The Annals of Statistics
2024, Vol. 52, No. 6, 2583–2612
https://doi.org/10.1214/24-AOS2396

TENSOR-ON-TENSOR REGRESSION: RIEMANNIAN OPTIMIZATION, OVER-PARAMETERIZATION, STATISTICAL-COMPUTATIONAL GAP AND THEIR INTERPLAY

BY YUETIAN LUO[1,a] AND ANRU R. ZHANG[2,b]

[1]*Data Science Institute, University of Chicago,* [a]*yuetian@uchicago.edu*
[2]*Department of Biostatistics & Bioinformatics and Department of Computer Science, Duke University,* [b]*anru.zhang@duke.edu*

We study the tensor-on-tensor regression, where the goal is to connect tensor responses to tensor covariates with a low Tucker rank parameter tensor/matrix without prior knowledge of its intrinsic rank. We propose the Riemannian gradient descent (RGD) and Riemannian Gauss–Newton (RGN) methods and cope with the challenge of unknown rank by studying the effect of rank over-parameterization. We provide the first convergence guarantee for the general tensor-on-tensor regression by showing that RGD and RGN respectively converge linearly and quadratically to a statistically optimal estimate in both rank correctly-parameterized and over-parameterized settings. Our theory reveals an intriguing phenomenon: Riemannian optimization methods naturally adapt to over-parameterization without modifications to their implementation. We also prove the statistical-computational gap in scalar-on-tensor regression by a direct low-degree polynomial argument. Our theory demonstrates a "blessing of statistical-computational gap" phenomenon: in a wide range of scenarios in tensor-on-tensor regression for tensors of order three or higher, the computationally required sample size matches what is needed by moderate rank over-parameterization when considering computationally feasible estimators, while there are no such benefits in the matrix settings. This shows moderate rank over-parameterization is essentially "cost-free" in terms of sample size in tensor-on-tensor regression of order three or higher. Finally, we conduct simulation studies to show the advantages of our proposed methods and to corroborate our theoretical findings.

1. Introduction. The analysis of tensor or multiway array data has emerged as a very active topic of research in statistics, applied mathematics, machine learning, and signal processing (Kolda and Bader (2009)), along with many important applications, such as neuroimaging analysis (Zhou, Li and Zhu (2013)), latent variable models (Anandkumar et al. (2014)) and collaborative filtering (Bi, Qu and Shen (2018)). This paper studies a general class of problems termed *tensor-on-tensor regression*, which aims to characterize the relationship between covariates and responses in the form of scalars, vectors, matrices, or high-order tensors:

$$\boldsymbol{\mathcal{Y}}_i = \langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* \rangle_* + \boldsymbol{\mathcal{E}}_i, \quad i = 1, \ldots, n. \tag{1}$$

Here, $\boldsymbol{\mathcal{A}}_i \in \mathbb{R}^{p_1 \times \cdots \times p_d}$, $i = 1, \ldots, n$ are the known order-d (or d-way) tensor covariates. $\boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{E}}_i \in \mathbb{R}^{p_{d+1} \times \cdots \times p_{d+m}}$ are both order-m tensors and are observations and unknown noise, respectively. $\boldsymbol{\mathcal{X}}^* \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_{d+1} \times \cdots \times p_{d+m}}$ is an order-$(d+m)$ tensor parameter of interest. $\langle \cdot, \cdot \rangle_*$ is the contracted tensor inner product defined as $\langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* \rangle_* \in \mathbb{R}^{p_{d+1} \times \cdots \times p_{d+m}}$,

$$\big(\langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* \rangle_*\big)_{[j_1, \ldots, j_m]} = \sum_{\substack{k_l = 1, \\ l = 1, \ldots, d}}^{p_l} \boldsymbol{\mathcal{A}}_{i[k_1, \ldots, k_d]} \boldsymbol{\mathcal{X}}^*_{[k_1, \ldots, k_d, j_1, \ldots, j_m]}.$$

Received November 2022; revised January 2024.
MSC2020 subject classifications. Primary 62H15; secondary 62C20.
Key words and phrases. Tensor-on-tensor regression, over-parameterization, Riemannian optimization, statistical-computational gaps, low-degree polynomials.

Throughout the paper, we consider d and m to be fixed constants. We also stack all responses and errors to $\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{E}} \in \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}}$, where $\boldsymbol{\mathcal{Y}}_{[i,:,\ldots,:]} = \boldsymbol{\mathcal{Y}}_i$ and $\boldsymbol{\mathcal{E}}_{[i,:,\ldots,:]} = \boldsymbol{\mathcal{E}}_i$. Then the tensor-on-tensor regression model can be written succinctly as $\boldsymbol{\mathcal{Y}} = \mathscr{A}(\boldsymbol{\mathcal{X}}^*) + \boldsymbol{\mathcal{E}}$, where $\mathscr{A}: \mathbb{R}^{p_1\times\cdots\times p_{d+m}} \to \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}}$ is a linear map such that

$$\mathscr{A}(\boldsymbol{\mathcal{X}}^*)_{[i,:,\ldots,:]} = \langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^*\rangle_* \quad \text{for } i = 1,\ldots,n. \tag{2}$$

Our goal is to estimate $\boldsymbol{\mathcal{X}}^*$ based on $(\boldsymbol{\mathcal{Y}}, \mathscr{A})$.

Tensor-on-tensor regression model was proposed and studied in Raskutti, Yuan and Chen (2019), Lock (2018). The generic tensor-on-tensor regression covers many special tensor regression models in the literature, such as:

- scalar-on-tensor regression (Zhou, Li and Zhu (2013), Mu et al. (2014)): $m = 0$;
- tensor-on-vector regression (Li and Zhang (2017), Sun and Li (2017)): $d = 1$;
- scalar-on-matrix regression (or matrix trace regression) (Recht, Fazel and Parrilo (2010)): $m = 0$, $d = 2$.

There is a great surge of interest in tensor-on-tensor regression for its applications (Lock (2018), Gahrooei et al. (2021), Llosa and Maitra (2022)). Specific examples include:

- *Neuroimaging data analysis*. Studies in neuroscience are greatly facilitated by a variety of neuroimaging technologies. Tensor-on-tensor regression provides interpretable analysis of such datasets (Zhou, Li and Zhu (2013), Li and Zhang (2017)). For example, tensor-on-vector regression has been applied to compare MRI scans across different autism spectrum disorder groups (Sun and Li (2017)), which has helped evaluate the effectiveness of a potential drug. Scalar-on-tensor regression has been used to predict neurological diseases, such as attention deficit hyperactivity disorder, and reveal regions of interest in the brain that affect the progression of diseases (Zhou, Li and Zhu (2013)).
- *Facial image data analysis*. Attributes prediction from facial images is popular in social data analysis. Oftentimes, each facial image is labeled only with the name of the individual, often a celebrity, while people are interested in inferring more features from that. Tensor-on-tensor regression and tensor-variate analysis of variance have been proposed to predict describable attributes from a facial image (Lock (2018)) and distinguish facial characteristics related to ethnic origin, age group and gender (Llosa and Maitra (2022)).
- *Longitudinal relational data analysis*. Longitudinal relational data among a set of objects can be represented as a time series of matrices, where each entry of the matrices represents a directed relationship involving pairs of objects at a given time. The relation between one pair of objects may have an effect on the relation between members of another pair, an effective tensor-on-tensor regression model has been developed to estimate such effects (Hoff (2015)).

Meanwhile, tensor datasets are often high-dimensional, that is, the ambient data dimension is substantially bigger than the sample size. It is thus crucial to exploit the hidden low-dimensional structures from the datasets to facilitate the follow-up analyses. In tensor data analysis, low-rankness is among the most commonly considered structural assumptions. In this paper, we assume the target parameter $\boldsymbol{\mathcal{X}}^*$ has an intrinsic low Tucker (or multilinear) rank $\mathbf{r}^* = (r_1^*,\ldots,r_d^*,r_{d+1}^*,\ldots,r_{d+m}^*)$, that is, all fibers[1] of $\boldsymbol{\mathcal{X}}^*$ along mode-k lie in a r_k^*-dimensional subspace of $\mathbb{R}^{p_k}$ for $k = 1,\ldots,d+m$.

[1]Fibers are bar-shaped vectors and are counterpart of matrix columns and rows in a tensors (Kolda and Bader (2009)).

1.1. *Central questions*. A natural question on low-rank tensor-on-tensor regression is:

QUESTION 1. Can we develop fast and statistically optimal solutions for the general low-rank tensor-on-tensor regression?

Various algorithms were proposed in the literature to solve specific instances of tensor-on-tensor regression with provable guarantees, such as variants of gradient descent methods (Rauhut, Schneider and Stojanac (2017), Yu and Liu (2016), Chen, Raskutti and Yuan (2019), Ahmed, Raja and Bajwa (2020), Han, Willett and Zhang (2022), Hao, Zhang and Cheng (2020), Tong et al. (2022)), alternating minimization (Zhou, Li and Zhu (2013)), Bayesian Markov chain Monte Carlo (Guhaniyogi, Qamar and Dunson (2017)) and Riemannian optimization methods (Kressner, Steinlechner and Vandereycken (2016), Luo and Zhang (2023)) for scalar-on-tensor regression; regularized rank constrained least squares (Rabusseau and Kadri (2016)), alternating minimization (Sun and Li (2017)) and envelope method (Li and Zhang (2017)) for tensor-on-vector regression. The theoretical guarantees of these methods were developed case-by-case under the assumption that the intrinsic tensor rank is known. In addition, Hoff (2015) proposed a Bayesian approach to solve the tensor-on-tensor regression when the mode numbers of the predictor and the response are equal. Lock (2018), Liu, Liu and Zhu (2020) proposed alternating least squares procedures for solving the general tensor-on-tensor regression, and a numerical study on the effect of rank misspecification was performed in Lock (2018) without theoretical exploration. Asymptotic analysis for the computationally intensive maximum likelihood estimator is provided in Llosa and Maitra (2022) for different low-rank tensor formats with known intrinsic ranks. The convex relaxation methods for tensor-on-tensor regression, including the computationally infeasible tensor nuclear norm relaxation, were studied in Raskutti, Yuan and Chen (2019). In summary, despite a great amount of effort in the literature, a general, fast and statistically optimal framework for tensor-on-tensor regression is still underdeveloped.

Moreover, the intrinsic rank $\mathbf{r}^*$ is usually unknown in practice, while tuning rank is even more challenging for tensors than matrices as $(d+m)$ parameter values need to be tuned simultaneously. Thus, an important question is:

QUESTION 2. Can we solve tensor-on-tensor regression robustly without knowing the intrinsic rank?

To this end, we adopt a rank over-parameterization scheme: we introduce a conservative guess of rank $\mathbf{r} := (r_1, \ldots, r_{d+m}) \geq (r_1^*, \ldots, r_{d+m}^*)$ and solve the following tensor-on-tensor regression under the possibly over-parameterized regime:

$$\widehat{\boldsymbol{\mathcal{X}}}_{\text{opt}} = \underset{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}}{\arg\min} \; f(\boldsymbol{\mathcal{X}}) := \frac{1}{2} \|\boldsymbol{\mathcal{Y}} - \mathscr{A}(\boldsymbol{\mathcal{X}})\|_{\text{F}}^2, \quad \text{subject to } \text{Tucrank}(\boldsymbol{\mathcal{X}}) \leq \mathbf{r}. \tag{3}$$

Here, $\text{Tucrank}(\boldsymbol{\mathcal{X}})$ is the Tucker rank of $\boldsymbol{\mathcal{X}}$ (see formal definition in the *notation and preliminaries* section). In most of the aforementioned literature, the ranks were assumed to be correctly specified and the results do not directly apply to the possibly over-parameterized scenario in (3). We will illustrate later that Riemannian optimization is an ideal scheme to treat rank-constrained optimization like (3). However, under the over-parameterized regime, the classic convergence theory of Riemannian optimization does not apply since the true parameter $\boldsymbol{\mathcal{X}}^*$ is merely a boundary point of the Riemannian manifold consisting of tensors with incorrectly specified rank.

In addition, tensor problems often exhibit statistical-computational gaps (Hillar and Lim (2013), Richard and Montanari (2014)). For example, in scalar-on-tensor regression, that is, $m = 0$, and suppose $p_1 = \cdots = p_d = p$ and $r_1^* = \cdots = r_d^* = r^*$ is known and the design is Gaussian ensemble (to be formally introduced in Section 3), it has been shown that rank minimization recovers $\boldsymbol{\mathcal{X}}^*$ with $\Omega(pr^* + r^{*d})$ samples (Mu et al. (2014)); but the rank minimization is generally NP-hard to compute (Hillar and Lim (2013)). On the other hand, all existing polynomial-time algorithms require at least $\Omega(p^{d/2}r^* + r^{*d})$ samples to guarantee recovery (Han, Willett and Zhang (2022)). So when $d \geq 3$, there exists a significant gap on the sample complexities between what can be achieved information theoretically and by existing polynomial-time algorithms. Xia, Zhang and Zhou (2022) leveraged this hypothetical gap to claim there is no need to debias in scalar-on-tensor regression inference. Intriguingly, this gap seems to close when $d = 2$, that is, in the matrix case, since $p^{d/2}r^* + r^{*d} = pr^* + r^{*d}$. So we ask:

QUESTION 3. Is there a statistical-computational gap in tensor-on-tensor regression? What is the difference between tensor and matrix settings?

In the era of big data, Riemannian optimization and over-parameterization have become a common remedy for nonconvexity in high-dimensional statistics and machine learning, where the statistical-computational gap is a prevalent phenomenon. As these ingredients nicely gather in tensor-on-tensor regression, a more open-ended question is:

QUESTION 4. Is there any interplay among Riemannian optimization, over-parameterization and statistical-computational gap?

1.2. *Our contributions*. We aim to answer the four questions above. Our specific contributions include:

(*Over-parameterization, algorithms, convergence theory and statistical optimality*). We address the unknown intrinsic rank through the rank over-parameterization scheme in (3). We introduce the Riemannian gradient descent (RGD) and Riemannian Gauss–Newton (RGN) algorithms for tensor-on-tensor regression and develop the corresponding convergence guarantees. We specifically show with proper initialization, RGD and RGN respectively converge linearly and quadratically to the true parameter $\boldsymbol{\mathcal{X}}^*$ up to some statistical error. Especially in the noiseless setting, that is, $\boldsymbol{\mathcal{E}} = 0$, RGD and RGN respectively converge linearly and quadratically to the exact parameter $\boldsymbol{\mathcal{X}}^*$. Our convergence theory for over-parameterized Riemannian optimization algorithms is novel, covers the rank under-parameterized cases as well, and cannot be inferred from the standard convergence theories in the Riemannian optimization literature, since the true parameter $\boldsymbol{\mathcal{X}}^*$ only lies on the boundary of the working Riemannian manifold consisting of tensors with incorrectly specified rank. We further show the estimation error achieved by RGD and RGN matches the minimax risk lower bound under the Gaussian ensemble design. To our best knowledge, this is the first algorithmic convergence result for tensor-on-tensor regression with optimal statistical error guarantees. In the specific over-parameterized matrix trace regression setting, our results yield the first linear/quadratic convergence guarantee for RGD/RGN. Compared to the existing results on factorized GD in the over-parameterized matrix trace regression (Zhuo et al. (2024), Zhang, Fattahi and Zhang (2021)), our second-order algorithm RGN and the corresponding theory are novel, which improve the results in literature in many ways.

Our convergence theory reveals an intriguing phenomenon: in tensor-on-tensor regression, *Riemannian optimization algorithms adapt to over-parameterized scenarios without modifications.* This is significantly different from the classic factorized gradient descent algorithm

TABLE 1
Riemannian gradient descent (RGD), Riemannian Gauss–Newton (RGN) versus factorized gradient descent (Factorized GD), preconditioned factorized GD for over-parameterized matrix trace regression

Algorithm	Statistical error rate	Convergence rate	Require tuning	Parameter matrix type
RGD (this work)	optimal	linear	no	general
RGN (this work)	optimal	quadratic	no	general
Factorized GD (Zhuo et al. (2024))	optimal	sublinear	yes	PSD
Preconditioned Factorized GD (Zhang, Fattahi and Zhang (2021))	suboptimal	linear	yes	PSD

where preconditioning is needed. Table 1 compares our results with the existing ones on over-parameterized matrix trace regression.

Although developing proper initialization for all cases of tensor-on-tensor regression is difficult, we introduce spectral methods that yield adequate initializations for both RGD and RGN in four prominent instances, *scalar-on-tensor regression*, *tensor-on-vector regression*, *matrix trace regression*, and *rank*-1 *tensor-on-tensor regression* under Gaussian ensemble design.

(*Statistical-computational gap and sample size requirement*). In this paper, we establish rigorous evidence on the statistical-computational gap in scalar-on-tensor regression via low-degree polynomials methods. Our argument shows $n = \Omega(p^{d/2})$ samples are necessary for any polynomial-time method to succeed. Existing hardness evidence from low-degree polynomials is often established for statistical problems with the simple "signal + noise" structure. Such a structure enables the decoupling of signal and noise that simplifies the analysis. To our best knowledge, our low-degree hardness evidence is the first one for problems with complex correlated structures.

Based on the computational lower bounds and algorithmic upper bounds developed in this paper, we draw Figure 1 to illustrate the sample size requirements in over-parameterized matrix trace regression with $d = 2$ (Panel (a)) and scalar-on-tensor regression, a prominent instance of tensor-on-tensor regression, with $d \geq 3$ (Panel (b)). When the input rank r is greater than $\sqrt{p}$, that is, in the heavily over-parameterized regime, we show that an extra sample complexity is needed for RGD and RGN to converge in both regressions. When the input rank r is between r^* and $\sqrt{p}$, that is, in the moderately over-parameterized regime,

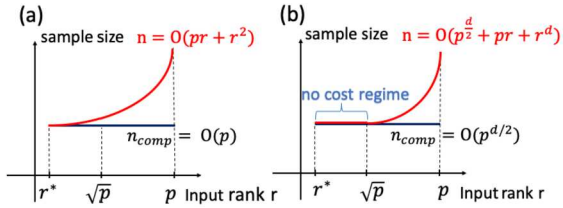


FIG. 1. *Comparison of sample size requirements in over-parameterized matrix trace (Panel (a)) and scalar-on-tensor regressions (Panel (b)) under Gaussian ensemble design. Here the red line denotes the sample size (n) requirements for the RGD and RGN to succeed with input rank r and spectral initialization and the black line (n_{comp}) is the sample complexity of the computational limit, that is, the minimum sample size requirement for any efficient algorithms. For simplicity, we assume $p_1 = \cdots = p_d = p$, $r_1 = \cdots = r_d = r$, $r_1^* = \cdots = r_d^* = r^*$, d and r^* are some fixed constants, $\boldsymbol{\mathcal{E}} = 0$ and $\boldsymbol{\mathcal{X}}^*$ is well-conditioned.*

extra sample complexity is still required in matrix trace regression (Figure 1(a)). On the other hand, in scalar-on-tensor regression (Figure 1(b)), no larger sample size is required to account for the inflated input rank, as the red line is flat in the "no extra cost" regime in Figure 1(b).

