Nonconvex Low-Rank Symmetric Tensor Completion from Noisy Data

Changxiao Cai Princeton University
Gen Li Tsinghua University
H. Vincent Poor Princeton University
Yuxin Chen Princeton University

Abstract

We study a completion problem of broad practical interest: the reconstruction of a low-rank symmetric tensor from highly incomplete and randomly corrupted observations of its entries. While a variety of prior work has been dedicated to this problem, prior algorithms either are computationally too expensive for large-scale applications, or come with sub-optimal statistical guarantees. Focusing on "incoherent" and well-conditioned tensors of a constant CP rank, we propose a two-stage nonconvex algorithm — (vanilla) gradient descent following a rough initialization — that achieves the best of both worlds. Specifically, the proposed nonconvex algorithm faithfully completes the tensor and retrieves individual tensor factors within nearly linear time, while at the same time enjoying near-optimal statistical guarantees (i.e. minimal sample complexity and optimal ℓ_2 and ℓ_∞ statistical accuracy). The insights conveyed through our analysis of nonconvex optimization might have implications for other tensor estimation problems.

1 Introduction

1.1 Tensor completion from noisy entries

Estimation of low-complexity models from highly incomplete observations is a fundamental task that spans a diverse array of engineering applications. Arguably one of the most extensively studied problems of this kind is matrix completion, where one wishes to recover a low-rank matrix given only partial entries [21, 14]. Moving beyond matrix-type data, a natural higher-order generalization is *low-rank tensor completion*, which aims to reconstruct a low-rank tensor when the vast majority of its entries are unseen. There is certainly no shortage of applications that motivate the investigation of tensor completion, examples including seismic data analysis [44, 24], visual data in-painting [47, 46], medical imaging [25, 58, 19], multi-dimensional harmonic retrieval [13, 72], to name just a few.

For the sake of clarity, we phrase the problem formally before we proceed, focusing on a simple model that already captures the intrinsic difficulty of tensor completion in many aspects.[1] Imagine we are asked to estimate a symmetric order-three tensor[2] $\boldsymbol{T}^\star \in \mathbb{R}^{d\times d\times d}$ from a few noisy entries

$$T_{j,k,l} = T^\star_{j,k,l} + E_{j,k,l}, \qquad \forall (j,k,l) \in \Omega, \tag{1}$$

where $T_{j,k,l}$ is the observed noisy entry at location (j,k,l), $E_{j,k,l}$ stands for the associated noise, and $\Omega \subseteq \{1,\cdots,d\}^3$ is a symmetric index subset to sample from. For notational simplicity, we set $\boldsymbol{T} = [T_{j,k,l}]_{1\le j,k,l\le d}$ and $\boldsymbol{E} = [E_{j,k,l}]_{1\le j,k,l\le d}$, with $T_{j,k,l} = E_{j,k,l} = 0$ for any $(j,k,l)\notin\Omega$. We adopt a *random sampling* model such that each index (j,k,l) $(j\le k\le l)$ is included in Ω independently with probability p. In addition, we know *a priori* that the unknown tensor $\boldsymbol{T}^\star \in \mathbb{R}^{d\times d\times d}$ is a superposition of r rank-one tensors (often termed canonical polyadic (CP) decomposition if r is minimal)

$$\boldsymbol{T}^\star = \sum\nolimits_{i=1}^{r} \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star, \qquad \text{or more concisely,} \qquad \boldsymbol{T}^\star = \sum\nolimits_{i=1}^{r} \boldsymbol{u}_i^{\star\otimes 3}, \tag{2}$$

[1] We focus on symmetric order-3 tensors primarily for simplicity of presentation. Many of our findings naturally extend to the more general case with asymmetric tensors of possibly higher order.

[2] Here, a tensor $\boldsymbol{T}\in\mathbb{R}^{d\times d\times d}$ is said to be symmetric if $T_{j,k,l} = T_{k,j,l} = T_{k,l,j} = T_{l,k,j} = T_{j,l,k} = T_{l,j,k}$.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

where each $\boldsymbol{u}_i^\star \in \mathbb{R}^d$ represents one of the r factors. The primary question is: can we hope to faithfully estimate $\boldsymbol{T}^\star$, as well as the factors $\{\boldsymbol{u}_i^\star\}_{1\le i\le r}$, from the partially revealed entries (1)?

1.2 Computational and statistical challenges

Even though tensor completion conceptually resembles matrix completion in various ways, it is considerably more challenging than the matrix counterpart. This is perhaps not surprising, given that a plethora of natural tensor problems are all notoriously hard [32]. As a notable example, while matrix completion is often efficiently solvable under nearly minimal sample complexity [8, 29], all polynomial-time algorithms developed so far for tensor completion — even in the noise-free case — require a sample size at least exceeding the order of $rd^{3/2}$. This is substantially larger than the degrees of freedom (i.e. rd) underlying the model (2). In fact, it is widely conjectured that there exists a large computational barrier away from the information-theoretic sampling limits [4].

With this fundamental gap in mind, the current paper focuses on the regime (in terms of the sample size) that enables reliable tensor completion in polynomial time. A variety of algorithms have been proposed that enjoy some sort of theoretical guarantees in (at least part of) this regime, including but not limited to spectral methods [50], sum-of-squares hierarchy [4, 53], nonconvex algorithms [36, 67], and also convex relaxation (based on proper unfolding) [25, 64, 34, 57, 47, 51, 28]. While these are all polynomial-time algorithms, most of the computational complexities supported by prior theory remain prohibitively high when dealing with large-scale tensor data. The only exception is the unfolding-based spectral method, which, however, fails to achieve exact recovery even when the noise vanishes. This leads to a critical question that this paper aims to explore:

Q1: *Is there any linear-time algorithm that is guaranteed to work for tensor completion?*

Going beyond such computational concerns, one might naturally wonder whether it is also possible for a fast algorithm to achieve a nearly un-improvable statistical accuracy in the presence of noise. Towards this end, intriguing stability guarantees have been established for sum-of-squares hierarchy in the noisy settings [4], although this paradigm is computationally prohibitive for large-scale data. The recent work [68] came up with a two-stage algorithm (i.e. spectral method followed by tensor power iterations) for noisy tensor completion. Its estimation accuracy, however, falls short of achieving exact recovery in the absence of noise. This gives rise to another question of fundamental importance:

Q2: *Can we achieve near-optimal statistical accuracy without compromising computational efficiency?*

1.3 A two-stage nonconvex algorithm

To address the above-mentioned challenges, a first impulse is to resort to the least squares formulation

$$\underset{\boldsymbol{u}_1,\cdots,\boldsymbol{u}_r\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{j,k,l\in\Omega} \Big(\Big[\sum\nolimits_{i=1}^r \boldsymbol{u}_i^{\otimes 3}\Big]_{j,k,l} - T_{j,k,l}\Big)^2, \tag{3}$$

or more concisely (up to proper re-scaling),

$$\underset{\boldsymbol{U}\in\mathbb{R}^{d\times r}}{\text{minimize}} \quad f(\boldsymbol{U}) \triangleq \frac{1}{6p}\Big\|\mathcal{P}_\Omega\Big(\sum\nolimits_{i=1}^r \boldsymbol{u}_i^{\otimes 3} - \boldsymbol{T}\Big)\Big\|_{\mathrm{F}}^2 \tag{4}$$

if we take $\boldsymbol{U} := [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_r] \in \mathbb{R}^{d\times r}$. Here, we denote by $\mathcal{P}_\Omega(\boldsymbol{T})$ the orthogonal projection of any tensor $\boldsymbol{T}$ onto the subspace of tensors which vanish outside of Ω. This optimization problem, however, is highly nonconvex, resulting in computational intractability in general.

