Higher order Matching Pursuit for Low Rank Tensor Learning

Yuning Yang, Siamak Mehrkanoon and Johan A.K. Suykens, *Fellow IEEE*

Abstract

Low rank tensor learning, such as tensor completion and multilinear multitask learning, has received much attention in recent years. In this paper, we propose higher order matching pursuit for low rank tensor learning problems with a convex or a nonconvex cost function, which is a generalization of the matching pursuit type methods. At each iteration, the main cost of the proposed methods is only to compute a rank-one tensor, which can be done efficiently, making the proposed methods scalable to large scale problems. Moreover, storing the resulting rank-one tensors is of low storage requirement, which can help to break the curse of dimensionality. The linear convergence rate of the proposed methods is established in various circumstances. Along with the main methods, we also provide a method of low computational complexity for approximately computing the rank-one tensors, with provable approximation ratio, which helps to improve the efficiency of the main methods and to analyze the convergence rate. Experimental results on synthetic as well as real datasets verify the efficiency and effectiveness of the proposed methods.

Key words: Matching pursuit, tensor completion, multitask learning, nonconvex, rank-one tensor, linear convergence.

I. Introduction

Tensors, appearing as the higher order generalization of vectors and matrices, make it possible to represent data that have intrinsically many dimensions, and give a better understanding of the relationship behind the information from a higher order perspective. In many machine learning problems such as tensor completion [1]–[4], multilinear multitask learning (MLMTL) [5]–[7] and tensor regression [8], one often aims at learning a tensor that has low rankness. For example, in tensor completion, the goal is to learn a low rank tensor provided that only partial observations are available. In the context of MLMTL, to allow for common information shared between tasks to pursuit better generalization, by learning several tasks simultaneously, where each task is indexed by more than two indices, all the tasks can be represented by a tensor assumed to lie in a low dimensional spaces. In tensor regression, to better understand the information behind high dimensionality data, the weight vector is represented by a low rank tensor. These applications give rise to low rank tensor learning.

Commonly speaking, to learn a low rank tensor, tensor learning minimizes a real-valued cost function $F : \mathbb{T} \to \mathbb{R}$ subject to some constraints or with regularizations to encourage the low rank property of the learned tensor. Here $\mathbb{T} := \mathbb{R}^{n_1 \times \cdots \times n_N}$ denotes an N-th order tensor space, and $F(\cdot)$ is a continuous function. A widely used regularization is the sum of mode-d matrix nuclear norms [1]–[3], [5], [6], [9], [10], which encourages the tensor to have low Tucker rank [11]. Some variations of the nuclear norm, such as the Schatten-p norm [3], [4], the latent norm and the scaled norm [6], as well as other variations [12], [13], have been studied. The advantage of the above approaches are their convexity, which enables them to be solved by many existing algorithms, while a main drawback is that all the approaches rely on solving singular value decompositions (SVD), which lacks scalability. Another category of approaches decomposes the tensor into several factors, and applies the alternating minimization rule to solve the resulting optimization problems, see, e.g., [5], [11], [14]. This type of approaches avoids computing SVDs, but lacks global convergence analysis. Recently, tensor nuclear norm based algorithms are proposed in [15], [16]. This type of algorithms solves a simple and efficient subproblem at each iteration, which is scalable to large-scale problems. However they are specifically designed for convex $F(\cdot)$, and the stepsizes have some restrictions.

In this paper, motivated by the simple and efficient matching pursuit (MP) methods for sparse approximation [17]–[19], signal recovery [20], matrix compoletion [21] and greedy method for tensor approximation [22], we propose higher order matching pursuit (HoMP) methods for solving low rank tensor learning problems, either with a convex or nonconvex cost function. At each iteration, the classical MP selects an "atom" from a given dictionary, and then updates the new trial based on the linear combination of the current trial and the selected atom, with suitably chosen weights, whereas in tensor learning setting, the atom is a rank-one tensor, which has to be learned based on the gradient information of $F(\cdot)$. Finding such a rank-one tensor reduces to computing a tensor spectral norm, or known as the tensor singular value problem [23], [24]. Although solving such a subproblem exactly is NP-hard in general [25], approximation methods exist, and fortunately, as MP, HoMP allows to use an approximation solution. This feature makes HoMP particularly suitable for tensor learning. When choosing the weights, if the cost function $F(\cdot)$ is associated with a least squares loss, then three strategies, which are in accordance with matching pursuit [17], economic orthogonal MP [21] (or relaxed MP, see, e.g., [26]) and orthogonal matching pursuit [18],

Y. Yang, S. Mehrkanoon and J.A.K. Suykens are with the Department of Electrical Engineering, STADIUS, KU Leuven, B-3001 Leuven, Belgium (email: yuning.yang, siamak.mehrkanoon, johan.suykens@esat.kuleuven.be).

can be applied. We then generalize HoMP to the case that $F(\cdot)$ is nonconvex, where the weights are chosen by minimizing a quadratic function which majorizes $F(\cdot)$. Along with the main HoMP methods, an efficient algorithm will be presented to approximately and efficiently solve the tensor singular value problem mentioned above, with a provable approximation ratio. This ratio is important in analyzing the convergence rate, which will be used extensively in Sect. IV. Besides the efficiency, another advantage of HoMP-type methods is its low storage requirement, as will be explained in Sect. III.

The convergence rate of HoMP is analyzed in two specific problems: tensor completion and MLMTL. Specifically, we first show that, if $F(\cdot)$ is associated with the least squares loss, then HoMP converges linearly for the two specific problems. We then generalize our analysis to a class of loss functions, which may be nonconvex and includes many robust losses as special cases. Interestingly, the linear convergence rate can still be established in these cases.

In a nutshell, our contribution is summarized as follows:

1) We propose efficient HoMP methods for tensor learning, which are applicable for problems with with convex or nonconvex cost functions;
2) We present an efficient method for selecting rank-one tensors, with provable approximation ratio. The ratio is important in analyzing the convergence rate.
3) We establish linear convergence of the proposed HoMP methods, either for problems with convex or nonconvex cost functions.

The rest of this paper is organized as follows. Sect. I-A gives preliminaries on tensors. Low rank tensor learning problems are formulated in Sect. II, and specified by tensor completion and MLMTL. The HoMP-type methods, their related work, and an efficient algorithm for selecting rank-one tensors will be detailed in Sect. III. Sect. IV is focused on analyzing the convergence rate. Numerical experiments will be conducted in Sect. V. Finally, Sect. VI draws some conclusions.

A. Preliminaries on tensors

Vectors are written as $(\mathbf{a}, \mathbf{b}, \ldots)$, matrices correspond to $(\mathbf{A}, \mathbf{B}, \ldots)$, and tensors are written as $(\mathcal{A}, \mathcal{B}, \cdots)$. $\mathbb{T} := \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$ denotes an N-th order tensor space.

For two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{T}$, their inner product is given by $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1=1}^{n_1} \cdots \sum_{i_N=1}^{n_N} \mathcal{A}_{i_1 \cdots i_N} \mathcal{B}_{i_1 \cdots i_N}$. The Frobenius norm of $\mathcal{A}$ is defined by $\|\mathcal{A}\|_F = \langle \mathcal{A}, \mathcal{A} \rangle^{1/2}$. The outer product of vectors $\mathbf{x}_i \in \mathbb{R}^{n_i}$, $i = 1, \ldots, N$ is denoted as $\mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_N$ and is a rank-one tensor in $\mathbb{T}$ defined by $(\mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_N)_{i_1 \cdots i_N} = \prod_{j=1}^{N} \mathbf{x}_{j,i_j}$. The mode-$d$ tensor-matrix multiplication of a tensor $\mathcal{X} \in \mathbb{T}$ with a matrix $U \in \mathbb{R}^{J_d \times n_d}$ is a tensor of size $n_1 \times \cdots \times n_{d-1} \times J_d \times n_{d+1} \times \cdots \times n_N$.

1) Tensor-matrix mode-d unfolding and mode-(p,q) unfolding: The mode-d unfolding of tensor $\mathcal{A}$ is denoted as $\mathbf{A}_{(d)}$ by arranging the mode-d fibers to be the columns of the resulting matrix. For an N-th order tensor $\mathcal{A}$, the mode-(p,q) unfolding is to choose the (p,q) modes and merge them into the first mode (row) of the unfolding matrix, and the remaining $N-p-q$ modes are merged into the second mode (column). The unfolding is denoted as $\mathbf{A}_{[p,q;N\setminus\{p,q\}]}$ and can be simplified as $\mathbf{A}_{(p,q)}$ if necessary, where the semicolon specifies a new mode. We explain it by an example. For a 4-th order tensor $\mathcal{A}$, its mode-$(1,2)$ unfolding is $\mathbf{A}_{[1,2;3,4]}$, defined by $(\mathbf{A}_{[1,2;3,4]})_{(i_1-1)n_2+i_2,(i_3-1)n_4+i_4} = \mathcal{A}_{i_1 i_2 i_3 i_4}$.

2) Tensor rank: There are mainly two types of tensor rank, namely the CP-rank and the Tucker-rank [11]. The CP-rank is defined as the minimum integer R such that for a tensor $\mathcal{X}$, it can be factorized as a sum of R rank-one tensors. $\mathrm{rank}_{\mathrm{CP}}(\cdot)$ will be used to denote the CP-rank in this paper. The Tucker-rank of an N-th order tensor $\mathcal{X}$ is an N tuple, whose i-th entry is the rank of the unfolding matrix $X_{(i)}$.

3) Tensor singular value problem: For a tensor $\mathcal{X} \in \mathbb{T}$, its largest singular value is defined as [23]

$$\max_{\|\mathcal{Y}\|_F=1, \mathrm{rank}_{\mathrm{CP}}(\mathcal{Y})=1} \langle \mathcal{X}, \mathcal{Y} \rangle, \tag{1}$$

which is equivalent to the tensor spectral norm $\|\mathcal{X}\|_2$ and is dual to the tensor spectral norm, see, e.g., [24]. Solving such a problem is NP-hard in general [25].

II. Low Rank Tensor Learning Formulation

As introduced in the introduction, low rank tensor learning seeks a low rank tensor solution via minimizing a cost function. Mathematically, the model considered in this work is of the following general form

$$\min_{\mathcal{W}} F(\mathcal{W}) \quad \text{s.t.} \quad \mathrm{rank}_{\mathrm{CP}}(\mathcal{W}) \leq K, \tag{2}$$

where $F : \mathbb{T} \to \mathbb{R}$ denotes the cost function which is continuously differentiable, either convex or nonconvex. The low CP-rank constraint encourages the learned tensor to have low rank structure. We specify the model (2) via two specific applications: tensor completion and MLMTL.

A. Tensor completion

The goal of tensor completion is to infer a tensor (possibly low rank) from its partial observations. Mathematically, given a partially observed tensor $\mathcal{B}_\Omega$ where Ω denotes the index set of observed entries, the problem can be formulated as

$$\min_{\mathcal{W}\in\mathbb{T}} \ r(\mathcal{W}) \ \ \text{s.t. } \mathcal{W}_\Omega = \mathcal{B}_\Omega,$$

where $r(\cdot)$ is used to control the low rankness of the tensor, such as $\text{rank}_{\text{CP}}(\cdot)$, and the sum of nuclear norms. In our setting, letting $F(\mathcal{W}) := \sum_{(i_1,\ldots,i_N)\in\Omega} \ell(\mathcal{W}_{i_1\cdots i_N} - \mathcal{B}_{i_1\cdots i_N})$ with a specific loss $\ell(\cdot)$, we model the problem as follows

$$\min_{\text{rank}_{\text{CP}}(\mathcal{W})\le K} F(\mathcal{W}),$$

with K being a positive integer to control the CP-rank of $\mathcal{W}$.

