PROVABLE SECOND-ORDER RIEMANNIAN GAUSS-NEWTON METHOD FOR LOW-RANK TENSOR ESTIMATION[‖]

Yuetian Luo[*] *Qin Ma*[†] *Chi Zhang*[‡] *Anru R. Zhang*[$]

ABSTRACT

In this paper, we consider the estimation of a low Tucker rank tensor from a number of noisy linear measurements. We propose a Riemannian Gauss-Newton (RGN) method with fast implementations for low Tucker rank tensor estimation. Different from the generic (super)linear convergence guarantee of RGN in the literature, we prove the first quadratic convergence guarantee of RGN for low-rank tensor estimation under some mild conditions. A deterministic estimation error lower bound, which matches the upper bound, is provided that demonstrates the statistical optimality of RGN. The merit of RGN is illustrated through applications of tensor regression and tensor SVD.

Index Terms— Low-rank tensor estimation, quadratic convergence, Riemannian optimization, statistical optimality

1. INTRODUCTION

We consider a prototypical model for tensor estimation:

$$\mathbf{y} = \mathscr{A}(\boldsymbol{\mathcal{X}}^*) + \boldsymbol{\varepsilon}. \tag{1}$$

Here, $\mathbf{y}, \boldsymbol{\varepsilon} \in \mathbb{R}^n$ are the observations and unknown noise and $\boldsymbol{\mathcal{X}}^* \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ is an order-d tensor parameter of interest. $\mathscr{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d} \to \mathbb{R}^n$ is a known linear map, which can be explicitly expressed as $\mathscr{A}(\boldsymbol{\mathcal{X}}^*) = [\langle \boldsymbol{\mathcal{A}}_1, \boldsymbol{\mathcal{X}}^* \rangle, \ldots, \langle \boldsymbol{\mathcal{A}}_n, \boldsymbol{\mathcal{X}}^* \rangle]^\top$, $\langle \boldsymbol{\mathcal{A}}_i, \boldsymbol{\mathcal{X}}^* \rangle = \sum_{1 \leqslant i_k \leqslant p_k, 1 \leqslant k \leqslant d} (\boldsymbol{\mathcal{A}}_i)_{[i_1,\ldots,i_d]} \boldsymbol{\mathcal{X}}^*_{[i_1,\ldots,i_d]}$ with the given measurement tensors $\{\boldsymbol{\mathcal{A}}_i\}_{i=1}^n \subseteq \mathbb{R}^{p_1 \times \cdots \times p_d}$. Our goal is to estimate $\boldsymbol{\mathcal{X}}^*$ based on $(\mathbf{y}, \mathscr{A})$.

In many applications, $\prod_{k=1}^d p_k$, i.e., the number of parameters in $\boldsymbol{\mathcal{X}}^*$, is much greater than the sample size n, so some structural conditions are often assumed to ensure the problem is well-posed. In the literature, the low-rankness assumption was widely considered [1, 2, 3]. In this work, we focus on the setting that the target parameter $\boldsymbol{\mathcal{X}}^*$ is low Tucker rank and admits the following Tucker (or multilinear) decomposition with Tucker rank $\mathbf{r} = (r_1, \ldots, r_d)$: $\boldsymbol{\mathcal{X}}^* = \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1 \times \cdots \times_d \mathbf{U}_d$. Here, $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ is the order-d core tensor; $\mathbf{U}_k$ is a p_k-by-r_k matrix with orthonormal columns, which represents the mode-k singular vectors of $\boldsymbol{\mathcal{X}}^*$; "$\times_k$" is the tensor-matrix product along mode k.

With different designs of $\mathscr{A}$, the general model (1) covers many specific settings arising from applications, such as

1. *Tensor regression.* Specifically, the Gaussian ensemble design ($\boldsymbol{\mathcal{A}}_i$ has i.i.d. Gaussian/sub-Gaussian entries) is widely studied in the literature.
2. *Tensor completion* [4, 5]: $\boldsymbol{\mathcal{A}}_i = \mathbf{e}_{a_1^{(i)}} \circ \cdots \circ \mathbf{e}_{a_d^{(i)}}$, $\mathbf{e}_{a_k^{(i)}}$ is the $a_k^{(i)}$th canonical vector and $\{a_1^{(i)}, \cdots, a_d^{(i)}\}_{i=1}^n$ are randomly selected integers from $[p_1] \times \cdots \times [p_d]$, "$\circ$" represents the outer product and $[p_k] = \{1, \ldots, p_k\}$;
3. *Tensor estimation via rank-1 projections* [6]: $\boldsymbol{\mathcal{A}}_i = \mathbf{a}_1^{(i)} \circ \cdots \circ \mathbf{a}_d^{(i)}$, where $\{\mathbf{a}_k^{(i)} \in \mathbb{R}^{p_k}\}_{k=1}^d$ are random vectors;
4. *Tensor PCA/SVD* [7, 8] is a special case of tensor completion where all entries are observable. In this particular setting, we can tensorize $\mathbf{y}, \boldsymbol{\varepsilon}$ and rewrite the model (1) equivalently to $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}}^* + \boldsymbol{\mathcal{E}}$. Here $\boldsymbol{\mathcal{X}}^*$ is the low Tucker rank signal tensor and $\boldsymbol{\mathcal{E}}$ is the noise.