Fortunately, not all nonconvex problems are as danuting as they may seem. For example, recent years have seen a flurry of activity in low-rank matrix factorization via nonconvex optimization, which achieves optimal statistical and computational efficiency at once [55, 39, 41, 35, 9, 12, 62, 20, 18, 11, 76, 49, 65, 78]. Motivated by this strand of work, we propose to solve (4) via a two-stage nonconvex paradigm, presented below in reverse order. The procedure is summarized in Algorithms 1-3.

Gradient descent (GD). Arguably one of the simplest optimization algorithms is gradient descent, which adopts a gradent update rule

$$\boldsymbol{U}^{t+1} = \boldsymbol{U}^t - \eta_t \nabla f(\boldsymbol{U}^t), \qquad t = 0, 1, \cdots \tag{5}$$

Algorithm 1 Gradient descent for nonconvex tensor completion

1: **Input:** observed entries $\{T_{j,k,l} \mid (j,k,l) \in \Omega\}$, sampling rate p, number of iterations t_0.
2: Generate an initial estimate $\boldsymbol{U}^0 \in \mathbb{R}^{d\times r}$ via Algorithm 2.
3: **for** $t = 0, 1, \ldots, t_0 - 1$ **do**
4: $\quad \boldsymbol{U}^{t+1} = \boldsymbol{U}^t - \eta_t \nabla f(\boldsymbol{U}^t) = \boldsymbol{U}^t - \frac{\eta_t}{p}\mathcal{P}_\Omega\big(\sum_{i=1}^r \big(\boldsymbol{u}_i^t\big)^{\otimes 3} - \boldsymbol{T}\big) \times_1^{\mathsf{seq}} \boldsymbol{U}^t \times_2^{\mathsf{seq}} \boldsymbol{U}^t$, where $\times_1^{\mathsf{seq}}$ and $\times_2^{\mathsf{seq}}$ are defined in Section 1.5.

Algorithm 2 Spectral initialization for nonconvex tensor completion

1: **Input:** sampling set Ω, observed entries $\{T_{i,j,k} \mid (i,j,k) \in \Omega\}$, sampling rate p.
2: Let $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ be the rank-r eigen-decomposition of

$$\boldsymbol{B} := \mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{A}\boldsymbol{A}^\top), \tag{6}$$

where $\boldsymbol{A} = \mathsf{unfold}\big(p^{-1}\boldsymbol{T}\big)$ is the mode-1 matricization of $p^{-1}\boldsymbol{T}$, and $\mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{Z})$ extracts out the off-diagonal entries of $\boldsymbol{Z}$.
3: **Output:** initial estimate $\boldsymbol{U}^0 \in \mathbb{R}^{d\times r}$ from $\boldsymbol{U} \in \mathbb{R}^{d\times r}$ using Algorithm 3.

where η_t is the learning rate. The main computational burden in each iteration lies in gradient evaluation, which, in this case, can be performed in time proportional to that taken to read the data.

Despite the simplicity of this algorithm, two critical issues stand out and might significantly affect its efficiency, which we shall bear in mind throughout the algorithmic and theoretical development.

(i) Local stationary points and initialization. As is well known, GD is guaranteed to find an approximate local stationary point, provided that the learning rates do not exceed the inverse Lipschitz constant of the gradient [5]. There exist, however, local stationary points (e.g. saddle points or spurious local minima) that might fall short of the desired statistical properties. This requires us to properly avoid such undesired points, while retaining computational efficiency. To address this issue, one strategy is to first identify a rough initial guess within a local region surrounding the global solution, which often helps rule out bad local minima. As a side remark, while careful initialization might not be crucial for several matrix recovery cases [45, 15, 27], it does seem to be critical in various tensor problems [56]. We shall elucidate this point in the full version [7].

(ii) Learning rates and regularization. Learning rates play a pivotal role in determining the convergence properties of GD. The challenge, however, is that the loss function (4) is overall not sufficiently smooth (i.e. its gradient often has a very large Lipschitz constant), and hence generic optimization theory recommends a pessimistically slow update rule (i.e. an extremely small learning rate) so as to guard against over-shooting. This, however, slows down the algorithm significantly, thus destroying the main computational advantage of GD (i.e. low per-iteration cost). With this issue in mind, prior literature suggests carefully designed regularization steps (e.g. proper projection, regularized loss functions) in order to improve the geometry of the optimization landscape [67]. By contrast, we argue that one is allowed to take a constant learning rate — which is as aggressive as it can possibly be — even without enforcing any regularization procedures.

Initialization. Motivated by the above-mentioned issue (i), we develop a procedure that guarantees a reasonable initial estimate. In a nutshell, the proposed procedure consists of two steps:

(a) Estimate the subspace spanned by the r tensor factors $\{\boldsymbol{u}_i^\star\}_{1\le i\le r}$ via a spectral method;

(b) Disentangle individual tensor factors from this subspace estimate.

The computational complexity of the proposed initialization is linear-time (i.e. $O(pd^3)$) when $r = O(1)$. Note, however, that these steps are more complicated to describe. We postpone the details to Section 2 and intuitions to [7]. The readers can catch a glimpse of these procedures in Algorithm 2-3.

1.4 Main results

Encouragingly, the proposed nonconvex algorithm provably achieves the best of both worlds — in terms of statistical accuracy and computational efficiency — for a broad class of problem instances.

Algorithm 3 Retrieval of low-rank tensor factors from a given subspace estimate.