B. Multilinear multitask learning (MLMTL)

MLMTL learns many tasks simultaneously, where each task is indexed by more than two indices [5], [6]. An example is to predict consumers' ratings for restaurants, where each rating contains several aspects. Then each task is indexed by consumer and aspect. Since each task can be represented by a weight vector, all the tasks jointly yield a third-order tensor, see, e.g., [5]. In the following, we restrict ourselves to multilinear multitask regression. Specifically, We consider T tasks, each of which is specified by a weight vector $\mathbf{w}^t \in \mathbb{R}^D$ that corresponds to a linear function $\langle \mathbf{x}, \mathbf{w}^t\rangle$, where $\mathbf{x}$ is an observed input. Provided that the associated observed output is y, we employ a specific loss $\ell(\langle \mathbf{x}, \mathbf{w}^t\rangle - y)$. For each task $\mathbf{w}^t$, a finite set of training samples $\{(\mathbf{x}_i^t, y_i^t)\}_{i=1}^{m_t}$ is available, and we aim at minimizing the empirical risk $F(\mathbf{W})$ defined as

$$F(\mathbf{W}) := \sum_{t=1}^{T} m_t^{-1} \sum_{i=1}^{m_t} \ell(\langle \mathbf{x}_i^t, \mathbf{w}^t\rangle - y_i^t),$$

where $\mathbf{W} = [\mathbf{w}^1, \ldots, \mathbf{w}^T] \in \mathbb{R}^{D\times T}$ is the weight matrix. For each task t, we assume that it is related to $N \ge 2$ indices, each of which varies from 1 to n_i, $i = 1, \ldots, N$. That is, the task $\mathbf{w}^t$ can be identified by the indices $(i_1, \ldots, i_N) \in [n_1] \times \cdots \times [n_N]$. In this case, we have $T = \prod_{i=1}^N n_i$. Correspondingly, the matrix $\mathbf{W}$ can be folded to an $(N+1)$-th order tensor $\mathcal{W}$, with size $D \times n_1 \times \cdots \times n_N$, and $\mathbf{W}$ can be regarded as the mode-1 unfolding of $\mathcal{W}$. We also denote $F(\mathcal{W}) = F(\mathbf{W})$. Assuming that the tasks share certain common structure, our model is defined as

$$\min_{\text{rank}_{\text{CP}}(\mathcal{W})\le K} F(\mathcal{W}),$$

Therefore, both of our models of tensor completion and MLMTL adopts the CP-rank to control the low rankness of the learned tensor, which is quite different from models based on nuclear norm regularizations [1]–[3], [5], [6], [9].

III. Higher Order Matching Pursuit

Having presenting our low rank tensor learning problem (2), the goal of this section is to introduce the Higher order Matching Pursuit (HoMP) methods to solve it. HoMP methods are presented in Algorithm 1.

Algorithm 1 Higher order Matching Pursuit (HoMP) for low rank tensor learning

Input: $\mathcal{W}^{(0)} = 0$; $\overline{\alpha}_0 = 0$; $K \ge 1$.
Output: the resulting tensor $\mathcal{W}^{(K)}$.
for $k = 1$ **to** K **do**

- Select a normalized rank-one tensor $\mathcal{S}^{(k)}$:

$$\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle \ge \beta \max_{\|\mathcal{S}\|_F=1, \text{rank}_{\text{CP}}(\mathcal{S})=1} \langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}\rangle \quad (0 < \beta \le 1) \tag{3}$$

- Update
  1) $\mathcal{W}^{(k+1)} = \mathcal{W}^{(k)} + \overline{\alpha}\mathcal{S}^{(k)}$, $\overline{\alpha} = \arg\min_\alpha F(\mathcal{W}^{(k)} + \alpha\mathcal{S}^{(k)})$ (HoMP-LS)
  2) $\mathcal{W}^{(k+1)} = \overline{\alpha_1}\mathcal{W}^{(k)} + \overline{\alpha_2}\mathcal{S}^{(k)}$, $(\overline{\alpha_1}, \overline{\alpha_2}) = \arg\min_{(\alpha_1,\alpha_2)} F(\alpha_1\mathcal{W}^{(k)} + \alpha_2\mathcal{S}^{(k)})$ (HoRMP-LS)
  3) $\mathcal{W}^{(k+1)} = \sum_{i=0}^k \overline{\alpha_i}\mathcal{S}^{(i)}$, $\overline{\boldsymbol{\alpha}} = (\overline{\alpha_0}, \ldots, \overline{\alpha_k})^\top = \arg\min_{\alpha\in\mathbb{R}^{k+1}} F(\sum_{i=0}^k \alpha_i\mathcal{S}^{(i)})$ (HoOMP-LS)
  4) $\mathcal{W}^{(k+1)} = \mathcal{W}^{(k)} + \overline{\alpha}\mathcal{S}^{(k)}$, $\overline{\alpha} = -\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle / L$ (HoMP-G) (L is a Lipschitz constant)

end for

We describe the method in more details. Given a cost function $F(\cdot)$ of low rank tensor learning, with initial guess $\mathcal{W}^{(0)}$ being the zero tensor, at each iteration, HoMP can be divided into two steps: the selection step and the updating step. The selection step chooses certain atom, which is a rank-one tensor $\mathcal{S}^{(k)}$ by solving the tensor singular value problem (1) approximately with an approximation ratio β, where $\mathcal{X} = \nabla F(\mathcal{W}^{(k)})$, as shown in (3). The approximation ratio is important in convergence rate analysis, as will be shown in Sect. IV. The updating step adaptively computes the weights and updates the new trial. This step has two cases:

1) If $F(\cdot)$ is associated with a least squares loss, and $F(\cdot)$ with respect to weights α is quadratic, then three strategies can be considered: higher order MP with least squares loss (HoMP-LS), higher order relaxed MP with least squares loss (HoRMP-LS) and higher order orthogonal MP with least squares loss (HoOMP-LS), as shown in Algorithm 1, where all the weights can be

computed by solving certain least squares problems. These strategies are respectively in accordance with MP [17], economic orthogonal MP [21] (or relaxed MP, see, e.g., [26]) and orthogonal MP [18], and generalize them to tensor learning setting.

2) If $F(\cdot)$ is associated with a general loss which is possibly nonconvex, while the gradient $\nabla F(\cdot)$ is Lipschitz continuous with Lipschitz constant L, then the strategy higher order MP with a general loss (HoMP-G) can be applied. In fact, the weight is computed by minimizing a quadratic function that majorizes $F(\cdot)$ at $\mathcal{W}^{(k+1)}$.

If Algorithm 1 stops within K iterations, then it generates a feasible solution to (2). Comparing between HoMP, HoRMP and HoOMP, one can see that HoOMP may obtain a better new trial, as it updates the weights most greedily, while HoMP updates the weights least greedily. However, computing the weights for HoOMP may be time consuming, as it may require to solve a linear equations system. HoRMP, which considers the linear combination between the current trial and the new atom, can be regarded as a trade-off between HoMP and HoOMP.

The main computational cost of HoMP-type methods is the selection step (3). Comparing with those methods that require to solve SVD, solving (3) will be more efficient, as will be detailed in subsection III-B.

Another advantage of HoMP-type methods is the low storage requirement. Suppose we work in a tensor space $\mathbb{T}$ and the methods stop within K iterations with K not too large; since the learned tensor is a combination of some rank-one tensors, and each rank-one tensor can be represented by the outer product of N vectors, the whole tensor can be stored by using $\sum_{i=1}^{N} n_i \cdot K$ storage only, against using $\prod_{i=1}^{N} n_i$ to store the whole tensor. This can help to break the curse of dimensionality.

A. Related work

As mentioned earlier, HoMP-type methods are motivated by MP-type methods for sparse approximation [17]–[19], signal recovery [20] and matrix completion [21]. The classical MPs iteratively select atoms from a redundant dictionary, one at a time, and then use their certain linear combination to approximate a given signal. Recently, [21] generalizes MPs to matrix completion, where the cost function is least squares based, and proves the linear convergence of their methods. Our work generalizes [21] in the following senses: 1) we generalize MPs to tensor learning problems; 2) our methods can be adapted to problems with a general loss; 3) the way of selecting atoms in tensor setting is more challenge than in matrix cases; 4) we establish linear convergence rate for a wide range of loss functions, which may possibly be nonconvex.

Another issue related to HoMPs is the approach of successive rank-one approximation to tensors (SR1A), see, [27, Sect. 3.3] and the references therein. In general, consider a linear system $\mathcal{A}(\mathcal{W}) \approx \mathbf{b}$ where $\mathcal{A}$ is a linear operator, $\mathbf{b}$ is a vector and $\mathcal{W}$ is a low rank tensor to be determined. SR1A updates the new trial as $\mathcal{W}^{(k+1)} = \mathcal{W}^{(k)} + \mathcal{S}^{(k)}$ where $\mathcal{S}^{(k)}$ is a rank-one tensor which minimizes some convex cost function $E(\mathcal{A}(\mathcal{X}) - \mathbf{b})$, see, e.g., [28], [29]. Comparing with SR1A, HoMPs are more general in terms of the problems to be solved, the cost function $F(\cdot)$ to be minimized, and the strategies of choosing the weights. Moreover, finding the rank-one term $\mathcal{S}$ in SR1A is either intractable or needs an alternating minimization strategy [22] without explicit approximation ratio, while we allow an approximation solution (3).

HoMPs are also closely related to conditional gradient (CG) methods for tensor learning [15], [16]. CG, also known as the Frank-Wolfe method [30], is a classical method for constrained convex optimization problems, and regains attention in recent years, see, e.g., [31]. At each iteration, CG also computes an atom. Then, different from MPs, CG performs a convex combination of the selected atom and the current trial, which restricts the new trial to still lie in the convex constraint. This difference leads to significant differences to the models behind MPs and CG, where MPs minimize the cost function constrained by a "hard" constraint, while CG minimizes the cost function constrained by its "soft" counterpart, such as L_0 norm versus L_1 norm, matrix/tensor rank versus matrix/tensor nuclear norm.

B. Efficiently computing the selection step (3)

Since solving (1) exactly is NP-hard in general [25], the goal of this subsection is to present a method to find an approximation solution of (3) with low computational complexity, and to explicitly derive the approximation ratio. In the literature, several approximation methods have been proposed, e.g., the power-type methods [32]–[34], the Newton method [35], and others [36]–[40]. However, these methods are not very efficient in our setting.

For a $2d$-th order tensor $\mathcal{A}$, our method is defined recursively as follows, where the output is a set of $2d$ normalized vectors, whose outer product yields the approximation solution.

Subroutine 1: $(\mathbf{x}_1, \ldots, \mathbf{x}_{2d}) =$ **ApproxSpectral2d**$(\mathcal{A})$

1) If the order is 2, return the singular vector pair $(\mathbf{x}_1, \mathbf{x}_2)$ corresponding to the leading singular value of $\mathcal{A}$;
2) Compute the (inexact) singular vector pair $(\mathbf{x}_{[1,2]}, \mathbf{x}_{[3,\ldots,2d]})$ corresponding to the leading singular value of matrix $\mathbf{A}_{[1,2;3,\ldots,2d]}$.
3) Fold the vector $\mathbf{x}_{[1,2]}$ to matrix $\mathbf{X}_{[1;2]}$ and compute the leading singular vector pair $\mathbf{x}_{[1]}$ and $\mathbf{x}_{[2]}$;
4) Denote the $(2d-2)$-th order tensor $\mathcal{Y} := \mathcal{A} \times_1 \mathbf{x}_1^\top \times_2 \mathbf{x}_2^\top$; compute $(\mathbf{x}_3, \ldots, \mathbf{x}_{2d}) =$ **ApproxSpectral2d**$(\mathcal{Y})$.
5) Return $(\mathbf{x}_1, \ldots, \mathbf{x}_{2d})$.

At each recursion of Subroutine 1, the dominant cost is Step 2), i.e., to compute the leading singular value of a matrix of size $n^2 \times n^{(2d-2k)}$ at the k-th recursion, with $1 \le k \le (d-1)$. Although the computational complexity of only computing the leading singular value is less than doing the full SVD, when the size goes high, this procedure still takes much time. Empirically we find that there is no need to compute the singular value precisely, and running a few power iterations is acceptable. That is, we can compute Step 2) inexactly. Suppose a few power iterations have been performed in Step 2), and we obtain a normalized vector pair $(\mathbf{x}_{[1,2]}, \mathbf{x}_{[3,\ldots,2d]})$ such that $\mathbf{A}_{(1,2)}\mathbf{x}_{[3,\ldots,2d]} = \alpha_{(d)}\|\mathbf{A}_{(1,2)}\|_2 \mathbf{x}_{[1,2]}$ where $0 < \alpha_{(d)} \le 1$, and $\mathbf{A}_{(1,2)}$ is short for $\mathbf{A}_{[1,2;3,\ldots,2d]}$. Then we can establish the following lower bound. For simplicity, we may assume $n_1 = n_2 = \cdots = n_{2d} = n$.