This alludes to an important message, *moderate rank over-parameterization is cost-free in terms of sample size for a computationally feasible optimal estimator in scalar-on-tensor regression.* The computational barrier, although being a tough scenario and is often referred to as the "curse of computability," becomes a *"blessing"* to over-parameterization here, as no extra samples are required if this large but essential sample size condition is met to guarantee that the computationally feasible estimator is achievable!

(*New technical tools*). We introduce a series of technical tools for theory development in this paper, including a tangent space projection error bound, a tensor decomposition perturbation bound under the over-parameterized setting, and a simple formula for computing expected values of Hermite polynomials on correlated multivariate Gaussian random variables while developing low-degree polynomials lower bounds. See Section 6 for a summary of our technical contributions.

(*Implementation details and numerical experiments*). Finally, we discuss the implementation details of RGD and RGN for tensor-on-tensor regression in Section 7. We specifically find a reduction from computing RGN update to solving $(m + 1)$ separate least squares. This reduction yields a fast implementation of RGN. We conduct numerical studies to show the convergence and required sample size of our proposed algorithms match our theoretical findings. We also compare the numerical performance of our algorithms with existing ones. The results show the proposed algorithms have significant advantages in both rank correctly-specified and overspecified tensor-on-tensor regression.

1.3. *Related prior work.* This work is related to several lines of research on over-parameterization, Riemannian optimization and computational barriers in tensor problems.

First, over-parameterization has attracted much attention in modern data science due to the great success of deep learning. The concept of over-parameterization generally refers to the scenario when learning problems include more model parameters than necessary. Recent studies show that over-parameterization brings both computational and statistical benefits when solving complex problems (Soltanolkotabi, Javanmard and Lee (2019), Bartlett et al. (2020), Belkin et al. (2019)). There is a vast amount of literature on studying the role of over-parameterization to demystify deep learning (Bartlett, Montanari and Rakhlin (2021), Belkin (2021)). This paper focuses on the effect of over-parameterization specifically in the rank-constrained tensor-on-tensor regression problem. In particular, we consider a special type of over-parameterization where the input rank to the model is overspecified.

Second, Riemannian manifold optimization methods have been powerful in solving optimization problems with geometric constraints (Absil, Mahony and Sepulchre (2008)). Many progress in this topic were made for the low-rank matrix estimation (Keshavan, Montanari and Oh (2010), Boumal and Absil (2011), Wei et al. (2016), Meyer, Bonnabel and Sepulchre (2011), Mishra et al. (2014), Vandereycken (2013), Huang and Hand (2018), Luo et al. (2024), Hou, Li and Zhang (2020)). Moreover, Riemannian manifold optimization methods under various Riemannian geometries have been explored in many tensor problems, such as tensor decomposition (Eldén and Savas (2009), Savas and Lim (2010), Ishteva et al. (2009), Breiding and Vannieuwenhoven (2018)), scalar-on-tensor regression (Kressner, Steinlechner and Vandereycken (2016), Luo and Zhang (2023)), tensor completion (Kasai and Mishra (2016), Dong et al. (2022), Kressner, Steinlechner and Vandereycken (2014), Heidel and Schulz (2018), Xia and Yuan (2019), Steinlechner (2016), Wang, Chen and Wei (2023), Cai, Li and Xia (2022)), and robust tensor PCA (Cai, Li and Xia (2023)).

Third, many high-dimensional tensor problems exhibit the statistical-computational gaps, that is, the gap between different signal-to-noise ratio thresholds that make the problem information-theoretically solvable versus polynomial-time solvable. Rigorous evidence for such gaps has been provided to tensor completion (Barak and Moitra (2016)), tensor PCA/SVD (Zhang and Xia (2018), Brennan and Bresler (2020), Dudeja and Hsu (2021), Choo and d'Orsi (2021)), tensor clustering (Luo and Zhang (2022), Han et al. (2022)) and tensor-on-tensor association detection (Diakonikolas et al. (2023)). This work provides a rigorous piece of evidence for the statistical-computational gap in scalar-on-tensor regression under the low-degree polynomials framework.

Finally, a special case of our setting, over-parameterized matrix trace regression, has attracted much attention recently. The results along this line include two categories: (1) $r \geq r^*$ and $n = O((p_1 + p_2)r)$: the problem is over-parameterized and identifiable Zhuo et al. (2024), Zhang, Fattahi and Zhang (2021), Ding et al. (2021a); (2) $r \geq r^*$ and $n = O((p_1 + p_2)r^*)$: as the sample size is smaller than the number of free parameters in the model, there can be infinitely many solutions to (3) and the model is unidentifiable. One important finding in Category (2) is that with small magnitude initialization, vanilla gradient descent under the factorization formulation tends to implicitly bias towards a low-rank solution (Gunasekar et al. (2017), Li, Ma and Zhang (2018), Li, Luo and Lyu (2020), Fan, Yang and Yu (2023), Stöger and Soltanolkotabi (2021), Ma and Fattahi (2023), Jiang, Chen and Ding (2023)). Our work provides a unified simple Riemannian optimization framework to solve the general tensor-on-tensor regression problem under the setting in Category (1). The implication of our results in over-parameterized matrix trace regression is further discussed in Remarks 6 and 8.

1.4. *Organization of the paper.* After a brief introduction of notation and preliminaries in Section 1.5, we introduce our main algorithms, Riemannian gradient descent and Riemannian Gauss–Newton in Section 2. The convergence results of RGD and RGN in the general tensor-on-tensor regression and applications in specific examples are discussed in Sections 3 and 4, respectively. Computational limits are discussed in Section 5. Technical contributions are summarized in Section 6. Implementation details of RGD/RGN and numerical studies are presented in Sections 7 and 8, respectively. Conclusion and future work are given in Section 9. Additional algorithms, numerical studies and all technical proofs are collected in Supplements 1–10.

1.5. *Notation and preliminaries.* Let $[r] = \{1, \ldots, r\}$ for any positive integer r. Lowercase letters (e.g., a), lowercase boldface letters (e.g., $\mathbf{u}$), uppercase boldface letters (e.g., $\mathbf{U}$) and boldface calligraphic letters (e.g., $\boldsymbol{\mathcal{A}}$) denote scalars, vectors, matrices and order-3-or-higher tensors, respectively. We use bracket subscripts to denote subvectors, submatrices and subtensors. For any matrix $\mathbf{D} \in \mathbb{R}^{p_1 \times p_2}$, let $\sigma_k(\mathbf{D})$ be the kth largest singular value of $\mathbf{D}$. We also denote $\mathrm{SVD}_r(\mathbf{D}) = [\mathbf{u}_1 \cdots \mathbf{u}_r]$ and $\mathrm{QR}(\mathbf{D})$ as the subspace composed of the leading r left singular vectors and the Q part of the QR decomposition of $\mathbf{D}$, respectively. $\mathbf{I}_r$ represents the r-by-r identity matrix. Let $\mathbb{O}_{p,r} = \{\mathbf{U} \in \mathbb{R}^{p \times r} : \mathbf{U}^\top \mathbf{U} = \mathbf{I}_r\}$ and for any $\mathbf{U} \in \mathbb{O}_{p,r}$, denote $P_{\mathbf{U}} = \mathbf{U}\mathbf{U}^\top$. The matricization operation $\mathcal{M}_k(\cdot)$ unfolds an order-d tensor along mode k to a matrix, say $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ to $\mathcal{M}_k(\boldsymbol{\mathcal{A}}) \in \mathbb{R}^{p_k \times p_{-k}}$, where $p_{-k} = \prod_{j \neq k} p_j$ and its detailed definition is provided in Supplement 2. The Frobenius norm of tensor $\boldsymbol{\mathcal{A}}$ is defined as $\|\boldsymbol{\mathcal{A}}\|_{\mathrm{F}} = (\sum_{i_1,\ldots,i_d} \boldsymbol{\mathcal{A}}^2_{[i_1,\ldots,i_d]})^{1/2}$. The Tucker rank of an order-d tensor $\boldsymbol{\mathcal{A}}$, denoted by $\mathrm{Tucrank}(\boldsymbol{\mathcal{A}})$, is defined as a d-tuple $\mathbf{r} := (r_1, \ldots, r_d)$, where $r_k = \mathrm{rank}(\mathcal{M}_k(\boldsymbol{\mathcal{A}}))$. Any Tucker rank-$(r_1, \ldots, r_d)$ tensor $\boldsymbol{\mathcal{A}}$ admits the following Tucker decomposition (Tucker (1966)): $\boldsymbol{\mathcal{A}} = [\![\boldsymbol{\mathcal{S}}; \mathbf{U}_1, \ldots, \mathbf{U}_d]\!] := \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1 \times \cdots \times_d \mathbf{U}_d$, where

$\boldsymbol{\mathcal{S}} \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ is the core tensor and $\mathbf{U}_k = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{A}}))$ is the mode-k top r_k left singular vectors. Here, the mode-k product of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ with a matrix $\mathbf{B} \in \mathbb{R}^{r_k \times p_k}$, denoted by $\boldsymbol{\mathcal{A}} \times_k \mathbf{B}$, is a $p_1 \times \cdots \times p_{k-1} \times r_k \times p_{k+1} \times \cdots \times p_d$-dimensional tensor, and its definition is provided in Supplement 2. The following abbreviations are used to denote the tensor-matrix product along multiple modes: $\boldsymbol{\mathcal{A}} \times_{k=1}^{d} \mathbf{U}_k := \boldsymbol{\mathcal{A}} \times_1 \mathbf{U}_1 \times \cdots \times_d \mathbf{U}_d$; $\boldsymbol{\mathcal{A}} \times_{l \neq k} \mathbf{U}_l := \boldsymbol{\mathcal{A}} \times_1 \mathbf{U}_1 \times \cdots \times_{k-1} \mathbf{U}_{k-1} \times_{k+1} \mathbf{U}_{k+1} \times \cdots \times_d \mathbf{U}_d$. For any order-$d$ tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ and a d-tuple $\mathbf{r} = (r_1, \ldots, r_d)$, let $\boldsymbol{\mathcal{Z}}_{\max(\mathbf{r})} := \boldsymbol{\mathcal{Z}} \times_{k=1}^{d} P_{\widehat{\mathbf{U}}_k}$ be the best Tucker rank $\mathbf{r}$ approximation of $\boldsymbol{\mathcal{Z}}$ in terms of Frobenius norm, where $(\widehat{\mathbf{U}}_1, \ldots, \widehat{\mathbf{U}}_d)$ is the solution to $\arg\max_{\mathbf{U}_k \in \mathbb{O}_{p_k, r_k}, k=1,\ldots,d} \|\boldsymbol{\mathcal{Z}} \times_{k=1}^{d} P_{\mathbf{U}_k}\|_{\mathrm{F}}$ (De Lathauwer, De Moor and Vandewalle ((2000b), Theorem 4.2)). Throughout the paper, let $c(d)$ be a constant that depends on d only, whose actual value varies from line to line; $c_1(m)$, $c_2(d, m)$ are noted similarly. Finally, we denote $\mathscr{A}^*$ as the adjoint of the linear operator $\mathscr{A}$.

2. Riemannian optimization for tensor-on-tensor regression. Riemannian optimization concerns optimizing a real-valued function f whose domain is a Riemannian manifold $\mathbb{M}$ (Absil, Mahony and Sepulchre (2008)). The continuous optimization on the Riemannian manifold often requires calculations on the tangent space due to its common nonlinearity. A typical procedure of a Riemannian optimization method includes three steps per iteration: 1. find the tangent space of $\mathbb{M}$; 2. update the point on the tangent space; 3. map the point from the tangent space back to the manifold, that is, retraction. A pictorial illustration for the three steps in Riemannian optimization is presented in Figure 2. The readers are also referred to Absil, Mahony and Sepulchre (2008) and Boumal (2023) for more discussions on Riemannian optimization.

2.1. *Geometry of low Tucker rank tensor manifolds.* Denote the collection of $(p_1, \ldots, p_d, p_{d+1}, \ldots, p_{d+m})$-dimensional tensors of Tucker rank $\mathbf{r} := (r_1, \ldots, r_d, r_{d+1}, \ldots, r_{d+m})$ by $\mathbb{M}_{\mathbf{r}} = \{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}, \mathrm{Tucrank}(\boldsymbol{\mathcal{X}}) = \mathbf{r}\}$. Then $\mathbb{M}_{\mathbf{r}}$ forms a $\{\prod_{j=1}^{d+m} r_j + \sum_{j=1}^{d+m} r_j(p_j - r_j)\}$-dimensional smooth submanifold embedded in $\mathbb{R}^{p_1 \times \cdots \times p_{d+m}}$ (Uschmajew and Vandereycken (2013)). Recall in the general over-parameterized scenario, $\mathbf{r}$ may be different from $\mathbf{r}^*$, the actual rank of the tensor of interest. Suppose $\boldsymbol{\mathcal{X}} \in \mathbb{M}_{\mathbf{r}}$ has Tucker decomposition $[\![\boldsymbol{\mathcal{S}}; \mathbf{U}_1, \ldots, \mathbf{U}_d, \mathbf{U}_{d+1}, \ldots, \mathbf{U}_{d+m}]\!]$. Define $\mathbf{V}_k = \mathrm{QR}(\mathcal{M}_k(\boldsymbol{\mathcal{S}})^\top)$, which corresponds to the row space of $\mathcal{M}_k(\boldsymbol{\mathcal{S}})$, and for $k = 1, \ldots, d+m$, define

$$\mathbf{W}_k := (\mathbf{U}_{d+m} \otimes \cdots \otimes \mathbf{U}_{k+1} \otimes \mathbf{U}_{k-1} \otimes \cdots \otimes \mathbf{U}_1)\mathbf{V}_k \in \mathbb{O}_{p_{-k}, r_k}, \tag{4}$$

where $p_{-k} = \prod_{j=1, j\neq k}^{d+m} p_j$. By the tensor matricization formula provided in Supplement 2, $\mathbf{U}_k$, $\mathbf{W}_k$ correspond to the subspaces of the column and row spans of $\mathcal{M}_k(\boldsymbol{\mathcal{X}})$, respectively. Koch and Lubich (2010) provided the explicit formulas for the tangent space of $\mathbb{M}_{\mathbf{r}}$ at $\boldsymbol{\mathcal{X}}$, denoted by $T_{\boldsymbol{\mathcal{X}}}\mathbb{M}_{\mathbf{r}}$ (see Supplement 2 for the expression). We equip $\mathbb{M}_{\mathbf{r}}$ with the Riemannian metric induced by the natural Euclidean inner product $\langle \cdot, \cdot \rangle$. Under this metric, the following

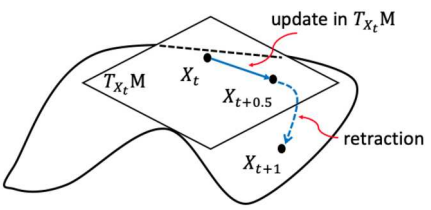


FIG. 2. *Pictorial illustration of steps in Riemannian optimization.*

operator $P_{T_{\mathcal{X}}}$ projects any tensor $\mathcal{Z} \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}$ onto the tangent space $T_{\mathcal{X}}\mathbb{M}_{\mathbf{r}}$,

$$P_{T_{\mathcal{X}}}(\mathcal{Z}) := \mathcal{Z} \times_{k=1}^{d+m} P_{\mathbf{U}_k} + \sum_{k=1}^{d+m} \mathcal{T}_k\big(P_{\mathbf{U}_{k\perp}} \mathcal{M}_k(\mathcal{Z}) P_{\mathbf{W}_k}\big), \tag{5}$$

where $\mathcal{T}_k(\cdot)$ denotes the mode-k tensorization, that is, the reverse operator of $\mathcal{M}_k(\cdot)$.

2.2. *Riemannian gradient descent and Gauss–Newton for tensor-on-tensor regression.* The Riemannian gradient of a smooth function $f : \mathbb{M}_{\mathbf{r}} \to \mathbb{R}$ at $\mathcal{X} \in \mathbb{M}_{\mathbf{r}}$ is defined as the unique tangent vector $\operatorname{grad} f(\mathcal{X}) \in T_{\mathcal{X}}\mathbb{M}_{\mathbf{r}}$ such that $\langle \operatorname{grad} f(\mathcal{X}), \mathcal{Z}\rangle = \mathrm{D} f(\mathcal{X})[\mathcal{Z}]$, $\forall \mathcal{Z} \in T_{\mathcal{X}}\mathbb{M}_{\mathbf{r}}$, where $\mathrm{D} f(\mathcal{X})[\mathcal{Z}]$ denotes the directional derivative of f at point $\mathcal{X}$ along direction $\mathcal{Z}$. We can calculate the Riemannian gradient for the tensor-on-tensor regression as follows.

LEMMA 1 (Riemannian gradient). For $f(\mathcal{X})$ in (3), $\operatorname{grad} f(\mathcal{X}) = P_{T_{\mathcal{X}}}(\mathscr{A}^*(\mathscr{A}(\mathcal{X}) - \mathcal{Y}))$, where $\mathscr{A}^*$ is the adjoint operator of $\mathscr{A}$.

By Lemma 1, a natural idea of RGD update is $\mathcal{X}^{t+0.5} = \mathcal{X}^t - \alpha_t P_{T_{\mathcal{X}^t}} \mathscr{A}^*(\mathscr{A}(\mathcal{X}^t) - \mathcal{Y})$, where the stepsize α_t is chosen as the local steepest descent direction with a closed form as

$$\begin{aligned} \alpha_t &:= \arg\min_{\alpha \in \mathbb{R}} \frac{1}{2} \|\mathcal{Y} - \mathscr{A}(\mathcal{X}^t - \alpha P_{T_{\mathcal{X}^t}} \mathscr{A}^*(\mathscr{A}(\mathcal{X}^t) - \mathcal{Y}))\|_{\mathrm{F}}^2 \\ &= \frac{\|P_{T_{\mathcal{X}^t}}(\mathscr{A}^*(\mathscr{A}(\mathcal{X}^t) - \mathcal{Y}))\|_{\mathrm{F}}^2}{\|\mathscr{A} P_{T_{\mathcal{X}^t}}(\mathscr{A}^*(\mathscr{A}(\mathcal{X}^t) - \mathcal{Y}))\|_{\mathrm{F}}^2}. \end{aligned} \tag{6}$$

As illustrated in Figure 2, the updated iterate $\mathcal{X}^{t+0.5}$ may not be on the Riemannian manifold $\mathbb{M}_{\mathbf{r}}$. We can apply two types of computationally efficient retractions to bring $\mathcal{X}^{t+0.5}$ back to $\mathbb{M}_{\mathbf{r}}$: truncated high-order singular value decomposition (T-HOSVD) (De Lathauwer, De Moor and Vandewalle (2000a)) or sequentially truncated high-order singular value decomposition (ST-HOSVD) (Vannieuwenhoven, Vandebril and Meerbergen (2012)). The pseudocode of T-HOSVD and ST-HOSVD are given in Algorithms 1 and 2 in Supplement 1, respectively.