Input: sampling set Ω, observed entries $\{T_{i,j,k} \mid (i,j,k) \in \Omega\}$, sampling rate p, number of restarts L, pruning threshold ϵ_{th}, subspace estimate $\boldsymbol{U} \in \mathbb{R}^{d \times r}$.
for $\tau = 1, \ldots, L$ **do**
  Generate an independent Gaussian vector $\boldsymbol{g}^\tau \sim \mathcal{N}(0, \boldsymbol{I}_d)$.
  $\big(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau\big) \leftarrow$ RETRIEVE-ONE-TENSOR-FACTOR$(\boldsymbol{T}, p, \boldsymbol{U}, \boldsymbol{g}^\tau)$.
Generate $\big\{(\boldsymbol{w}^1, \lambda_1), \ldots, (\boldsymbol{w}^r, \lambda_r)\big\} \leftarrow$ PRUNE$(\big\{\big(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau\big)\big\}_{\tau=1}^{L}, \epsilon_{\mathsf{th}})$.
Output: initial estimate $\boldsymbol{U}^0 = \big[\lambda_1^{1/3}\boldsymbol{w}^1, \ldots, \lambda_r^{1/3}\boldsymbol{w}^r\big]$.

function RETRIEVE-ONE-TENSOR-FACTOR$(\boldsymbol{T}, p, \boldsymbol{U}, \boldsymbol{g})$
Compute

$$\boldsymbol{\theta} = \boldsymbol{U}\boldsymbol{U}^\top \boldsymbol{g} =: \mathcal{P}_{\boldsymbol{U}}(\boldsymbol{g}), \tag{7a}$$

$$\boldsymbol{M} = p^{-1}\boldsymbol{T} \times_3 \boldsymbol{\theta}, \tag{7b}$$

where $\times_3$ is defined in Section 1.5.
Let $\boldsymbol{\nu}$ be the leading singular vector of $\boldsymbol{M}$ obeying $\langle \boldsymbol{T}, \boldsymbol{\nu}^{\otimes 3}\rangle \geq 0$; set $\lambda = \langle p^{-1}\boldsymbol{T}, \boldsymbol{\nu}^{\otimes 3}\rangle$.
return $\big(\boldsymbol{\nu}, \lambda, \sigma_1(\boldsymbol{M}) - \sigma_2(\boldsymbol{M})\big)$.

Before continuing, we note that one cannot hope to recover an arbitrary tensor from highly subsampled and arbitrarily corrupted entries. In order to enable provably valid recovery, the present paper focuses on a tractable model by imposing the following assumptions.

Assumption 1.1 (Incohrence and well-conditionedness)**.** *The tensor factors $\{\boldsymbol{u}_i^\star\}_{1\le i\le r}$ satisfy*

$$(\mathbf{A1}) \qquad \|\boldsymbol{T}^\star\|_\infty \le \sqrt{\mu_0/d^3}\, \|\boldsymbol{T}^\star\|_{\mathrm{F}}\,, \tag{8a}$$

$$(\mathbf{A2}) \qquad \|\boldsymbol{u}_i^\star\|_\infty \le \sqrt{\mu_1/d}\, \|\boldsymbol{u}_i^\star\|_2\,, \qquad 1 \le i \le d; \tag{8b}$$

$$(\mathbf{A3}) \qquad \big|\big\langle \boldsymbol{u}_i^\star, \boldsymbol{u}_j^\star\big\rangle\big| \le \sqrt{\mu_2/d}\, \|\boldsymbol{u}_i^\star\|_2 \big\|\boldsymbol{u}_j^\star\big\|_2\,, \qquad 1 \le i \ne j \le d; \tag{8c}$$

$$(\mathbf{A4}) \qquad \kappa \triangleq \big(\max_i \|\boldsymbol{u}_i^\star\|_2^3\big) \,/\, \big(\min_i \|\boldsymbol{u}_i^\star\|_2^3\big) = O(1). \tag{8d}$$

Remark 1.2. *Here, μ_0, μ_1 and μ_2 are termed the incoherence parameters. Assumptions* $\mathbf{A1}$*,* $\mathbf{A2}$ *and* $\mathbf{A3}$ *can be viewed as some sort of incoherence conditions for the tensor. For instance, when μ_0, μ_1 and μ_2 are small, these conditions say that (1) the energy of tensor $\boldsymbol{T}^\star$ is (nearly) evenly spread across all entries; (2) each factor $\boldsymbol{u}_i^\star$ is de-localized; (3) the factors $\{\boldsymbol{u}_i^\star\}$ are nearly orthogonal to each other. Assumption* $\mathbf{A4}$ *is concerned with the "well-conditionedness" of the tensor, meaning that each rank-1 component is of roughly the same strength.*

For notational simplicity, we shall set $\mu := \max\{\mu_0, \mu_1, \mu_2\}$.

Assumption 1.3 (Random noise)**.** *Suppose that $\boldsymbol{E}$ is a symmetric random tensor, where $\{E_{j,k,l}\}_{1\le j\le k\le l\le d}$ (cf.* (1)*) are independently generated symmetric sub-Gaussian random variables with mean zero and variance $\mathsf{Var}(E_{j,k,l}) \le \sigma^2$.*

In addition, recognizing that there is a global permutational ambiguity issue (namely, one cannot distinguish $\boldsymbol{u}_1^\star, \cdots, \boldsymbol{u}_r^\star$ from an arbitrary permutation of them), we introduce the following loss metrics to account for this ambiguity

$$\mathsf{dist}_{\mathrm{F}}(\boldsymbol{U}, \boldsymbol{U}^\star) := \min_{\boldsymbol{\Pi} \in \mathsf{perm}_r} \|\boldsymbol{U}\boldsymbol{\Pi} - \boldsymbol{U}^\star\|_{\mathrm{F}}, \tag{9a}$$

function PRUNE$(\big\{\big(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau\big)\big\}_{\tau=1}^{L}, \epsilon_{\mathsf{th}})$
  Set $\Theta = \big\{\big(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau\big)\big\}_{\tau=1}^{L}$.
  for $i = 1, \ldots, r$ **do**
    Choose $(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau)$ from Θ with the largest $\mathsf{spec\text{-}gap}_\tau$; set $\boldsymbol{w}^i = \boldsymbol{\nu}^\tau$, $\lambda_i = \lambda_\tau$.
    Update $\Theta \leftarrow \Theta \setminus \big\{\big(\boldsymbol{\nu}^\tau, \lambda_\tau, \mathsf{spec\text{-}gap}_\tau\big) \in \Theta : |\langle \boldsymbol{\nu}^\tau, \boldsymbol{w}^i\rangle| > 1 - \epsilon_{\mathsf{th}}\big\}$.
  return $\big\{(\boldsymbol{w}^1, \lambda_1), \ldots, (\boldsymbol{w}^r, \lambda_r)\big\}$.

$$\mathsf{dist}_{\infty}(\boldsymbol{U},\boldsymbol{U}^{\star}) := \min_{\boldsymbol{\Pi}\in\mathsf{perm}_r} \|\boldsymbol{U}\boldsymbol{\Pi}-\boldsymbol{U}^{\star}\|_{\infty}, \tag{9b}$$

$$\mathsf{dist}_{2,\infty}(\boldsymbol{U},\boldsymbol{U}^{\star}) := \min_{\boldsymbol{\Pi}\in\mathsf{perm}_r} \|\boldsymbol{U}\boldsymbol{\Pi}-\boldsymbol{U}^{\star}\|_{2,\infty}, \tag{9c}$$

where perm_r stands for the set of $r\times r$ permutation matrices. For notational simplicity, we also take $\lambda_{\min}^{\star} := \min_{1\le i\le r}\|\boldsymbol{u}_i^{\star}\|_2^3$ and $\lambda_{\max}^{\star} := \max_{1\le i\le r}\|\boldsymbol{u}_i^{\star}\|_2^3$.