Proposition 1: Let the order of the tensor be $2d$. Suppose at the k-th recursion, the vectors obtained in Step 2) are normalized and satisfy

$$\mathbf{A}_{(1,2)}\mathbf{x}_{[3,\ldots,2d-2k+2]} = \alpha_{(k)}\|\mathbf{A}_{(1,2)}\|_2 \mathbf{x}_{[1,2]}, \tag{4}$$

where $0 < c \le \alpha_{(k)} \le 1$, $1 \le k \le d-1$ and c is a constant. Then there holds

$$\langle \mathcal{A}, \mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_{2d} \rangle \ge \frac{\prod_{k=1}^{d-1} \alpha_{(k)} \|\mathbf{A}_{(1,2)}\|_2}{n^{\max\{0, 3d/2-2\}}} \ge \frac{\prod_{k=1}^{d-1} \alpha_{(k)} \|\mathcal{A}\|_2}{n^{\max\{0, 3d/2-2\}}}. \tag{5}$$

Proof: We denote $v = \langle \mathcal{A}, \mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_{2d} \rangle$, tensor $\mathcal{M}_1 := \mathcal{A} \times_1 \mathbf{x}_1^\top \times_2 \mathbf{x}_2^\top$ and matrix $\mathbf{M}_2 := \langle \mathcal{A}, \cdot \otimes \cdot \otimes \mathcal{X}_{[3;\cdots;2d]} \rangle$. Here $\mathcal{X}_{[3;\cdots;2d]}$ is a $(2d-2)$-th order tensor folded by the vector $\mathbf{x}_{[3,\ldots,2d]}$ generated in Step 2) of Subroutine 1, either exactly or inexactly, and the (i,j)-th entry of matrix $\mathbf{M}_2$ is given by the inner product of $\mathcal{A}(i,j,:,:,\ldots,:)$ and $\mathcal{X}_{[3;\cdots;2d]}$. For ease of notation, we also denote $\mathbf{M}_1$ the mode-$(1,2)$ unfolding of $\mathcal{M}_1$. We use the induction method on d. When $d = 1$, (5) holds. Suppose (5) holds when $d = l \ge 2$. When $d = l+1$, there holds

$$\begin{aligned}
v &= \langle \mathcal{M}_1, \mathbf{x}_3 \otimes \cdots \otimes \mathbf{x}_{2(l+1)} \rangle \\
&\ge \frac{\prod_{k=2}^{d-1} \alpha_{(k)} \|\mathbf{M}_1\|_2}{n^{3/2l-2}} \text{ (from Step 4) and the induction)} \\
&\ge \frac{\prod_{k=2}^{d-1} \alpha_{(k)} \|\mathbf{M}_1\|_F}{n^{3/2l-2} \cdot n} = \max_{\|\mathcal{X}\|_F=1} \frac{\prod_{k=2}^{d-1} \alpha_{(k)} \langle \mathcal{M}_1, \mathcal{X} \rangle}{n^{3/2l-1}} \\
&\ge \frac{\prod_{k=2}^{d-1} \alpha_{(k)} \langle \mathcal{M}_1, \mathcal{X}_{[3;\cdots;2(l+1)]} \rangle}{n^{3/2l-1}} \\
&= \frac{\prod_{k=2}^{d-1} \alpha_{(k)} \langle \mathbf{M}_2, \mathbf{x}_1 \otimes \mathbf{x}_2 \rangle}{n^{3/2l-1}} \\
&= \frac{\prod_{k=1}^{d-1} \alpha_{(k)} \|\mathbf{A}_{(1,2)}\|_2 \langle \mathbf{x}_1^\top \mathbf{X}_{[1;2]} \mathbf{x}_2 \rangle}{n^{3/2l-1}} \qquad \text{(from (4))} \\
&\ge \frac{\prod_{k=1}^{d-1} \alpha_{(k)} \|\mathbf{A}_{(1,2)}\|_2}{n^{3/2l-1/2}} \ge \frac{\prod_{k=1}^{d-1} \alpha_{(k)} \|\mathcal{A}\|_2}{n^{3/2(l+1)-2}},
\end{aligned}$$

where the second and the last inequalities follows from the relationship between matrix spectral norm and Frobenius norm. Therefore, (5) has been proved. ∎

After applying Subroutine 1, we can perform the following block coordinate updating subroutine [32] to further improve the solution quality. That is to say, we can get a larger value $\langle \mathcal{A}, \tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{2d} \rangle$ than $\langle \mathcal{A}, \mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_{2d} \rangle$ obtain in (5).

Subroutine 2: $(\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{2d}) =$ **BCU**$(\mathcal{A}, \mathbf{x}_1, \ldots, \mathbf{x}_{2d})$
for $i = 1, \ldots, d$
Compute the singular vector pair $(\mathbf{x}_{2i-1}, \mathbf{x}_{2i})$ corresponding to the leading singular value of the matrix $\mathcal{A} \times_1 \tilde{\mathbf{x}}_1 \times_2 \cdots \times_{2i-2} \tilde{\mathbf{x}}_{2i-2} \times_{2i+1} \mathbf{x}_{2i+1} \times_{2i+2} \cdots \times_{2d} \mathbf{x}_{2d}$.
end for

Of course, Subroutine 2 can be applied several times after Subroutine 1 to get a better solution. However, considering the trade-off between the computational cost and the solution quality, performing it a few times is enough to get an acceptable solution.

We discuss the computational complexity. At the k-th recursion of Subroutine 1, the complexity is $O(n^{2(d-k)}) + O(n^2)$; Subroutine 2 is $O(n^{2d}) + O(n^2)$. Thus the total complexity of Subroutine 1 together with Subroutine 2 is $\sum_{i=1}^{d} O(n^{2i})$, which is slightly worse than the power method for computing the leading singular value of a matrix of the same size of the tensor considered here. Therefore, comparing with methods based on SVD, e.g., [1], [2], HoMPs may be more efficient: for N-th order tensor space, at each iteration, methods based on SVD require to perform N SVD, with complexity at least $O(Nn^{N+1})$, which is higher than ours.

Finally, we note that tensors whose order is odd can always be treated as a tensor of even order, e.g., a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ can be seen as in the space $\mathbb{R}^{1 \times n_1 \times n_2 \times n_3}$.

IV. Convergence rate analysis

In this section, we will establish the linear convergence rate of Algorithm 1 for tensor completion and MLMTL, either with a convex cost function with least squares loss, or with a possibly nonconvex cost function. Tensors in this section are assumed to be of order N, $\mathcal{W} \in \mathbb{T}$. A key property for the analysis is inequality (5). To fit into the language of Algorithm 1, we write it as

$$\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)} \rangle \geq \rho \|\nabla F(\mathcal{W}^{(k)})_{(1,2)}\|_2, \tag{6}$$

where $0 < \rho \leq 1$ is a ratio depending on the size of the tensor, as that derived in (5), and $\nabla F(\mathcal{W}^{(k)})_{(1,2)}$ is the mode-$(1,2)$ unfolding of $\nabla F(\mathcal{W}^{(k)})$.

This section is organized as follows: in subsection IV-A we prove the linear convergence for tensor completion and MLMTL when $F(\cdot)$ is associated with a least squares loss; in subsection IV-B we extend the linear convergence results to $F(\cdot)$ with a possibly nonconvex loss.

A. Least squares

1) Tensor completion: In the least squares sense, the cost function of tensor completion can be written as

$$F(\mathcal{W}) = 1/2 \|\mathcal{W}_\Omega - \mathcal{B}_\Omega\|_F^2.$$

The following theorem will establish the linear convergence rate of HoMP-LS, HoRMP-LS and HoOMP-LS uniformly in terms of the objective value $F(\mathcal{W})$.

Theorem 1 (Linear convergence rate for tensor completion): Denote $\mathcal{R}^{(k)} := \mathcal{W}_\Omega^{(k)} - \mathcal{B}_\Omega$, and let $\mathcal{S}^{(k)}$ be generated by Subroutine 1 with input $\nabla F(\mathcal{W}^{(k)})$, and (6) holds. If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-LS, HoRMP-LS and HoOMP-LS, then there holds

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho^2}{n_1 n_2}\right) F(\mathcal{W}^{(k)}).$$

Proof: We first consider HoOMP-LS. For clarity we denote the weights $\overline{\boldsymbol{\alpha}}$ at the k-th step as $\overline{\boldsymbol{\alpha}}^k = (\overline{\alpha_0^k}, \ldots, \overline{\alpha_k^k})$. For convenience we denote $\|\mathcal{X}_\Omega\|_F = \|\mathcal{X}\|_\Omega$. By the definition of $\mathcal{R}^{(k)}$ and $\mathcal{W}^{(k+1)}$, there holds

$$\begin{aligned} 2F(\mathcal{W}^{(k+1)}) = \|\mathcal{R}^{(k+1)}\|_F^2 &= \|\mathcal{W}^{(k+1)} - \mathcal{B}\|_\Omega^2 = \min_{\boldsymbol{\alpha}} \|\sum_{i=0}^{k} \alpha_i \mathcal{S}^{(i)} - \mathcal{B}\|_\Omega^2 \\ &\leq \min_{\alpha} \|\sum_{i=0}^{k-1} \overline{\alpha_i^{k-1}} \mathcal{S}^{(i)} + \alpha \mathcal{S}^{(k)} - \mathcal{B}\|_\Omega^2 \\ &= \min_{\alpha} \|\mathcal{W}^{(k)} - \mathcal{B} + \alpha \mathcal{S}^{(k)}\|_\Omega^2 = \min_{\alpha} \|\mathcal{R}^{(k)} + \alpha \mathcal{S}^{(k)}\|_\Omega^2. \end{aligned}$$

For HoRMP-LS, we have

$$\begin{aligned} \|\mathcal{R}^{(k+1)}\|_F^2 &= \min_{(\alpha_1, \alpha_2)} \|\alpha_1 \mathcal{W}^{(k)} + \alpha_2 \mathcal{S}^{(k)} - \mathcal{B}\|_\Omega^2 \\ &\leq \min_{\alpha} \|\mathcal{W}^{(k)} - \mathcal{B} + \alpha \mathcal{S}^{(k)}\|_\Omega^2 = \min_{\alpha} \|\mathcal{R}^{(k)} + \alpha \mathcal{S}^{(k)}\|_\Omega^2. \end{aligned}$$

It follows that for HoMP-LS, it naturally holds $\|\mathcal{R}^{(k+1)}\|_F^2 = \min_\alpha \|\mathcal{R}^{(k)} + \alpha \mathcal{S}^{(k)}\|_\Omega^2$. Therefore, we can uniformly analyze the upper bound in terms of $\min_\alpha \|\mathcal{R}^{(k)} + \alpha \mathcal{S}^{(k)}\|_\Omega^2$. We have

$$\min_{\alpha} \|\mathcal{R}^{(k)} + \alpha \mathcal{S}^{(k)}\|_\Omega^2 = \|\mathcal{R}^{(k)}\|_F^2 - \frac{\langle \mathcal{R}^{(k)}, S^{(k)} \rangle_\Omega^2}{\|\mathcal{S}^{(k)}\|_\Omega^2} \leq \|\mathcal{R}^{(k)}\|_F^2 - \rho^2 \|\mathbf{R}_{(1,2)}^{(k)}\|_2^2 \leq \left(1 - \frac{\rho^2}{n_1 n_2}\right) \|\mathcal{R}^{(k)}\|_F^2,$$

where the first inequality follows from $\langle \mathcal{R}^{(k)}, \mathcal{S}^{(k)} \rangle_\Omega = \langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)} \rangle \geq \rho \|\nabla F(\mathcal{W}^{(k)})_{(1,2)}\|_2^2 = \rho \|\mathbf{R}_{(1,2)}^{(k)}\|_2^2$, and the last inequality is due to the relationship between the spectral norm and the Frobenius norm. ∎

2) Multilinear multitask learning: In the least squares sense, the cost function of multilinear multitask learning is given by

$$F(\mathcal{W}) = 1/2 \sum_{t=1}^{T} m_t^{-1} \sum_{i=1}^{m_t} (\langle \mathbf{x}_i^t, \mathbf{w}^t \rangle - y_i^t)^2.$$

Letting $\mathbf{X}^t \in \mathbb{R}^{m_t \times D}$ be formed by stacking the samples (transposed to rows) corresponding to the t-th task row by row, i.e., $(\mathbf{X}^t)^\top = [\mathbf{x}_1^t, \ldots, \mathbf{x}_{m_t}^t]$, we derive a compact form of F

$$F(\mathcal{W}) = 1/2 \sum_{t=1}^{T} m_t^{-1} \|\mathbf{X}^t \mathbf{w}^t - \mathbf{y}^t\|_F^2,$$

and the corresponding gradient, which is represented by the mode-1 unfolding of $\nabla F(\mathcal{W})$, can be written as

$$\nabla F(\mathcal{W})_{(1)} = \left[(\mathbf{X}^1)^\top (\mathbf{X}^1 \mathbf{w}^1 - \mathbf{y}^1)/m_1, \ldots, (\mathbf{X}^T)^\top (\mathbf{X}^T \mathbf{w}^T - \mathbf{y}^T)/m_T \right]. \tag{7}$$

We have the following results.