In view of model (1), a natural estimator of $\boldsymbol{\mathcal{X}}^*$ is

$$\begin{aligned} \widehat{\boldsymbol{\mathcal{X}}} = & \mathop{\arg\min}_{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}} f(\boldsymbol{\mathcal{X}}) := \frac{1}{2} \|\mathbf{y} - \mathscr{A}(\boldsymbol{\mathcal{X}})\|_2^2, \\ & \text{subject to} \quad \mathrm{Tucrank}(\boldsymbol{\mathcal{X}}) = \mathbf{r}. \end{aligned} \tag{2}$$

Here $\mathrm{Tucrank}(\boldsymbol{\mathcal{X}})$ is the Tucker rank of $\boldsymbol{\mathcal{X}}$. However, the optimization problem in (2) is non-convex and NP-hard in general. To tame the non-convexity, a common scheme is the convex relaxation [9, 10]. However, this scheme may either obtain suboptimal statistical guarantees or require evaluating the tensor nuclear norm, which is NP-hard to compute in general [11].

Our Contributions. In this paper, we develop a new Riemannian Gauss-Newton (RGN) algorithm for low-rank tensor estimation. The proposed algorithm is tuning free and generally has the same per-iteration computational complexity as the alternating minimization [2, 12] and comparable complexity to the other first-order methods including projected gradient descent [13] and gradient descent [14].

Moreover, assuming $\mathscr{A}$ satisfies the tensor restricted isometry property (TRIP) (see Definition 2), we prove that

[*]University of Wisconsin-Madison

[†]Ohio State University

[‡]Indiana University School of Medicine

[§]Duke University. To whom correspondence should be addressed.

[‖]The research of Y. Luo and A. R. Zhang was supported in part by NSF Grant CAREER-1944904. The research of Y. Luo, Q. Ma, C. Zhang, A. R. Zhang was supported in part by NIH Grant R01 GM131399.

978-1-6654-0540-9/22/$31.00 ©2022 IEEE

with some proper initialization, the iterates generated by RGN converge quadratically to $\boldsymbol{\mathcal{X}}^*$ up to some statistical error. Especially in the noiseless setting, i.e., $\varepsilon = 0$, RGN converges quadratically to the exact parameter $\boldsymbol{\mathcal{X}}^*$. Since RGN generally converges to a point with nonzero function value in the noisy setting, the generic theory on RGN can only guarantee a (super)linear convergence rate to a stationary point [15]. Our result complements the classic theory of RGN: we show RGN converges quadratically to a neighborhood of the true parameter of interest, which achieves a statistically optimal estimation error rate. To our best knowledge, such a result is new and our RGN is the first algorithm with a provable guarantee of second-order convergence for the low-rank tensor estimation. Furthermore, we provide a deterministic minimax lower bound for the estimation error under model (1). The lower bound matches the estimation error upper bound, which demonstrates the statistical rate-optimality of RGN.

Next, we apply RGN to two problems arising from applications in machine learning and statistics: tensor regression and tensor SVD. In both problems, we prove the iterates of RGN converge quadratically to a neighborhood of $\boldsymbol{\mathcal{X}}^*$ that achieves the minimax optimal estimation error.

Notation and Preliminaries. 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., $\boldsymbol{\mathcal{A}}$) are used to denote scalars, vectors, matrices, and order-3-or-higher tensors, respectively. We use bracket subscripts to denote sub-vectors, sub-matrices, and sub-tensors. 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 orthogonalization of $\mathbf{D}$, respectively. Let $\mathbb{O}_{p,r} = \{\mathbf{U} : \mathbf{U}^\top \mathbf{U} = \mathbf{I}_r\}$ be the set of all p-by-r matrices with orthonormal columns. For any $\mathbf{U} \in \mathbb{O}_{p,r}$, $P_{\mathbf{U}} = \mathbf{U}\mathbf{U}^\top$ represents the projection matrix onto the column space of $\mathbf{U}$; we use $\mathbf{U}_\perp \in \mathbb{O}_{p,p-r}$ to represent the orthonormal complement of $\mathbf{U}$.

The matricization $\mathcal{M}_k(\cdot)$ is the operation that unfolds the order-d tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ along mode k into the matrix $\mathcal{M}_k(\boldsymbol{\mathcal{A}}) \in \mathbb{R}^{p_k \times p_{-k}}$ where $p_{-k} = \prod_{j \neq k} p_j$. We also use notation $\mathcal{T}_k(\cdot)$ to denote the mode-k tensorization or reverse operator of $\mathcal{M}_k(\cdot)$. Throughout the paper, $\mathcal{T}_k$, as a reversed operation of $\mathcal{M}_k(\cdot)$, maps an $\mathbb{R}^{p_k \times p_{-k}}$ matrix back to an $\mathbb{R}^{p_1 \times \cdots \times p_d}$ tensor. The Hilbert-Schmidt norm of $\boldsymbol{\mathcal{A}}$ is defined as $\|\boldsymbol{\mathcal{A}}\|_{\mathrm{HS}} = (\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}} \rangle)^{1/2}$. The Tucker rank of a tensor $\boldsymbol{\mathcal{A}}$ is denoted by $\mathrm{Tucrank}(\boldsymbol{\mathcal{A}})$ and defined as a d-tuple $\mathbf{r} := (r_1, \ldots, r_d)$, where $r_k = \mathrm{rank}(\mathcal{M}_k(\boldsymbol{\mathcal{A}}))$. For any Tucker rank-$(r_1, \ldots, r_d)$ tensor $\boldsymbol{\mathcal{A}}$, it has Tucker decomposition [16]: $\boldsymbol{\mathcal{A}} = [\![\boldsymbol{\mathcal{S}}; \mathbf{U}_1, \ldots, \mathbf{U}_d]\!] := \boldsymbol{\mathcal{S}} \times_1 \mathbf{U}_1 \times \cdots \times_d \mathbf{U}_d$, where $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ is the core tensor; $\mathbf{U}_k = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{A}}))$ is the mode-k singular vectors. Here, the mode-k product of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ with a matrix $\mathbf{B} \in \mathbb{R}^{r_k \times p_k}$ is denoted by $\boldsymbol{\mathcal{A}} \times_k \mathbf{B}$ and is of size $p_1 \times \cdots \times p_{k-1} \times r_k \times p_{k+1} \times \cdots \times p_d$.