Moreover, the first-order methods, such as RGD described above, can suffer from slow convergence and low precision in large-scale settings. A natural remedy is to apply second-order methods, such as the Newton algorithm. For tensor-on-tensor regression, the Riemannian Newton relies on the construction and inversion of Riemannian Hessian, which is analytically difficult to develop and computationally intensive. Alternatively, the following Riemannian Gauss–Newton update is a nice approximation of the Riemannian Newton for the nonlinear least squares objective (Absil, Mahony and Sepulchre ((2008), Section 8.4.1)):

$$-\operatorname{grad} f\big(\mathcal{X}^t\big) = P_{T_{\mathcal{X}^t}}\big(\mathscr{A}^*\big(\mathscr{A}(\eta)\big)\big) \quad \text{where } \eta \in T_{\mathcal{X}^t}\mathbb{M}_{\mathbf{r}}. \tag{7}$$

Gauss–Newton has a similar per-iteration complexity as first-order methods but requires much fewer iterations to converge in several other tensor decomposition problems (Sorber, Van Barel and De Lathauwer (2013)). The direct calculation of (7) is still complicated. Surprisingly, we can show the Gauss–Newton equation (7) for tensor-on-tenor regression is equivalent to the following least squares equation.

LEMMA 2. For $f(\mathcal{X})$ in (3), suppose the current iterate is $\mathcal{X}^t$. Then the Riemannian Gauss–Newton update is $\eta^{\mathrm{RGN}} = \arg\min_{\eta \in T_{\mathcal{X}^t}\mathbb{M}_{\mathbf{r}}} \frac{1}{2}\|\mathcal{Y} - \mathscr{A} P_{T_{\mathcal{X}^t}}(\mathcal{X}^t + \eta)\|_{\mathrm{F}}^2$.

Algorithm 1 Riemannian gradient descent/Gauss–Newton for (over-parameterized) tensor-on-tensor regression

1: **Input**: $\mathcal{Y} \in \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}}$, $\mathcal{A}_1, \ldots, \mathcal{A}_n \in \mathbb{R}^{p_1\times\cdots\times p_d}$, $t_{\max}$, input Tucker rank $\mathbf{r}$ and initialization $\mathcal{X}^0$ of Tucker rank $\mathbf{r}$.
2: **for** $t = 0, 1, \ldots, t_{\max} - 1$ **do**
3: (RGD Update) Compute $\mathcal{X}^{t+0.5} = \mathcal{X}^t - \alpha_t P_{T_{\mathcal{X}^t}} \mathscr{A}^*(\mathscr{A}(\mathcal{X}^t) - \mathcal{Y})$, where α_t is given in (6).
(RGN Update) Solve the least squares problem

$$\mathcal{X}^{t+0.5} = \mathop{\arg\min}_{\mathcal{X}\in T_{\mathcal{X}^t}\mathbb{M}_{\mathbf{r}}} \frac{1}{2}\|\mathcal{Y} - \mathscr{A} P_{T_{\mathcal{X}^t}}(\mathcal{X})\|_2^2. \tag{8}$$

4: Update $\mathcal{X}^{t+1} = \mathcal{H}_{\mathbf{r}}(\mathcal{X}^{t+0.5})$. Here $\mathcal{H}_{\mathbf{r}}(\cdot)$ is the retraction map onto $\mathbb{M}_{\mathbf{r}}$, for example, ST-HOSVD and T-HOSVD.
5: **end for**
6: **Output**: $\mathcal{X}^{t_{\max}}$.

As we will discuss in Section 7 that under some mild condition on $\mathscr{A}$, the least squares problem in (8) has a unique solution and can be implemented and solved efficiently via solving $(m + 1)$ separate least squares based on Lemma 2. The pseudocode of the overall RGD and RGN procedures are summarized in Algorithm 1.

REMARK 1 (Riemannian optimization for bounded rank constraint). The classic RGD/RGN methods are designed to optimize on smooth manifolds. This corresponds to minimizing the objective function in (3) with the fixed Tucker rank constraint $\text{Tucrank}(\mathcal{X}) = \mathbf{r}$ since $\mathbb{M}_{\mathbf{r}}$ is a smooth manifold. Note that $\{\mathcal{X} : \text{Tucrank}(\mathcal{X}) \le \mathbf{r}\}$ is not a smooth manifold while such bounded rank constraint is essential to handle over-parameterization, the classic theory no longer applies. Regardless, we propose to continue using Algorithm 1 even with the bounded rank constraint.

3. Theory of RGD/RGN in tensor-on-tensor regression. For technical convenience in the convergence analysis of RGD and RGN, we first introduce the tensor restricted isometry property (TRIP).

DEFINITION 1 (Tensor restricted isometry property (TRIP)). Let $\mathscr{A} : \mathbb{R}^{p_1\times\cdots\times p_{d+m}} \to \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}}$ be a linear map. For a fixed $(d + m)$-tuple $\mathbf{r} = (r_1, \ldots, r_{d+m})$ with $1 \le r_k \le p_k$, define the $\mathbf{r}$-tensor restricted isometry constant to be the smallest number $R_{\mathbf{r}}$ such that $(1 - R_{\mathbf{r}})\|\mathcal{Z}\|_{\text{F}}^2 \le \|\mathscr{A}(\mathcal{Z})\|_{\text{F}}^2 \le (1 + R_{\mathbf{r}})\|\mathcal{Z}\|_{\text{F}}^2$ holds for all $\mathcal{Z}$ of Tucker rank at most $\mathbf{r}$. If $0 \le R_{\mathbf{r}} < 1$, we say $\mathscr{A}$ satisfies $\mathbf{r}$-tensor restricted isometry property ($\mathbf{r}$-TRIP).

TRIP can be seen as a tensor generalization of the popular restricted isometry property (RIP) (Candès and Plan (2011)). TRIP was used in various tensor inverse problems (Rauhut, Schneider and Stojanac (2017)). The next Proposition 1 shows $\mathscr{A}$ satisfies TRIP with high probability when $\mathscr{A}$ is generated from a sufficient number of sub-Gaussian measurements.

PROPOSITION 1 (TRIP under sub-Gaussian). *Suppose $\mathscr{A}$ is defined as (2) and each entry of $\mathcal{A}_i$ is independently drawn from mean zero variance $1/n$ sub-Gaussian distributions. There exists universal constants $C, c > 0$ such that for any Tucker rank $\mathbf{r} = (r_1, \ldots, r_{d+m})$ and $0 \le R_{\mathbf{r}} < 1$, as long as $n \ge C(\sum_{i=1}^d (p_i - r_i)r_i + \prod_{i=1}^d r_i)\log(d)/R_{\mathbf{r}}^2$, $\mathscr{A}$ satisfies the TRIP with $\mathbf{r}$-TRIP constant $R_{\mathbf{r}}$ with probability at least $1 - \exp(-c(\sum_{i=1}^d p_i))$.*

Now, we are ready to present the convergence theories for RGD and RGN.

THEOREM 1 (Convergence of RGD). *Assume the tensor rank of $\boldsymbol{\mathcal{X}}^*$ is $\mathbf{r}^*$ and the input rank to Algorithm 1 is $\mathbf{r} \geq \mathbf{r}^*$. Suppose $\mathscr{A}$ satisfies $2\mathbf{r}$-TRIP, and the initialization $\boldsymbol{\mathcal{X}}^0$ satisfies $\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq \frac{R_{2\mathbf{r}}}{(d+m)(1+R_{2\mathbf{r}+\mathbf{r}^*}-R_{2\mathbf{r}})}\underline{\lambda}$, where $\underline{\lambda} := \min_{k=1,\ldots,d+m} \sigma_{r_k^*}(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$ is the minimum of least singular values at each matricization of $\boldsymbol{\mathcal{X}}^*$. In addition, we assume $R_{2\mathbf{r}} \leq \frac{1}{8(\sqrt{d+m}+1)+1}$ and $\underline{\lambda} \geq \frac{2(1+R_{2\mathbf{r}+\mathbf{r}^*}-R_{2\mathbf{r}})(\sqrt{d+m}+1)(d+m)}{R_{2\mathbf{r}}(1-R_{2\mathbf{r}})}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}}$. Then for all $t \geq 0$,*

$$\begin{aligned}(9)\qquad &\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \\ &\leq 2^{-t}\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} + \frac{2(\sqrt{d+m}+1)}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}}.\end{aligned}$$

Recall $(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}$ denotes the best Tucker rank $2\mathbf{r}$ approximation of the tensor $\mathscr{A}^*(\boldsymbol{\mathcal{E}})$.

Especially if $\boldsymbol{\mathcal{E}} = 0$, $\{\boldsymbol{\mathcal{X}}^t\}$ converges linearly to $\boldsymbol{\mathcal{X}}^*$:

$$\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq 2^{-t}\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \quad \forall t \geq 0.$$

THEOREM 2 (Convergence of RGN). *Assume the tensor rank of $\boldsymbol{\mathcal{X}}^*$ is $\mathbf{r}^*$ and the input rank to Algorithm 1 is $\mathbf{r} \geq \mathbf{r}^*$. Suppose $\mathscr{A}$ satisfies $2\mathbf{r}$-TRIP and the initialization $\boldsymbol{\mathcal{X}}^0$ satisfies $\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq \frac{1-R_{2\mathbf{r}}}{4(d+m)(\sqrt{d+m}+1)(1+R_{2\mathbf{r}+\mathbf{r}^*}-R_{2\mathbf{r}})}\underline{\lambda}$. Then for all $t \geq 0$,*

$$\begin{aligned}&\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \\ &\leq 2^{-2^t}\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} + \frac{2(\sqrt{d+m}+1)}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}}.\end{aligned}$$

Especially if $\boldsymbol{\mathcal{E}} = 0$, $\{\boldsymbol{\mathcal{X}}^t\}$ converges quadratically to $\boldsymbol{\mathcal{X}}^*$:

$$\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq 2^{-2^t}\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \quad \forall t \geq 0.$$

Theorems 1 and 2 show that with proper assumptions on $\mathscr{A}$ and initialization, iterates of RGD and RGN converge linearly and quadratically to the ball of center $\boldsymbol{\mathcal{X}}^*$ and radius $O(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$, respectively. If $\boldsymbol{\mathcal{E}} = 0$, that is, in the noiseless case, $\boldsymbol{\mathcal{X}}^t$ generated by RGD/RGN converges linearly/quadratically to the exact $\boldsymbol{\mathcal{X}}^*$. These results show the convergence of RGD and RGN are both robust against rank over-parameterization. We note that the error bound $O(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$, which is achievable by RGD and RGN, depends on the input rank $\mathbf{r}$ and will increase as $\mathbf{r}$ increases. This is confirmed by the simulation study in Section 8.2, indicating that selecting an appropriate input rank $\mathbf{r}$ remains crucial for the accuracy of the estimators.

One challenge in establishing Theorems 1 and 2 is to show the contraction of the iterates in the rank overspecified scenario. Standard analysis will result in a condition which requires $\underline{\lambda}' := \min_{k=1,\ldots,d+m} \sigma_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$ to be larger than some positive threshold. However, it can never be satisfied since $\underline{\lambda}'$ is zero in the rank overspecified scenario. Instead, we show via a refined analysis that lower bounding $\underline{\lambda}$ is still enough. One such example is Lemma 3 in Section 6, where we obtain a projection error bound proportional to $\underline{\lambda}$ rather than $\underline{\lambda}'$ even in the rank overspecified scenario.

REMARK 2 (General input rank and under-parameterization). Suppose $\mathbf{r}$ is a general input rank (possibly under-parameterized, e.g., $r_k < r_k^*$ for some k), we can rewrite (1) into $\boldsymbol{\mathcal{Y}}_i = \langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}' \rangle_* + \boldsymbol{\mathcal{E}}_i'$, where $\boldsymbol{\mathcal{X}}'$ is the best rank $\mathbf{r}$ approximation of $\boldsymbol{\mathcal{X}}^*$ and $\boldsymbol{\mathcal{E}}_i' = \boldsymbol{\mathcal{E}}_i + \langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* - \boldsymbol{\mathcal{X}}' \rangle_*$. Similar results to Theorems 1 and 2 hold if $\boldsymbol{\mathcal{E}}_i$ is replaced by $\boldsymbol{\mathcal{E}}_i'$. We have

the following contraction error bounds for RGD and RGN for general input rank and under-parameterized cases:

$$\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}'\|_{\mathrm{F}} \leq \frac{\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}'\|_{\mathrm{F}}}{2^t} + \frac{2(\sqrt{d+m}+1)}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}'))_{\max(2\mathbf{r})}\|_{\mathrm{F}},$$

$$\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}'\|_{\mathrm{F}} \leq \frac{\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}'\|_{\mathrm{F}}}{2^{2^t}} + \frac{2(\sqrt{d+m}+1)}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}'))_{\max(2\mathbf{r})}\|_{\mathrm{F}}.$$

REMARK 3 (Convergence guarantees under over-parameterized scenario compared with literature). When $\boldsymbol{\mathcal{E}} = 0$, the convergent point $\boldsymbol{\mathcal{X}}^*$ of RGD and RGN has Tucker rank $\mathbf{r}^*$, which falls out of the manifold $\mathbb{M}_{\mathbf{r}}$ when $\mathbf{r} > \mathbf{r}^*$, that is, the over-parameterized scenario. Because of this, the standard convergence theory of RGD/RGN does not imply the convergence results in Theorems 1 and 2 to our best knowledge. Especially in the low-rank matrix trace regression setting, (Barber and Ha ((2018), Theorem 4.1)) established a local convergence result of RGD with a bounded rank constraint for a general objective f satisfying restricted strong convexity and smoothness. However, the local convergence radius implied by their theory shrinks to 0 in our setting and does not directly apply. Also see more discussions on the convergence of various Riemannian optimization algorithms with bounded rank constraints in Schneider and Uschmajew (2015), Levin, Kileel and Boumal (2023), Olikier and Absil (2023).

REMARK 4 (Conditions). We impose the mild condition $\underline{\lambda} \geq \Omega(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$ while analyzing RGD. Since the forthcoming Theorem 4 shows $\Omega(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$ is the essential statistical error, $\underline{\lambda} \leq O(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$ can be a trivial case from a statistical perspective because the initialization $\boldsymbol{\mathcal{X}}^0$ is already optimal and no further refinement is needed in such the scenario. Another key condition on initialization will be discussed in Section 4.

Next, we show in two ways that the statistical error $O(\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}})$ achieved by RGD and RGN is essential. First, in Theorem 3, we show the estimators with small loss, such as the global minimizer of the loss function (3), achieve the same error rate.

THEOREM 3 (Upper bound for estimators with small loss and global minimizers). *Suppose $\mathscr{A}$ satisfies $2\mathbf{r}$-TRIP with TRIP constant $R_{2\mathbf{r}}$ (Definition 1). Let $\widehat{\boldsymbol{\mathcal{X}}}$ be any estimator such that* $\mathrm{Tucrank}(\widehat{\boldsymbol{\mathcal{X}}}) \leq \mathbf{r}$ *and* $\|\boldsymbol{\mathcal{Y}} - \mathscr{A}(\widehat{\boldsymbol{\mathcal{X}}})\|_{\mathrm{F}}^2 \leq \|\boldsymbol{\mathcal{Y}} - \mathscr{A}(\boldsymbol{\mathcal{X}}^*)\|_{\mathrm{F}}^2$, *that is, the loss function value of $\widehat{\boldsymbol{\mathcal{X}}}$ is no bigger than $\boldsymbol{\mathcal{X}}^*$. Then* $\|\widehat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq \frac{2}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}}$.

Second, we focus on the Gaussian ensemble design, which has been widely considered as a benchmark-setting in the literature on compressed sensing, and matrix/tensor regression (Candès and Plan (2011), Raskutti, Yuan and Chen (2019)). In Theorem 4, we establish the minimax estimation error rate under Gaussian ensemble design, which demonstrates the statistical optimality of RGD and RGN when d and m are constants.

DEFINITION 2 (Tensor-on-tensor regression under gaussian ensemble design). We say the tensor-on-tensor regression (1) is generated from the Gaussian ensemble design if $\{\boldsymbol{\mathcal{A}}_i\}_{i=1}^n$ and $\{\boldsymbol{\mathcal{E}}_i\}_{i=1}^n$ are generated independently, $\boldsymbol{\mathcal{A}}_i$ has i.i.d. $N(0, 1/n)$ entries, and $\boldsymbol{\mathcal{E}}_i$ has i.i.d. $N(0, \sigma^2/n)$ entries.

THEOREM 4 (Error bound under Gaussian ensemble and minimax risk upper and lower bounds). *Consider the tensor-on-tensor regression problem (1) under Gaussian ensemble design (Definition 2) and let* $df = \sum_{i=1}^{d+m} r_i(p_i - r_i) + \prod_{i=1}^{d+m} r_i$.

- (*Upper bound*) *When* $n \geq C(\sum_{i=1}^{d}(p_i - r_i)r_i + \prod_{i=1}^{d} r_i)\log(d)$ *for some large positive constant* C, *with probability at least* $1 - \exp(-c_1(d,m)\underline{p})$, $\|(\mathscr{A}^*(\boldsymbol{\mathcal{E}}))_{\max(2\mathbf{r})}\|_{\mathrm{F}} \leq c_2(d,m)\sigma\sqrt{\frac{df}{n}}$ *for some* $c_1(d,m), c_2(d,m) > 0$, *where* $\underline{p} := \min_j p_j$. *Furthermore, for* $\widehat{\boldsymbol{\mathcal{X}}}$ *in Theorem 3, we have* $\mathbb{E}\|\widehat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}\|_{\mathrm{F}} \leq C_2(d,m)\sigma\sqrt{\frac{df}{n}}$.
- (*Lower bound*) *Consider the parameter space of all* $p_1 \times \cdots \times p_{d+m}$*-dimensional tensors of Tucker rank at most* $\mathbf{r} = (r_1, \ldots, r_{d+m})$:

$$\mathcal{F}_{\mathbf{p},\mathbf{r}} := \{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}, \mathrm{Tucrank}(\boldsymbol{\mathcal{X}}) \leq \mathbf{r}\}.$$

Suppose $\min_k r_k \geq C'$ *for some absolute constant* C'. *Then there exists a absolute constant* $c > 0$ *that does not depend on* $\mathbf{r}$ *and* $\mathbf{p}$ *such that* $\inf_{\widehat{\boldsymbol{\mathcal{X}}}} \sup_{\boldsymbol{\mathcal{X}} \in \mathcal{F}_{\mathbf{p},\mathbf{r}}} \mathbb{E}\|\widehat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}\|_{\mathrm{F}} \geq c\sigma\sqrt{\frac{df}{n}}$.

4. Applications, initialization and guarantees in specific scenarios. The convergence theory in Theorems 1 and 2 rely on a good initialization. As it is challenging to develop a universal initialization algorithm that handles all settings of tensor-on-tensor regression with provable guarantees, we focus on the four most representative cases appearing in applications and literature, *scalar-on-tensor regression*, *tensor-on-vector regression*, *matrix trace regression* and *rank*-1 *tensor-on-tensor regression* to show various spectral methods yield adequate initializations.