Theorem 2 (Linear convergence rate for MLMTL): Assume that the matrices $\mathbf{X}^t (\mathbf{X}^t)^\top$ are all positive definite, $1 \le t \le T$, whose smallest eigenvalues are uniformly lower bounded by $\lambda_{\min}$, while the largest eigenvalues are uniformly upper bounded by $\lambda_{\max}$. Let $\mathcal{S}^{(k)}$ be generated by Subroutine 1 with input $\nabla F(\mathcal{W}^{(k)})$, and (6) holds. Let $m_{\max} := \max_{1 \le t \le T} m_t$. If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-LS, HoRMP-LS or HoOMP-LS, then there holds

$$F(\mathcal{W}^{(k+1)}) \le \left(1 - \frac{\rho \lambda_{\min} m_{\max}^{-1}}{n_1 n_2 \lambda_{\max} \sum_{t=1}^{T} m_t^{-1}} \right) F(\mathcal{W}^{(k)}).$$

Proof: We denote $\mathbf{r}^{t,(k)} := \mathbf{X}^t \mathbf{w}^{t,(k)} - \mathbf{y}^t$, $1 \le t \le T$. We first consider HoOMP-LS. Similar to the previous analysis, we have

$$\begin{aligned}
F(\mathcal{W}^{(k+1)}) &= \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k+1)}\|_F^2 = \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t \mathbf{w}^{t,(k+1)} - \mathbf{y}^t\|_F^2 \\
&= \min_{\boldsymbol{\alpha}} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t (\sum_{i=0}^{k} \alpha_i \mathbf{s}^{t,(i)}) - \mathbf{y}^t\|_F^2 \\
&\le \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t (\sum_{i=0}^{k-1} \overline{\alpha_i^k} \mathbf{s}^{t,(i)} + \alpha \mathbf{s}^{t,(k)}) - \mathbf{y}^t\|_F^2 \\
&= \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t \mathbf{w}^{t,(k)} - \mathbf{y}^t + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2 = \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2.
\end{aligned}$$

We then consider HoRMP-LS as follows

$$\begin{aligned}
F(\mathcal{W}^{(k+1)}) &= \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k+1)}\|_F^2 = \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t \mathbf{w}^{t,(k+1)} - \mathbf{y}^t\|_F^2 \\
&= \min_{(\alpha_1, \alpha_2)} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t (\alpha_1 \mathbf{w}^{t,(k)} + \alpha_2 \mathbf{s}^{t,(k)}) - \mathbf{y}^t\|_F^2 \\
&\le \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{X}^t \mathbf{w}^{t,(k)} - \mathbf{y}^t + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2 = \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2.
\end{aligned}$$

And it naturally holds $F(\mathcal{W}^{(k+1)}) = \min_\alpha \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2$ for HoMP-LS. We then analyze the upper bound of $\min_\alpha \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2$. From the optimality condition

$$\sum_{t=1}^{T} \frac{1}{m_t} (\mathbf{X}^t \mathbf{s}^{t,(k)})^\top (\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}) = 0$$

we get

$$\alpha = -\frac{\sum_{t=1}^{T} \frac{1}{m_t} \langle \mathbf{X}^t \mathbf{s}^{t,(k)}, \mathbf{r}^{t,(k)} \rangle}{\sum_{t=1}^{T} m_t^{-1} \|\mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2}$$

and so

$$F(\mathcal{W}^{(k+1)}) \leq \min_{\alpha} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)} + \alpha \mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2 = \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)}\|_F^2 - \frac{(\sum_{t=1}^{T} m_t^{-1} \langle \mathbf{X}^t \mathbf{s}^{t,(k)}, \mathbf{r}^{t,(k)} \rangle)^2}{2 \sum_{t=1}^{T} m_t^{-1} \|\mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2}. \tag{8}$$

Recalling the definition of $\mathbf{r}^{t,(k)}$ and $\nabla F(\mathcal{W}^{(k)})$, the numerator of the second term is exactly $\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)} \rangle^2$. Using the inequalities

$$\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)} \rangle \geq \rho \|\nabla F(\mathcal{W}^{(k)})_{(1,2)}\|_2^2 \geq \frac{\rho}{n_1 n_2} \|\nabla F(\mathcal{W}^{(k)})\|_F^2,$$

and from the assumption that $\|\mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2 \leq \lambda_{\max} \|\mathbf{s}^{t,(k)}\|_F^2$ and $\|(\mathbf{X}^t)^\top \mathbf{r}^{t,(k)}\|_F^2 \geq \lambda_{\min} \|\mathbf{r}^{t,(k)}\|_F^2$, the second term of (8) can be lower bounded as follows

$$\begin{aligned}
\frac{(\sum_{t=1}^{T} m_t^{-1} \langle \mathbf{X}^t \mathbf{s}^{t,(k)}, \mathbf{r}^{t,(k)} \rangle)^2}{2 \sum_{t=1}^{T} m_t^{-1} \|\mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2} &\geq \frac{\rho}{2n_1 n_2} \frac{\|\nabla F(\mathcal{W}^{(k)})\|_F^2}{\sum_{t=1}^{T} m_t^{-1} \|\mathbf{X}^t \mathbf{s}^{t,(k)}\|_F^2} \\
&\geq \frac{\rho \lambda_{\min}}{2n_1 n_2 \lambda_{\max}} \frac{\sum_{t=1}^{T} m_t^{-2} \|\mathbf{r}^{t,(k)}\|_F^2}{\sum_{t=1}^{T} m_t^{-1} \|\mathbf{s}^{t,(k)}\|_F^2} \\
&\geq \frac{\rho \lambda_{\min} m_{\max}^{-1}}{n_1 n_2 \lambda_{\max} \sum_{t=1}^{T} m_t^{-1}} \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)}\|_F^2.
\end{aligned}$$

This together with (8) yields

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho \lambda_{\min} m_{\max}^{-1}}{n_1 n_2 \lambda_{\max} \sum_{t=1}^{T} m_t^{-1}}\right) \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{r}^{t,(k)}\|_F^2 = \left(1 - \frac{\rho \lambda_{\min} m_{\max}^{-1}}{n_1 n_2 \lambda_{\max} \sum_{t=1}^{T} m_t^{-1}}\right) F(\mathcal{W}^{(k)}),$$

as desired. ∎

In real world applications, however, the assumption that every matrix $\mathbf{X}^t(\mathbf{X}^t)^\top$ is positive definite may not hold, e.g., when the sample size m_t is larger than the size of the feature space D. To establish the linear convergence in this setting, we consider the following two cases:

1) $(\mathbf{X}^t)^\top \mathbf{X}^t$ is positive semidefinite; we add the L_2 regularization to the cost function, i.e., now the new cost function is

$$\hat{F}(\mathcal{W}) = F(\mathcal{W}) + \lambda \sum_{t=1}^{T} \frac{1}{2m_t} \|\mathbf{w}^t\|_F^2,$$

where $\lambda > 0$ is a regularization parameter. In this case, we rewrite $\hat{F}$ as

$$\begin{aligned}
\hat{F}(\mathcal{W}) &= \hat{G}(\mathcal{W}) + C \\
&= \sum_{t=1}^{T} \frac{1}{2m_t} \|\hat{\mathbf{X}}^t \mathbf{w}^t - \mathbf{z}^t\|_F^2 + C,
\end{aligned}$$

where $\hat{\mathbf{X}}^t \in \mathbb{R}^{D \times D}$ is the square root of $(\mathbf{X}^t)^\top \mathbf{X}^t + \lambda I$ and is a symmetric matrix, $\mathbf{z}^t = (\hat{\mathbf{X}}^t)^{-1} (\mathbf{X}^t)^\top \mathbf{y}^t$, and $C := \sum_{t=1}^{T} (2m_t)^{-1} (\|\mathbf{y}^t\|_F^2 - \|\mathbf{z}^t\|_F^2)$ denotes the constant term. Now $\hat{\mathbf{X}}^t (\hat{\mathbf{X}}^t)^\top$ is positive definite, which meets the assumption of Theorem 2.

2) $(\mathbf{X}^t)^\top \mathbf{X}^t$ is positive definite; we simply set $\lambda = 0$ above, and also obtain that $\hat{\mathbf{X}}^t (\hat{\mathbf{X}}^t)^\top$ is positive definite.

Therefore, we have linear convergence rate in the sense of $\hat{G}(\cdot)$.

Corollary 1 (Linear convergence for MLMTL without the positive definiteness assumption): Assume that $\mathcal{W} \in \mathbb{T}$. Let $\hat{F}(\cdot)$, $\hat{G}(\cdot)$ and $\hat{\mathbf{X}}^t$ be defined as above. Assume that the smallest and the largest eigenvalues of matrices $(\hat{\mathbf{X}}^t)^2$ are respectively lower bounded by $\lambda_{\min}$ and upper bounded by $\lambda_{\max}$. Let $0 < \rho \leq 1$ be defined as in (6), and let $m_{\max} := \max_{1 \leq t \leq T} m_t$. If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-LS, HoRMP-LS or HoOMP-LS with the cost function given by $\hat{F}(\cdot)$, then there holds

$$\hat{G}(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho \lambda_{\min} m_{\max}^{-1}}{n_1 n_2 \lambda_{\max} \sum_{t=1}^{T} m_t^{-1}}\right) \hat{G}(\mathcal{W}^{(k)}).$$

B. A class of loss functions

In this subsection, we conduct the convergence rate analysis for a class of loss functions which may be nonconvex. To this end, we first present some assumptions that characterize such class of functions $\ell(t)$:

Assumption 1:

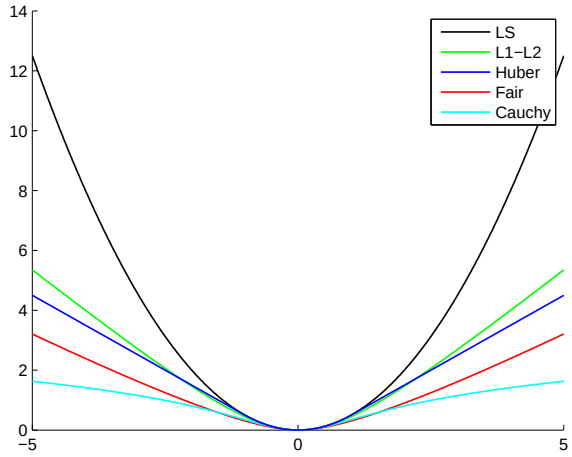
1) $\ell(t) \geq 0$, $\ell(0) = 0$, $\ell(t) = \ell(-t)$, $\ell(t) \leq t^2/2$, $\forall t \in \mathbb{R}$, and $\ell(t)$ is coercive;
2) $\ell(t)$ is continuously differentiable;

3) $|\ell^{'}(s)-\ell^{'}(t)|\leq|s-t|$, $\forall s,t\in\mathbb{R}$;
4) denote $\psi(t):=\ell^{'}(t)/t$; then $0\leq\psi(t)\leq 1$, $\psi(-t)=\psi(t)$, and $\psi(0)$ exists and is finite;
5) $\psi(t)\not\to 0$ if $t\not\to\infty$.

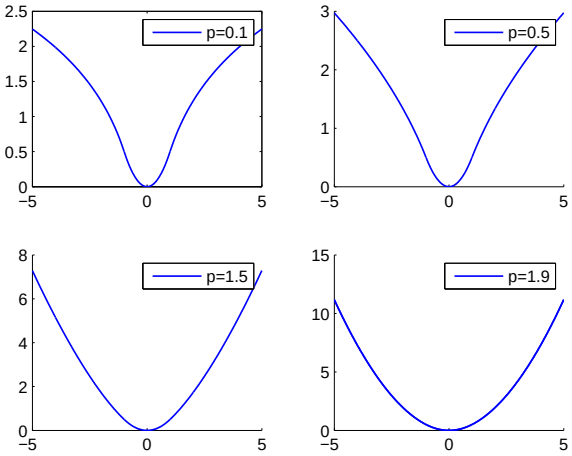
The above assumptions are not restrictive, as they are met for a variety of loss functions, e.g., the Huber's loss [41], the L_1-L_2 loss $2(\sqrt{1+t^2/2}-1)$, the Fair loss $\sigma^2(|t|/\sigma-\log(1+|t|/\sigma))$, and the Cauchy loss $\sigma^2/2\log(1+t^2/\sigma^2)$. Here σ is a parameter. Note that the Cauchy loss is nonconvex. We can also define the generalized Huber's functions that satisfies Assumption 1:

$$\ell(t):=\begin{cases} t^2/2 & |t|\leq\delta \\ \delta^{2-p}(|t|^p/p+\delta^p/2-\delta^p/p) & |t|>\delta, \end{cases}$$

where $\delta>0$ is a parameter and $0<p\leq 2$. When $p=1$ it reduces to the Huber's loss; when $p=2$ it is exactly the least squares loss. If $p<1$ then this class of functions is also nonconvex. Fig. 1 plots the losses mentioned above.



(a) Different loss functions.



(b) Generalized Huber's loss functions with different p values, with $\delta=1$.

Fig. 1: Different loss functions.

We still begin our analysis from the tensor completion setting.