2. ALGORITHM

Next, we introduce the geometry of low Tucker rank tensor Riemannian manifolds and present the procedure of RGN.

Geometry for Low Tucker Rank Tensor Manifolds. Denote the collection of $(p_1, \ldots, p_d)$-dimensional tensors of Tucker rank $\mathbf{r}$ by $\mathbb{M}_{\mathbf{r}} = \{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}, \mathrm{Tucrank}(\boldsymbol{\mathcal{X}}) = \mathbf{r}\}$. Then $\mathbb{M}_{\mathbf{r}}$ forms a smooth submanifold embedded in $\mathbb{R}^{p_1 \times \cdots \times p_d}$. Throughout the paper, we use the natural Euclidean inner product as the Riemannian metric. Suppose $\boldsymbol{\mathcal{X}} \in \mathbb{M}_{\mathbf{r}}$ has Tucker decomposition $[\![\boldsymbol{\mathcal{S}}; \mathbf{U}_1, \ldots, \mathbf{U}_d]\!]$; [17] showed the tangent space of $\mathbb{M}_{\mathbf{r}}$ at $\boldsymbol{\mathcal{X}}$, $T_{\boldsymbol{\mathcal{X}}}\mathbb{M}_{\mathbf{r}}$ can be represented as

$$T_{\boldsymbol{\mathcal{X}}}\mathbb{M}_{\mathbf{r}} = \Big\{ \boldsymbol{\mathcal{B}} \times_{k=1}^d \mathbf{U}_k + \sum_{k=1}^d \boldsymbol{\mathcal{S}} \times_k \bar{\mathbf{D}}_k \times_{j \neq k} \mathbf{U}_j : \begin{array}{l} \boldsymbol{\mathcal{B}} \in \mathbb{R}^{r_1 \times \cdots \times r_d}, \bar{\mathbf{D}}_k \in \mathbb{R}^{p_k \times r_k}, \\ \bar{\mathbf{D}}_k^\top \mathbf{U}_k = 0, k = 1, \ldots, d \end{array} \Big\}. \tag{3}$$

For $\boldsymbol{\mathcal{X}} = [\![\boldsymbol{\mathcal{S}}; \mathbf{U}_1, \ldots, \mathbf{U}_d]\!]$, we let $\mathbf{V}_k = \mathrm{QR}(\mathcal{M}_k(\boldsymbol{\mathcal{S}})^\top)$, which corresponds to the row space of $\mathcal{M}_k(\boldsymbol{\mathcal{S}})$, and define

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

$\mathbf{U}_k, \mathbf{W}_k$ correspond to the subspaces of the column and row spans of $\mathcal{M}_k(\boldsymbol{\mathcal{X}})$, respectively.

By Lemma 3.1 of [17] and the tangent space representation, we can write the projection operator $P_{T_{\boldsymbol{\mathcal{X}}}}$ that projects any tensor $\boldsymbol{\mathcal{Z}}$ onto the tangent space of $\mathbb{M}_{\mathbf{r}}$ at $\boldsymbol{\mathcal{X}}$ as:

$$\begin{aligned} P_{T_{\boldsymbol{\mathcal{X}}}}(\boldsymbol{\mathcal{Z}}) &:= \mathcal{L}\mathcal{L}^*(\boldsymbol{\mathcal{Z}}) \\ &= \boldsymbol{\mathcal{Z}} \times_{k=1}^d P_{\mathbf{U}_k} + \sum_{k=1}^d \mathcal{T}_k(P_{\mathbf{U}_{k\perp}} \mathcal{M}_k(\boldsymbol{\mathcal{Z}}) P_{\mathbf{W}_k}), \end{aligned} \tag{5}$$

where $\mathcal{L}^*$ and $\mathcal{L}$ are respectively the contraction map and extension map defined as follows:

$$\begin{aligned} &\mathcal{L} : (\boldsymbol{\mathcal{B}}, \{\mathbf{D}_k\}_{k=1}^d) \mapsto \boldsymbol{\mathcal{B}} \times_{k=1}^d \mathbf{U}_k + \sum_{k=1}^d \mathcal{T}_k(\mathbf{U}_{k\perp} \mathbf{D}_k \mathbf{W}_k^\top), \\ &\mathcal{L}^* : \boldsymbol{\mathcal{Z}} \mapsto (\boldsymbol{\mathcal{Z}} \times_{k=1}^d \mathbf{U}_k^\top, \{\mathbf{U}_{k\perp}^\top \mathcal{M}_k(\boldsymbol{\mathcal{Z}}) \mathbf{W}_k\}_{k=1}^d). \end{aligned} \tag{6}$$

In particular, $\mathcal{L}^*$ is the adjoint operator of $\mathcal{L}$.