4.1. *Scalar-on-tensor regression*. The scalar-on-tensor regression corresponds to the general tensor-on-tensor regression model (1) with $m = 0$. It can be written as

$$\mathbf{y} = \mathscr{A}(\boldsymbol{\mathcal{X}}^*) + \boldsymbol{\varepsilon}, \quad \text{or} \quad \mathbf{y}_i = \langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* \rangle + \boldsymbol{\varepsilon}_i, \quad i \in [n]. \tag{10}$$

Here, $\mathbf{y} \in \mathbb{R}^n$ are observations, $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ are unknown noise, and $\boldsymbol{\mathcal{X}}^* \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ is an order-d Tucker rank $\mathbf{r}^*$ tensor that links response $\mathbf{y}_i$ to tensor covariates $\boldsymbol{\mathcal{A}}_i$, which is the parameter of interest. $\mathscr{A}(\boldsymbol{\mathcal{X}}^*) = (\langle \boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{X}}^* \rangle, \ldots, \langle \boldsymbol{\mathcal{A}}_n, \boldsymbol{\mathcal{X}}^* \rangle)^\top$. We propose the following Algorithm 2 on initialization.

THEOREM 5 (Initialization and overall guarantees in scalar-on-tensor regression). *Consider the over-parameterized scalar-on-tensor regression under Gaussian ensemble design. Denote* $df = \sum_{i=1}^{d}(p_i - r_i)r_i + \prod_{i=1}^{d} r_i$ *and suppose* $n \geq c(d)(\frac{(\|\boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}}^2 + \sigma^2)}{\underline{\lambda}^2}((\prod_{i=1}^{d} p_i)^{1/2} + df))$ *for some constant* $c(d)$. *Then with probability at least* $1 - \underline{p}^{-C}$ *for some* $C > 0$:

- $\boldsymbol{\mathcal{X}}^0$ *returned from Algorithm 2 satisfies the initialization conditions in Theorems 1 and 2*;
- *consider RGD and RGN initialized with* $\boldsymbol{\mathcal{X}}^0$, *then as long as* $t_{\max} \geq \log(\frac{\underline{\lambda}\sqrt{n/df}}{c_1(d)\sigma}) \vee 0$ *for RGD or* $t_{\max} \geq \log\log(\frac{\underline{\lambda}\sqrt{n/df}}{c_2(d)\sigma}) \vee 0$ *for RGN, we have the output of RGD or RGN satisfies* $\|\boldsymbol{\mathcal{X}}^{t_{\max}} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} \leq c_3(d)\sigma\sqrt{\frac{df}{n}}$.

Algorithm 2 Initialization for (over-parameterized) scalar-on-tensor regression

1: **Input:** $\mathbf{y}_i \in \mathbb{R}$, $\boldsymbol{\mathcal{A}}_i \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ for $i = 1, \ldots, n$ and input Tucker rank $\mathbf{r} = (r_1, \ldots, r_d)$.
2: Calculate $\widetilde{\mathbf{U}}_k^0 = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\mathscr{A}^*(\mathbf{y})))$, $k = 1, \ldots, d$.
3: For $k = 1$ to d, apply one-iteration HOOI, that is, calculate

$$\widetilde{\mathbf{U}}_k^1 = \mathrm{SVD}_{r_k}\big(\mathcal{M}_k\big(\mathscr{A}^*(\mathbf{y}) \times_{j<k} (\widetilde{\mathbf{U}}_j^0)^\top \times_{j>k} (\widetilde{\mathbf{U}}_j^0)^\top\big)\big).$$

Recall $\mathrm{SVD}_r(\cdot)$ returns the matrix composed of the leading r left singular vectors of matrix "$\cdot$".
4: **Output**: $\boldsymbol{\mathcal{X}}^0 = \mathscr{A}^*(\mathbf{y}) \times_{k=1}^{d} \widetilde{\mathbf{U}}_k^1 (\widetilde{\mathbf{U}}_k^1)^\top$.

In establishing Theorem 5, we introduce a new perturbation bound for over-parameterized tensor decomposition. See Theorem 10 in Section 6 for more details. Compared with RGD, RGN only requires a *double logarithmic* number of iterations to achieve the same $O(\sigma\sqrt{\frac{df}{n}})$ error rate.

REMARK 5 (Sample complexity for over-parameterized scalar-on-tensor regression). Suppose $\|\boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}}^2 \geq C\sigma^2$ for some $C > 0$, $\kappa := \bar{\lambda}/\underline{\lambda} = O(1)$ where $\bar{\lambda} = \max_{k=1,\ldots,d} \sigma_1 \times (\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$ and $p_1 = p_2 = \cdots = p$, $r_1 = r_2 = \cdots = r$, $r_1^* = r_2^* \ldots = r^*$, then the overall sample complexity for RGD/RGN in over-parameterized scalar-on-tensor regression with spectral initialization is $\Omega(r^*(p^{d/2} + pr + r^d))$. Compared to the sample complexity required for the global minimizer (see Theorem 3) in this example, that is, $\Omega(pr + r^d)$ proved in Theorem 4, there is a significant gap between what can be achieved by the inefficient global minimizer and efficient RGD/RGN algorithms. Rigorous evidence for this statistical-computational gap will be provided in Section 5.

4.2. *Tensor-on-vector regression*. In this section, we consider the tensor-on-vector regression model:

$$\boldsymbol{\mathcal{Y}}_i = \boldsymbol{\mathcal{X}}^* \times_1 \mathbf{a}_i^\top + \boldsymbol{\mathcal{E}}_i \quad \text{for } i = 1, \ldots, n, \tag{11}$$

where $\boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{E}}_i \in \mathbb{R}^{p_2 \times \cdots \times p_{m+1}}$ are the observation and noise, $\boldsymbol{\mathcal{X}}^* \in \mathbb{R}^{p_1 \times \cdots \times p_{m+1}}$ is the parameter tensor of interest with Tucker rank $\mathbf{r}^*$ and $\mathbf{a}_i \in \mathbb{R}^{p_1}$ is the covariate vector. We can also write the model compactly as $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}}^* \times_1 \mathbf{A} + \boldsymbol{\mathcal{E}}$ where $\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{E}} \in \mathbb{R}^{n \times \cdots \times p_{m+1}}$, $\boldsymbol{\mathcal{Y}}_{[i,:,\ldots,:]} = \boldsymbol{\mathcal{Y}}_i$, $\boldsymbol{\mathcal{E}}_{[i,:,\ldots,:]} = \boldsymbol{\mathcal{E}}_i$ and $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n]^\top \in \mathbb{R}^{n \times p_1}$ is the collection of covariate vectors. We propose the following Algorithm 3 for initialization and its guarantee is provided in Theorem 6.

THEOREM 6 (Initialization and overall guarantees in tensor-on-vector regression). *Consider the over-parameterized tensor-on-vector regression under Gaussian ensemble design. Denote* $df = \sum_{i=1}^{m+1}(p_i - r_i)r_i + \prod_{i=1}^{m+1} r_i$. *Suppose*

$$n \geq c(m)\left(\left(\left(\prod_{i=1}^{m+1} p_i\right)^{1/2} + df\right)\sigma^2/\underline{\lambda}^2 + p_1\right)$$

for some constant $c(m)$. *Then with probability at least* $1 - \exp(-c\underline{p})$ *for some* $c > 0$:

- $\boldsymbol{\mathcal{X}}^0$ *returned from Algorithm 3 satisfies the initialization conditions in Theorems 1 and 2*;

Algorithm 3 Initialization for (over-parameterized) tensor-on-vector regression

1: **Input:** $\boldsymbol{\mathcal{Y}}_i \in \mathbb{R}^{p_2 \times \cdots \times p_{m+1}}$, $\mathbf{a}_i \in \mathbb{R}^{p_1}$ for $i = 1, \ldots, n$ and input Tucker rank $\mathbf{r} = (r_1, \ldots, r_{m+1})$.
2: Compute the QR decomposition of $\mathbf{A}$ and denote it by $\mathbf{Q}_\mathbf{A}\mathbf{R}_\mathbf{A}$.
3: Calculate $\widetilde{\mathbf{U}}_k^0 = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{Y}} \times_1 \mathbf{Q}_\mathbf{A}^\top))$, $k = 1, \ldots, m+1$.
4: For $k = 1$ to $m+1$, apply one-iteration HOOI, that is, calculate

$$\widetilde{\mathbf{U}}_k^1 = \mathrm{SVD}_{r_k}\big(\mathcal{M}_k\big((\boldsymbol{\mathcal{Y}} \times_1 \mathbf{Q}_\mathbf{A}^\top) \times_{j<k} (\widetilde{\mathbf{U}}_j^0)^\top \times_{j>k} (\widetilde{\mathbf{U}}_j^0)^\top\big)\big).$$

5: Compute $\overline{\boldsymbol{\mathcal{X}}}^0 = (\boldsymbol{\mathcal{Y}} \times_1 \mathbf{Q}_\mathbf{A}^\top) \times_{k=1}^{m+1} \widetilde{\mathbf{U}}_k^1 (\widetilde{\mathbf{U}}_k^1)^\top$.
6: Return $\boldsymbol{\mathcal{X}}^0 = \overline{\boldsymbol{\mathcal{X}}}^0 \times_1 \mathbf{R}_\mathbf{A}^{-1}$.
7: **Output:** $\boldsymbol{\mathcal{X}}^0$.

- *moreover, consider RGD and RGN initialized with* $\boldsymbol{\mathcal{X}}^0$, *then as long as* $t_{\max} \geq \log(\frac{\lambda\sqrt{n/df}}{c_1(m)\sigma}) \vee 0$ *for RGD or* $t_{\max} \geq \log\log(\frac{\lambda\sqrt{n/df}}{c_2(m)\sigma}) \vee 0$ *for RGN, we have the output of RGD or RGN satisfies*

$$\left\|\boldsymbol{\mathcal{X}}^{t_{\max}} - \boldsymbol{\mathcal{X}}^*\right\|_{\mathrm{F}} \leq c_3(m)\sigma\sqrt{\frac{df}{n}}.$$

4.3. *Matrix trace regression.* In this model, we observe

$$\mathbf{y}_i = \langle \mathbf{A}_i, \mathbf{X}^* \rangle + \boldsymbol{\varepsilon}_i, \quad i = 1, \ldots, n; \quad \text{or} \quad \mathbf{y} = \mathscr{A}(\mathbf{X}^*) + \boldsymbol{\varepsilon}, \tag{12}$$

where $\mathbf{y}, \boldsymbol{\varepsilon} \in \mathbb{R}^n$ are observations and unknown noise and $\mathbf{X}^* \in \mathbb{R}^{p_1 \times p_2}$ is a rank r^* parameter matrix of interest.

In matrix trace regression, we can take the retraction map $\mathcal{H}_r$ in RGD and RGN as the best rank r matrix projection operator: $\mathcal{P}_r(\mathbf{B}) = \mathbf{U}_{[:,1:r]}\boldsymbol{\Sigma}_{[1:r,1:r]}\mathbf{V}_{[:,1:r]}^\top$, where $\mathbf{B} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ is the SVD. Different from the low-rank projection for tensor of order 3 or higher, $\mathcal{P}_r$ can be computed efficiently by truncated SVD. Moreover, suppose $\mathbf{X}^t$ has economic SVD $\mathbf{U}^t\boldsymbol{\Sigma}^t\mathbf{V}^{t\top}$, then the projection of $\mathbf{Z} \in \mathbb{R}^{p_1 \times p_2}$ onto the tangent space $T_{\mathbf{X}^t}\mathbb{M}_r$ can be written succinctly as $P_{T_{\mathbf{X}^t}}(\mathbf{Z}) = P_{\mathbf{U}^t}\mathbf{Z}P_{\mathbf{V}^t} + P_{\mathbf{U}_\perp^t}\mathbf{Z}P_{\mathbf{V}^t} + P_{\mathbf{U}^t}\mathbf{Z}P_{\mathbf{V}_\perp^t}$.

We have the following corollary on the guarantees of RGD and RGN in over-parameterized matrix trace regression.

COROLLARY 1 (Convergence of RGD/RGN in matrix trace regression). *Consider the (over-parameterized) matrix trace regression model in (12) with* $r \geq r^*$. *Let* $\mathcal{H}_r$ *be the rank* r *truncated SVD. Suppose* $\mathscr{A}$ *satisfies* $2r$*-RIP.*

(RGD) Suppose the initialization $\mathbf{X}^0$ *satisfies* $\|\mathbf{X}^0 - \mathbf{X}^*\|_{\mathrm{F}} \leq \frac{R_{2r}}{(1+R_{2r+r^*}-R_{2r})}\sigma_{r^*}(\mathbf{X}^*)$. *In addition, we assume* $R_{2r} \leq \frac{1}{17}$ *and* $\sigma_{r^*}(\mathbf{X}^*) \geq \frac{4(1+R_{2r+r^*}-R_{2r})}{R_{2r}(1-R_{2r})}\|(\mathscr{A}^*(\boldsymbol{\varepsilon}))_{\max(2r)}\|_{\mathrm{F}}$. *Then* $\{\mathbf{X}^t\}$ *generated by RGD satisfy for all* $t \geq 0$,

$$\left\|\mathbf{X}^t - \mathbf{X}^*\right\|_{\mathrm{F}} \leq 2^{-t}\left\|\mathbf{X}^0 - \mathbf{X}^*\right\|_{\mathrm{F}} + \frac{4}{1-R_{2r}}\left\|\left(\mathscr{A}^*(\boldsymbol{\varepsilon})\right)_{\max(2r)}\right\|_{\mathrm{F}}.$$

(RGN) If the initialization $\mathbf{X}^0$ *satisfies* $\|\mathbf{X}^0 - \mathbf{X}^*\|_{\mathrm{F}} \leq \frac{1-R_{2r}}{8(1+R_{2r+r^*}-R_{2r})}\sigma_{r^*}(\mathbf{X}^*)$. *Then* $\{\mathbf{X}^t\}$ *generated by RGN satisfy for all* $t \geq 0$,

$$\left\|\mathbf{X}^t - \mathbf{X}^*\right\|_{\mathrm{F}} \leq 2^{-2^t}\left\|\mathbf{X}^0 - \mathbf{X}^*\right\|_{\mathrm{F}} + \frac{4}{1-R_{2r}}\left\|\left(\mathscr{A}^*(\boldsymbol{\varepsilon})\right)_{\max(2r)}\right\|_{\mathrm{F}}.$$

Especially if $\boldsymbol{\varepsilon} = 0$, $\|\mathbf{X}^t - \mathbf{X}^*\|_{\mathrm{F}} \leq 2^{-t}\|\mathbf{X}^0 - \mathbf{X}^*\|_{\mathrm{F}}$ *for RGD and* $\|\mathbf{X}^t - \mathbf{X}^*\|_{\mathrm{F}} \leq 2^{-2^t}\|\mathbf{X}^0 - \mathbf{X}^*\|_{\mathrm{F}}$ *for RGN.*

An efficient initialization for the matrix trace regression is $\mathbf{X}^0 = \mathcal{P}_r(\mathscr{A}^*(\mathbf{y}))$. The guarantee of $\mathbf{X}^0$ and overall performance of RGD and RGN in matrix trace regression are given in Theorem 7.

THEOREM 7 (Initialization and overall guarantees in over-parameterized matrix trace regression). *Consider the over-parameterized matrix trace regression under Gaussian ensemble design. Denote* $df = (p_1 + p_2 - r)r$ *and suppose* $n \geq \frac{C(\sigma^2 + \|\mathbf{X}^*\|_{\mathrm{F}}^2)}{\sigma_{r^*}^2(\mathbf{X}^*)}df$ *for some* $C > 0$. *Then with probability at least* $1 - \exp(-c\underline{p})$:

- $\mathbf{X}^0 = \mathcal{P}_r(\mathscr{A}^*(\mathbf{y}))$ *satisfies the initialization conditions in Corollary* 1;
- *moreover, consider RGD and RGN initialized with* $\mathbf{X}^0$, *then as long as* $t_{\max} \geq \log(\frac{\sigma_{r^*}(\mathbf{X}^*)}{c_1\sigma} \times \sqrt{\frac{n}{df}}) \vee 0$ *for RGD or* $t_{\max} \geq \log\log(\frac{\sigma_{r^*}(\mathbf{X}^*)}{c_2\sigma}\sqrt{\frac{n}{df}}) \vee 0$ *for RGN, we have the output of RGD or RGN satisfies*

$$\left\|\boldsymbol{\mathcal{X}}^{t_{\max}} - \boldsymbol{\mathcal{X}}^*\right\|_{\mathrm{F}} \leq c_3\sigma\sqrt{\frac{df}{n}}.$$

REMARK 6 (Comparison with existing results on over-parameterized matrix trace regression). Recently, Zhuo et al. (2024), Zhang, Fattahi and Zhang (2021) studied the local convergence of factorized gradient descent (GD) in the same setting as ours. In particular, Zhuo et al. (2024) showed the convergence rate of the original factorized GD slows down to being sublinear when the input rank r is greater than the actual rank r^*. Zhang, Fattahi and Zhang (2021) proposed to overcome that by preconditioning the factorized GD; they showed that the convergence rate of preconditioned factorized GD can be boosted back to linear for all $r \geq r^*$. However, the preconditioning step in Zhang, Fattahi and Zhang (2021) requires a carefully chosen damping parameter in each iteration and such the choice depends on the unknown noise variance. In contrast, our proposed RGD and RGN algorithms are easy to implement, tuning-free and are unified in both rank correctly-specified and overspecified settings. In addition, in terms of the theoretical guarantees, the estimation error bound in Zhang, Fattahi and Zhang (2021) is suboptimal in the noisy setting, while our bound is minimax optimal as shown in Theorem 4. Finally, our result is also more general since our $\mathbf{X}^*$ can be a general rank r^* matrix while existing works only focus on positive-semidefinite $\mathbf{X}^*$. The readers are referred to Table 1 for a summary of comparisons.

Meanwhile, to satisfy r-RIP, we need $n = \Omega((p_1 + p_2)r)$, so our theory is still based on the "sample size (n) $\geq$ parameter degree of freedom (df)" scenario. A follow-up question is whether the "implicit regularization" phenomenon discussed in the *related prior work* section appears in Riemannian formulated matrix trace regression in the highly over-parameterized regime, that is, "$df > n$," as such phenomenon was recently observed in factorized gradient descent (Gunasekar et al. (2017), Li, Ma and Zhang (2018)). In fact, the direct application of RGD proposed in this paper does not enjoy implicit regularization in the highly over-parameterized regime because when the input rank r is equal to $p_1 \wedge p_2$, RGD reduces to gradient descent in the whole p_1-by-p_2 matrix parameter space, which does not enjoy implicit regularization as it will converge to the minimum Frobenius norm solution in this over-parameterized setting with near origin initialization (Gunasekar et al. (2017)). Our theory so far does not cover the highly over-parameterized regime and further investigation is left as future work.

