, oracle, RGrad, RGrad with normalized non-conformity score, and RGrad with two-sided non-conformity score. We set $r = 3$ in all cases, and the results are based on 30 iterations.

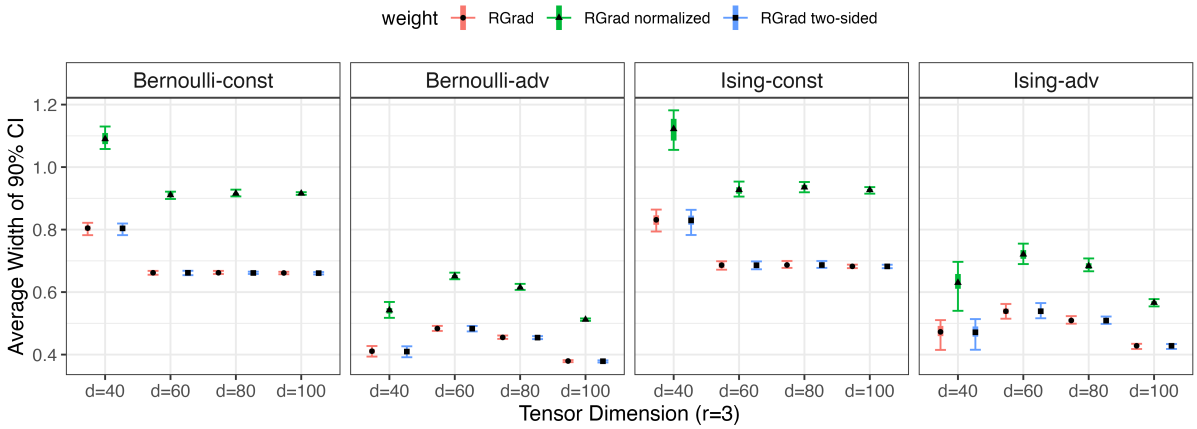

Figure S.4: The average width of the 90% conformal interval of conformal prediction methods under three different non-conformity scores, with $d \in \{40, 60, 80, 100\}$, $r = 3$ under the Bernoulli and Ising model. Two uncertainty regimes: constant noise (const) and adversarial noise (adv) are considered. Results are based on 30 repetitions, error bars show the $2.5\%, 97.5\%$ quantiles, and the thicker lines show the range of 25% to 75% quantiles.

C.6 Algorithm Stepsize Selection, Runtime and Convergence

In this subsection, we further explore the convergence, running time, and step size choice of the Riemannian Gradient descent algorithm. In Algorithm 2, we set a fixed step size $\eta = 0.1$, which can be quite restrictive. Here, we explore an alternative step size scheme based on linearized line search. Specifically, at iteration $(l+1)$, we start with a relatively large stepsize η', and check if the following Armijo condition holds for a prespecified hyper-parameter α:

$$\ell\left(\boldsymbol{\mathcal{W}}_{tr}|\boldsymbol{\mathcal{B}}_l\right) - \ell\left(\boldsymbol{\mathcal{W}}_{tr}|\mathrm{SVD}^{\mathrm{tt}}_{\mathbf{r}}(\boldsymbol{\mathcal{B}}_l - \eta'\mathcal{P}_{\mathbb{T}_l}(\boldsymbol{\mathcal{G}}_l))\right) \geq \alpha \cdot \eta' \|\mathcal{P}_{\mathbb{T}_l}(\boldsymbol{\mathcal{G}}_l)\|_{\mathrm{F}}^2. \tag{S.19}$$

If it does not hold, we set $\eta' \leftarrow \eta'/2$ and continue checking. Basically, we are selecting a step size that reaches sufficient descent of the target negative pseudo-likelihood $\ell(\cdot)$. In Figure S.5, we compare the runtime per iteration of our Algorithm 2 with fixed $\eta = 0.1$ (RGrad), Algorithm 2 with adaptive step size (RGrad-adaptive), and the binary tensor decomposition with generalized CP-decomposition (Wang & Li 2020, Hong et al. 2020) (GCP), where the GCP is essentially based on a gradient descent algorithm. For GCP, we use the MATLAB Tensor Toolbox for implementation and set the CP rank such that the

number of parameters is roughly the same as our tensor-train decomposition parameter.

We see that the RGrad-based algorithm is much faster than the GCP per iteration. It is also our empirical observation that RGrad-adaptive requires a much smaller number of iterations on average than the other two algorithms, indicating the potential of further speeding up our algorithm.