1) Tensor completion with a general loss: With the loss function $\ell(t)$ satisfying Assumption 1, the cost function of tensor completion is given by

$$F(\mathcal{W})=\sum_{(i_1,\ldots,i_N)\in\Omega}\ell(\mathcal{W}_{i_1\cdots i_N}-\mathcal{B}_{i_1\cdots i_N}). \tag{9}$$

That is to say, we penalize the tensors entry by entry with $\ell(\cdot)$. Accordingly, the gradient of $F(\cdot)$ at $\mathcal{W}$ can be derived as

$$\nabla F(\mathcal{W})=\Psi_\Omega\circ(\mathcal{W}-\mathcal{B})_\Omega, \tag{10}$$

where $\circ$ denotes the Hadamard operator, i.e., entry-wise product, and Ψ is defined as $\Psi_{i_1\cdots i_N} = \psi(\mathcal{R}_{i_1\cdots i_N})$ where we denote $\mathcal{R} := \mathcal{W} - \mathcal{B}$, and ψ is defined in Assumption 1. In robust statistics [41], with an appropriate loss such as those mentioned below Assumption 1, each entry of Ψ can be seen as a weight assigned to the corresponding entry of $\mathcal{R}$. As the entry of $\mathcal{R}$ goes large, the corresponding entry of Ψ will decrease to control the influence of the large deviations.

If the cost function of tensor completion is given by (9), then one can use Assumption 1–3) to verify that

$$\|\nabla F(\mathcal{U}) - \nabla F(\mathcal{V})\|_F \leq \|\mathcal{U} - \mathcal{V}\|_F, \tag{11}$$

i.e., the gradient of $F(\cdot)$ is Lipschitz continuous with constant being 1. Therefore, we can apply strategy 4) of Algorithm 1, HoMP-G to solve it. In the following, we present our results and analysis for tensor completion solved by HoMP-G.

Theorem 3 (Linear convergence for tensor completion with a general loss): Assume that the cost function is given by (9) with a loss function $\ell(\cdot)$ satisfying Assumption 1. Let ρ be defined as in (6). If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-G, then there holds

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho q^2}{n_1 n_2}\right) F(\mathcal{W}^{(k)}),$$

where $0 < q \leq 1$ is some constant.

Proof: Let $\mathcal{W}^{(k+1)} = \mathcal{W}^{(k)} + \overline{\alpha}\mathcal{S}^{(k)}$ where $\overline{\alpha} = -\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle$. According to (11), we have

$$\begin{aligned} F(\mathcal{W}^{(k+1)}) &\leq F(\mathcal{W}^{(k)}) + \langle \nabla F(\mathcal{W}^{(k)}), \mathcal{W}^{(k+1)} - \mathcal{W}^{(k)}\rangle + 2^{-1}\|\mathcal{W}^{(k+1)} - \mathcal{W}^{(k)}\|_F^2 \\ &= F(\mathcal{W}^{(k)}) - 2^{-1}\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle^2 \\ &\leq F(\mathcal{W}^{(k)}) - \frac{\rho}{2n_1 n_2}\|\nabla F(\mathcal{W}^{(k)})\|_F^2. \end{aligned}$$

Therefore, $\{F(\mathcal{W}^{(k)})\}$ is nonincreasing. From the definition of $F(\cdot)$ and Assumption 1–1), it follows that $\{\mathcal{W}_\Omega^{(k)}\}$ is uniformly bounded. We consider the term $\|\nabla F(\mathcal{W}^{(k)})\|_F^2$. From (10), it follows

$$\|\nabla F(\mathcal{W}^{(k)})\|_F^2 = \sum_{(i_1,\ldots,i_N)\in\Omega} \psi(\mathcal{R}^{(k)}_{i_1,\ldots,i_N})^2 (\mathcal{R}^{(k)}_{i_1,\ldots,i_N})^2,$$

where $\mathcal{R}^{(k)} = \mathcal{W}_\Omega^{(k)} - \mathcal{B}_\Omega$. From Assumption 1–5), the boundedness of $\{\mathcal{W}_\Omega^{(k)}\}$ and $\mathcal{B}_\Omega$ implies that $\psi(\mathcal{R}^{(k)}_{i_1,\ldots,i_N})$ is uniformly lower bounded away from zero for all $k \geq 0$ and for all $(i_1,\ldots,i_N) \in \Omega$. Without loss of generality we assume that $\psi(\mathcal{R}^{(k)}_{i_1,\ldots,i_N}) \geq q > 0$, where the magnitude of q only depends on the magnitude of $\mathcal{B}_\Omega$. On the other hand, Assumption 1–1) tells us that $(\mathcal{R}^{(k)}_{i_1,\ldots,i_N})^2 \geq 2\ell(\mathcal{R}^{(k)}_{i_1,\ldots,i_N})$. As a consequence,

$$\|\nabla F(\mathcal{W}^{(k+1)})\|_F^2 \geq 2q^2 F(\mathcal{W}^{(k)}),$$

and so

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho q^2}{n_1 n_2}\right) F(\mathcal{W}^{(k)}),$$

as desired. ■

2) Mutilinear multitask learning with a general loss: In this setting, the cost function is given by

$$F(\mathcal{W}) = \sum_{t=1}^{T} m_t^{-1} \sum_{i=1}^{m_t} \ell(\langle \mathbf{x}_i^t, \mathbf{w}^t\rangle - y_i^t). \tag{12}$$

That is, the noise or outliers are penalized sample-wisely by using $\ell(\cdot)$. The gradient of $F(\cdot)$ at $\mathcal{W}$ can be derived as follows: its gradient at $\mathbf{w}^t$ is given by

$$(\mathbf{X}^t)^\top \mathbf{\Lambda}^t (\mathbf{X}^t\mathbf{w}^t - \mathbf{y}^t)/m_t,$$

where $\mathbf{X}^t$ is the same as that defined in subsection IV-A2, and $\mathbf{\Lambda}^t \in \mathbb{R}^{m_t \times m_t}$ is a diagonal matrix, whose i-th diagonal entry is $\mathbf{\Lambda}^t_{ii} = \psi(\langle \mathbf{x}_i^t, \mathbf{w}^t\rangle - y_i^t)$. Therefore, the gradient of $F(\cdot)$ at $\mathcal{W}$ in terms of its mode-1 unfolding can be written as

$$\nabla F(\mathcal{W})_{(1)} = \left[(\mathbf{X}^1)^\top \mathbf{\Lambda}^1 (\mathbf{X}^1\mathbf{w}^1 - \mathbf{y}^1)/m_1, \ldots, (\mathbf{X}^T)^\top \mathbf{\Lambda}^T (\mathbf{X}^T\mathbf{w}^T - \mathbf{y}^T)/m_T\right]. \tag{13}$$

The difference between (13) and its least squares counterpart (7) is the matrix $\mathbf{\Lambda}^t$, which acts as a weight matrix to remove the large deviation between $\mathbf{X}^t\mathbf{w}^t$ and $\mathbf{y}^t$ if necessary. Using Assumption 1–3), one can verify that its gradient is Lipschitz continuous,

$$\|\nabla F(\mathcal{U}) - \nabla F(\mathcal{V})\|_F \leq \left(\sum_{t=1}^{T} \|\mathbf{X}^t\|_2^4 \|\mathbf{u}^t - \mathbf{v}^t\|_F^2\right)^{1/2} \leq \lambda_{\max}\|\mathcal{U} - \mathcal{V}\|_F, \tag{14}$$

where we use $\lambda_{\max} = \max_{1\leq t\leq T} \|\mathbf{X}^t\|_2^2$ as a Lipschitz constant. Thus HoMP-G can also be applied. We present our convergence rate results in the following.

Theorem 4 (Linear convergence for multilinear multitask learning with a general loss): Assume that the cost function is given by (12) with a loss function $\ell(\cdot)$ satisfying Assumption 1. Assume that the matrices $\mathbf{X}^t(\mathbf{X}^t)^\top$ are all positive definite, $1 \leq t \leq T$, whose smallest eigenvalues are uniformly lower bounded by $\lambda_{\min}$, while the largest eigenvalues are uniformly upper bounded by $\lambda_{\max}$. Let ρ be defined as in (6), and let $m_{\max} := \max_{1\leq t\leq T} m_t$. If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-G, then there holds

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho\lambda_{\min}m_{\max}^{-1}q^2}{n_1n_2\lambda_{\max}}\right) F(\mathcal{W}^{(k)}),$$

where $0 < q \leq 1$ is a some constant.

Proof: We denote $\mathbf{r}^{t,(k)} := \mathbf{X}^t\mathbf{w}^{t,(k)} - \mathbf{y}^t$, $1 \leq t \leq T$. According to the strategy MP-non-LS in Algorithm 1, we let $\mathcal{W}^{(k+1)} = \mathcal{W}^{(k)} + \overline{\alpha}\mathcal{S}^{(k)}$, where

$$\overline{\alpha} = -\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle/\lambda_{\max}.$$

Using (14) and similar to the proof of Theorem 3, we have

$$\begin{aligned} F(\mathcal{W}^{(k+1)}) &\leq F(\mathcal{W}^{(k)}) + \langle \nabla F(\mathcal{W}^{(k)}), \mathcal{W}^{(k+1)} - \mathcal{W}^{(k)}\rangle + \frac{\lambda_{\max}}{2}\|\mathcal{W}^{(k+1)} - \mathcal{W}^{(k)}\|_F^2 \\ &= F(\mathcal{W}^{(k)}) - \frac{1}{2\lambda_{\max}}\langle \nabla F(\mathcal{W}^{(k)}), \mathcal{S}^{(k)}\rangle^2 \\ &\leq F(\mathcal{W}^{(k)}) - \frac{\rho}{2n_1n_2\lambda_{\max}}\|\nabla F(\mathcal{W}^{(k)})\|_F^2. \end{aligned}$$

It follows that $\{F(\mathcal{W}^{(k)}\}$ is a nonincreasing sequence. This together with the definition of $F(\cdot)$ and Assumption 1–1) implies that all the sequence $\{\mathbf{r}^{t,(k)}\}$ are uniformly bounded, $1 \leq t \leq T$. Recalling Assumption 1–5) and recalling $\mathbf{\Lambda}^t_{ii} = \psi(\langle \mathbf{x}^t_i, \mathbf{w}^t\rangle - y^t_i)$, the boundedness of $\{\mathbf{r}^{t,(k)}\}$ also implies that there is a universal constant $q > 0$ such that $\mathbf{\Lambda}^t_{ii} \geq q$ for $1 \leq i \leq m_t$, $1 \leq t \leq T$, and $k \geq 0$, where the magnitude of q only depends on the magnitude of the samples $\{\mathbf{x}^t_i, y^t_i\}$. Now we can consider the term $\|\nabla F(\mathcal{W}^{(k)})\|_F^2$. From (13), it follows

$$\begin{aligned} \|\nabla F(\mathcal{W}^{(k)})\|_F^2 &= \sum_{t=1}^{T} m_t^{-2}\|(\mathbf{X}^t)^\top\mathbf{\Lambda}^t(\mathbf{X}^t\mathbf{w}^{t,(k)} - \mathbf{y}^t)\|_F^2 \\ &\geq \lambda_{\min}m_{\max}^{-1}\sum_{t=1}^{T} m_t^{-1}\|\mathbf{\Lambda}^t(\mathbf{X}^t\mathbf{w}^{t,(k)} - \mathbf{y}^t)\|_F^2 \\ &\geq \lambda_{\min}m_{\max}^{-1}q^2\sum_{t=1}^{T} m_t^{-1}\|\mathbf{r}^{t,(k)}\|_F^2 \\ &\geq \lambda_{\min}m_{\max}^{-1}q^2\sum_{t=1}^{T} m_t^{-1}\sum_{i=1}^{m_t}\ell(\langle \mathbf{x}^t_i, \mathbf{w}^t\rangle - y^t_i) \\ &= 2\lambda_{\min}m_{\max}^{-1}q^2F(\mathcal{W}^{(k)}), \end{aligned}$$

where the first inequality is due to $\|\mathbf{X}^t\|_2^2 \geq \lambda_{\min}$, and the last inequality follows from Assumption 1–1), and so

$$F(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho\lambda_{\min}m_{\max}^{-1}q^2}{n_1n_2\lambda_{\max}}\right) F(\mathcal{W}^{(k)}).$$

The proof is completed. ■

To study the case that the matrices $\mathbf{X}^t(\mathbf{X}^t)^\top$ may not be all positive definite, we also make some modifications to the cost function as that discussed in subsection IV-A2. We let $(\hat{\mathbf{X}}^t)^2 := (\mathbf{X}^t)^\top\mathbf{X}^t + \lambda I$, where $\lambda > 0$ if $(\mathbf{X}^t)^\top\mathbf{X}^t$ is positive semidefinite, and $\lambda = 0$ if $(\mathbf{X}^t)^\top\mathbf{X}^t$ is positive definite. We denote $\mathbf{z}^t = (\hat{\mathbf{X}}^t)^{-1}(\mathbf{X}^t)^\top\mathbf{y}^t$, and reconstruct the cost function as

$$\hat{G}(\mathcal{W}) = \sum_{t=1}^{T} m_t^{-1}\sum_{i=1}^{m_t}\ell(\langle \hat{\mathbf{x}}^t_i, \mathbf{w}^t\rangle - z^t_i), \tag{15}$$

where $\hat{\mathbf{x}}^t_i \in \mathbb{R}^D$ is the i-th column of $\hat{\mathbf{X}}^t$. Similar to Corollary 1, we have

Corollary 2 (Linear convergence for multilinear multitask learning with a general loss and without the positive definiteness assumption): Assume that the cost function is given by (15) with a loss function $\ell(\cdot)$ satisfying Assumption 1. Let $\hat{G}(\cdot)$ and $\hat{\mathbf{X}}^t$ be defined as above. Assume that the smallest and the largest eigenvalues of matrices $(\hat{\mathbf{X}}^t)^2$ are respectively lower bounded by $\lambda_{\min}$ and upper bounded by $\lambda_{\max}$. Let ρ be defined as in (6), and let $m_{\max} := \max_{1\leq t\leq T} m_t$. If $\{\mathcal{W}^{(k)}\}$ is generated by HoMP-G, then there holds

$$\hat{G}(\mathcal{W}^{(k+1)}) \leq \left(1 - \frac{\rho\lambda_{\min}m_{\max}^{-1}q^2}{n_1n_2\lambda_{\max}}\right) \hat{G}(\mathcal{W}^{(k)}).$$

Remark 1: Before ending this section, we remark that 1) inequality (3) is very important in deriving the linear convergence. Without it, only sublinear convergence rate can be obtained; 2) all the convergence rates obtained in this section can be improved in theory by using, e.g., the methods in [38], [39] to improve the ratio ρ in (6), while getting less efficiency.