4.4. *Rank-*1 *tensor-on-tensor regression*. For the general tensor-on-tensor regression model, although $\mathbb{E}(\mathscr{A}^*(\boldsymbol{\mathcal{Y}})) = \boldsymbol{\mathcal{X}}^*$ is low-rank, the noise structure of $\mathscr{A}^*(\boldsymbol{\mathcal{Y}}) - \boldsymbol{\mathcal{X}}^*$ is complicated that significantly deviates from the commonly studied additive tensor PCA model in the literature. It is thus challenging to provide an optimal theoretical guarantee for the initialization schemes T-HOSVD and ST-HOSVD in general.

In this section, we introduce a modified initialization scheme with theoretical guarantees for general d and m when $\boldsymbol{\mathcal{X}}^*$ is a rank-1 tensor and input rank is also 1. For simplicity, we assume n is even. Suppose $\boldsymbol{\mathcal{X}}^* = \lambda\mathbf{u}_1 \circ \mathbf{u}_2 \circ \cdots \circ \mathbf{u}_{d+m} \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}$, where "$\circ$" denotes the outer product of vectors. Then in this special setting, the model (1) can be rewritten as

$$\boldsymbol{\mathcal{Y}}_i = \lambda\langle \boldsymbol{\mathcal{A}}_i, \mathbf{u}_1 \circ \cdots \circ \mathbf{u}_d\rangle \mathbf{u}_{d+1} \circ \cdots \circ \mathbf{u}_{d+m} + \boldsymbol{\mathcal{E}}_i. \tag{13}$$

Algorithm 4 Initialization for rank-1 tensor-on-tensor regression

1: **Input:** $\mathcal{Y}_i \in \mathbb{R}^{p_{d+1}\times\cdots\times p_{d+m}}$, $\mathcal{A}_i \in \mathbb{R}^{p_1\times\cdots\times p_d}$ for $i = 1,\dots,n$.
2: Calculate $\widetilde{\mathbf{u}}_k^0 = \mathrm{SVD}_1(\mathcal{M}_{k-d+1}(\mathcal{Y}^1))$, $k = d+1,\dots,d+m$.
3: Compute $\mathbf{y}_i' = \langle \mathcal{Y}_i, \widetilde{\mathbf{u}}_{d+1}^0 \circ\cdots\circ \widetilde{\mathbf{u}}_{d+m}^0\rangle$ for $i = n/2+1,\dots,n$.
4: Calculate $\widetilde{\mathbf{u}}_k^0 = \mathrm{SVD}_1(\mathcal{M}_k(\sum_{i=n/2+1}^n \mathbf{y}_i'\mathcal{A}_i))$, $k = 1,\dots,d$.
5: For $k = 1$ to $d+m$, apply one-iteration HOOI, that is, calculate

$$\widetilde{\mathbf{u}}_k^1 = \mathrm{SVD}_1\big(\mathcal{M}_k\big(\mathscr{A}^*(\mathcal{Y}) \times_{j<k} (\widetilde{\mathbf{u}}_j^0)^\top \times_{j>k} (\widetilde{\mathbf{u}}_j^0)^\top\big)\big).$$

6: **Output**: $\mathcal{X}^0 = \mathscr{A}^*(\mathcal{Y}) \times_{k=1}^{d+m} \widetilde{\mathbf{u}}_k^1 (\widetilde{\mathbf{u}}_k^1)^\top$.

Let $\mathcal{Y}^1$ and $\mathcal{Y}^2$ collect $\mathcal{Y}_i$s in the first and second halves of the data: $\mathcal{Y}^1_{[i,:,\dots,:]} = \mathcal{Y}_i$ and $\mathcal{Y}^2_{[i,:,\dots,:]} = \mathcal{Y}_{n/2+i}$ for $i = 1,\dots,n/2$. We propose an initialization procedure in Algorithm 4 and provide its theoretical guarantee in Theorem 8. The high-level idea for Algorithm 4 is as follows: we use the first half of the data $\mathcal{Y}^1$ to get estimates $\widehat{\mathbf{u}}_k$ for $k = d+1,\dots,d+m$ and then use the second half of the data $\mathcal{Y}^2$ to estimate $\mathbf{u}_k$ for $k = 1,\dots,d$ after projecting the data to the subspace spanned by $\{\widehat{\mathbf{u}}_k\}_{k=d+1}^{d+m}$; finally, a one-iteration HOOI is applied to obtain the initialization.

THEOREM 8 (Initialization and overall guarantees in rank-1 tensor-on-tensor regression). *Consider the rank-1 tensor-on-tensor regression under Gaussian ensemble design* (13). *Denote* $df = \sum_{i=1}^{d+m} p_i$ *and suppose* $\lambda > C'\sigma$ *for some* $C' > 0$. *If* $n \geq c(d,m)(\frac{(\lambda^2+\sigma^2)}{\lambda^2} \times ((\prod_{i=1}^d p_i)^{1/2} + \bar{p}) + \frac{\sigma^4}{\lambda^4}(\prod_{i=d+1}^{d+m} p_i + \bar{p}))$ *for some constant* $c(d,m)$ *depending on* d *and* m *only, where* $\bar{p} = \max_{k=1,\dots,d+m} p_i$. *Then with probability at least* $1 - \underline{p}^{-C}$ *for some* $C > 0$:

- $\mathcal{X}^0$ *returned from Algorithm 4 satisfies the initialization conditions in Theorems 1 and 2*;
- *Considering RGD and RGN initialized with* $\mathcal{X}^0$, *as long as* $t_{\max} \geq \log(\frac{\lambda\sqrt{n/df}}{c_1(d,m)\sigma}) \vee 0$ *for RGD or* $t_{\max} \geq \log\log(\frac{\lambda\sqrt{n/df}}{c_2(d,m)\sigma}) \vee 0$ *for RGN, we have the output of RGD or RGN satisfies* $\|\mathcal{X}^{t_{\max}} - \mathcal{X}^*\|_{\mathrm{F}} \leq c_3(d,m)\sigma\sqrt{\frac{df}{n}}$.

5. Computational limits. In this section, we provide rigorous evidence for the computational barrier in scalar-on-tensor regression via the low-degree polynomials method. Without loss of generality, we assume $\boldsymbol{\varepsilon}_i \overset{i.i.d.}{\sim} N(0,\sigma^2)$ with $0 \leq \sigma^2 < 1$, $\mathcal{A}_i \overset{i.i.d.}{\sim} N(0,1)$ and $\|\mathcal{X}^*\|_{\mathrm{F}} + \sigma^2 = 1$ in establishing the computational lower bound for scalar-on-tensor regression (10) (see Supplement 7.1 for a proof). We also consider the setting $p_1 = \cdots = p_d = p$ and $r_1^* = \cdots = r_d^* = r^*$ throughout this section.

We consider a canonical hypothesis testing formulation of scalar-on-tensor regression:

$$\begin{aligned}
&H_0 : \{(\mathbf{y}_i, \mathrm{vec}(\mathcal{A}_i))\}_{i=1}^n \overset{i.i.d.}{\sim} N(0, \mathbf{I}_{1+p^d}),\\
&H_1 : \{(\mathbf{y}_i, \mathrm{vec}(\mathcal{A}_i))\}_{i=1}^n\colon \mathcal{X}^* = \sqrt{1-\sigma^2}\mathbf{x}^{*\otimes d},\\
&\quad \mathbf{x}^* = (x_1^*,\dots,x_p^*),\ x_j^* \overset{i.i.d.}{\sim} \mathrm{Uniform}(\{p^{-1/2}, -p^{-1/2}\});\\
&\quad \text{for } i \in [n],\ \mathcal{A}_i \overset{i.i.d.}{\sim} N(0,1),\ \mathbf{y}_i \text{ is i.i.d. generated}\\
&\quad \text{via } \mathbf{y}_i = \langle \mathcal{X}^*, \mathcal{A}_i\rangle + \boldsymbol{\varepsilon}_i,\ \boldsymbol{\varepsilon}_i \overset{i.i.d.}{\sim} N(0,\sigma^2).
\end{aligned} \tag{14}$$

Since we aim to develop a lower bound, the hardness result for (14) also implies the hardness result for a bigger class in the sense of minimax. The idea of using low-degree polynomials to predict the statistical-computational gaps is recently developed in a line of work (Hopkins and Steurer (2017), Hopkins (2018)). In comparison to sum-of-squares (SOS) computational lower bounds, the low-degree polynomials method is simpler to establish and appears to always yield the same results for natural average-case hardness problems. Low-degree polynomials computational hardness results have been provided to a number of problems, such as the planted clique detection (Hopkins (2018), Barak et al. (2019)), community detection in stochastic block model (Hopkins and Steurer (2017), Hopkins (2018)), the spiked tensor model (Hopkins et al. (2017), Hopkins (2018), Kunisky, Wein and Bandeira (2022)), the spiked Wishart model (Bandeira, Kunisky and Wein (2020)), sparse PCA (Ding et al. (2024)), spiked Wigner model (Kunisky, Wein and Bandeira (2022)), clustering (Löffler, Wein and Bandeira (2022), Davis, Diaz and Wang (2021), Lyu and Xia (2023)), planted vector recovery (Mao and Wein (2021)), certifying RIP (Ding et al. (2021b)) and random k-SAT (Bresler and Huang (2022)). It is gradually believed that the low-degree polynomials method is able to capture the essence of what makes sum-of-squares algorithms succeed or fail (Hopkins (2018), Kunisky, Wein and Bandeira (2022)). Our results on the computational hardness of distinguishing between H_0 and H_1 in scalar-on-tensor regression based on low-degree polynomials are given below.

THEOREM 9 (Low-degree hardness for ccalar-on-tensor regression). *Consider the hypothesis test* (14). *For any* $0 < \delta < 1$, *if* $n \le \frac{(p/dD)^{d/2}\delta}{2(1-\sigma^2)}$, *we have*

$$\sup_{\substack{\deg(f)\le D \\ \text{polynomial } f:\, \mathbb{E}_{H_0} f(\{\mathbf{y}_i, \boldsymbol{\mathcal{A}}_i\}_{i=1}^n)=0, \\ Var_{H_0} f(\{\mathbf{y}_i, \boldsymbol{\mathcal{A}}_i\}_{i=1}^n)=1}} \mathbb{E}_{H_1} f\big(\{\mathbf{y}_i, \boldsymbol{\mathcal{A}}_i\}_{i=1}^n\big) \le \frac{\delta}{1-\delta}. \tag{15}$$

It has been widely conjectured in the literature that for a broad class of hypothesis testing problems: H_0 versus H_1, there is a test with runtime $n^{\tilde{O}(D)}$ and type I + II error tending to zero if and only if there is a successful D-simple statistic, that is, a polynomial f of degree at most D, such that $\mathbb{E}_{H_0} f(X) = 0$, $\mathrm{Var}_{H_0}(f^2(X)) = 1$ and $\mathbb{E}_{H_1} f(X) \to \infty$ (Hopkins (2018), Kunisky, Wein and Bandeira (2022)). Therefore, by setting $D = C \log p$ for any $C > 0$, Theorem 9 provides firm evidence for the statistical-computational gap when $n = O(p^{d/2-\varepsilon})$ for any $\epsilon > 0$. Compared to the sample size requirement in the upper bound mentioned in Remark 5, the computational lower bound established in Theorem 9 is sharp when $r^* = O(1)$, $r \le \sqrt{p}$. Our Theorem 9 answers the question raised by Rauhut, Schneider and Stojanac (2017) on the sample complexity requirement for efficient estimators in scalar-on-tensor regression. We note the first computational hardness evidence for scalar-on-tensor regression was provided recently in Diakonikolas et al. (2023) in the statistical query model. We complement their results by providing a direct low-degree polynomials argument and figuring out the explicit dependence of the sample complexity on the degrees tolerated in low-degree polynomials. Finally, we also show in the Supplement 7.4, Proposition 3, that the hardness of testing H_0 versus H_1 implies the hardness of estimating $\boldsymbol{\mathcal{X}}^*$.

REMARK 7 (Proof ideas and comparison with existing arguments). Here we briefly discuss the proof idea of Theorem 9 and the key technical novelty therein. A detailed proof and preliminaries of low-degree polynomials are provided in Supplement 7. First, it has been established in Hopkins (2018), Kunisky, Wein and Bandeira (2022) that the left-hand side of

(15) is equal to the norm of the truncated likelihood ratio under the null:

$$\sup_{\substack{\text{polynomial } f\colon \deg(f)\le D\\ \mathbb{E}_{H_0} f(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n)=0,\\ \operatorname{Var}_{H_0} f(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n)=1}} \mathbb{E}_{H_1} f\big(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n\big) = \sqrt{\mathbb{E}_{H_0}\bigg(\bigg(\frac{p_{H_1}(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n)}{p_{H_0}(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n)}\bigg)^{\le D}-1\bigg)^2}, \tag{16}$$

where p_{H_0} and p_{H_1} denote the likelihood under the null and alternative, respectively, and $f^{\le D}$ is the projection of a function f to the linear subspace of degree-D polynomials, where the projection is orthonormal with respect to the inner product induced under H_0. A standard trick to bound the right-hand of (16) is to evaluate it separately under the orthogonal basis functions $\{f_j\}_{j\ge 1}$ under the null, and then the argument boils down to bound $\sum_{j=1}^D(\mathbb{E}_{H_1} f_j(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n))^2$, which is the sum of second moments of the orthogonal basis functions under the *alternative*. See (36) in Supplement 7.2 for details. There have been many successes in bounding $\sum_{j=1}^D(\mathbb{E}_{H_1} f_j(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n))^2$ when the testing problem under H_1 has the "signal + noise" structure (Hopkins (2018), Kunisky, Wein and Bandeira (2022)). Such a structure simplifies the analysis as the noise part and signal part are decoupled. In contrast, there is little low-degree polynomial hardness evidence when the problem under H_1 has correlated structures, such as the regression problem considered in this paper. One of our main technical contributions in tackling this challenge is a formula for computing the expectation of Hermite polynomials for correlated multivariate Gaussian random variables (Lemma 4 in Section 6). With this key technical tool, we can bound $\sum_{j=1}^D(\mathbb{E}_{H_1} f_j(\{y_i,\boldsymbol{\mathcal{A}}_i\}_{i=1}^n))^2$ under the H_1 in (14) to prove the result. See Supplement 7.3 for the detailed calculation.

REMARK 8 (Comparing rank overspecification in matrix trace regression and scalar-on-tensor regression). Suppose $r^*=O(1)$. In matrix trace regression, the sample size requirement of the "spectral initialization + local refinement" estimation scheme is $O(pr)$, where r is the input rank. Thus, the sample complexity increases linearly as the input rank r increases. Meanwhile, the sample complexity of the scalar-on-tensor regression under the same estimation scheme is $O(p^{d/2})$ when $r\le\sqrt{p}$ (see Remark 5). Due to the computational lower bound of scalar-on-tensor regression in Theorem 9, the sample complexity $\Omega(p^{d/2})$ is essential for any polynomial-time algorithm to succeed under proper assumptions. Therefore, no extra samples are needed for efficient estimators in moderate over-parameterized scalar-on-tensor regression; while such a phenomenon does not exist in its matrix counterpart. See Figure 1 for a pictorial illustration of this distinction.

In addition to the "spectral initialization + local refinement", random initialization + refinement by some simple local methods is another effective approach for solving matrix and tensor problems. Such a "random initialization + local refinement" scheme has been shown to be effective in over-parameterized matrix trace regression, where only $O(pr^{*2})$ samples are needed (Li, Ma and Zhang (2018)). However, initialization with a small enough magnitude and the factorization formulation seem to be critical there. Due to the space limit, we leave a thorough comparison of these two popular approaches for over-parameterized tensor-on-tensor regression problems as future work.

6. Technical contributions. We develop several technical tools to establish the theoretical results in this paper. We summarize them in this section.

Tackle over-parameterization in the convergence analysis. In the proof of Theorems 1 and 2, we first observe that for any $k\in[d+m]$, the mode-k singular subspace of $\boldsymbol{\mathcal{X}}^t$, denoted by $\mathbf{U}_k^t$, can be decomposed as $\mathbf{U}_k^t=[\overline{\mathbf{U}}_k^t \quad \check{\mathbf{U}}_k^t]$ where $\overline{\mathbf{U}}_k^t$ is composed of the first r_k^* columns

of $\mathbf{U}_k^t$ and $\check{\mathbf{U}}_k^t$ is composed of the rest of the $(r_k - r_k^*)$ columns of $\mathbf{U}_k^t$. Then the projection operator onto $\mathbf{U}_{k\perp}^t$, the orthogonal complement of $\mathbf{U}_k^t$, satisfies

$$P_{\mathbf{U}_{k\perp}^t} = \mathbf{I}_{p_k} - P_{\mathbf{U}_k^t} = \mathbf{I}_{p_k} - P_{\overline{\mathbf{U}}_k^t} - P_{\check{\mathbf{U}}_k^t} = (\mathbf{I}_{p_k} - P_{\check{\mathbf{U}}_k^t})(\mathbf{I}_{p_k} - P_{\overline{\mathbf{U}}_k^t}). \tag{17}$$

This implies $\|(\mathbf{I}_{p_k} - P_{\mathbf{U}_k^t})\mathbf{Z}\| \le \|(\mathbf{I}_{p_k} - P_{\overline{\mathbf{U}}_k^t})\mathbf{Z}\|$ for any matrix $\mathbf{Z}$ with compatible dimension. Based on this property, we can focus on the first r_k^* columns of $\mathbf{U}_k^t$ and establish the following lemma, which plays a key role in establishing the convergence of RGD and RGN.

LEMMA 3 (An over-parameterized projection error bound). Suppose $\boldsymbol{\mathcal{X}}^t \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}$ is an order-$(d+m)$ Tucker rank $\mathbf{r} := (r_1, \ldots, r_{d+m})$ tensor and $\boldsymbol{\mathcal{X}}^* \in \mathbb{R}^{p_1 \times \cdots \times p_{d+m}}$ is an order-$(d+m)$ Tucker rank $\mathbf{r}^* := (r_1^*, \ldots, r_{d+m}^*)$ tensor with $\mathbf{r}^* \le \mathbf{r}$. Then we have

$$\left\| P_{(T_{\boldsymbol{\mathcal{X}}^t})_\perp} \boldsymbol{\mathcal{X}}^* \right\|_{\mathrm{F}} \le \frac{2(d+m)\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}}^2}{\underline{\lambda}},$$

where $P_{(T_{\boldsymbol{\mathcal{X}}})_\perp} := \mathbf{I} - P_{T_{\boldsymbol{\mathcal{X}}}}$ is the orthogonal complement of the projector $P_{T_{\boldsymbol{\mathcal{X}}}}$ (5) and $\underline{\lambda} := \min_{k=1,\ldots,d+m} \sigma_{r_k^*}(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$. Especially in the matrix setting, that is, $d+m=2$, a sharper upper bound holds: $\|P_{(T_{\mathbf{X}^t})_\perp}\mathbf{X}^*\|_{\mathrm{F}} \le \frac{2\|\mathbf{X}^t - \mathbf{X}^*\|_{\mathrm{F}}^2}{\sigma_{r^*}(\mathbf{X}^*)}$.