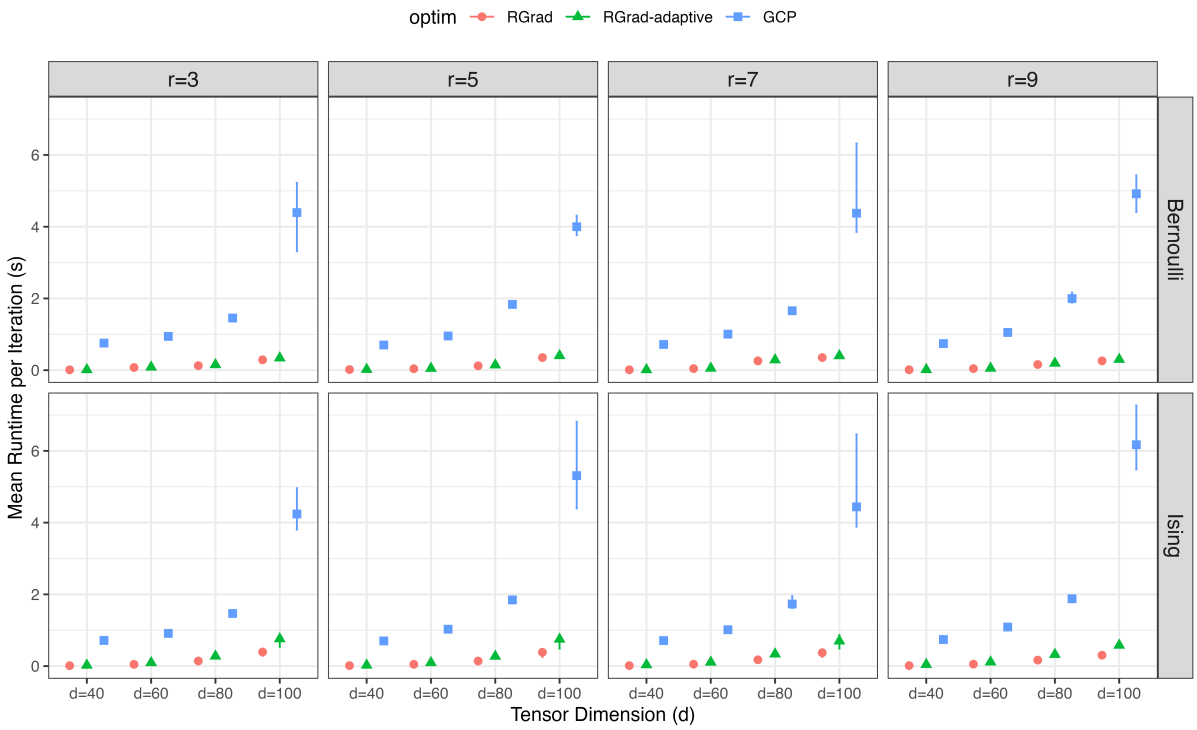

Figure S.5: Runtime in seconds per iteration comparison. All three algorithms are run on a 2x 3.0 GHz Intel Xeon Gold 6154 CPU. Errorbar show the 2.5% and 97.5% quantile across 30 iterations.

We further check the convergence of our RGrad algorithm and the robustness of our initialization method. For convergence, we fit our RGrad under $d \in \{40, 60, 80, 100\}$ with $r = 3$, and measure the relative squared error (RSE) of $\widehat{\boldsymbol{\mathcal{B}}}$ across both Bernoulli and Ising models. The RSE is simply $\|\widehat{\boldsymbol{\mathcal{B}}} - \boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}/\|\boldsymbol{\mathcal{B}}^*\|_{\mathrm{F}}$. We show the result in the left panel of Figure S.6. We generally see that the higher the tensor dimension, the lower the error is. There is an error bound for the RSE, and generally, one requires a higher dimension relative to rank to converge to the true tensor $\boldsymbol{\mathcal{B}}^*$. In the middle and right panel of Figure S.6, we

plot the initialization RSE and final RSE of the RGrad algorithm under various standard deviations of the error $\boldsymbol{\mathcal{E}}$ that we apply to the binary tensor $\boldsymbol{\mathcal{W}}$ for the spectral initialization. We found that regardless of the initialization error, the final error is the same, and thus we could simply do a low-rank TT-SVD over $\boldsymbol{\mathcal{W}}$ as initialization.

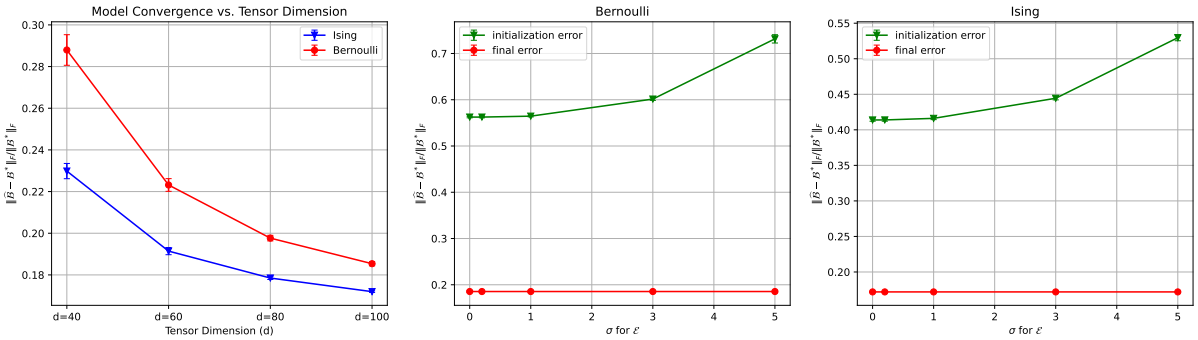

Figure S.6: Left: Final estimator RSE vs. Tensor Dimension for RGrad. Middle & Right: Initialization and Final RSE of the estimator across various levels of random perturbation of the binary tensor during spectral initialization. All results based on 30 iterations and confidence bands are ± 1.96 std.