V. Numerical Experiments

In this section, we present some numerical experiments on synthetic data as well as real data, focusing on the applications: (robust) tensor completion and MLMTL. All the numerical computations are conducted on an Intel i7-3770 CPU desktop computer with 16 GB of RAM. The supporting software is MATLAB R2013a.

A. Tensor completion

Our HoMP-type methods with least squares loss are compared with 4 state-of-the-art methods: generalized conditional gradient (GCG) [15], HaLRTC [1], factor priors (FP) [14] and TMac [42]. GCG also solves an approximate tensor singular value problem (3) at each iteration; HaLRTC solves a convex relaxation of tensor completion with sum of matrix nuclear norms; FP uses some prior knowledges of the tensors, and TMac is based on tensor factorization. The stopping criterion for HoMPS is when the residual $\mathcal{R}^{(k)}$ used in Sect. IV is less than a threshold. For all the methods except HaLRTC, the threshold of the stopping criterion is $\epsilon = 10^{-5}$; for HaLRTC, we set 10^{-6} because otherwise it cannot generate a good result. The max iteration for all the methods is 500, whereas for HoMPs, the max iteration K is the only parameter, which is tuned by 10-fold cross validation over $K \in \{100, 200, 300, 400, 500\}$. All the results are averaged over ten instances.

1) Synthetic data: Third order tensors of size $200 \times 200 \times 200$ are randomly generated, with CP-rank 10. Some entries are randomly missing, with missing ratio (MR) varies in $\{0.5, 0.6, 0.7, 0.8, 0.9, 0.95, 0.99\}$. The relative error $\|\mathcal{X}^* - \mathcal{B}\|_F / \|\mathcal{B}\|_F$ and the computational time (in Second) are respectively reported in Fig. 2 and Table I. The results in Fig. 2 show that HoMPs and GCG have better performances, particularly when the MR value is very high. The results in the subfigure of Fig. 2 show that HoMPs perform better than GCG when the MR values in $[0.5, 0.9]$. Table I shows that HoMPs are more efficient than other methods, particularly when the MR value is very high. That is because we have optimized our codes by utilizing the sparsity of the data and using the sparsity manipulation in Matlab. Comparing between the three HoMPs, it is interesting to see that the relative error of HoOMP is not as good as the other two. That may be because HoOMP learns the tensor more greedily and leads to overfitting. And HoOMP is the slowest among the three methods, as it has to solve a larger linear equation system to obtain the weights.

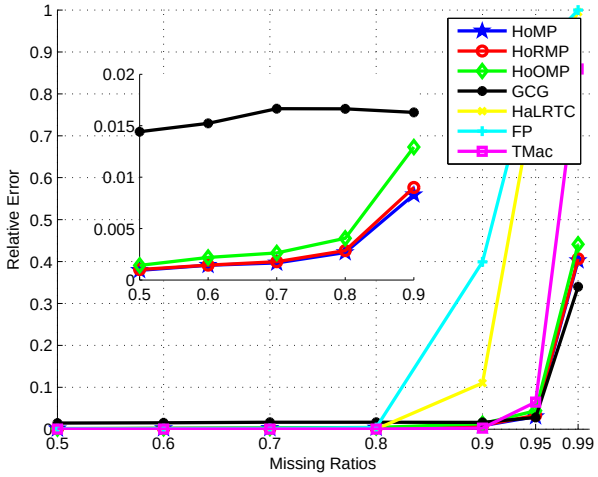


Fig. 2: Tensor completion results on synthetic data ($200 \times 200 \times 200$, CP-rank $= 10$) in terms of relative error.

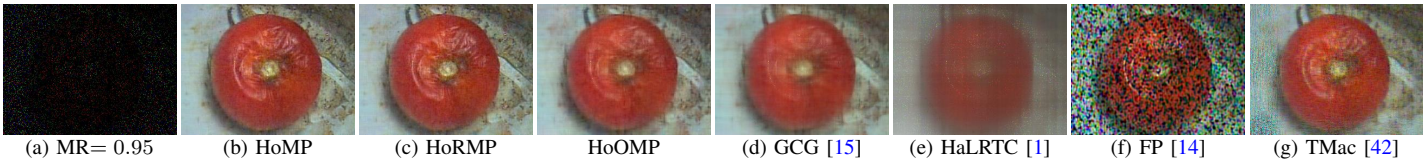
TABLE I: Efficiency comparison of different methods on tensor completion on synthetic data ($200 \times 200 \times 200$, CP-rank $= 10$).

MR (%)	HoMP	HoRMP	HoOMP	GCG [15]	HaLRTC [1]	FP [14]	TMac [42]
50	16.09	20.95	45.02	126.14	42.89	88.18	23.00
60	13.85	17.77	37.38	106.60	54.38	97.64	26.56
70	11.41	14.27	28.29	85.72	72.21	121.94	32.75
80	8.63	10.66	19.84	64.13	98.76	149.37	42.78
90	5.25	6.28	10.79	38.89	97.03	100.28	49.41
95	3.45	3.94	6.05	24.69	96.72	71.72	46.44
99	1.98	2.05	2.35	14.41	1.98	54.65	46.06

TABLE II: Comparisons of different methods on tensor completion real datasets

Datasets	MR (%)	HoMP		HoRMP		HoOMP		GCG [15]		HaLRTC [1]		FP [14]		TMac [42]	
		Relerr	Time (S.)	Relerr	Time (S.)	Relerr	Time (S.)	Relerr	Time (S.)	Relerr	Time (S.)	Relerr	Time (S.)	Relerr	Time (S.)
Lena (512×512×3)	70	8.10E-02	1.17	8.16E-02	1.50	8.05E-02	2.64	7.85E-02	6.47	6.93E-02	32.09	5.47E-02	195.12	2.45E-01	61.62
	80	1.06E-01	0.92	1.07E-01	1.08	1.05E-01	1.78	9.70E-02	4.67	9.59E-02	44.14	8.31E-02	199.84	4.01E-01	57.88
	90	1.56E-01	0.31	1.56E-01	0.36	1.58E-01	0.42	1.42E-01	2.28	1.71E-01	68.94	1.34E-01	64.42	7.15E-01	45.83
	95	2.17E-01	0.24	2.19E-01	0.27	2.44E-01	0.32	2.02E-01	1.66	3.77E-01	75.53	5.92E-01	32.04	9.09E-01	40.40
	99	4.10E-01	0.21	4.38E-01	0.20	4.90E-01	0.22	4.00E-01	1.05	8.61E-01	72.94	9.75E-01	14.03	1.01E+00	18.11
Spectral (205×246×96)	70	1.17E-02	43.37	1.18E-02	51.36	1.17E-02	259.93	3.41E-02	47.62	3.00E-02	35.33	3.67E-02	442.16	1.81E-02	244.31
	80	1.45E-02	34.76	1.45E-02	40.13	1.44E-02	175.54	3.68E-02	35.24	4.24E-02	35.19	6.65E-02	444.89	2.65E-02	227.77
	90	2.24E-02	24.40	2.23E-02	26.87	2.20E-02	90.14	3.90E-02	21.97	6.46E-02	56.31	6.74E-02	210.08	6.31E-02	208.89
	95	3.74E-02	7.34	3.80E-02	7.54	3.68E-02	11.83	4.38E-02	14.78	1.05E-01	132.35	1.74E-01	91.23	1.71E-01	198.56
	99	8.74E-02	2.25	8.97E-02	2.35	9.40E-02	2.47	8.28E-02	9.59	8.50E-01	129.95	9.56E-01	33.66	6.94E-01	183.16
MRI (181×217×181)	70	1.55E-02	42.00	1.54E-02	51.41	1.53E-02	331.32	5.98E-02	65.01	3.22E-05	133.70	1.30E-03	86.05	5.81E-02	286.55
	80	2.11E-02	33.20	2.14E-02	39.97	2.10E-02	177.95	6.54E-02	47.25	2.89E-03	148.18	2.04E-03	116.64	9.68E-02	322.64
	90	3.96E-02	21.44	3.97E-02	24.47	3.83E-02	89.69	7.32E-02	27.37	9.03E-02	158.40	4.20E-03	150.70	2.26E-01	309.16
	95	8.35E-02	7.13	8.30E-02	7.84	7.95E-02	15.20	8.59E-02	16.90	3.44E-01	186.62	2.69E-01	120.94	4.47E-01	308.42
	99	3.18E-01	2.03	3.33E-01	2.24	3.46E-01	2.51	3.30E-01	9.17	9.95E-01	2.77	9.54E-01	52.91	9.08E-01	273.60
Knix1 (512×512×3×22)	70	1.12E-01	90.95	1.14E-01	116.51	1.88E-01	57.08	1.79E-01	328.28	1.14E-01	288.78	7.35E-02	697.61	1.89E-01	530.81
	80	1.26E-01	64.18	1.28E-01	80.70	2.03E-01	41.70	1.92E-01	281.14	1.60E-01	359.90	7.85E-02	716.28	2.37E-01	904.77
	90	1.62E-01	37.69	1.62E-01	44.26	2.23E-01	23.14	2.09E-01	211.19	2.70E-01	381.45	9.57E-02	714.48	3.25E-01	1237.13
	95	2.21E-01	9.27	2.20E-01	10.51	2.16E-01	38.80	2.24E-01	173.88	4.36E-01	383.07	6.05E-01	438.44	4.02E-01	1208.53
	99	4.22E-01	2.26	4.14E-01	2.02	3.94E-01	5.92	3.65E-01	134.75	9.27E-01	356.27	9.86E-01	138.46	8.26E-01	1098.88
Knix2 (512×512×3×24)	70	1.34E-01	94.85	1.35E-01	121.01	1.93E-01	63.89	1.85E-01	337.07	1.50E-01	396.27	8.05E-02	726.74	1.50E-01	562.98
	80	1.52E-01	66.01	1.54E-01	83.02	2.08E-01	41.80	2.00E-01	276.68	2.04E-01	390.28	8.47E-02	732.19	1.92E-01	918.55
	90	1.99E-01	38.34	1.99E-01	45.50	2.28E-01	23.64	2.18E-01	208.10	2.98E-01	391.50	1.19E-01	732.63	2.68E-01	1278.58
	95	2.53E-01	9.57	2.53E-01	10.74	2.42E-01	38.86	2.46E-01	167.50	3.89E-01	385.79	6.10E-01	441.75	3.86E-01	1230.64
	99	3.85E-01	2.21	3.78E-01	2.00	3.80E-01	5.76	3.52E-01	131.51	9.95E-01	7.80	9.85E-01	148.51	1.12E+00	1197.74
Tomato (242×320×3×167)	70	7.11E-02	179.43	7.20E-02	211.71	1.07E-01	45.52	9.98E-02	360.14	7.42E-02	327.47	7.56E-02	1191.23	7.75E-02	1688.45
	80	7.74E-02	133.91	7.77E-02	153.38	1.13E-01	29.72	1.07E-01	264.58	9.63E-02	436.98	7.98E-02	1203.06	8.33E-02	1597.08
	90	8.51E-02	81.57	8.54E-02	87.78	1.21E-01	16.21	1.14E-01	158.06	1.35E-01	646.06	9.51E-02	1212.66	8.90E-02	1496.16
	95	9.55E-02	48.74	9.55E-02	50.38	1.13E-01	25.52	1.17E-01	98.14	2.44E-01	634.48	4.32E-01	825.22	1.93E-01	1435.33
	99	1.40E-01	14.01	1.40E-01	12.29	1.43E-01	28.90	1.30E-01	41.69	8.81E-01	634.68	9.75E-01	293.10	7.41E-01	1389.03

2) Real data: Several real data sets have been chosen: the Knix datasets, the Tomato video, the Hyperspectral images, the brain MRI and a color image Lena[1]. Knix consists of two tensors of order-4, Tomato is also a tensor of order-4, and the remaining datases are 3rd order tensors. Since the order of magnitude of tensors in Knix and Tomato is 10^7, for these datasets we set the max iteration of FP as 50, because it is too time-consuming at each iteration. For the same datasets we also restrict the max iteration of HoOMP by 100 because at each iteration it has to solve a relatively large linear equation system. For HaLRTC on Tomato, the max iteration is set to 100 due to efficiency. The results are reported in Table II, where we can observe that in most cases, HoMPs have acceptable or better performances, but need less time. HoMPs perform particularly well on the Hyperspectral images: even the MR value is 99%, the relative error is less than 0.1, which maybe due to the fact that the tensor in this dataset is indeed low rank. On Tomato, which is of size $O(10^7)$, our methods take much less time than competing methods and obtain better performances. On Knix and Lenna, FP outperforms other methods when the MR value is not high, which maybe because it uses some prior knowledges, however, mining the knowledges might be time-consuming, making it slower than other methods. When recovering the color image, HoMPs have a 30x speedup comparing with HaLRTC, and 100x speedup comparing with FP, which may be useful in real world applications. In Fig. 3, results recovered by different methods from one slide of the Tomato dataset with 0.95 missing ratio are illustrated to intuitively evaluate the performances. From the results we can observe that HoMP and HoRMP recover more details than other methods.