With these in place, we are ready to present our main results.

Theorem 1.4. *Fix an arbitrary small constant $\delta>0$. Suppose that $r,\kappa,\mu=O(1)$,*

$$p\ge c_0\tfrac{\log^5 d}{d^{3/2}},\qquad \tfrac{\sigma}{\lambda_{\min}^{\star}}\le c_1\tfrac{\sqrt{p}}{\sqrt{d^{3/2}\log^5 d}},$$

$$L\ge c_3 \quad \textit{and} \quad \epsilon_{\mathsf{th}} = c_4\left(\tfrac{\log d}{d\sqrt{p}}+\tfrac{\sigma}{\lambda_{\min}^{\star}}\sqrt{\tfrac{d\log^2 d}{p}}+\sqrt{\tfrac{\log d}{d}}\right)$$

for some sufficiently large constants $c_0,c_3>0$ and some sufficiently small constants $c_1,c_4>0$. The learning rate $\eta_t\equiv\eta$ is taken to be a constant obeying $0<\eta\le\lambda_{\min}^{\star 4/3}/\big(32\lambda_{\max}^{\star 8/3}\big)$. Then with probability at least $1-\delta$,

$$\mathsf{dist}_{\mathrm{F}}(\boldsymbol{U}^t,\boldsymbol{U}^{\star})\le\left(C_1\rho^t+C_2\tfrac{\sigma}{\lambda_{\min}^{\star}}\sqrt{\tfrac{d\log d}{p}}\right)\|\boldsymbol{U}^{\star}\|_{\mathrm{F}} \tag{10a}$$

$$\mathsf{dist}_{\infty}(\boldsymbol{U}^t,\boldsymbol{U}^{\star})\le\mathsf{dist}_{2,\infty}(\boldsymbol{U}^t,\boldsymbol{U}^{\star})\le\left(C_3\rho^t+C_4\tfrac{\sigma}{\lambda_{\min}^{\star}}\sqrt{\tfrac{d\log d}{p}}\right)\|\boldsymbol{U}^{\star}\|_{2,\infty} \tag{10b}$$

hold simultaneously for all $0\le t\le t_0=d^5$. Here, $0<C_1,C_3,\rho<1$ and $C_2,C_4>0$ are some absolute constants.

Proof. The proof of this theorem is built upon a powerful statistical technique — called the leave-one-out analysis [23, 16, 1, 49, 79, 15, 22, 17, 52]. The proof can be found in our full version [7]. □

Several important implications are as follows. The discussion below assumes $\lambda_{\max}^{\star}\asymp\lambda_{\min}^{\star}\asymp 1$ for notational simplicity.

- *Linear convergence.* In the absence of noise, the proposed algorithm converges linearly, namely, it provably attains ε accuracy within $O(\log(1/\varepsilon))$ iterations. Given the inexpensiveness of each gradient iteration, this algorithm can be viewed as a linear-time algorithm, which can almost be implemented as long as we can read the data.
- *Near-optimal sample complexity.* The fast convergence is guaranteed as soon as the sample size exceeds the order of $d^{3/2}\mathrm{poly}\log(d)$. This matches the minimal sample complexity — modulo some logarithmic factor — known so far for any polynomial-time algorithm.
- *Near-optimal statistical accuracy.* The proposed algorithm converges geometrically fast to a point with Euclidean error $O\big(\sigma\sqrt{(d\log d)/p}\big)$. This matches the lower bound established in [68, Theorem 5] up to some logarithmic factor.
- *Entrywise estimation accuracy.* In addition to the Euclidean error bound, we have also established an entrywise error bound which, to the best of our knowledge, has not been established in any of the prior works. When t is sufficiently large, the iterates reach an entrywise error bound $O\big(\sigma\sqrt{(\log d)/p}\big)$. This entrywise error bound is about $\sqrt{d}$ times smaller than the above ℓ_2 norm bound, implying that the estimation errors are evenly spread out across all entries.
- *Implicit regularization.* One appealing feature of our finding is the simplicity of the algorithm. All of the above statistical and computational benefits hold for vanilla gradient descent (when properly initialized). This should be contrasted with prior work (e.g. [67]) that requires extra regularization to stabilize the optimization landscape. In principle, vanilla GD implicitly constrains itself within a region of well-conditioned landscape, thus enabling fast convergence without regularization.
- *No sample splitting.* The theory developed herein does not require fresh samples in each iteration. We note that sample splitting has been frequently adopted in other context primarily to simplify analysis. Nevertheless, it typically does not exploit the data in an efficient manner (i.e. each data sample is used only once), thus resulting in the need of a much larger sample size in practice.

As an immediate consequence of Theorem 1.4, we obtain optimal ℓ_∞ statistical guarantees for estimating tensor entries, which are previously rarely available (see Table 1). Specifically, let our tensor estimate in the t-th iteration be $\boldsymbol{T}^t := \sum_{i=1}^r \boldsymbol{u}_i^t\otimes\boldsymbol{u}_i^t\otimes\boldsymbol{u}_i^t$, where $\boldsymbol{U}^t=[\boldsymbol{u}_1^t,\cdots,\boldsymbol{u}_r^t]\in\mathbb{R}^{d\times r}$.

	algorithm	sample complexity	comput. complexity	ℓ_2 error (noisy)	ℓ_∞ error (noisy)	recovery type (noiseless)
ours	spectral method + (vanilla) GD	$d^{1.5}$	pd^3	$\sigma\sqrt{\frac{d}{p}}$	$\sigma\sqrt{\frac{1}{p}}$	exact
[68]	spectral initialization + tensor power method	$d^{1.5}$	pd^3	$\frac{(\|\boldsymbol{T}^\star\|_\infty+\sigma)\sqrt{d}}{\sqrt{p}}$	n/a	approximate
[67]	spectral method + GD on manifold	$d^{1.5}$	$\mathrm{poly}(d)$	n/a	n/a	exact
[50]	spectral method	$d^{1.5}$	d^3	n/a	n/a	approximate
[4]	sum-of-squares	$d^{1.5}$	d^{15}	$\frac{\|\boldsymbol{T}^\star\|_{\mathrm{F}}}{\sqrt{pd^{1.5}}}+\sigma d^{1.5}$	n/a	approximate
[53]	sum-of-squares	$d^{1.5}$	d^{10}	n/a	n/a	exact
[73] [74]	tensor nuclear norm minimization	d	NP-hard	n/a	n/a	exact

Table 1: Comparison with theory for existing methods when $r, \mu, \kappa \asymp 1$ (neglecting log factors).