Initialization guarantees for scalar-on-tensor regression and tensor-on-vector regression. A key step of Algorithms 2 and 3 is the one-iteration HOOI (OHOOI) algorithm (Algorithm 5 below). Such one loop update improves the dependence of r^* in sample complexity compared to the vanilla T-HOSVD based initialization in both scalar-on-tensor and tensor-on-vector regressions. In the proofs of Theorems 5 and 6, we develop the following deterministic tensor perturbation bound for OHOOI in the over-parameterized regime.

THEOREM 10 (Perturbation bound for over-parameterized tensor decomposition). *Suppose $\widetilde{\boldsymbol{\mathcal{T}}}, \boldsymbol{\mathcal{T}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$, $\boldsymbol{\mathcal{T}}$ is of Tucker rank $\mathbf{r}^* = (r_1^*, \ldots, r_d^*)$ with Tucker decomposition $\boldsymbol{\mathcal{B}} \times_1 \mathbf{U}_1 \times \cdots \times_d \mathbf{U}_d$, where $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{r_1^* \times \cdots \times r_d^*}$ and $\mathbf{U}_k \in \mathbb{O}_{p_k, r_k^*}$ for $k = 1, \ldots, d$. Let $\boldsymbol{\mathcal{Z}} = \widetilde{\boldsymbol{\mathcal{T}}} - \boldsymbol{\mathcal{T}}$. Suppose the inputs of the OHOOI algorithm are $\widetilde{\boldsymbol{\mathcal{T}}}$, Tucker rank $\mathbf{r} = (r_1, \ldots, r_d)$ with $\mathbf{r} \ge \mathbf{r}^*$ and initializations $\widetilde{\mathbf{U}}_k^0 \in \mathbb{O}_{p_k, r_k}$ for $k = 1, \ldots, d$. If the initialization error satisfies $\max_{k=1,\ldots,d} \|\widetilde{\mathbf{U}}_{k\perp}^{0\top}\mathbf{U}_k\| \le \frac{\sqrt{2}}{2}$. Then the output of Algorithm 5, $\widehat{\boldsymbol{\mathcal{T}}}$, satisfies $\|\widehat{\boldsymbol{\mathcal{T}}} - \boldsymbol{\mathcal{T}}\|_{\mathrm{F}} \le (2^{\frac{d+1}{2}} \cdot d + 1)\|\boldsymbol{\mathcal{Z}}_{\max(\mathbf{r})}\|_{\mathrm{F}}$.*

Low-degree polynomials evidence for problems with correlated structures. As we have mentioned in Remark 7, the main task in the proof of Theorem 9 is to compute the norm of the truncated likelihood ratio. See Supplement 7.2 for a preliminary of low-degree polynomials method. Since the data are i.i.d. Gaussian under the null hypothesis of (14), the main challenge boils down to computing the expected Hermite polynomials on correlated multivariate Gaussian. In the following Lemma 4, we provide a simple formula for that. This lemma can be

Algorithm 5 One-iteration higher-order orthogonal iteration (OHOOI)

1: **Input:** $\widetilde{\boldsymbol{\mathcal{T}}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$, initialization $\widetilde{\mathbf{U}}_k^0 \in \mathbb{O}_{p_k, r_k}$, $k = 1, \ldots, d$, input Tucker rank $\mathbf{r} = (r_1, \ldots, r_d)$.
2: For $k = 1$ to d, update $\widetilde{\mathbf{U}}_k^1 = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\widetilde{\boldsymbol{\mathcal{T}}} \times_{j<k} (\widetilde{\mathbf{U}}_j^0)^\top \times_{j>k} (\widetilde{\mathbf{U}}_j^0)^\top))$.
3: **Output:** $\widehat{\boldsymbol{\mathcal{T}}} = \widetilde{\boldsymbol{\mathcal{T}}} \times_{k=1}^d P_{\widetilde{\mathbf{U}}_k^1}$.

useful in establishing low-degree polynomial hardness evidence for other problems with complex structures. Let $\{h_k\}_{k\in\mathbb{N}}$ be the normalized univariate Hermite polynomials $h_k = \frac{1}{\sqrt{k!}}H_k$ where $\{H_k\}_{k\in\mathbb{N}}$ are univariate Hermite polynomials which are defined by the following recurrence: $H_0(x) = 1$, $H_1(x) = x$, $H_{k+1}(x) = xH_k(x) - kH_{k-1}(x)$ for $k \geq 1$.

LEMMA 4 (Expected Hermitian polynomials on correlated multivariate Gaussian). Suppose w is a positive integer, $Y \in \mathbb{R}$, $\mathbf{X} = (X_1, \ldots, X_w) \in \mathbb{R}^w$ are random variable and random vectors, respectively, and $(Y, \mathbf{X}) \sim \mathcal{N}(0, \left[\begin{smallmatrix} 1 & \mathbf{u}^\top \\ \mathbf{u} & \mathbf{I}_w \end{smallmatrix}\right])$ with $\mathbf{u} = (u_1, \ldots, u_w)$. For any integers $\alpha, \beta_1, \ldots, \beta_w \geq 0$, $\mathbb{E}(h_\alpha(Y)\prod_{j=1}^w h_{\beta_j}(X_j)) = \sqrt{\frac{\alpha!}{\prod_{j=1}^w \beta_j!}} \cdot \prod_{j=1}^w u_j^{\beta_j} 1(\alpha = \sum_{j=1}^w \beta_j)$, where $1(\cdot)$ in the indicator function.

7. Implementation details of RGD and RGN. In this section, we complement the implementation details of RGD and RGN proposed in Section 2.2.

Implementation of RGD. First, by the definition of the adjoint map, $\mathscr{A}^* : \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}} \to \mathbb{R}^{p_1\times\cdots\times p_{d+m}}$ satisfies $\langle \mathscr{A}(\boldsymbol{\mathcal{Z}}_1), \boldsymbol{\mathcal{Z}}_2\rangle = \langle \boldsymbol{\mathcal{Z}}_1, \mathscr{A}^*(\boldsymbol{\mathcal{Z}}_2)\rangle$ for any $\boldsymbol{\mathcal{Z}}_1 \in \mathbb{R}^{p_1\times\cdots\times p_{d+m}}$, $\boldsymbol{\mathcal{Z}}_2 \in \mathbb{R}^{n\times p_{d+1}\times\cdots\times p_{d+m}}$. Simple manipulation yields:

$$\mathscr{A}^*(\boldsymbol{\mathcal{Z}}_2)_{[k_1,\ldots,k_d,j_1,\ldots,j_m]} = \sum_{i=1}^n \boldsymbol{\mathcal{Z}}_{2[i,j_1,\ldots,j_m]}\boldsymbol{\mathcal{A}}_{i[k_1,\ldots,k_d]}.$$

Combining this with the formula of projection $P_{T_{\boldsymbol{\mathcal{X}}^t}}$ in (5), we can calculate $\boldsymbol{\mathcal{X}}^{t+0.5} = \boldsymbol{\mathcal{X}}^t - \alpha_t P_{T_{\boldsymbol{\mathcal{X}}^t}}\mathscr{A}^*(\mathscr{A}(\boldsymbol{\mathcal{X}}^t) - \boldsymbol{\mathcal{Y}})$ and implement the RGD update.

Implementation of RGN. To illustrate the implementation details of RGN, we first introduce the following lemma.

LEMMA 5 (Spectrum of $P_{T_{\boldsymbol{\mathcal{X}}^t}}\mathscr{A}^*\mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}$). Suppose $\boldsymbol{\mathcal{X}}^t$ is of Tucker rank at most $\mathbf{r}$ and the linear map $\mathscr{A}$ satisfies the $2\mathbf{r}$-TRIP. Then for any tensor $\boldsymbol{\mathcal{Z}} \in T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}$,

$$(1 - R_{2\mathbf{r}})\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{F}} \leq \left\|P_{T_{\boldsymbol{\mathcal{X}}^t}}\mathscr{A}^*\mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}(\boldsymbol{\mathcal{Z}})\right\|_{\mathrm{F}} \leq (1 + R_{2\mathbf{r}})\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{F}}, \tag{18}$$

and

$$\frac{\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{F}}}{1 + R_{2\mathbf{r}}} \leq \left\|\left(P_{T_{\boldsymbol{\mathcal{X}}^t}}\mathscr{A}^*\mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}\right)^{-1}(\boldsymbol{\mathcal{Z}})\right\|_{\mathrm{F}} \leq \frac{\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{F}}}{1 - R_{2\mathbf{r}}}. \tag{19}$$

Lemma 5 shows the linear operator $P_{T_{\boldsymbol{\mathcal{X}}^t}}\mathscr{A}^*\mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}$, which is a mapping from $T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}$ to itself, is provably invertible under TRIP condition, which further implies the least squares in RGN update, $\boldsymbol{\mathcal{X}}^{t+0.5} = \arg\min_{\boldsymbol{\mathcal{X}}\in T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}} \frac{1}{2}\|\boldsymbol{\mathcal{Y}} - \mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}(\boldsymbol{\mathcal{X}})\|_{\mathrm{F}}^2$, has a unique solution. In the following Proposition 2, we show that the RGN update can be reduced to solving $(m + 1)$ least squares, which renders a fast implementation of RGN.

PROPOSITION 2 (Efficient implementation of RGN update). *Suppose $\boldsymbol{\mathcal{X}}^t$ has Tucker decomposition $\boldsymbol{\mathcal{S}}^t \times_{k=1}^{d+m} \mathbf{U}_k^t$. Then the RGN update, that is, $\boldsymbol{\mathcal{X}}^{t+0.5} = \arg\min_{\boldsymbol{\mathcal{X}}\in T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}} \frac{1}{2}\|\boldsymbol{\mathcal{Y}} - \mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}(\boldsymbol{\mathcal{X}})\|_{\mathrm{F}}^2$, is equal to $\boldsymbol{\mathcal{X}}^{t+0.5} = \boldsymbol{\mathcal{B}}^t \times_{k=1}^{d+m} \mathbf{U}_k^t + \sum_{k=1}^{d+m} \boldsymbol{\mathcal{S}}^t \times_k \mathbf{U}_{k\perp}^t\mathbf{D}_k^t \times_{j\neq k} \mathbf{U}_j^t$, where:*

- $(\boldsymbol{\mathcal{B}}^t, \{\mathbf{D}_k^t\}_{k=1}^d)$ *is the solution of the following least squares with design matrix size* $n\prod_{l=d+1}^{d+m} r_l \times (\prod_{k=1}^{d+m} r_k + \sum_{k=1}^d r_k(p_k - r_k))$:

$$(\boldsymbol{\mathcal{B}}^t, \{\mathbf{D}_k^t\}_{k=1}^d)$$

$$= \underset{\substack{\boldsymbol{\mathcal{B}}\in\mathbb{R}^{r_1\times\cdots\times r_{d+m}},\\ \mathbf{D}_k\in\mathbb{R}^{(p_k-r_k)\times r_k},k=1,\ldots,d}}{\arg\min} \sum_{i=1}^{n}\Bigg\|\mathcal{Y}_i\times_{l=1}^{m}\mathbf{U}_{l+d}^{t\top}-\langle\boldsymbol{\mathcal{A}}_i\times_{j=1}^{d}\mathbf{U}_j^{t\top},\boldsymbol{\mathcal{B}}\rangle_*$$

$$-\sum_{k=1}^{d}\langle\boldsymbol{\mathcal{A}}\times_k\mathbf{U}_{k\perp}^{t\top}\times_{j\neq k}\mathbf{U}_j^{t\top},\boldsymbol{\mathcal{S}}^t\times_k\mathbf{D}_k\rangle_*\Bigg\|_{\mathrm{F}}^2$$

$$= \underset{\substack{\boldsymbol{\mathcal{B}}\in\mathbb{R}^{r_1\times\cdots\times r_{d+m}},\\ \mathbf{D}_k\in\mathbb{R}^{(p_k-r_k)\times r_k},k=1,\ldots,d}}{\arg\min} \sum_{i=1}^{n}\sum_{j_l\in[r_{d+l}],l=1,\ldots,m}$$

$$\Bigg((\mathcal{Y}_i\times_{l=1}^{m}\mathbf{U}_{l+d}^{t\top})_{[j_1,\ldots,j_m]}-\langle\boldsymbol{\mathcal{A}}_i\times_{j=1}^{d}\mathbf{U}_j^{t\top},\boldsymbol{\mathcal{B}}_{[:,\ldots,:,j_1,\ldots,j_m]}\rangle-$$

$$-\sum_{k=1}^{d}\langle\mathbf{U}_{k\perp}^{t\top}\mathcal{M}_k(\boldsymbol{\mathcal{A}}_i\times_{j\neq k}\mathbf{U}_j^{t\top})(\mathcal{M}_k(\boldsymbol{\mathcal{S}}^t_{[:,\ldots,:,j_1,\ldots,j_m]}))^\top,\mathbf{D}_k\rangle\Bigg)^2$$

- *for* $k=d+1,\ldots,d+m$,

$$\mathbf{D}_k^{t\top}=\underset{\mathbf{D}_k^\top\in\mathbb{R}^{r_k\times(p_k-r_k)}}{\arg\min}\|\mathbf{Y}_{ki}-\mathbf{A}_{ki}\mathbf{D}_k^\top\|_{\mathrm{F}}^2,$$

where

$$\mathbf{A}_{ki}=\big(\mathcal{M}_{k-d}\big(\langle\boldsymbol{\mathcal{A}}_i\times_{j=1}^{d}\mathbf{U}_j^{t\top},\boldsymbol{\mathcal{S}}^t\rangle_*\big)\big)^\top\in\mathbb{R}^{\prod_{l=d+1,l\neq k}^{d+m}r_l\times r_k},$$

$$\mathbf{Y}_{ki}=\big(\mathcal{M}_{k-d}\big(\mathcal{Y}_i\times_{l\neq k-d}\mathbf{U}_{l+d}^{t\top}\big)\big)^\top\mathbf{U}_{k\perp}^t\in\mathbb{R}^{\prod_{l=d+1,l\neq k}^{d+m}r_l\times(p_k-r_k)}.$$

In the tensor-on-vector regression ($d=1$), the update of RGN has a cleaner and fully closed expression as follows.

LEMMA 6 (RGN update in tensor-on-vector regression). Consider the RGN for tensor-on-vector regression in (11). Suppose $\mathbf{A}^\top\mathbf{A}$ is invertible where $\mathbf{A}$ is the collection of covariate vectors and the iterate at iteration t is $\boldsymbol{\mathcal{X}}^t=[\![\boldsymbol{\mathcal{S}}^t;\mathbf{U}_1^t,\mathbf{U}_2^t,\ldots,\mathbf{U}_{m+1}^t]\!]$. Then the solution $\boldsymbol{\mathcal{X}}^{t+0.5}$ in (8) has a closed-form expression:

$$\boldsymbol{\mathcal{X}}^{t+0.5}=\boldsymbol{\mathcal{B}}^t\times_{k=1}^{1+m}\mathbf{U}_k^t+\sum_{k=1}^{1+m}\boldsymbol{\mathcal{S}}^t\times_k\mathbf{U}_{k\perp}^t\mathbf{D}_k^t\times_{j\neq k}\mathbf{U}_j^t,$$

where

$$\begin{aligned}\mathcal{M}_1(\boldsymbol{\mathcal{B}}^t)=&(\mathbf{U}_1^{t\top}\mathbf{A}^\top\mathbf{A}\mathbf{U}_1^t)^{-1}\mathbf{U}_1^{t\top}\mathbf{A}^\top\\&\cdot(\mathcal{M}_1(\boldsymbol{\mathcal{Y}})\otimes_{j=(1+m)}^{2}\mathbf{U}_j^t\\&-\mathbf{A}\mathbf{U}_{1\perp}^t\mathbf{U}_{1\perp}^{t\top}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathcal{M}_1(\boldsymbol{\mathcal{Y}})\mathbf{W}_1^t\mathbf{V}_1^{t\top});\end{aligned}$$

$$\mathcal{M}_1(\boldsymbol{\mathcal{S}}^t\times_1\mathbf{U}_{1\perp}^t\mathbf{D}_1^t\times_{j\neq1}\mathbf{U}_j^t)=\mathbf{U}_{1\perp}^t\mathbf{U}_{1\perp}^{t\top}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathcal{M}_1(\boldsymbol{\mathcal{Y}})\mathbf{W}_1^t\mathbf{W}_1^{t\top};$$

and

$$\begin{aligned}&\mathcal{M}_k(\boldsymbol{\mathcal{S}}^t\times_k\mathbf{U}_{k\perp}^t\mathbf{D}_k^t\times_{j\neq k}\mathbf{U}_j^t)\\&\quad=\mathbf{U}_{k\perp}^t\mathbf{U}_{k\perp}^{t\top}\mathcal{M}_k(\boldsymbol{\mathcal{Y}}\times_1\mathbf{A}^\top)\mathbf{W}_k^t(\mathbf{V}_k^{t\top}(\otimes_{\substack{i\neq k,\\ i\neq1}}\mathbf{I}_{r_i}\otimes(\mathbf{U}_1^{t\top}\mathbf{A}^\top\mathbf{A}\mathbf{U}_1^t))\mathbf{V}_k^t)^{-1}\mathbf{W}_k^{t\top}\end{aligned}$$

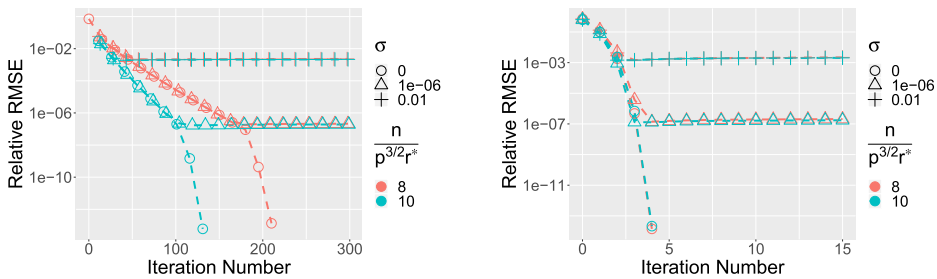
for $k=2,\ldots,(1+m)$. Recall $\mathbf{V}_k^t=\mathrm{QR}(\mathcal{M}_k(\boldsymbol{\mathcal{S}}^t)^\top)$ and $\mathbf{W}_k^t$ is defined in (4).

8. Numerical studies. We conduct simulation studies to investigate the numerical performance of RGD/RGN in tensor-on-tensor regression and to verify our theoretical findings. In each simulation, we generate $\boldsymbol{\mathcal{E}}_i$ with i.i.d. $N(0, \sigma^2)$ entries, $\boldsymbol{\mathcal{A}}_i$ with i.i.d. $N(0, 1)$ entries, $\{\mathbf{U}_k\}_{k=1}^{d+m}$ uniformly at random from $\mathbb{O}_{p,r^*}$ for some to-be-specified p and r^*, and $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{r^* \times \cdots \times r^*}$ with i.i.d. $N(0, 1)$ entries; then we form $\boldsymbol{\mathcal{X}}^* = \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1 \times \cdots \times_{d+m} \mathbf{U}_{d+m}$ and generate $\boldsymbol{\mathcal{Y}}_i$ for $i = 1, \ldots, n$. The input rank of RGD and RGN is set to be $\mathbf{r} = (r, \ldots, r)$ and $r \geq r^*$. In the simulation study, we will experiment with various values of r. Additionally, r can be chosen by a data-driven approach. See Supplement 3 for details. For simplicity, we mainly focus on two examples: scalar-on-tensor regression and tensor-on-vector regression. In the scalar-on-tensor regression, we consider $d = 3$; in the tensor-on-vector regression, we consider $m = 3$. Spectral initializations discussed in Section 4 are applied in both examples.