Riemannian Optimization and Gauss-Newton. Riemannian optimization concerns optimizing a real-valued function f defined on a Riemannian manifold $\mathbb{M}$, for which the readers are referred to [15] and [18] for an introduction. A typical procedure of a Riemannian optimization method contains three steps per iteration: Step 1. find the tangent space; Step 2. update the point on the tangent space; Step 3. map the point from the tangent space back to the manifold.

Low-rank tensor Riemannian manifold (Step 1). We have already discussed the tangent space of low Tucker rank tensor manifolds earlier.

Update on tangent space (Step 2). Next, we describe the procedure of RGN in the tangent space. We begin by introducing a few more preliminaries for Riemannian manifold optimization. The Riemannian gradient of a smooth function $f : \mathbb{M}_{\mathbf{r}} \to \mathbb{R}$ at $\boldsymbol{\mathcal{X}} \in \mathbb{M}_{\mathbf{r}}$ is defined as the unique tangent vector $\operatorname{grad} f(\boldsymbol{\mathcal{X}}) \in T_{\boldsymbol{\mathcal{X}}}\mathbb{M}_{\mathbf{r}}$ such that $\langle \operatorname{grad} f(\boldsymbol{\mathcal{X}}), \boldsymbol{\mathcal{Z}} \rangle = \mathrm{D} f(\boldsymbol{\mathcal{X}})[\boldsymbol{\mathcal{Z}}], \forall \boldsymbol{\mathcal{Z}} \in T_{\boldsymbol{\mathcal{X}}}\mathbb{M}_{\mathbf{r}}$, where $\mathrm{D}f(\boldsymbol{\mathcal{X}})[\boldsymbol{\mathcal{Z}}]$ denotes the directional derivative of f at point $\boldsymbol{\mathcal{X}}$ along direction $\boldsymbol{\mathcal{Z}}$. Specifically for the embedded submanifold $\mathbb{M}_{\mathbf{r}}$, we have:

Lemma 1 *For $f(\boldsymbol{\mathcal{X}})$ in* (2), $\operatorname{grad} f(\boldsymbol{\mathcal{X}}) = P_{T_{\boldsymbol{\mathcal{X}}}}(\mathscr{A}^*(\mathscr{A}(\boldsymbol{\mathcal{X}})-\mathbf{y}))$, *where $P_{T_{\boldsymbol{\mathcal{X}}}}(\cdot)$ is the projection operator onto the tangent space of $\mathbb{M}_{\mathbf{r}}$ at $\boldsymbol{\mathcal{X}}$ defined in* (5).

A common way to derive RGN update in the literature is to first write down the Riemannian Newton equation, then replace the Riemannian Hessian by its Gauss-Newton approximation [15, Chapter 8.4.1], and finally solve the modified Riemannian Newton equation, i.e., the Riemannian Gauss-Newton equation. In our low-rank tensor estimation problem with the objective function (2), suppose the current iterate is $\boldsymbol{\mathcal{X}}^t$, the RGN update $\eta^{RGN} \in T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}$ should solve the following RGN equation [15, Chapter 8.4],

$$-\operatorname{grad} f(\boldsymbol{\mathcal{X}}^t) = P_{T_{\boldsymbol{\mathcal{X}}^t}}\left(\mathscr{A}^*(\mathscr{A}(\eta^{RGN}))\right). \tag{7}$$

However, it is unclear how to solve (7) directly in practice.

Inspired by the classical Gauss-Newton (GN) algorithm, we instead introduce another scheme to derive RGN. Recall in solving the nonlinear least squares problem in the Euclidean space $\min_x \frac{1}{2}\|h(x)\|_2^2$, the classic Gauss-Newton can be viewed as a modified Newton method, and can also be derived by replacing the non-linear function $h(x)$ by its local linear approximation at the current iterate x_k [19, Chapter 10.3]. These two ways of interpretations are equivalent. Similar local linearization idea can be extended to the manifold setting except that the linearization needs to be taken in the tangent space in each iterate. Specifically, consider the objective function $f(\boldsymbol{\mathcal{X}})$ in (2), the linearization of $\mathbf{y} - \mathscr{A}(\boldsymbol{\mathcal{X}})$ at $\boldsymbol{\mathcal{X}}^t$ in $T_{\boldsymbol{\mathcal{X}}^t}\mathbb{M}_{\mathbf{r}}$ is $\mathbf{y} - \mathscr{A}(\boldsymbol{\mathcal{X}}^t) - \mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}(\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}^t)$, which can be simplified to $\mathbf{y} - \mathscr{A}P_{T_{\boldsymbol{\mathcal{X}}^t}}(\boldsymbol{\mathcal{X}})$. So the update can be calculated by

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

We can show the proposed update derived in (8) actually matches the standard RGN update (7) and (8).