Corollary 1.5. *Fix an arbitrarily small constant $\delta > 0$. Instate the assumptions of Theorem 1.4. Then with probability at least $1-\delta$,*

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\mathrm{F}} \lesssim \left(C_1\rho^t + C_2\frac{\sigma}{\lambda^\star_{\min}}\sqrt{\frac{d\log d}{p}}\right)\|\boldsymbol{T}^\star\|_{\mathrm{F}} \tag{11a}$$

$$\left\|\boldsymbol{T}^t - \boldsymbol{T}^\star\right\|_{\infty} \lesssim \left(C_3\rho^t + C_4\frac{\sigma}{\lambda^\star_{\min}}\sqrt{\frac{d\log d}{p}}\right)\|\boldsymbol{T}^\star\|_{\infty} \tag{11b}$$

hold simultaneously for all $0 \le t \le t_0 = d^5$. Here, $0 < C_1, C_3, \rho < 1$ and $C_2, C_4 > 0$ are some absolute constants.

We shall take a moment to discuss the merits of our approach in comparison to prior work. One of the best-known polynomial-time algorithms is the degree-6 level of the sum-of-squares hierarchy, which seems to match the computationally feasible limit in terms of the sample complexity [4]. However, this approach has a well-documented limitation in that it involves solving a semidefinite program of dimensions $d^3 \times d^3$, which requires enormous storage and computation power. Yuan et al. [73, 74] proposed to consider tensor nuclear norm minimization, which provably allows for reduced sample complexity. The issue, however, is that computing the tensor nuclear norm itself is already computationally intractable. The work [50] alleviates this computational burden by resorting to a clever unfolding-based spectral algorithm; it is a nearly linear-time procedure that enables near-minimal sample complexity (among polynomial-time algorithms), although it does not achieve exact recovery even in the absence of noise. The two-stage algorithm developed by [68] — which is based on spectral initialization followed by tensor power methods — shares similar advantages and drawbacks as [50]. The work [36] used tensor power methods for initialization, which, however, requires a large number of restart attempts; see discussions in [7]. Further, [67] proposes a polynomial-time nonconvex algorithm based on gradient descent over Grassmann manifold (with a properly regularized objective function), which is an extension of the nonconvex matrix completion algorithm proposed by [40, 41] to tensor data. The theory provided in [67], however, does not provide explicit computational complexities. The recent work [59] attempts tensor estimation via a collaborative filtering approach, which, however, does not enable exact recovery even in the absence of noise.

1.5 Notations

Before proceeding, we gather a few notations that will be used throughout this paper. For any tensors $\boldsymbol{T}, \boldsymbol{R} \in \mathbb{R}^{d\times d\times d}$, the inner product is defined as $\langle \boldsymbol{T}, \boldsymbol{R}\rangle = \sum_{j,k,l} T_{j,k,l}R_{j,k,l}$. The Frobenius norm of $\boldsymbol{T}$ is defined as $\|\boldsymbol{T}\|_{\mathrm{F}} := \sqrt{\langle \boldsymbol{T}, \boldsymbol{T}\rangle}$. For any vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$, we define the vector products of a tensor $\boldsymbol{T} \in \mathbb{R}^{d\times d\times d}$ — denoted by $\boldsymbol{T} \times_3 \boldsymbol{u} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{T} \times_1 \boldsymbol{u} \times_2 \boldsymbol{v} \in \mathbb{R}^d$ — such that

$$\left[\boldsymbol{T} \times_3 \boldsymbol{u}\right]_{ij} := \sum\nolimits_{1\le k\le d} T_{i,j,k}u_k, \qquad 1 \le i, j \le d; \tag{12a}$$

$$\left[\boldsymbol{T} \times_1 \boldsymbol{u} \times_2 \boldsymbol{v}\right]_{k} := \sum\nolimits_{1\le i,j\le d} T_{i,j,k}u_iv_j, \qquad 1 \le k \le d. \tag{12b}$$

For any $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_r] \in \mathbb{R}^{d\times r}$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_r] \in \mathbb{R}^{d\times r}$, we define

$$\boldsymbol{T} \times_1^{\mathsf{seq}} \boldsymbol{U} \times_2^{\mathsf{seq}} \boldsymbol{V} := [\boldsymbol{T} \times_1 \boldsymbol{u}_i \times_2 \boldsymbol{v}_i]_{1\le i\le r} \in \mathbb{R}^{d\times r}. \tag{13}$$

Further, $f(n) \lesssim g(n)$ or $f(n) = O(g(n))$ means that $|f(n)/g(n)| \le C_1$ for some constant $C_1 > 0$; $f(n) \gtrsim g(n)$ means that $|f(n)/g(n)| \ge C_2$ for some constant $C_2 > 0$; $f(n) \asymp g(n)$ means that $C_1 \le |f(n)/g(n)| \le C_2$ for some constants $C_1, C_2 > 0$.

2 Initialization

This section presents formal details of the proposed two-step initialization. Recall that the proposed initialization procedures consist of two steps, which we detail separately.

2.1 Step 1: subspace estimation via a spectral method

The spectral algorithm is often applied in conjunction with simple "unfolding" (or "matricization") to estimate the *subspace* spanned by the r factors $\{\boldsymbol{u}_i^\star\}_{1\le i\le r}$. This strategy is partly motivated by prior approaches developed for covariance estimation with missing data [48, 50], and has been investigated in detail in [6]. For self-containedness, we provide a brief introduction below, and refer the interested reader to [6] for in-depth discussions.

Let

$$\boldsymbol{A} = \mathsf{unfold}^{1\times 2}\big(\tfrac{1}{p}\boldsymbol{T}\big) \in \mathbb{R}^{d\times d^2}, \quad \text{or more concisely } \boldsymbol{A} = \mathsf{unfold}\big(\tfrac{1}{p}\boldsymbol{T}\big) \in \mathbb{R}^{d\times d^2} \tag{14}$$

be the mode-1 matricization of $p^{-1}\boldsymbol{T}$ (namely, $\frac{1}{p}T_{i,j,k} = A_{i,(j-1)d+k}$ for any $1 \le i,j,k \le d$) [43]. The rationale of this step is that: under our model, the unfolded matrix $\boldsymbol{A}$ obeys

$$\mathbb{E}[\boldsymbol{A}] = \mathsf{unfold}\big(\boldsymbol{T}^\star\big) = \sum_{i=1}^{r} \boldsymbol{u}_i^\star \left(\boldsymbol{u}_i^\star \otimes \boldsymbol{u}_i^\star\right)^\top =: \boldsymbol{A}^\star, \tag{15}$$

whose column space is precisely the span of $\{\boldsymbol{u}^\star\}_{1\le i\le r}$. This motivates one to estimate the r-dimensional column space of $\mathbb{E}[\boldsymbol{A}]$ from $\boldsymbol{A}$. Towards this, a natural strategy is to look at the principal subspace of $\boldsymbol{A}\boldsymbol{A}^\top$. However, the diagonal entries of $\boldsymbol{A}\boldsymbol{A}^\top$ bear too much influence on the principal directions and need to be properly down-weighed. The current paper chooses to work with the principal subspace of the following matrix that zeros out all diagonal components:

$$\boldsymbol{B} := \mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{A}\boldsymbol{A}^\top), \tag{16}$$

where $\mathcal{P}_{\mathsf{off\text{-}diag}}(\boldsymbol{Z})$ extracts out the off-diagonal entries of a squared matrix $\boldsymbol{Z}$. If we let $\boldsymbol{U} \in \mathbb{R}^{d\times r}$ be an orthonormal matrix whose columns are the top-r eigenvectors of $\boldsymbol{B}$, then $\boldsymbol{U}$ serves as our subspace estimate. See Algorithm 2 for a summary of the procedure, and [6] for in-depth discussions.