Throughout the simulation studies, the error metric we consider is the relative root mean squared error (Relative RMSE) $\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}} / \|\boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}}$. The algorithm is terminated when it reaches the maximum number of iterations $t_{\max} = 300$ or the corresponding error metric is less than 10^{-13}. Unless otherwise noted, the reported results are based on averages of 100 simulations and on a computer with Intel Xeon E5-2680 2.5 GHz CPU.

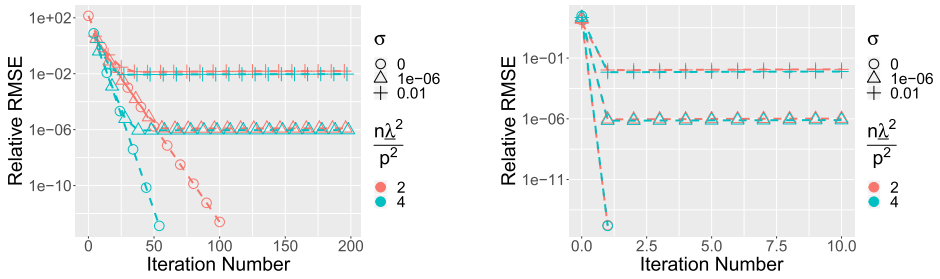
8.1. *Numerical performance of RGD and RGN.* In this simulation, we examine the convergence rate of RGD/RGN in over-parameterized scalar-on-tensor regression and tensor-on-vector regression. We set $\sigma \in \{0, 10^{-6}, 10^{-2}\}$, $p = 30$, $r^* = 3$ and $r = 10$. In scalar-on-tensor regression, we choose n such that $\frac{n}{p^{3/2} r^*} \in \{8, 10\}$; in tensor-on-vector regression, we let $\frac{n\underline{\lambda}^2}{p^2} \in \{2, 4\}$ where $\underline{\lambda} = \min_k \sigma_{r^*}(\mathcal{M}_k(\boldsymbol{\mathcal{S}}))$. The convergence performance of RGD and RGN in scalar-on-tensor regression and tensor-on-vector regression are presented in Figures 3 and 4, respectively. In both examples, we find the estimation error of RGD converges linearly to the minimum precision in the noiseless setting and converges linearly to a limit determined by the noise level in the noisy setting. In scalar-on-tensor regression, we find RGN converges quadratically and in tensor-on-vector regression, we observe RGN converges with almost one iteration. We tried several other simulation settings and observed the similar phenomenon.

8.2. *Effect of input rank and sample size on the performance of RGD and RGN.* We also examine the effect of input rank r and sample size n on the convergence of RGD and RGN and we focus on the scalar-on-tensor regression example. We let $p = 30$, $r^* = 3$, $\sigma = 10^{-6}$, $n \in [500, 8000]$ and input rank $r \in \{3, 6, 9, 12, 15\}$. The performance of RGD and RGN in this simulation study is given in Figure 5. We can see that for both RGD and RGN, the sample size requirement for convergence increases as the input rank r increases. For a fixed n, the relative RMSE attainable by RGD and RGN increases as the input rank increases.



(a) Scalar-on-tensor regression: RGD (b) Scalar-on-tensor regression: RGN

FIG. 3. *Convergence performance of RGD/RGN in over-parameterized scalar-on-tensor regression with spectral initialization. Here,* $p = 30$, $r^* = 3$, $r = 10$.



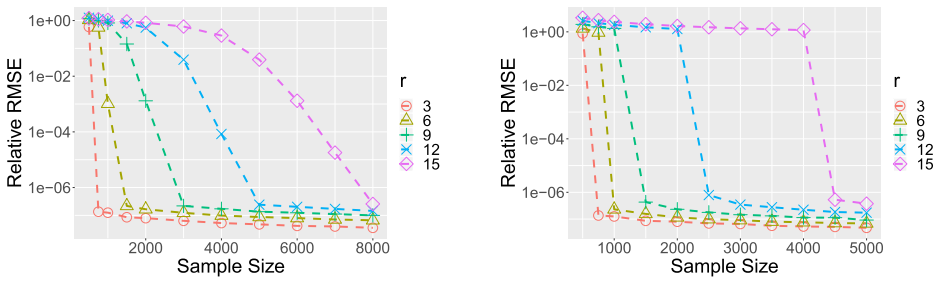
(a) Tensor-on-vector regression: RGD (b) Tensor-on-vector regression: RGN

FIG. 4. *Convergence performance of RGD/RGN in over-parameterized tensor-on-vector regression with spectral initialization. Here* $p = 30$, $r^* = 3$, $r = 10$.

In addition, the phase transition on the sample complexity for the failure/success in RGN is sharper than the one in RGD. This is because RGN enjoys a higher-order convergence compared to RGD and RGD converges slowly when the number of samples is around the threshold. This matches our main theoretical results in Sections 3 and 4. Moreover, our results suggest that the number of samples needed for the convergence of RGD and RGN increases at the scale of r^d for large r (here $d = 3$) and this is indeed suggested in Figure 6 after we plot the cubic root of the sample size with respect to Relative RMSE.

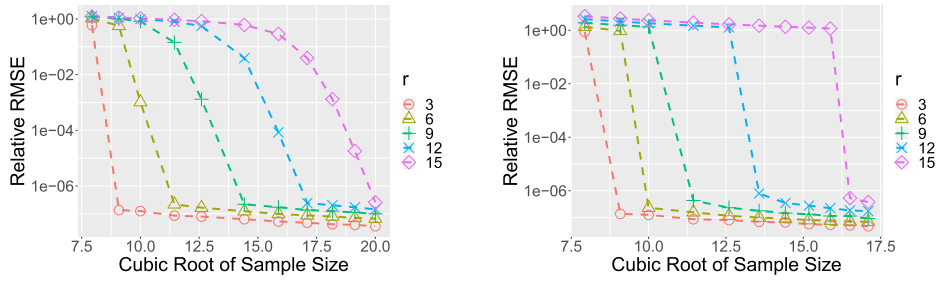
8.3. *Scalar-on-tensor regression versus matrix trace regression under over-parameterization.* In this simulation, we compare the sample size requirements to ensure successful recovery in over-parameterized scalar-on-tensor regression and matrix trace regression with an increasing input rank via RGD. We focus on the noiseless setting, that is, $\sigma = 0$. We say an algorithm achieves successful recovery if the averaged relative root mean squared error (Relative RMSE) $\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_F / \|\boldsymbol{\mathcal{X}}^*\|_F$ is smaller than 0.01. In scalar-on-tensor regression, we set $p = 90$, $r^* = 1$, $r \in [1, \ldots, 8]$, $n = [800, 900, \ldots, 3500]$ and in the matrix trace regression, we set $p = 100$, $r^* = 1$, $r \in [1, \ldots, 8]$ and $n = [200, \ldots, 3000]$. For every input rank r, we increase the sample size by 100 at each time from the one that ensures the successful recovery with input rank $r - 1$ until RGD succeeds.

Figure 7 shows as the input rank increases, the line of triangles for the sample size requirement of successful recovery in scalar-on-tensor regression is flat at the beginning stage while increases for large input r. In contrast, the sample size requirement for successful recovery of RGD in the matrix trace regression always increases linearly as input rank increases. This matches our theoretical findings in Section 5 that a "free lunch" on the sample complexity appears in over-parameterized scalar-on-tensor regression, but not in the matrix trace regression. Meanwhile, Figure 7 shows when the input rank is equal to r^*, the phase transitions



(a) RGD (b) RGN

FIG. 5. *Convergence performance of RGD/RGN in over-parameterized scalar-on-tensor regression with spectral initialization. Here* $p = 30$, $r^* = 3$, $n \in [500, 8000]$, $r \in \{3, 6, 9, 12, 15\}$.



(a) RGD (b) RGN

FIG. 6. *Rescaled plot for the convergence performance of RGD/RGN in over-parameterized scalar-on-tensor regression with spectral initialization. Here* $p = 30$, $r^* = 3$, $n \in [500, 8000]$, $r \in \{3, 6, 9, 12, 15\}$.

on sample complexity for the failure/success of RGD in matrix trace regression and scalar-on-tensor regression appear around $n = 300 \approx 2pr^*$ and $n = 1000 \approx p^{3/2}r^*$, respectively. This matches the results in Section 5 that there is a statistical-computational gap in scalar-on-tensor regression and $\Omega(p^{d/2})$ (here $d = 3$) samples are needed for any polynomial-time algorithm to succeed.

8.4. *Comparison of Riemannian optimization methods with existing algorithms.* In the second simulation, we compare RGN with other existing algorithms, including alternating minimization (Alter Mini) (Zhou, Li and Zhu (2013)), projected gradient descent (PGD) (Rauhut, Schneider and Stojanac (2017)), gradient descent (GD) (Han, Willett and Zhang (2022)) and scaled gradient descent (Tong et al. (2022)), in both exact and over-parameterized scalar-on-tensor regression. While implementing PGD, GD and scaled GD, we evaluate five choices of step size, $\frac{1}{n} \cdot \{0.1, 0.25, 0.5, 0.75, 1\}$, then choose the best one following Zheng and Lafferty (2015). We set $p = 30$, $r^* = 3$, $r \in \{3, 10\}$, $n = 8p^{3/2}r^*$ and consider the noiseless case ($\sigma = 0$). Figure 8 shows RGN converges quadratically in both settings, while the other baseline algorithms converge at a much slower linear rate. Moreover, when we go from exact-parameterization (Panel (a)) to over-parameterization (Panel (b)), the convergence rate of all baseline algorithms slows down significantly while RGN maintains its robust and fast second-order convergence performance.

9. Conclusion and discussions. In this work, we propose Riemannian gradient descent and Riemannian Gauss–Newton methods for solving the general tensor-on-tensor regression. We provide optimal statistical and computational guarantees for these algorithms in both rank correctly-specified and overspecified settings and discover an intriguing blessing

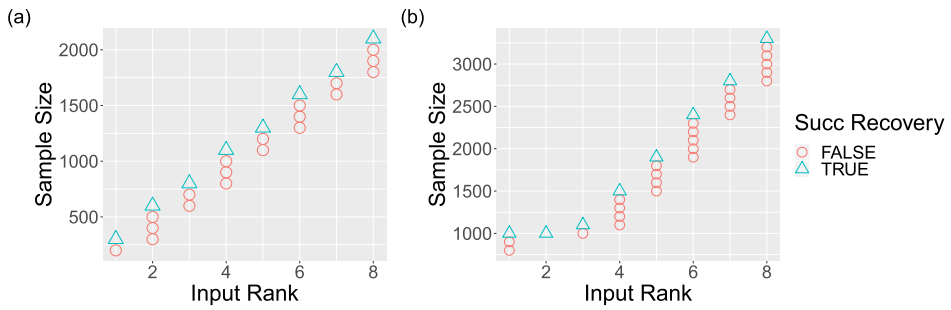


FIG. 7. *Comparison of successful recovery of RGD under over-parameterized matrix trace regression (Panel (a)) and scalar-on-tensor regression (Panel (b)).*

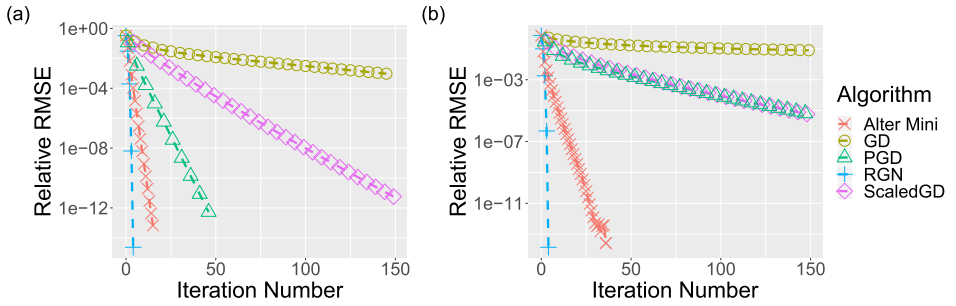


FIG. 8. *Panel* (*a*): $r = 3$; *Panel* (*b*): $r = 10$. *Relative RMSE of RGN* (*this work*), *alternating minimization* (*Alter Mini*), *projected gradient descent* (*PGD*), *gradient descent* (*GD*) *and scaled gradient descent* (*ScaledGD*) *in noiseless scalar-on-tensor regression*.

of the statistical-computational gap in the over-parameterized scalar-on-tensor regression. Our current initialization and computational results are established for several representative examples. It is of great interest to see whether these results can be extended to the general tensor-on-tensor regression problem. Moreover, the rank overspecification studied in this paper falls in the moderate over-parameterized regime in the sense that the model still includes more samples than the degree of freedom of parameters. It is interesting to consider the highly over-parameterized regime and study the analogy of implicit regularization effect (Gunasekar et al. (2017), Li, Ma and Zhang (2018)) in factorization formulated tensor problems. Some progress has been made recently in the tensor decomposition setting (Razin, Maman and Cohen (2021), Ge et al. (2021)).

Acknowledgements. The authors would like to thank Ilias Diakonikolas and Daniel Kane for helpful discussions. Diakonikolas and Kane developed a computational lower bound in the statistical query model (which further yields a low-degree polynomial computational lower bound) for low-rank scalar-on-tensor rank-one regression before this work; and the proof was later incorporated into a full paper in Diakonikolas et al. (2023). However, the low-degree polynomial computational lower bound in Theorem 9 of this paper is tighter and its proof is direct and arguably simpler. We also thank the Editor, the Associated Editor and two anonymous referees for their helpful suggestions, which helped improve the presentation and quality of this paper.

Funding. The research is supported in part by NSF Grant CAREER-2203741.

SUPPLEMENTARY MATERIAL

Supplement to "Tensor-on-tensor regression: Riemannian optimization, over-parameterization, statistical-computational gap and their interplay" (DOI: 10.1214/24-AOS2396SUPP; .pdf). The supplement (Luo and Zhang (2024)) contains a table of contents, detailed algorithms and all technical proofs.

REFERENCES

ABSIL, P.-A., MAHONY, R. and SEPULCHRE, R. (2008). *Optimization Algorithms on Matrix Manifolds*. Princeton Univ. Press, Princeton, NJ. With a foreword by Paul Van Dooren. MR2364186 https://doi.org/10.1515/9781400830244

AHMED, T., RAJA, H. and BAJWA, W. U. (2020). Tensor regression using low-rank and sparse Tucker decompositions. *SIAM J. Math. Data Sci.* **2** 944–966. MR4161310 https://doi.org/10.1137/19M1299335

ANANDKUMAR, A., GE, R., HSU, D., KAKADE, S. M. and TELGARSKY, M. (2014). Tensor decompositions for learning latent variable models. *J. Mach. Learn. Res.* **15** 2773–2832. MR3270750

BANDEIRA, A. S., KUNISKY, D. and WEIN, A. S. (2020). Computational hardness of certifying bounds on constrained PCA problems. In 11*th Innovations in Theoretical Computer Science Conference*. *LIPIcs*. *Leibniz Int. Proc. Inform.* **151** Art. No. 78, 29. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern. MR4048181

BARAK, B., HOPKINS, S., KELNER, J., KOTHARI, P. K., MOITRA, A. and POTECHIN, A. (2019). A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM J. Comput.* **48** 687–735. MR3945259 https://doi.org/10.1137/17M1138236

BARAK, B. and MOITRA, A. (2016). Noisy tensor completion via the sum-of-squares hierarchy. In *Conference on Learning Theory* 417–445.

BARBER, R. F. and HA, W. (2018). Gradient descent with non-convex constraints: Local concavity determines convergence. *Inf. Inference* **7** 755–806. MR4023770 https://doi.org/10.1093/imaiai/iay002

BARTLETT, P. L., LONG, P. M., LUGOSI, G. and TSIGLER, A. (2020). Benign overfitting in linear regression. *Proc. Natl. Acad. Sci. USA* **117** 30063–30070. MR4263288 https://doi.org/10.1073/pnas.1907378117

BARTLETT, P. L., MONTANARI, A. and RAKHLIN, A. (2021). Deep learning: A statistical viewpoint. *Acta Numer.* **30** 87–201. MR4295218 https://doi.org/10.1017/S0962492921000027

BELKIN, M. (2021). Fit without fear: Remarkable mathematical phenomena of deep learning through the prism of interpolation. *Acta Numer.* **30** 203–248. MR4298218 https://doi.org/10.1017/S0962492921000039

BELKIN, M., HSU, D., MA, S. and MANDAL, S. (2019). Reconciling modern machine-learning practice and the classical bias-variance trade-off. *Proc. Natl. Acad. Sci. USA* **116** 15849–15854. MR3997901 https://doi.org/10.1073/pnas.1903070116

BI, X., QU, A. and SHEN, X. (2018). Multilayer tensor factorization with applications to recommender systems. *Ann. Statist.* **46** 3308–3333. MR3852653 https://doi.org/10.1214/17-AOS1659

BOUMAL, N. (2023). *An Introduction to Optimization on Smooth Manifolds*. Cambridge Univ. Press, Cambridge. MR4533407

BOUMAL, N. and ABSIL, P.-A. (2011). Rtrmc: A Riemannian trust-region method for low-rank matrix completion. In *Advances in Neural Information Processing Systems* 406–414.

BREIDING, P. and VANNIEUWENHOVEN, N. (2018). A Riemannian trust region method for the canonical tensor rank approximation problem. *SIAM J. Optim.* **28** 2435–2465. MR3852721 https://doi.org/10.1137/17M114618X

BRENNAN, M. and BRESLER, G. (2020). Reducibility and statistical-computational gaps from secret leakage. In *Conference on Learning Theory* 648–847. PMLR.

BRESLER, G. and HUANG, B. (2022). The algorithmic phase transition of random k-SAT for low degree polynomials. In 2021 *IEEE* 62*nd Annual Symposium on Foundations of Computer Science—FOCS* 2021 298–309. IEEE Comput. Soc., Los Alamitos, CA. MR4399691

CAI, J.-F., LI, J. and XIA, D. (2022). Provable tensor-train format tensor completion by Riemannian optimization. *J. Mach. Learn. Res.* **23** 5365–5441. MR4577075

CAI, J.-F., LI, J. and XIA, D. (2023). Generalized low-rank plus sparse tensor estimation by fast Riemannian optimization. *J. Amer. Statist. Assoc.* **118** 2588–2604. MR4681606 https://doi.org/10.1080/01621459.2022.2063131

CANDÈS, E. J. and PLAN, Y. (2011). Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Trans. Inf. Theory* **57** 2342–2359. MR2809094 https://doi.org/10.1109/TIT.2011.2111771

CHEN, H., RASKUTTI, G. and YUAN, M. (2019). Non-convex projected gradient descent for generalized low-rank tensor regression. *J. Mach. Learn. Res.* **20** 172–208. MR3911412

CHOO, D. and D'ORSI, T. (2021). The complexity of sparse tensor pca. *Adv. Neural Inf. Process. Syst.* **34**.