(a) MR= 0.95 (b) HoMP (c) HoRMP HoOMP (d) GCG [15] (e) HaLRTC [1] (f) FP [14] (g) TMac [42]

Fig. 3: This example intuitively shows slides recovered by different methods from the 37-th slide of the Tomato dataset with MR= 0.95

B. Multilinear multitask learning

Our HoMP-type methods with least squares loss are compared with 4 state-of-the-art methods: sum of (overlapped) nuclear norm (Overlapped for short) [5], [6] , latent nuclear norm (Latent) [6] , scaled latent nuclear norm (Scaled) [6], and a nonconvex approach (Nonconvex) [5]. The first three methods are based on nuclear norm regularized convex optimization, while the last method factorizes the tensor into factors, which is nonconvex. We also note here that the difference between the Overlapped method in [5] and [6] is that [5] uses the sum of nuclear norm as a regularization, while [6] treats it as a constraint. In the following we consider them as the same method. The stopping criterion is the same as the previous experiment. Parameters are tuned via 10-fold cross validation. Specifically, $K \in \{15, 20, 25, \ldots, 50\}$. The following datasets are chosen:

1) School dataset. This datasets is made available by the Inner London Education Authority (ILEA), which consists of examination records from 139 schools in years 1985, 1986 and 1987, with 15362 students. Each task is to predict exam scores for students in each school, where the size of the input space is 25, with one indicating the bias term. Following [6], we model it as a MLMTL problem, where each task has two indices: the school index and the year index. Therefore, the tasks jointly

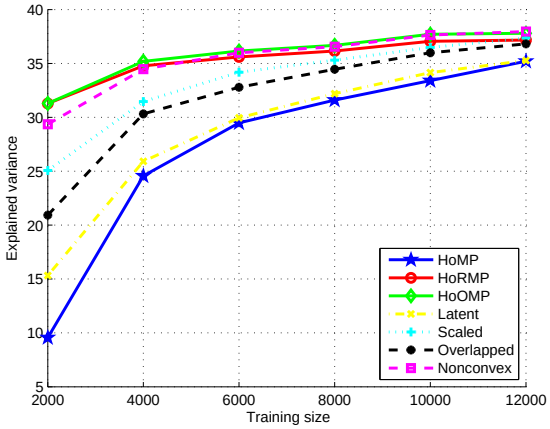
[1]Knix can be downloaded from http://www.osirix-viewer.com/datasets/, and Tomato, Hyperspectral images and brain MRI are available at https://code.google.com/p/tensor-related-code/source/browse/trunk/Model/Tensor+Completion/LRTC_Package_Ji/?r=6.

gives a $25 \times 3 \times 139$ weight tensor to be learned. The size of the training set varies from 2000 to 12000. Similar to [6], we use the explained variance $100 \cdot (1 - \mathrm{MSE}_{\mathrm{test}}/\mathrm{var}(\mathbf{y}))$ to evaluate the performance of the compared methods.

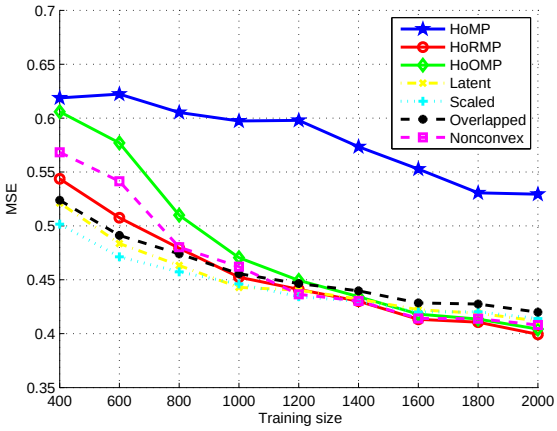
2) Restaurant & consumer dataset [43][2]. This dataset consists of rating scores from 138 consumers to different restaurant from 3 aspects, with 3483 instances. Each task is to predict the rating given by a consumer from one aspect, provided a restaurant as an input. The size of the input space is 45, with one indicating the bias term. Since each task can be indexed by two indices: the consumer index and the aspect index, the tasks jointly yields a $45 \times 3 \times 138$ weight tensor to be learned. The size of the training set varies from 400 to 2000. The test MSE is used to evaluate the performance of the compared methods.

The results averaged over 20 instances on the school and the restaurant datasets are respectively reported in Fig. 4a and 4b. From Fig. 4a, we can observe that HoRMP and HoOMP perform comparable with or better than other methods on the school dataset, while HoMP does not perform well. Scaled [6] performs best among Latent [6], Scaled [6] and Overlapped [5], [6], which is in accordance with the observations in [6]. The method Nonconvex [5] is slightly worse than HoRMP and HoOMP when the size of the sample is small, and then it catches up our methods as the samples increase. For the restaurant dataset, HoMP is still worse than other methods, see Fig. 4b; HoRMP and HoOMP are not as good as other methods when the sample size is small, which may be due to the overfitting in these cases. However, they eventually catch up other methods when the samples increase, and are slightly better when the sample size is larger than 1600. To give a clearer view on their performances, we also use the bar plot to show the MSE of the compared methods in Fig. 4c. The efficiency comparisons are reported in Table III, from which one can again see the efficiency of HoMP-type methods.

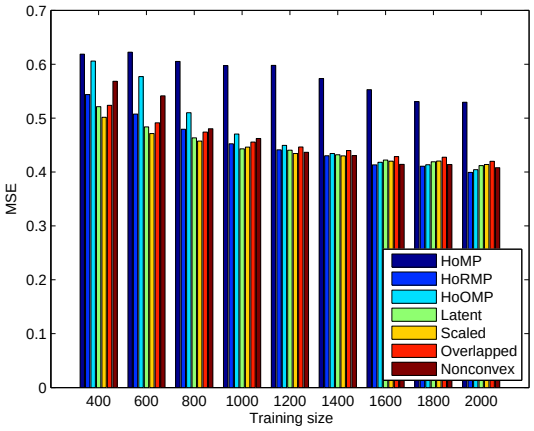
During the experiments, we also observe that for our methods, around $K = 25$ iterations can give desirable results, which implies that the weight tensor $\mathcal{W}$ can indeed be approximated by a tensor of rank lower than 25.



(a) Explained variance (the larger, the better) for different methods on the school dataset.



(b) Test MSE for different methods on the restaurant dataset.



(c) Test MSE on the restaurant data using bar plot.

Fig. 4: MLMTL results on two real datasets with different methods: HoMP, HoRMP, HoOMP, Latent [6], Scaled [6], Overlapped [5], [6] and Nonconvex [5].

[2]We would like to thank the first author of [5] to provide us the cleaned dataset.

TABLE III: Efficiency comparison of different methods on school and restaurant datasets. The results are averaged over all sample sizes and all the instances.

Dataset	HoMP	HoRMP	HoOMP	Latent [6]	Scaled [6]	Overlapped [5], [6]	Nonconvex [5]
School	0.33	0.35	0.36	4.92	1.44	8.55	110.31
Restaurant	0.45	0.46	0.47	1.35	5.81	8.37	110.17

C. Robust tensor completion

The goal of this subsection is to examine the effectiveness of the HoMP-G strategy. A suitable setting is the robust tensor completion, with the loss function instantiated by the Cauchy loss $\ell_\sigma(t) = \sigma^2/2 \log(1 + t^2/\sigma^2)$ with σ fixed to 0.08. The compared methods are HoRPCA [44, Alg. 2.2] and RPCA [45]. HoRPCA and RPCA are designed to solve convex optimization problems, which employ the ℓ_1 loss to penalize the noise or outliers, where RPCA is focused on robust matrix completion, while HoRPCA is for robust tensor completion. The stopping criterion and other settings are the same as the previous experiments.

1) Synthetic data: Third order tensors of size $100 \times 100 \times 100$ are randomly generated, with CP-rank 10. Some entries are randomly missing, with missing ratio (MR) varies from 0.3 to 0.99. 10% of the entries are contaminated by outliers drawn form $[-1, 1]$. We compare with HoRPCA in this experiment, and report the results in Fig. 5. We can observe that when the MR value is less than 0.6, HoRPCA is better than our method; when the MR value increases, the performance of HoRPCA decreases rapidly, while our method is more stable. This observation is similar to that in the experiment of tensor completion.

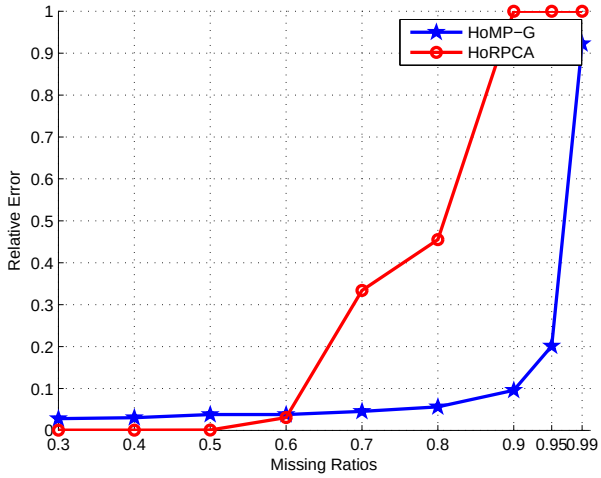


Fig. 5: Robust tensor completion results on synthetic data ($100 \times 100 \times 100$, CP-rank $= 10$) in terms of relative error. The compared method is HoRPCA [44].

2) Yale face: We compare our method and RPCA on removing shadows and specularities from face images, as that done in [45]. The shadows and specularities can be seen as noise or outliers [45]. The selected dataset consists of 64 face images of size 192×168, giving a $192 \times 168 \times 64$ tensor. Previously RPCA treats it as a 32256×64 matrix [45], while we directly treat it as a tensor. We consider two settings: there are no missing entries and there are 60% missing entries. The maximum iteration for both two methods is 200. Part of the results are shown in Fig. 6. In the figure, row (a) shows some original images; (b) presents images recovered by HoMP-G (HoMP for short in this paragraph) and (c) are those recovered by RPCA. We can observe that both of the two methods can remove the shadows, while it seems from the first column that HoMP performs better, as it can remove the lines. However, images recovered by HoMP are not as clear as those recovered by RPCA. This may be because that HoMP yields a tensor of CP-rank 200, say $\sum_{i=1}^{200} \alpha^i \mathbf{x}_i \otimes \mathbf{y}_i \otimes \mathbf{z}_i$, to approximate the original data tensor, which is equal to that for the j-th image, $1 \le j \le 64$, it is approximated by a matrix of rank at most 200, $\sum_{i=1}^{200} \alpha^i z_{i,j} \mathbf{x}_i \otimes \mathbf{y}_i$. Since these $\mathbf{x}_i$ and $\mathbf{y}_i$ are not orthogonal, they may not be principle components, and hence may lead to loss of information. Rows (d)–(f) show the images with 60% entries missing and those recovered by HoMP and RPCA, respectively. In this case HoMP still performs stable, while RPCA cannot successfully impute all the missing entries. Finally, HoMP only requires $(192+168+64) \cdot 200 = 169600$ size to store the recovered tensor, while RPCA has to use 2064384 size to store the recovered matrix. Totally speaking, our method has the following features: it can remove shadows and specularities, impute missing entires, as well as generate a set of common basis $\{\mathbf{x}_i, \mathbf{y}_i\}$ for all the images, which has lower storage requirement, and is as efficient as the previous HoMPs.