2.2 Step 2: retrieval of low-rank tensor factors from the subspace estimate

2.2.1 Procedure

As it turns out, it is possible to obtain rough (but reasonable) estimates of all low-rank tensor factors $\{\boldsymbol{u}_i^\star\}_{1\le i\le r}$ — up to global permutation — given a reliable subspace estimate $\boldsymbol{U}$. This is in stark contrast to the low-rank matrix recovery case, where there exists some global rotational ambiguity that prevents us from disentangling the r factors of interest.

We begin by describing how to retrieve *one* tensor factor from the subspace estimate — a procedure summarized in RETRIEVE-ONE-TENSOR-FACTOR(). Let us generate a random vector from the provided subspace $\boldsymbol{U}$ (which has orthonormal columns), that is,

$$\boldsymbol{\theta} = \boldsymbol{U}\boldsymbol{U}^\top \boldsymbol{g}, \qquad \boldsymbol{g} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d). \tag{17}$$

The rescaled tensor data $p^{-1}\boldsymbol{T}$ is then transformed into a matrix via proper "projection" along this random direction $\boldsymbol{\theta}$, namely,

$$\boldsymbol{M} = \tfrac{1}{p}\boldsymbol{T} \times_3 \boldsymbol{\theta} \in \mathbb{R}^{d\times d}. \tag{18}$$

Our estimate for a tensor factor is then given by $\lambda^{1/3}\boldsymbol{\nu}$, where $\boldsymbol{\nu}$ is the leading singular vector of $\boldsymbol{M}$ obeying $\langle \boldsymbol{T}, \boldsymbol{\nu}^{\otimes 3}\rangle \ge 0$, and λ is taken as $\lambda = \big\langle p^{-1}\boldsymbol{T}, \boldsymbol{\nu}^{\otimes 3}\big\rangle$. Informally, $\boldsymbol{\nu}$ reflects the direction of the component $\boldsymbol{u}_i^\star$ that exhibits the largest correlation with the random direction $\boldsymbol{\theta}$, and λ forms an estimate of the corresponding size $\|\boldsymbol{u}_i^\star\|_2$. We shall provide intuition in the full version [7].

A challenge remains, however, as there are oftentimes more than one tensor factors to estimate. To address this issue, we propose to re-run the aforementioned procedure multiple times, so as to ensure that we get to retrieve each tensor factor of interest at least once. We will then apply a careful pruning procedure (i.e. PRUNE()) to remove redundancy.

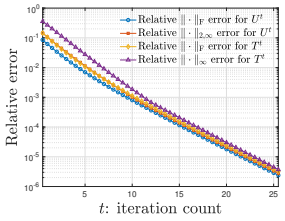

Figure 1: Relative error of the estimate $\boldsymbol{U}^t$ and $\boldsymbol{T}^t$ vs. the iteration count t for the noiseless case, where $d = 100$, $r = 4$, $p = 0.1$.

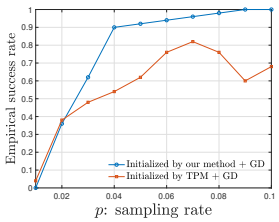

Figure 2: Empirical success rate vs. sampling rate. Each point is averaged over 100 trials.

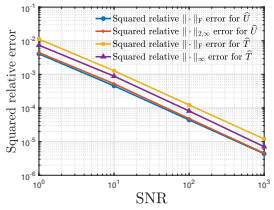

Figure 3: Squared relative errors vs. SNR for noisy settings. Here, $d = 100$, $r = 4$, $p = 0.1$. Each point is averaged over 100 trials.

3 Numerical experiments

We carry out a series of numerical experiments to corroborate our theoretical findings. We generate the truth $\boldsymbol{T}^\star = \sum_{1\le i\le r} \boldsymbol{u}_i^{\star\otimes 3}$ randomly with $\boldsymbol{u}_i^\star \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$. The learning rates, the restart number and the pruning threshold are taken to be $\eta_t \equiv 0.2, L = 64, \epsilon_{\mathsf{th}} = 0.4$.

We start with numerical convergence rates of our algorithm in the absence of noise. Set $d = 100$, $r = 4$ and $p = 0.1$. Fig. 1 shows the numerical estimation errors vs. iteration count t in a typical Monte Carlo trial. Here, 4 kinds of estimation errors are reported: (1) the relative Euclidean error $\frac{\mathsf{dist}_{\mathrm{F}}(\boldsymbol{U}^t,\boldsymbol{U}^\star)}{\|\boldsymbol{U}^\star\|_{\mathrm{F}}}$; (2) the relative $\|\cdot\|_{2,\infty}$ error $\frac{\mathsf{dist}_{2,\infty}(\boldsymbol{U}^t,\boldsymbol{U}^\star)}{\|\boldsymbol{U}^\star\|_{2,\infty}}$; (3) the relative Frobenius norm error $\frac{\|\boldsymbol{T}^t-\boldsymbol{T}^\star\|_{\mathrm{F}}}{\|\boldsymbol{T}^\star\|_{\mathrm{F}}}$; (4) the relative ℓ_∞ error $\frac{\|\boldsymbol{T}^t-\boldsymbol{T}^\star\|_\infty}{\|\boldsymbol{T}^\star\|_\infty}$. Here, we set $\boldsymbol{T}^t = \sum_{i=1}^r \boldsymbol{u}_i^t \otimes \boldsymbol{u}_i^t \otimes \boldsymbol{u}_i^t$ with $\boldsymbol{U}^t = [\boldsymbol{u}_1^t, \cdots, \boldsymbol{u}_r^t]$. For all these metrics, the numerical estimation errors decay geometrically fast.

Next, we study the phase transition (in terms of the success rates for exact recovery) in the noise-free settings. For the sake of comparisons, we also report the numerical performance of tensor power method (TPM) followed by gradient descent. When running the tensor power method, we set the iteration number and restart number to be 16 and 64 respectively. Set $r = 4$. Each trial is claimed to succeed if the relative ℓ_2 error $\frac{\mathsf{dist}_{\mathrm{F}}(\widehat{\boldsymbol{U}},\boldsymbol{U}^\star)}{\|\boldsymbol{U}^\star\|_{\mathrm{F}}} \le 0.01$. Fig. 2 plots the empirical success rates over 100 independent trials. As can be seen, our initialization algorithm outperforms the tensor power method.