Proposition 1 $\boldsymbol{\mathcal{X}}^{t+0.5} - \boldsymbol{\mathcal{X}}^t$ *is the Riemannian Gauss-Newton update, i.e., it solves the Riemannian Gauss-Newton equation* (7). *Moreover,* (8) *can be equivalently solved by*

$$\boldsymbol{\mathcal{X}}^{t+0.5} = \mathcal{L}_t(\mathbf{B}^{t+1}, \mathbf{D}_1^{t+1}, \cdots, \mathbf{D}_d^{t+1}),$$

where $\mathcal{L}_t$ and $\mathcal{L}_t^*$ are defined in the similar way as in (6) *except evaluated on $\boldsymbol{\mathcal{X}}^t = [\![\boldsymbol{\mathcal{S}}^t; \mathbf{U}_1^t, \ldots, \mathbf{U}_d^t]\!]$,*

$$\begin{aligned}&(\boldsymbol{\mathcal{B}}^{t+1}, \mathbf{D}_1^{t+1}, \cdots, \mathbf{D}_d^{t+1})\\ &= \underset{\substack{\boldsymbol{\mathcal{B}} \in \mathbb{R}^{r_1 \times \cdots \times r_d},\\ \mathbf{D}_k \in \mathbb{R}^{(p_k - r_k) \times r_k}}}{\arg\min} \left\| \mathbf{y} - \mathscr{A}_{\boldsymbol{\mathcal{B}}}(\boldsymbol{\mathcal{B}}) - \sum_{k=1}^d \mathscr{A}_{\mathbf{D}_k}(\mathbf{D}_k) \right\|_2^2.\end{aligned} \tag{9}$$

Retraction (Step 3). We apply *retraction* [15, Chapter 4] to map the point $\boldsymbol{\mathcal{X}}^{t+0.5}$ from the tangent space back to the manifold. In the low Tucker rank tensor manifolds, Proposition 2.3 of [20] showed that the truncated high-order singular value decomposition (T-HOSVD) [21] is a retraction. We further show that the sequentially truncated HOSVD (ST-HOSVD) [22], a computationally more efficient procedure than T-HOSVD, also satisfies the retraction properties.

Summary of RGN. We give the complete RGN algorithm for low-rank tensor estimation in Algorithm 1.

```
Algorithm 1 Riemannian Gauss-Newton for Tensor
Input: $\mathbf{y} \in \mathbb{R}^n, \boldsymbol{\mathcal{A}}_1, \ldots, \boldsymbol{\mathcal{A}}_n \in \mathbb{R}^{p_1 \times \cdots \times p_d}, t_{\max}$, Tucker rank $\mathbf{r}$, initialization $\boldsymbol{\mathcal{X}}^0 = [\![\boldsymbol{\mathcal{S}}^0; \mathbf{U}_1^0, \ldots, \mathbf{U}_d^0]\!]$, $\mathbf{W}_k^0$ defined as (4).
for $t = 0, 1, \ldots, t_{\max} - 1$ do
    Construct the covariates maps $\mathscr{A}_{\boldsymbol{\mathcal{B}}} : \mathbb{R}^{r_1 \times \cdots \times r_d} \to \mathbb{R}^n, \mathscr{A}_{\mathbf{D}_k} : \mathbb{R}^{(p_k - r_k) r_k} \to \mathbb{R}^n, k = 1, \cdots, d$, where
        $(\mathscr{A}_{\boldsymbol{\mathcal{B}}})_i = \boldsymbol{\mathcal{A}}_i \times_{k=1}^d \mathbf{U}_k^{t\top}, (\mathscr{A}_{\mathbf{D}_k})_i = \mathbf{U}_{k\perp}^{t\top} \mathcal{M}_k(\boldsymbol{\mathcal{A}}_i) \mathbf{W}_k^t.$
    Solve the unconstrained least squares problem (9).
    Update
        $\boldsymbol{\mathcal{X}}^{t+1} = [\![\boldsymbol{\mathcal{S}}^{t+1}; \mathbf{U}_1^{t+1}, \ldots, \mathbf{U}_d^{t+1}]\!]$
        $= \mathcal{H}_{\mathbf{r}}\left(\boldsymbol{\mathcal{B}}^{t+1} \times_{k=1}^d \mathbf{U}_k^t + \sum_{k=1}^d \mathcal{T}_k(\mathbf{U}_{k\perp}^t \mathbf{D}_k^{t+1} \mathbf{W}_k^{t\top})\right)$
    and $\mathbf{W}_k^{t+1}$ via (4). Here $\mathcal{H}_{\mathbf{r}}(\cdot)$ is the retraction map onto $\mathbb{M}_{\mathbf{r}}$ (two choices are ST-HOSVD and T-HOSVD).
end for
Output: $\boldsymbol{\mathcal{X}}^{t_{\max}}$
```

3. THEORETICAL ANALYSIS

We analyze the convergence rate of RGN in this section. Different from the low-rank matrix projection, which can be efficiently and exactly computed via truncated SVD, performing low-rank tensor projection exactly, even for $\mathbf{r} = 1$, can be NP hard in general [11]. We thus introduce the following quasi-projection property and the approximation constant $\delta(d)$.