DAVIS, D., DIAZ, M. and WANG, K. (2021). Clustering a mixture of Gaussians with unknown covariance. ArXiv preprint. Available at arXiv:2110.01602.

DE LATHAUWER, L., DE MOOR, B. and VANDEWALLE, J. (2000a). A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* **21** 1253–1278. MR1780272 https://doi.org/10.1137/S0895479896305696

DE LATHAUWER, L., DE MOOR, B. and VANDEWALLE, J. (2000b). On the best rank-1 and rank-$(R_1, R_2, \ldots, R_N)$ approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.* **21** 1324–1342. MR1780276 https://doi.org/10.1137/S0895479898346995

DIAKONIKOLAS, I., KANE, D. M., LUO, Y. and ZHANG, A. (2023). Statistical and computational limits for tensor-on-tensor association detection. In *The Thirty Sixth Annual Conference on Learning Theory* 5260–5310. PMLR.

DING, L., JIANG, L., CHEN, Y., QU, Q. and ZHU, Z. (2021a). Rank overspecified robust matrix recovery: Subgradient method and exact recovery. *Adv. Neural Inf. Process. Syst.* **34** 26767–26778.

DING, Y., KUNISKY, D., WEIN, A. S. and BANDEIRA, A. S. (2021b). The average-case time complexity of certifying the restricted isometry property. *IEEE Trans. Inf. Theory* **67** 7355–7361. MR4345126 https://doi.org/10.1109/TIT.2021.3112823

DING, Y., KUNISKY, D., WEIN, A. S. and BANDEIRA, A. S. (2024). Subexponential-time algorithms for sparse PCA. *Found. Comput. Math.* **24** 865–914. MR4760356 https://doi.org/10.1007/s10208-023-09603-0

DONG, S., GAO, B., GUAN, Y. and GLINEUR, F. (2022). New Riemannian preconditioned algorithms for tensor completion via polyadic decomposition. *SIAM J. Matrix Anal. Appl.* **43** 840–866. MR4426891 https://doi.org/10.1137/21M1394734

DUDEJA, R. and HSU, D. (2021). Statistical query lower bounds for tensor PCA. *J. Mach. Learn. Res.* **22** Paper No. 83, 51. MR4253776

ELDÉN, L. and SAVAS, B. (2009). A Newton–Grassmann method for computing the best multilinear rank-(r_1, r_2, r_3) approximation of a tensor. *SIAM J. Matrix Anal. Appl.* **31** 248–271. MR2496418 https://doi.org/10.1137/070688316

FAN, J., YANG, Z. and YU, M. (2023). Understanding implicit regularization in over-parameterized single index model. *J. Amer. Statist. Assoc.* **118** 2315–2328. MR4681585 https://doi.org/10.1080/01621459.2022.2044824

GAHROOEI, M. R., YAN, H., PAYNABAR, K. and SHI, J. (2021). Multiple tensor-on-tensor regression: An approach for modeling processes with heterogeneous sources of data. *Technometrics* **63** 147–159. MR4251490 https://doi.org/10.1080/00401706.2019.1708463

GE, R., REN, Y., WANG, X. and ZHOU, M. (2021). Understanding deflation process in over-parametrized tensor decomposition. *Adv. Neural Inf. Process. Syst.* **34**.

GUHANIYOGI, R., QAMAR, S. and DUNSON, D. B. (2017). Bayesian tensor regression. *J. Mach. Learn. Res.* **18** Paper No. 79, 31. MR3714242

GUNASEKAR, S., WOODWORTH, B. E., BHOJANAPALLI, S., NEYSHABUR, B. and SREBRO, N. (2017). Implicit regularization in matrix factorization. *Adv. Neural Inf. Process. Syst.* **30**.

HAN, R., LUO, Y., WANG, M. and ZHANG, A. R. (2022). Exact clustering in tensor block model: Statistical optimality and computational limit. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **84** 1666–1698. MR4515554

HAN, R., WILLETT, R. and ZHANG, A. R. (2022). An optimal statistical and computational framework for generalized tensor estimation. *Ann. Statist.* **50** 1–29. MR4382094 https://doi.org/10.1214/21-AOS2061

HAO, B., ZHANG, A. and CHENG, G. (2020). Sparse and low-rank tensor estimation via cubic sketchings. *IEEE Trans. Inf. Theory* **66** 5927–5964. MR4158653 https://doi.org/10.1109/TIT.2020.2982499

HEIDEL, G. and SCHULZ, V. (2018). A Riemannian trust-region method for low-rank tensor completion. *Numer. Linear Algebra Appl.* **25** e2175, 16. MR3890978 https://doi.org/10.1002/nla.2175

HILLAR, C. J. and LIM, L.-H. (2013). Most tensor problems are NP-hard. *J. ACM* **60** Art. 45, 39. MR3144915 https://doi.org/10.1145/2512329

HOFF, P. D. (2015). Multilinear tensor regression for longitudinal relational data. *Ann. Appl. Stat.* **9** 1169–1193. MR3418719 https://doi.org/10.1214/15-AOAS839

HOPKINS, S. (2018). Statistical inference and the sum of squares method. Ph.D. thesis. MR3864930

HOPKINS, S. B., KOTHARI, P. K., POTECHIN, A., RAGHAVENDRA, P., SCHRAMM, T. and STEURER, D. (2017). The power of sum-of-squares for detecting hidden structures. In 58*th Annual IEEE Symposium on Foundations of Computer Science—FOCS* 2017 720–731. IEEE Comput. Soc., Los Alamitos, CA. MR3734275 https://doi.org/10.1109/FOCS.2017.72

HOPKINS, S. B. and STEURER, D. (2017). Efficient Bayesian estimation from few samples: Community detection and related problems. In 58*th Annual IEEE Symposium on Foundations of Computer Science—FOCS* 2017 379–390. IEEE Comput. Soc., Los Alamitos, CA. MR3734245 https://doi.org/10.1109/FOCS.2017.42

HOU, T. Y., LI, Z. and ZHANG, Z. (2020). Fast global convergence for low-rank matrix recovery via Riemannian gradient descent with random initialization. ArXiv preprint. Available at arXiv:2012.15467.

HUANG, W. and HAND, P. (2018). Blind deconvolution by a steepest descent algorithm on a quotient manifold. *SIAM J. Imaging Sci.* **11** 2757–2785. MR3882949 https://doi.org/10.1137/17M1151390

ISHTEVA, M., DE LATHAUWER, L., ABSIL, P.-A. and VAN HUFFEL, S. (2009). Differential-geometric Newton method for the best rank-(R_1, R_2, R_3) approximation of tensors. *Numer. Algorithms* **51** 179–194. MR2505840 https://doi.org/10.1007/s11075-008-9251-2

JIANG, L., CHEN, Y. and DING, L. (2023). Algorithmic regularization in model-free overparametrized asymmetric matrix factorization. *SIAM J. Math. Data Sci.* **5** 723–744. MR4626339 https://doi.org/10.1137/22M1519833

KASAI, H. and MISHRA, B. (2016). Low-rank tensor completion: A Riemannian manifold preconditioning approach. In *International Conference on Machine Learning* 1012–1021. PMLR.

KESHAVAN, R. H., MONTANARI, A. and OH, S. (2010). Matrix completion from a few entries. *IEEE Trans. Inf. Theory* **56** 2980–2998. MR2683452 https://doi.org/10.1109/TIT.2010.2046205

KOCH, O. and LUBICH, C. (2010). Dynamical tensor approximation. *SIAM J. Matrix Anal. Appl.* **31** 2360–2375. MR2685162 https://doi.org/10.1137/09076578X

KOLDA, T. G. and BADER, B. W. (2009). Tensor decompositions and applications. *SIAM Rev.* **51** 455–500. MR2535056 https://doi.org/10.1137/07070111X

KRESSNER, D., STEINLECHNER, M. and VANDEREYCKEN, B. (2014). Low-rank tensor completion by Riemannian optimization. *BIT* **54** 447–468. MR3223510 https://doi.org/10.1007/s10543-013-0455-z

KRESSNER, D., STEINLECHNER, M. and VANDEREYCKEN, B. (2016). Preconditioned low-rank Riemannian optimization for linear systems with tensor product structure. *SIAM J. Sci. Comput.* **38** A2018–A2044. MR3519141 https://doi.org/10.1137/15M1032909

KUNISKY, D., WEIN, A. S. and BANDEIRA, A. S. (2022). Notes on computational hardness of hypothesis testing: Predictions using the low-degree likelihood ratio. In *Mathematical Analysis, Its Applications and Computation. Springer Proc. Math. Stat.* **385** 1–50. Springer, Cham. MR4461037 https://doi.org/10.1007/978-3-030-97127-4_1

LEVIN, E., KILEEL, J. and BOUMAL, N. (2023). Finding stationary points on bounded-rank matrices: A geometric hurdle and a smooth remedy. *Math. Program.* **199** 831–864. MR4578384 https://doi.org/10.1007/s10107-022-01851-2

LI, L. and ZHANG, X. (2017). Parsimonious tensor response regression. *J. Amer. Statist. Assoc.* **112** 1131–1146. MR3735365 https://doi.org/10.1080/01621459.2016.1193022

LI, Y., MA, T. and ZHANG, H. (2018). Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *Conference on Learning Theory* 2–47. PMLR.

LI, Z., LUO, Y. and LYU, K. (2020). Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*.

LIU, Y., LIU, J. and ZHU, C. (2020). Low-rank tensor train coefficient array estimation for tensor-on-tensor regression. *IEEE Trans. Neural Netw. Learn. Syst.* **31** 5402–5411. MR4189257 https://doi.org/10.1109/tnnls.2020.2967022

LLOSA, C. and MAITRA, R. (2022). Reduced-rank tensor-on-tensor regression and tensor-variate analysis of variance. *IEEE Trans. Pattern Anal. Mach. Intell.*

LOCK, E. F. (2018). Tensor-on-tensor regression. *J. Comput. Graph. Statist.* **27** 638–647. MR3863764 https://doi.org/10.1080/10618600.2017.1401544

LÖFFLER, M., WEIN, A. S. and BANDEIRA, A. S. (2022). Computationally efficient sparse clustering. *Inf. Inference* **11** 1255–1286. MR4526323 https://doi.org/10.1093/imaiai/iaac019

LUO, Y., HUANG, W., LI, X. and ZHANG, A. (2024). Recursive importance sketching for rank constrained least squares: Algorithms and high-order convergence. *Oper. Res.* **72** 237–256. MR4705836

LUO, Y. and ZHANG, A. R. (2022). Tensor clustering with planted structures: Statistical optimality and computational limits. *Ann. Statist.* **50** 584–613. MR4382029 https://doi.org/10.1214/21-aos2123

LUO, Y. and ZHANG, A. R. (2023). Low-rank tensor estimation via Riemannian Gauss–Newton: Statistical optimality and second-order convergence. *J. Mach. Learn. Res.* **24** Paper No. 381, 48. MR4720837 https://doi.org/10.59277/pra-ser.a.24.4.09

LUO, Y. and ZHANG, A. R. (2024). Supplement to "Tensor-on-Tensor Regression: Riemannian Optimization, over-parameterization, Statistical-computational Gap, and Their Interplay." https://doi.org/10.1214/24-AOS2396SUPP

LYU, Z. and XIA, D. (2023). Optimal estimation and computational limit of low-rank Gaussian mixtures. *Ann. Statist.* **51** 646–667. MR4600996 https://doi.org/10.1214/23-aos2264

MA, J. and FATTAHI, S. (2023). Global convergence of sub-gradient method for robust matrix recovery: Small initialization, noisy measurements, and over-parameterization. *J. Mach. Learn. Res.* **24** Paper No. [96], 84. MR4582518

MAO, C. and WEIN, A. S. (2021). Optimal spectral recovery of a planted vector in a subspace. ArXiv preprint. Available at arXiv:2105.15081.

MEYER, G., BONNABEL, S. and SEPULCHRE, R. (2011). Linear regression under fixed-rank constraints: A Riemannian approach. In *Proceedings of the* 28*th International Conference on Machine Learning*.

MISHRA, B., MEYER, G., BONNABEL, S. and SEPULCHRE, R. (2014). Fixed-rank matrix factorizations and Riemannian low-rank optimization. *Comput. Statist.* **29** 591–621. MR3261830 https://doi.org/10.1007/s00180-013-0464-z

MU, C., HUANG, B., WRIGHT, J. and GOLDFARB, D. (2014). Square deal: Lower bounds and improved relaxations for tensor recovery. In *ICML* 73–81.

OLIKIER, G. and ABSIL, P.-A. (2023). An apocalypse-free first-order low-rank optimization algorithm with at most one rank reduction attempt per iteration. *SIAM J. Matrix Anal. Appl.* **44** 1421–1435. MR4644394 https://doi.org/10.1137/22M1518256

RABUSSEAU, G. and KADRI, H. (2016). Low-rank regression with tensor responses. *Adv. Neural Inf. Process. Syst.* **29**.

RASKUTTI, G., YUAN, M. and CHEN, H. (2019). Convex regularization for high-dimensional multiresponse tensor regression. *Ann. Statist.* **47** 1554–1584. MR3911122 https://doi.org/10.1214/18-AOS1725

RAUHUT, H., SCHNEIDER, R. and STOJANAC, Ž. (2017). Low rank tensor recovery via iterative hard thresholding. *Linear Algebra Appl.* **523** 220–262. MR3624675 https://doi.org/10.1016/j.laa.2017.02.028

RAZIN, N., MAMAN, A. and COHEN, N. (2021). Implicit regularization in tensor factorization. In *International Conference on Machine Learning* 8913–8924. PMLR.

RECHT, B., FAZEL, M. and PARRILO, P. A. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.* **52** 471–501. MR2680543 https://doi.org/10.1137/070697835

RICHARD, E. and MONTANARI, A. (2014). A statistical model for tensor pca. *Adv. Neural Inf. Process. Syst.* 2897–2905.

SAVAS, B. and LIM, L.-H. (2010). Quasi-Newton methods on Grassmannians and multilinear approximations of tensors. *SIAM J. Sci. Comput.* **32** 3352–3393. MR2746624 https://doi.org/10.1137/090763172

SCHNEIDER, R. and USCHMAJEW, A. (2015). Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality. *SIAM J. Optim.* **25** 622–646. MR3323551 https://doi.org/10.1137/140957822

SOLTANOLKOTABI, M., JAVANMARD, A. and LEE, J. D. (2019). Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Trans. Inf. Theory* **65** 742–769. MR3904911 https://doi.org/10.1109/TIT.2018.2854560

SORBER, L., VAN BAREL, M. and DE LATHAUWER, L. (2013). Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-$(L_r, L_r, 1)$ terms, and a new generalization. *SIAM J. Optim.* **23** 695–720. MR3044107 https://doi.org/10.1137/120868323

STEINLECHNER, M. (2016). Riemannian optimization for high-dimensional tensor completion. *SIAM J. Sci. Comput.* **38** S461–S484. MR3565572 https://doi.org/10.1137/15M1010506

STÖGER, D. and SOLTANOLKOTABI, M. (2021). Small random initialization is akin to spectral learning: Optimization and generalization guarantees for overparameterized low-rank matrix reconstruction. *Adv. Neural Inf. Process. Syst.* **34**.

SUN, W. W. and LI, L. (2017). STORE: Sparse tensor response regression and neuroimaging analysis. *J. Mach. Learn. Res.* **18** Paper No. 135, 37. MR3763769

TONG, T., MA, C., PRATER-BENNETTE, A., TRIPP, E. and CHI, Y. (2022). Scaling and scalability: Provable nonconvex low-rank tensor estimation from incomplete measurements. *J. Mach. Learn. Res.* **23** Paper No. [163], 77. MR4577115

TUCKER, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika* **31** 279–311. MR0205395 https://doi.org/10.1007/BF02289464

USCHMAJEW, A. and VANDEREYCKEN, B. (2013). The geometry of algorithms using hierarchical tensors. *Linear Algebra Appl.* **439** 133–166. MR3045227 https://doi.org/10.1016/j.laa.2013.03.016

VANDEREYCKEN, B. (2013). Low-rank matrix completion by Riemannian optimization. *SIAM J. Optim.* **23** 1214–1236. MR3069099 https://doi.org/10.1137/110845768

VANNIEUWENHOVEN, N., VANDEBRIL, R. and MEERBERGEN, K. (2012). A new truncation strategy for the higher-order singular value decomposition. *SIAM J. Sci. Comput.* **34** A1027–A1052. MR2914314 https://doi.org/10.1137/110836067

WANG, H., CHEN, J. and WEI, K. (2023). Implicit regularization and entrywise convergence of Riemannian optimization for low Tucker-rank tensor completion. *J. Mach. Learn. Res.* **24** Paper No. [347], 84. MR4690296

YU, R. and LIU, Y. (2016). Learning from multiway data: Simple and efficient tensor regression. In *International Conference on Machine Learning* 373–381. PMLR.

WEI, K., CAI, J.-F., CHAN, T. F. and LEUNG, S. (2016). Guarantees of Riemannian optimization for low rank matrix recovery. *SIAM J. Matrix Anal. Appl.* **37** 1198–1222. MR3543156 https://doi.org/10.1137/15M1050525

XIA, D. and YUAN, M. (2019). On polynomial time methods for exact low-rank tensor completion. *Found. Comput. Math.* **19** 1265–1313. MR4029842 https://doi.org/10.1007/s10208-018-09408-6

XIA, D., ZHANG, A. R. and ZHOU, Y. (2022). Inference for low-rank tensors—no need to debias. *Ann. Statist.* **50** 1220–1245. MR4404934 https://doi.org/10.1214/21-aos2146

ZHANG, A. and XIA, D. (2018). Tensor SVD: Statistical and computational limits. *IEEE Trans. Inf. Theory* **64** 7311–7338. MR3876445 https://doi.org/10.1109/TIT.2018.2841377

ZHANG, J., FATTAHI, S. and ZHANG, R. (2021). Preconditioned gradient descent for over-parameterized nonconvex matrix factorization. *Adv. Neural Inf. Process. Syst.* **34**.

ZHENG, Q. and LAFFERTY, J. (2015). A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. In *Advances in Neural Information Processing Systems* 109–117.

ZHOU, H., LI, L. and ZHU, H. (2013). Tensor regression with applications in neuroimaging data analysis. *J. Amer. Statist. Assoc.* **108** 540–552. MR3174640 https://doi.org/10.1080/01621459.2013.776499

ZHUO, J., KWON, J., HO, N. and CARAMANIS, C. (2024). On the computational and statistical complexity of over-parameterized matrix sensing. *J. Mach. Learn. Res.* **25** Paper No. [169], 47. MR4777411