The third series of experiments concerns the statistical accuracy of our algorithm. Take $t_0 = 100$, $d = 100$, $r = 4$ and $p = 0.1$. Define the signal-to-noise ratio (SNR) to be $\mathrm{SNR} = \frac{\|\boldsymbol{T}^\star\|_{\mathrm{F}}^2/d^3}{\sigma^2}$. We report in Fig. 3 three types of squared relative errors (namely, $\frac{\mathsf{dist}_{\mathrm{F}}^2(\widehat{\boldsymbol{U}},\boldsymbol{U}^\star)}{\|\boldsymbol{U}^\star\|_{\mathrm{F}}^2}$, $\frac{\mathsf{dist}_{2,\infty}^2(\widehat{\boldsymbol{U}},\boldsymbol{U}^\star)}{\|\boldsymbol{U}^\star\|_{2,\infty}^2}$ and $\frac{\|\widehat{\boldsymbol{T}}-\boldsymbol{T}^\star\|_\infty^2}{\|\boldsymbol{T}^\star\|_\infty^2}$) vs. SNR. Here, the SNR varies from 1 to 1000. Figure 3 illustrates that all three types of relative squared errors scale inversely proportional to the SNR, which is consistent with our theory.

4 Discussion

The current paper uncovers the possibility of efficiently and stably completing a low-CP-rank tensor from partial and noisy entries. Perhaps somewhat unexpectedly, despite the high degree of nonconvexity, this problem can be solved to optimal statistical accuracy within nearly linear time. To the best of our knowledge, this intriguing message has not been shown in the prior literature. The insights and analysis techniques developed in this paper might also have implications for other nonconvex algorithms [36, 66, 69, 54, 67, 38, 61, 71, 37, 70] and other tensor recovery problems [2, 3, 63, 33, 60, 26, 75, 42, 77, 31, 10, 30].

Acknowledgements

Y. Chen is supported in part by the AFOSR YIP award FA9550-19-1-0030, by the ONR grant N00014-19-1-2120, by the ARO grant W911NF-18-1-0303, by the NSF grants CCF-1907661 and IIS-1900140. H. V. Poor is supported in part by the NSF grant DMS-1736417.

References

[1] E. Abbe, J. Fan, K. Wang, and Y. Zhong. Entrywise eigenvector analysis of random matrices with low expected rank. *arXiv preprint arXiv:1709.09565*, 2017.

[2] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky. Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research*, 15(1):2773–2832, 2014.

[3] A. Anandkumar, R. Ge, and M. Janzamin. Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates. *arXiv preprint arXiv:1402.5180*, 2014.

[4] B. Barak and A. Moitra. Noisy tensor completion via the sum-of-squares hierarchy. In *Conference on Learning Theory*, pages 417–445, 2016.

[5] S. Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[6] C. Cai, G. Li, Y. Chi, H. V. Poor, and Y. Chen. Subspace estimation from unbalanced and incomplete data matrices: $\ell_{2,\infty}$ statistical guarantees. *arXiv preprint arXiv:1910.04267*, 2019.

[7] C. Cai, G. Li, H. V. Poor, and Y. Chen. Supplementary materials for " nonconvex low-rank symmetric tensor completion from noisy data ". `http://www.princeton.edu/~yc5/publications/NonconvexTC.pdf`.

[8] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, April 2009.

[9] E. J. Candes, X. Li, and M. Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

[10] H. Chen, G. Raskutti, and M. Yuan. Non-convex projected gradient descent for generalized low-rank tensor regression. *The Journal of Machine Learning Research*, 20(1):172–208, 2019.

[11] Y. Chen and E. Candès. The projected power method: An efficient algorithm for joint alignment from pairwise differences. *Comm. Pure and Appl. Math.*, 71(8):1648–1714, 2018.

[12] Y. Chen and E. J. Candès. Solving random quadratic systems of equations is nearly as easy as solving linear systems. *Comm. Pure Appl. Math.*, 70(5):822–883, 2017.

[13] Y. Chen and Y. Chi. Robust spectral compressed sensing via structured matrix completion. *IEEE Transactions on Information Theory*, 60(10):6576 – 6601, 2014.

[14] Y. Chen and Y. Chi. Harnessing structures in big data via guaranteed low-rank matrix estimation: Recent theory and fast algorithms via convex and nonconvex optimization. *IEEE Signal Processing Magazine*, 35(4):14–31, July 2018.

[15] Y. Chen, Y. Chi, J. Fan, and C. Ma. Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval. *Mathematical Programming*, pages 1–33, 2018.

[16] Y. Chen, J. Fan, C. Ma, and K. Wang. Spectral method and regularized MLE are both optimal for top-K ranking. *Annals of Statistics*, 47(4):2204–2235, August 2019.

[17] Y. Chen, J. Fan, C. Ma, and Y. Yan. Inference and uncertainty quantification for noisy matrix completion. *accepted to the Proceedings of the National Academy of Sciences (PNAS)*, 2019.

[18] Y. Chen and M. J. Wainwright. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

[19] J. Y. Cheng, T. Zhang, M. T. Alley, M. Uecker, M. Lustig, J. M. Pauly, and S. S. Vasanawala. Comprehensive multi-dimensional MRI for the simultaneous assessment of cardiopulmonary anatomy and physiology. *Scientific reports*, 7(1):5330, 2017.

[20] Y. Chi, Y. M. Lu, and Y. Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269, 2019.

[21] M. A. Davenport and J. Romberg. An overview of low-rank matrix recovery from incomplete observations. *IEEE Journal of Selected Topics in Signal Processing*, 10(4):608–622, 2016.

[22] L. Ding and Y. Chen. The leave-one-out approach for matrix completion: Primal and dual analysis. *arXiv preprint arXiv:1803.07554*, 2018.

[23] N. El Karoui. On the impact of predictor geometry on the performance on high-dimensional ridge-regularized generalized robust regression estimators. *Probability Theory and Related Fields*, pages 1–81, 2015.

[24] G. Ely, S. Aeron, N. Hao, and M. E. Kilmer. 5D and 4D pre-stack seismic data completion using tensor nuclear norm (TNN). In *SEG Technical Program Expanded Abstracts 2013*, pages 3639–3644. Society of Exploration Geophysicists, 2013.

[25] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010, 2011.

[26] R. Ge and T. Ma. On the optimization landscape of tensor decompositions. In *Advances in Neural Information Processing Systems*, pages 3653–3663, 2017.

[27] D. Gilboa, S. Buchanan, and J. Wright. Efficient dictionary learning with gradient descent. *arXiv preprint arXiv:1809.10313*, 2018.

[28] D. Goldfarb and Z. Qin. Robust low-rank tensor recovery: Models and algorithms. *SIAM Journal on Matrix Analysis and Applications*, 35(1):225–253, 2014.

[29] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *IEEE Transactions on Information Theory*, 57(3):1548–1566, March 2011.

[30] B. Hao, B. Wang, P. Wang, J. Zhang, J. Yang, and W. W. Sun. Sparse tensor additive regression. *arXiv preprint arXiv:1904.00479*, 2019.

[31] B. Hao, A. Zhang, and G. Cheng. Sparse and low-rank tensor estimation via cubic sketchings. *arXiv preprint arXiv:1801.09326*, 2018.