Definition 1 *Let $P_{\mathbb{M}_{\mathbf{r}}}(\cdot)$ be the projection map from $\mathbb{R}^{p_1 \times \cdots \times p_d}$ to the tensor space of Tucker rank at most $\mathbf{r}$, i.e., for any*

$\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$ *and* $\widehat{\boldsymbol{\mathcal{Z}}}$ *of Tucker rank at most* $\mathbf{r}$*, one always has* $\|\boldsymbol{\mathcal{Z}} - \widehat{\boldsymbol{\mathcal{Z}}}\|_{\mathrm{HS}} \geqslant \|\boldsymbol{\mathcal{Z}} - P_{\mathbb{M}_{\mathbf{r}}}(\boldsymbol{\mathcal{Z}})\|_{\mathrm{HS}}$.

We say $\mathcal{H}_{\mathbf{r}}$ *satisfies the quasi-projection property with approximation constant* $\delta(d)$ *if* $\|\boldsymbol{\mathcal{Z}} - \mathcal{H}_{\mathbf{r}}(\boldsymbol{\mathcal{Z}})\|_{\mathrm{HS}} \leqslant \delta(d)\|\boldsymbol{\mathcal{Z}} - P_{\mathbb{M}_{\mathbf{r}}}(\boldsymbol{\mathcal{Z}})\|_{\mathrm{HS}}$ *for any* $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{p_1 \times \cdots \times p_d}$.

It is known that T-HOSVD and ST-HOSVD satisfy the *quasi-projection property* with approximation constant $\delta(d) = \sqrt{d}$ (see Chapter 10 in [23]).

For technical convenience, we also assume $\mathscr{A}$ satisfies the following Tensor Restricted Isometry Property (TRIP) [24]. TRIP condition can be seen as a tensor generalization of the restricted isometry property (RIP). In the compressed-sensing and low-rank matrix recovery literature, the RIP condition has been widely used as one standard assumption [25, 26]. [24] showed that TRIP condition holds if $\mathscr{A}$ is randomly designed with a sufficient sample size n.

Definition 2 *Let* $\mathscr{A} : \mathbb{R}^{p_1 \times \cdots \times p_d} \to \mathbb{R}^n$ *be a linear map. For a fixed d-tuple* $\mathbf{r} = (r_1, \ldots, r_d)$ *with* $1 \leqslant r_k \leqslant p_k$ *for* $k = 1, \ldots, d$*, define the* $\mathbf{r}$*-tensor restricted isometry constant to be the smallest number* $R_{\mathbf{r}}$ *such that* $(1 - R_{\mathbf{r}})\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{HS}}^2 \leqslant \|\mathscr{A}(\boldsymbol{\mathcal{Z}})\|_2^2 \leqslant (1+R_{\mathbf{r}})\|\boldsymbol{\mathcal{Z}}\|_{\mathrm{HS}}^2$ *holds for all* $\boldsymbol{\mathcal{Z}}$ *of Tucker rank at most* $\mathbf{r}$*. If* $0 \leqslant R_{\mathbf{r}} < 1$*, we say* $\mathscr{A}$ *satisfies* $\mathbf{r}$*-tensor restricted isometry property (*$\mathbf{r}-$*TRIP).*

Theorem 1 (Convergence of RGN) *Suppose* $\mathcal{H}_{\mathbf{r}}$ *is either T-HOSVD or ST-HOSVD,* $\mathscr{A}$ *satisfies the* $3\mathbf{r}$*-TRIP, and initialization* $\boldsymbol{\mathcal{X}}^0$ *satisfies* $\|\boldsymbol{\mathcal{X}}^0 - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}} \leqslant \frac{\underline{\lambda}}{4d(\sqrt{d}+1)(R_{3\mathbf{r}}/(1-R_{2\mathbf{r}})+1)}$*, where* $\underline{\lambda} := \min_{k=1,\ldots,d} \sigma_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$ *is the minimum of least singular values at matricizations of* $\boldsymbol{\mathcal{X}}^*$*. Then* $\forall t \geqslant 0$*,*

$$\begin{aligned}\|\boldsymbol{\mathcal{X}}^{t+1} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}} \leqslant & d(\sqrt{d}+1)\left(\frac{R_{3\mathbf{r}}}{1-R_{2\mathbf{r}}}+1\right)\frac{\|\boldsymbol{\mathcal{X}}^t - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}}^2}{\underline{\lambda}} \\ & + \frac{\sqrt{d}+1}{1-R_{2\mathbf{r}}}\|(\mathscr{A}^*(\boldsymbol{\varepsilon}))_{\max(2\mathbf{r})}\|_{\mathrm{HS}}.\end{aligned}$$

Here, $(\cdot)_{\max(\mathbf{r})}$ *denotes the best Tucker rank* $\mathbf{r}$ *approximation of the tensor “$\cdot$".*

Theorem 1 shows with some proper assumptions on $\mathscr{A}$ and initialization, the iterates of RGN converge quadratically to the ball centered at $\boldsymbol{\mathcal{X}}^*$ and of radius $O(\|(\mathscr{A}^*(\boldsymbol{\varepsilon}))_{\max(2\mathbf{r})}\|_{\mathrm{HS}})$. Especially if $\boldsymbol{\varepsilon} = 0$, i.e., the observations are noiseless, $\boldsymbol{\mathcal{X}}^t$ converges quadratically to the exact $\boldsymbol{\mathcal{X}}^*$. To the best of our knowledge, this is the first provable quadratic convergence guarantee for both low-rank tensor estimation and recovery.