D. Linear convergence

Last, we examine the linear convergence of HoMPs on three experiments, as shown in Fig. 7. In the figures, the y-axes is the logarithm of the cost function at iteration k, $\log(F(\mathcal{W}^{(k)}))$, while x-axes stands for iteration. Specifically, Fig. 7a plots the

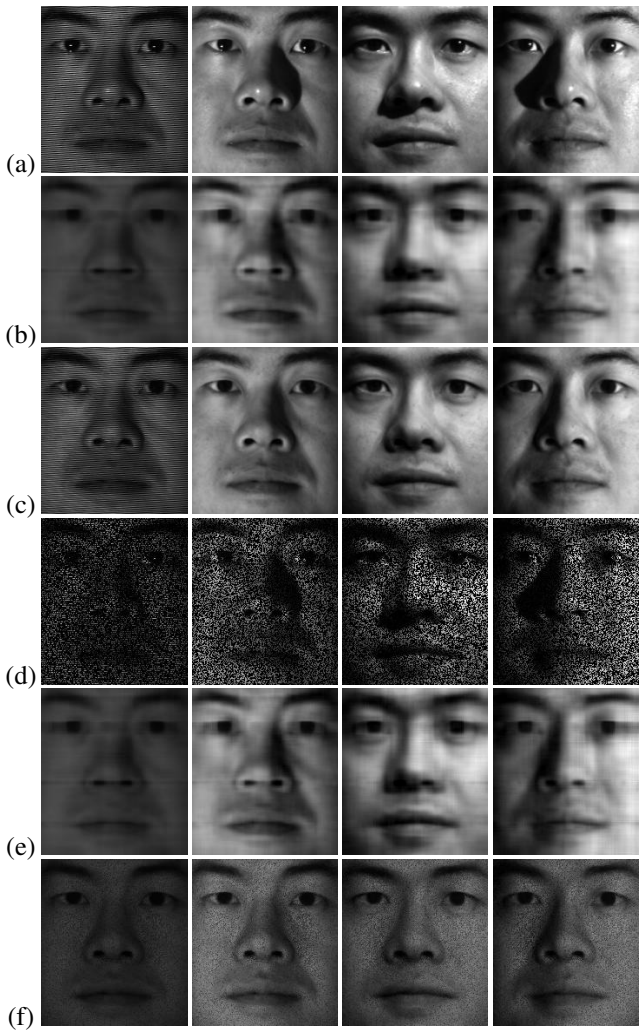


Fig. 6: Comparison of HoMP-G and RPCA on removing shadows and specularities from face images. (a): Some original images. (b) Images recovered by HoMP-G. (c) Images recovered by RPCA. (d) MR $= 60\%$. (e) Recovered by HoMP-G. (f) Recovered by RPCA.

curves of HoMP-LS (blue), HoRMP-LS (red) and HoOMP-LS (green) on tensor completion; Fig. 7b plots those on MLMTL, while Fig. 7c plots the curve of HoMP-G on robust tensor completion. From the figures, we can observe that the curves confirm the theoretical results derived in Sect. IV.

VI. Conclusion

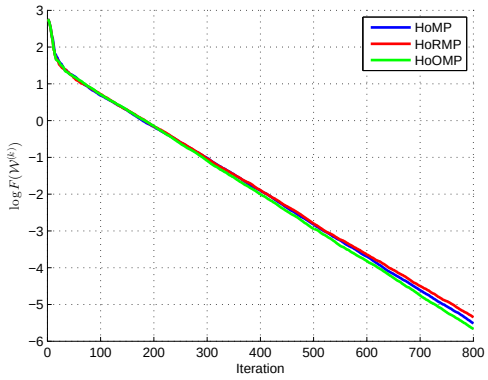
In this paper, we proposed higher order matching pursuit for low rank tensor learning. Comparing with some state-of-the-art methods, HoMPs have three important features: low computational complexity, low storage requirement, and linear convergence. Furthermore, HoMP can also be applied to problems with a nonconvex cost function, sharing the same convergence rate as those with a convex cost function. Numerical experiments on synthetic as well as real datasets verify the efficiency and effectiveness of HoMPs.

Acknowledgement

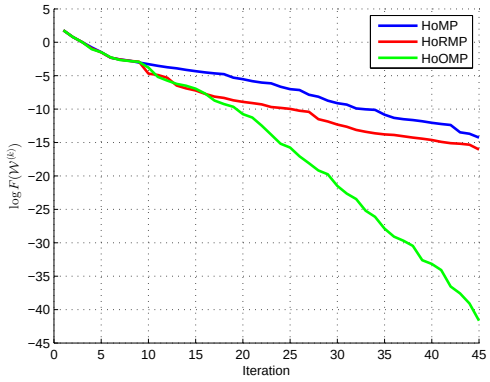
The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC AdG A-DATADRIVE-B (290923). This paper reflects only the authors' views, the Union is not liable for any use that may be made of the contained information; Research Council KUL: GOA/10/09 MaNet, CoE PFV/10/002 (OPTEC), BIL12/11T; PhD/Postdoc grants; Flemish Government: FWO: PhD/Postdoc grants, projects: G.0377.12 (Structured systems), G.088114N (Tensor based data similarity); IWT: PhD/Postdoc grants, projects: SBO POM (100031); iMinds Medical Information Technologies SBO 2014; Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017). Johan Suykens is a professor at KU Leuven, Belgium.

References

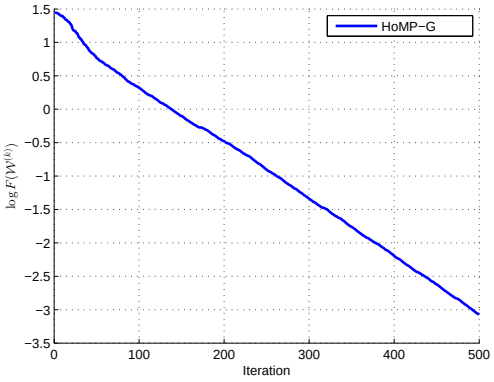
[1] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 208–220, 2013.



(a) Convergence rate on tensor completion



(b) Convergence rate on MLMTL



(c) Convergence rate on robust tensor completion

Fig. 7: Convergence rate of HoMPs on different experiments. Fig. 7a: tensor completion. Fig. 7b: MLMTL. Fig. 7c: robust tensor completion. y-axes: the logarithm of the cost function at iteration k, $\log(F(\mathcal{W}^{(k)}))$; x-axes:iteration. The plots confirms the theoretical results in Sect. IV.

[2] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Prob.*, vol. 27, no. 2, p. 025010, 2011.

[3] M. Signoretto, Q. T. Dinh, L. De Lathauwer, and J. A. K. Suykens, "Learning with tensors: a framework based on convex optimization and spectral regularization," *Mach. Learn.*, vol. 94, pp. 303–351, 2014.

[4] R. Tomioka and T. Suzuki, "Convex tensor decomposition via structured schatten norm regularization," in *Advances in Neural Information Processing Systems*, 2013, pp. 1331–1339.

[5] B. Romera-Paredes, H. Aung, N. Bianchi-Berthouze, and M. Pontil, "Multilinear multitask learning," in *Proceedings of The 30th International Conference on Machine Learning*, 2013, pp. 1444–1452.

[6] K. Wimalawarne, M. Sugiyama, and R. Tomioka, "Multitask learning meets tensor factorization: task imputation via convex optimization," in *Advances in Neural Information Processing Systems*, 2014, pp. 2825–2833.

[7] M. Signoretto, R. Langone, M. Pontil, and J. Suykens, "Graph based regularization for multilinear multitask learning," *Internal Report 14-138, ESAT-SISTA, KU Leuven (Leuven, Belgium)*, 2014.

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

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

[10] L. Yang, Z.-H. Huang, and X. Shi, "A fixed point iterative method for low n-rank tensor pursuit," *Signal Processing, IEEE Transactions on*, vol. 61, no. 11, pp. 2952–2962, 2013.

[11] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, pp. 455–500, 2009.

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

[13] X. Zhang, Z. Zhou, D. Wang, and Y. Ma, "Hybrid singular value thresholding for tensor completion," in *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[14] Y.-L. Chen, C.-T. Hsu, and H.-Y. Liao, "Simultaneous tensor decomposition and completion using factor priors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 3, pp. 577–591, 2014.

[15] Y. Yu, H. Cheng, and X. Zhang, "Approximate low-rank tensor learning," *7th NIPS Workshop on Optimization for Machine Learning*, 2014.

[16] Y. Yang, Y. Feng, and J. A. K. Suykens, "A rank-one tensor updating algorithm for tensor completion," *Internal Report 14-203, ESAT-SISTA, KU Leuven (Leuven, Belgium)*, 2014.

[17] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3397–3415, 1993.

[18] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, 1993, pp. 40–44.

[19] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *Information Theory, IEEE Transactions on*, vol. 50, no. 10, pp. 2231–2242, 2004.

[20] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, 2007.

[21] Z. Wang, M.-J. Lai, Z. Lu, and J. Ye, "Orthogonal rank-one matrix pursuit for low rank matrix completion," *arXiv preprint arXiv:1404.1377*, 2014.

[22] T. G. Kolda, "Orthogonal tensor decompositions," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 1, pp. 243–255, 2001.

[23] L.-H. Lim, "Singular values and eigenvalues of tensors: a variational approach," in *Computational Advances in Multi-Sensor Adaptive Processing, 2005 1st IEEE International Workshop on*, vol. 1, 2005, pp. 129–132.

[24] L.-H. Lim and P. Comon, "Blind multilinear identification," *Information Theory, IEEE Transactions on*, vol. 60, no. 2, pp. 1260–1280, Feb 2014.

[25] C. J. Hillar and L.-H. Lim, "Most tensor problems are NP-hard," *J. ACM*, vol. 60, no. 6, pp. 45:1–45:39, 2013.

[26] A. R. Barron, A. Cohen, W. Dahmen, and R. A. DeVore, "Approximation and learning by greedy algorithms," *The annals of statistics*, pp. 64–94, 2008.

[27] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, 2013.

[28] L. E. Figueroa and E. Süli, "Greedy approximation of high-dimensional ornstein–uhlenbeck operators," *Foundations of Computational Mathematics*, vol. 12, no. 5, pp. 573–623, 2012.

[29] A. Ammar, F. Chinesta, and A. Falcó, "On the convergence of a greedy rank-one update algorithm for a class of linear systems," *Archives of Computational Methods in Engineering*, vol. 17, no. 4, pp. 473–486, 2010.

[30] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval research logistics quarterly*, vol. 3, no. 1-2, pp. 95–110, 1956.

[31] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, pp. 427–435.

[32] L. De Lathauwer, B. De Moor, and J. Vandewalle, "On the best rank-1 and rank-$(R_1, R_2, \ldots, R_n)$ approximation of higer-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1324–1342, 2000.

[33] B. Chen, S. He, Z. Li, and S. Zhang, "Maximum block improvement and polynomial optimization," *SIAM J. Optim.*, vol. 22, pp. 87–107, 2012.

[34] X. Zhang, C. Ling, and L. Qi, "The best rank-1 approximation of a symmetric tensor and related spherical optimization problems," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 3, pp. 806–821, 2012.

[35] T. Zhang and G. H. Golub, "Rank-one approximation to high order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 23, no. 2, pp. 534–550, 2001.

[36] X. Zhang, L. Qi, and Y. Ye, "The cubic spherical optimization problems," *Math. Comput.*, vol. 81, no. 279, pp. 1513–1525, 2012.

[37] S. He, Z. Li, and S. Zhang, "Approximation algorithms for homogeneous polynomial optimization with quadratic constraints," *Math. Program.*, vol. 125, pp. 353–383, 2010.

[38] A. M.-C. So, "Deterministic approximation algorithms for sphere constrained homogeneous polynomial optimization problems," *Math. Program.*, vol. 129, no. 2, pp. 357–382, 2011.

[39] S. He, B. Jiang, Z. Li, and S. Zhang, "Probability bounds for polynomial functions in random variables," *Math. Oper. Res.*, 2014.

[40] Y. Yang, Y. Feng, X. Huang, and J. A. K. Suykens, "Rank-one tensor properties with applications to a class of tensor optimization problems," *Internal Report 13-245, ESAT-SISTA, KU Leuven (Leuven, Belgium)*, 2013.

[41] P. J. Huber, *Robust statistics*. Springer, 2011.

[42] Y. Xu, R. Hao, W. Yin, and Z. Su, "Parallel matrix factorization for low-rank tensor completion," *arXiv preprint arXiv:1312.1254*, 2013.

[43] B. Vargas-Govea, G. González-Serna, and R. Ponce-Medellın, "Effects of relevant contextual features in the performance of a restaurant recommender system," *ACM RecSys*, vol. 11, 2011.

[44] D. Goldfarb and Z. Qin, "Robust low-rank tensor recovery: Models and algorithms," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 1, pp. 225–253, 2014.

[45] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM (JACM)*, vol. 58, no. 3, p. 11, 2011.