[32] C. J. Hillar and L.-H. Lim. Most tensor problems are np-hard. *Journal of the ACM (JACM)*, 60(6):45, 2013.

[33] S. B. Hopkins, T. Schramm, J. Shi, and D. Steurer. Fast spectral algorithms from sum-of-squares proofs: tensor decomposition and planted sparse vectors. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 178–191. ACM, 2016.

[34] B. Huang, C. Mu, D. Goldfarb, and J. Wright. Provable models for robust low-rank tensor completion. *Pacific Journal of Optimization*, 11(2):339–364, 2015.

[35] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *ACM symposium on Theory of computing*, pages 665–674, 2013.

[36] P. Jain and S. Oh. Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems*, pages 1431–1439, 2014.

[37] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, and G. Liu. Tensor completion using total variation and low-rank matrix factorization. *Information Sciences*, 326:243–257, 2016.

[38] H. Kasai and B. Mishra. Low-rank tensor completion: a riemannian manifold preconditioning approach. In *International Conference on Machine Learning*, pages 1012–1021, 2016.

[39] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 56(6):2980 –2998, June 2010.

[40] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *IEEE transactions on information theory*, 56(6):2980–2998, 2010.

[41] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *J. Mach. Learn. Res.*, 11:2057–2078, 2010.

[42] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover. Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM Journal on Matrix Analysis and Applications*, 34(1):148–172, 2013.

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

[44] N. Kreimer, A. Stanton, and M. D. Sacchi. Tensor completion based on nuclear norm minimization for 5d seismic data reconstruction. *Geophysics*, 78(6):V273–V284, 2013.

[45] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient descent only converges to minimizers. In *Conference on learning theory*, pages 1246–1257, 2016.

[46] X. Li, Y. Ye, and X. Xu. Low-rank tensor completion with total variation for visual data inpainting. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[47] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):208–220, 2013.

[48] K. Lounici. High-dimensional covariance matrix estimation with missing observations. *Bernoulli*, 20(3):1029–1058, 2014.

[49] C. Ma, K. Wang, Y. Chi, and Y. Chen. Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval, matrix completion and blind deconvolution. *accepted to Foundations of Computational Mathematics*, 2018.

[50] A. Montanari and N. Sun. Spectral algorithms for tensor completion. *Communications on Pure and Applied Mathematics*, 71(11):2381–2425, 2018.

[51] C. Mu, B. Huang, J. Wright, and D. Goldfarb. Square deal: Lower bounds and improved relaxations for tensor recovery. In *International conference on machine learning*, pages 73–81, 2014.

[52] A. Pananjady and M. J. Wainwright. Value function estimation in markov reward processes: Instance-dependent ℓ_∞-bounds for policy evaluation. *arXiv preprint arXiv:1909.08749*, 2019.

[53] A. Potechin and D. Steurer. Exact tensor completion with sum-of-squares. In *Conference on Learning Theory*, pages 1619–1673, 2017.

[54] H. Rauhut, R. Schneider, and Ž. Stojanac. Low rank tensor recovery via iterative hard thresholding. *Linear Algebra and its Applications*, 523:220–262, 2017.

[55] J. D. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *International conference on Machine learning*, pages 713–719. ACM, 2005.

[56] E. Richard and A. Montanari. A statistical model for tensor PCA. In *Advances in Neural Information Processing Systems*, pages 2897–2905, 2014.

[57] B. Romera-Paredes and M. Pontil. A new convex relaxation for tensor completion. In *Advances in Neural Information Processing Systems*, pages 2967–2975, 2013.

[58] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller. Tensor-based formulation and nuclear norm regularization for multienergy computed tomography. *IEEE Transactions on Image Processing*, 23(4):1678–1693, 2014.

[59] D. Shah and C. L. Yu. Iterative collaborative filtering for sparse noisy tensor estimation. *arXiv preprint arXiv:1908.01241*, 2019.

[60] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.

[61] M. Steinlechner. Riemannian optimization for high-dimensional tensor completion. *SIAM Journal on Scientific Computing*, 38(5):S461–S484, 2016.

[62] R. Sun and Z.-Q. Luo. Guaranteed matrix completion via non-convex factorization. *IEEE Transactions on Information Theory*, 62(11):6535–6579, 2016.

[63] G. Tang and P. Shah. Guaranteed tensor decomposition: A moment approach. In *International Conference on Machine Learning*, pages 1491–1500, 2015.

[64] R. Tomioka, K. Hayashi, and H. Kashima. Estimation of low-rank tensors via convex optimization. *arXiv preprint arXiv:1010.0789*, 2010.

[65] G. Wang, G. B. Giannakis, and Y. C. Eldar. Solving systems of random quadratic equations via truncated amplitude flow. *IEEE Transactions on Information Theory*, 2017.

[66] W. Wang, V. Aggarwal, and S. Aeron. Tensor completion by alternating minimization under the tensor train (tt) model. *arXiv preprint arXiv:1609.05587*, 2016.

[67] D. Xia and M. Yuan. On polynomial time methods for exact low rank tensor completion. *arXiv preprint arXiv:1702.06980*, 2017.

[68] D. Xia, M. Yuan, and C.-H. Zhang. Statistically optimal and computationally efficient low rank tensor completion from noisy entries. *arXiv preprint arXiv:1711.04934*, 2017.

[69] Y. Xu, R. Hao, W. Yin, and Z. Su. Parallel matrix factorization for low-rank tensor completion. *Inverse Problems & Imaging*, 9(2):601–624, 2015.

[70] Y. Xu and W. Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.

[71] Q. Yao. Scalable tensor completion with nonconvex regularization. *arXiv preprint arXiv:1807.08725*, 2018.

[72] J. Ying, H. Lu, Q. Wei, J.-F. Cai, D. Guo, J. Wu, Z. Chen, and X. Qu. Hankel matrix nuclear norm regularized tensor completion for n-dimensional exponential signals. *IEEE Transactions on Signal Processing*, 65(14):3702–3717, 2017.

[73] M. Yuan and C.-H. Zhang. On tensor completion via nuclear norm minimization. *Foundations of Computational Mathematics*, 16(4):1031–1068, 2016.

[74] M. Yuan and C.-H. Zhang. Incoherent tensor norms and their applications in higher order tensor completion. *IEEE Transactions on Information Theory*, 63(10):6753–6766, 2017.

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

[76] H. Zhang, Y. Zhou, Y. Liang, and Y. Chi. A nonconvex approach for phase retrieval: Reshaped wirtinger flow and incremental algorithms. *The Journal of Machine Learning Research*, 18(1):5164–5198, 2017.

[77] Z. Zhang and S. Aeron. Exact tensor completion using t-svd. *IEEE Trans. Signal Processing*, 65(6):1511–1526, 2017.

[78] Q. Zheng and J. Lafferty. Convergence analysis for rectangular matrix completion using Burer-Monteiro factorization and gradient descent. *arXiv:1605.07051*, 2016.

[79] Y. Zhong and N. Boumal. Near-optimal bound for phase synchronization. *SIAM Journal on Optimization*, 2018.