Next, we further introduce a lower bound to show $\xi := \|(\mathscr{A}^*(\boldsymbol{\varepsilon}))_{\max(2\mathbf{r})}\|_{\mathrm{HS}}$ is essential in the estimation error upper bound of Theorem 1.

Theorem 2 *Consider the following class of* $(\widetilde{\mathscr{A}}, \widetilde{\boldsymbol{\mathcal{X}}}, \widetilde{\boldsymbol{\varepsilon}})$*:*

$$\mathcal{F}_{\mathbf{r}}(\xi) = \left\{ \left(\widetilde{\mathscr{A}}, \widetilde{\boldsymbol{\mathcal{X}}}, \widetilde{\boldsymbol{\varepsilon}}\right) : \begin{array}{c} \widetilde{\mathscr{A}} \textit{ satisfies } 3\mathbf{r}\textit{-TRIP}, \\ \widetilde{\boldsymbol{\mathcal{X}}} \textit{ is of Tucker rank at most } \mathbf{r}, \\ \|(\widetilde{\mathscr{A}}^*(\widetilde{\boldsymbol{\varepsilon}}))_{\max(2\mathbf{r})}\|_{\mathrm{HS}} \leqslant \xi \end{array} \right\}.$$

Under the low-rank tensor estimation model (1)*, we have*

$$\inf_{\widehat{\boldsymbol{\mathcal{X}}}} \sup_{(\widetilde{\mathscr{A}}, \widetilde{\boldsymbol{\mathcal{X}}}, \widetilde{\boldsymbol{\varepsilon}}) \in \mathcal{F}_{\mathbf{r}}(\xi)} \|\widehat{\boldsymbol{\mathcal{X}}} - \widetilde{\boldsymbol{\mathcal{X}}}\|_{\mathrm{HS}} \geqslant 2^{-1/2}\xi.$$

4. IMPLICATIONS IN MACHINE LEARNING

Throughout this section, we denote $\bar{p} := \max_k p_k, \underline{p} := \min_k p_k, \bar{r} = \max_k r_k$, $\underline{\lambda} := \min_k \sigma_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$, $\bar{\lambda} := \max_k \sigma_1(\mathcal{M}_k(\boldsymbol{\mathcal{X}}^*))$ and $\kappa := \bar{\lambda}/\underline{\lambda}$.

Tensor Regression is a basic problem for the supervised tensor learning. We assume $\{\boldsymbol{\mathcal{A}}_i\}_{i=1}^n$ are independent and have i.i.d. $N(0, 1/n)$ entries; $\varepsilon_i \overset{i.i.d.}{\sim} N(0, \sigma^2/n)$ in model (1). Under this design, [24] showed that the $\mathbf{r}$-TRIP condition with TRIP constant $R_{\mathbf{r}}$ is satisfied with high probability as long as the sample size is greater than $CR_{\mathbf{r}}^{-2}(\bar{r}^d + d\bar{p}\bar{r})\log(d)$ for some $C > 0$. Suppose the initialization is obtained by T-HOSVD: $\boldsymbol{\mathcal{X}}^0 = \mathscr{A}^*(\mathbf{y}) \times_{k=1}^d P_{\mathbf{U}_k^0}$, where $\mathbf{U}_k^0 = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\mathscr{A}^*(\mathbf{y})))$. Then we have the following theoretical guarantee for the outcome of RGN.

Theorem 3 (RGN for Tensor Regression) *Consider RGN for tensor regression. Suppose* $\bar{r} \leqslant \underline{p}^{1/2}$*,* $\mathcal{H}_{\mathbf{r}}$ *is either T-HOSVD or ST-HOSVD. If* $n \geqslant c(d)(\|\boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}}^2 + \sigma^2)\kappa^2\sqrt{\bar{r}}\bar{p}^{d/2}/\underline{\lambda}^2$*, and* $t_{\max} \geqslant C(d)\log\log\left(\frac{\underline{\lambda}\sqrt{n}}{\sigma\sqrt{\sum_{k=1}^d r_k p_k + \prod_{k=1}^d r_k}}\right)$*, then*

$$\|\boldsymbol{\mathcal{X}}^{t_{\max}} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}} \leqslant c(\sqrt{d}+1)\sigma\sqrt{\left(\sum_{k=1}^d r_k p_k + \prod_{k=1}^d r_k\right)/n}$$

holds with probability at least $1 - \exp(-C\underline{p})$.

Tensor SVD is a specific model covered by the prototypical model (1), which can be equivalently written as $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}}^* + \boldsymbol{\mathcal{E}}$, where $\boldsymbol{\mathcal{X}}^*$ has Tucker decomposition and $\boldsymbol{\mathcal{E}}$ has i.i.d. $N(0, \sigma^2)$ entries. The goal is to estimate $\boldsymbol{\mathcal{X}}^*$.

The following Theorem 4 gives the theoretical guarantee of RGN initialized with T-HOSVD for the tensor SVD.

Theorem 4 (RGN for Tensor SVD) *Consider RGN for tensor SVD. Suppose* $\bar{r} \leqslant \underline{p}^{1/2}$*,* $\mathcal{H}_{\mathbf{r}}$ *is either T-HOSVD or ST-HOSVD and the algorithm is initialized by T-HOSVD, i.e.,* $\boldsymbol{\mathcal{X}}^0 = \boldsymbol{\mathcal{Y}} \times_{k=1}^d P_{\mathbf{U}_k^0}$ *where* $\mathbf{U}_k^0 = \mathrm{SVD}_{r_k}(\mathcal{M}_k(\boldsymbol{\mathcal{Y}}))$*. If the least singular value* $\underline{\lambda} \geqslant c(d)\kappa\bar{p}^{d/4}\bar{r}^{1/4}\sigma$*, and* $t_{\max} \geqslant C(d)\log\log\left\{\underline{\lambda}\Big/\left(\sigma\sqrt{\sum_{k=1}^d r_k p_k + \prod_{k=1}^d r_k}\right)\right\}$*,*

$$\|\boldsymbol{\mathcal{X}}^{t_{\max}} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{HS}} \leqslant c \cdot (\sqrt{d}+1)\sigma\sqrt{\sum_{k=1}^d r_k p_k + \prod_{k=1}^d r_k}$$

holds with probability at least $1 - \exp(-C\underline{p})$.

5. REFERENCES

[1] Tamara G Kolda and Brett W Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[2] Hua Zhou, Lexin Li, and Hongtu Zhu, "Tensor regression with applications in neuroimaging data analysis," *Journal of the American Statistical Association*, vol. 108, no. 502, pp. 540–552, 2013.

[3] Animashree Anandkumar, Rong Ge, and Majid Janzamin, "Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates," *arXiv preprint arXiv:1402.5180*, 2014.

[4] Silvia Gandy, Benjamin Recht, and Isao Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, pp. 025010, 2011.

[5] Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye, "Tensor completion for estimating missing values in visual data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 208–220, 2013.

[6] Botao Hao, Anru Zhang, and Guang Cheng, "Sparse and low-rank tensor estimation via cubic sketchings," *IEEE Transactions on Information Theory*, 2020.

[7] Emile Richard and Andrea Montanari, "A statistical model for tensor pca," in *Advances in Neural Information Processing Systems*, 2014, pp. 2897–2905.

[8] Anru Zhang and Dong Xia, "Tensor svd: Statistical and computational limits," *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7311–7338, 2018.

[9] Ryota Tomioka, Taiji Suzuki, Kohei Hayashi, and Hisashi Kashima, "Statistical performance of convex tensor decomposition," in *Advances in Neural Information Processing Systems*, 2011, pp. 972–980.

[10] Cun Mu, Bo Huang, John Wright, and Donald Goldfarb, "Square deal: Lower bounds and improved relaxations for tensor recovery.," in *ICML*, 2014, pp. 73–81.

[11] Christopher J Hillar and Lek-Heng Lim, "Most tensor problems are np-hard," *Journal of the ACM (JACM)*, vol. 60, no. 6, pp. 1–39, 2013.

[12] Xiaoshan Li, Da Xu, Hua Zhou, and Lexin Li, "Tucker tensor regression and neuroimaging analysis," *Statistics in Biosciences*, pp. 1–26, 2018.

[13] Han Chen, Garvesh Raskutti, and Ming Yuan, "Nonconvex projected gradient descent for generalized low-rank tensor regression," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 172–208, 2019.

[14] Rungang Han, Rebecca Willett, and Anru Zhang, "An optimal statistical and computational framework for generalized tensor estimation," *The Annals of Statistics, to appear*, 2020.

[15] P-A Absil, Robert Mahony, and Rodolphe Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, 2009.

[16] Ledyard R Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[17] Othmar Koch and Christian Lubich, "Dynamical tensor approximation," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 5, pp. 2360–2375, 2010.

[18] Nicolas Boumal, "An introduction to optimization on smooth manifolds," *http://sma.epfl.ch/ nboumal/#book*, 2020.

[19] Jorge Nocedal and Stephen Wright, *Numerical optimization*, Springer Science & Business Media, 2006.

[20] Daniel Kressner, Michael Steinlechner, and Bart Vandereycken, "Low-rank tensor completion by Riemannian optimization," *BIT Numerical Mathematics*, vol. 54, no. 2, pp. 447–468, 2014.

[21] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[22] Nick Vannieuwenhoven, Raf Vandebril, and Karl Meerbergen, "A new truncation strategy for the higher-order singular value decomposition," *SIAM Journal on Scientific Computing*, vol. 34, no. 2, pp. A1027–A1052, 2012.

[23] Wolfgang Hackbusch, *Tensor spaces and numerical tensor calculus*, vol. 42, Springer, 2012.

[24] Holger Rauhut, Reinhold Schneider, and Zeljka Stojanac, "Low rank tensor recovery via iterative hard thresholding," *Linear Algebra and its Applications*, vol. 523, pp. 220–262, 2017.

[25] Emmanuel J Candès and Yaniv Plan, "Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements," *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 2342–2359, 2011.

[26] T Tony Cai and Anru Zhang, "Sharp RIP bound for sparse signal and low-rank matrix recovery," *Applied and Computational Harmonic Analysis*, vol. 35, no. 1, pp. 74–93, 2013.