On the extension of trace norm to tensors

Ryota Tomioka, Kohei Hayashi, and Hisashi Kashima

Abstract

In this paper, we propose three extensions of trace norm for the minimization of tensor rank via convex optimization. The alternating direction method of multipliers is used to efficiently solve the optimization problems. One of the proposed extensions recovers partially observed tensor almost perfectly from a small fraction of observations.

I. Introduction

Higher order tensor decompositions have recently been studied intensively motivated by their usefulness in various fields including chemometrics, neuroimaging, and graph analysis [1]. Tensor decomposition methods aim to separate the factors spanning each modality of a given tensor and the interactions among the factors. The smaller the number of factors or the smaller the number of interactions, the more compact and succinct the decomposition is.

Matrix completion, or matrix estimation, has recently witnessed a great advance driven by powerful theory [2], [3] from compressed sensing and tools from convex optimization [4].

In this paper, we consider the problem of tensor completion, or more generally tensor estimation, which aims to recover an unknown tensor from partial (noisy) observations under the assumption that the underlying tensor admits a compact decomposition. This setting obviously includes the case of decomposing a fully observed tensor.

The aim of this paper is to extend the trace norm, which is a key component in matrix completion via convex optimization, for the tensor completion problem.

In the next section, we first review the matrix rank and its relation to the trace norm. Then we review the definition of tensor k-rank, which suggests that a low rank tensor *is* a low rank matrix when appropriately unfolded. In Sec. III, we propose three approaches to extend the trace norm for the

Department of Mathematical Informatics, The University of Tokyo, {tomioka,kashima}@mist.i.u-tokyo.ac.jp
Graduate School of Information Science, Nara Institute of Science and Technology kohei-h@is.naist.jp

estimation of low-rank tensors. In Sec. IV, we show that the optimization problems associated to the proposed extensions can be solved by the alternating direction method of multipliers. In Sec. V, we show through numerical experiments that one of the proposed approaches can recover a partly observed low-rank tensor almost perfectly from smaller fraction of observations compared to the conventional EM-based Tucker decomposition algorithm. The proposed algorithm shows a sharp threshold behaviour from a poor fit to a nearly perfect fit; we numerically show that the fraction of samples at the threshold is roughly proportional to the sum of the k-ranks of the underlying tensor when the tensor dimension is fixed. Finally we summarize the paper in Sec. VI

II. Low rank matrix and tensor

A. Rank of a matrix and the trace norm

The rank r of an $R \times C$ matrix $\boldsymbol{X}$ is defined via the singular-value decomposition (SVD)

$$\boldsymbol{X} = \boldsymbol{U}\mathrm{diag}(\sigma_1(\boldsymbol{X}), \sigma_2(\boldsymbol{X}), \ldots, \sigma_r(\boldsymbol{X}))\boldsymbol{V}^\top,$$

where $\boldsymbol{U} \in \mathbb{R}^{R\times r}$ and $\boldsymbol{V} \in \mathbb{R}^{C\times r}$ are orthogonal matrices, and $\sigma_j(\boldsymbol{X})$ is the jth largest singular-value of $\boldsymbol{X}$. The matrix $\boldsymbol{X}$ is called *low-rank* if the rank r is less than $\min(R, C)$. Unfortunately, the rank of a matrix is a nonconvex function, and the direct minimization of the rank is an NP-hard problem.

The trace norm is known to be the tightest convex lower bound of matrix rank [3] and is defined as the linear sum of singular values as follows:

$$\|\boldsymbol{X}\|_* = \sum_{j=1}^{r} \sigma_j(\boldsymbol{X}).$$

The trace norm allows us to estimate low-rank matrices via convex optimization with a theoretical guarantee [2]. Intuitively, the trace norm plays the role of the ℓ_1-norm in the subset selection problem, for the estimation of low-rank matrices.

B. Rank of a tensor

We consider the k-rank of tensors, which is a direct generalization of the above definition of the matrix rank; see [1] for other definitions of tensor rank.

The k-rank of an Kth-order tensor $\boldsymbol{\mathcal{X}}$, denoted $\mathrm{rank}_k(\boldsymbol{\mathcal{X}})$, is defined as the rank of the mode-k unfolding $\boldsymbol{X}_{(k)}$ of $\boldsymbol{\mathcal{X}}$. The tensor $\boldsymbol{\mathcal{X}}$ is called low-rank if any of its unfoldings is a low-rank matrix.

A rank-$(r_1, \ldots, r_k, \ldots, r_K)$ tensor $\boldsymbol{\mathcal{X}}$ of dimensions $n_1 \times \cdots \times n_K$ can be written as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K,$$

where $\times_k$ denotes the k-mode matrix product, $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ is called the *core tensor*, and $\boldsymbol{U}_k \in \mathbb{R}^{n_k \times r_k}$ ($n = 1, \ldots, K$) are left singular-vectors from the SVD of the mode-k unfolding of $\boldsymbol{\mathcal{X}}$. The above decomposition is called the Tucker decomposition [1].

Since the core tensor $\boldsymbol{\mathcal{G}}$ that corresponds to singular-values in the matrix case is not diagonal in general, it is not straightforward to generalize the trace norm from matrices to tensors.

III. Three strategies to extend the trace-norm regularization to tensors

In this section, we first consider a given tensor as a matrix and propose to minimize the trace norm of one of its unfoldings. Next, we extend this to the minimization of the weighted sum of the trace norms of the unfoldings. Finally, relaxing the condition that the tensor is *jointly* low-rank in every mode in the second approach, we propose a mixture approach.

A. Tensor as a matrix

The definition of a low-rank tensor in the previous section implies that a low-rank tensor *is* a low-rank matrix when unfolded appropriately.

Therefore, for the reconstruction of partly observed tensor, we can solve the following problem:

$$\underset{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times \cdots \times n_K}}{\text{minimize}} \qquad \frac{1}{2\lambda} \|\Omega(\boldsymbol{\mathcal{X}}) - \boldsymbol{y}\|^2 + \|\boldsymbol{X}_{(k)}\|_*, \tag{1}$$

where $\boldsymbol{X}_{(k)}$ is the mode-k unfolding of $\boldsymbol{\mathcal{X}}$, $\boldsymbol{y} \in \mathbb{R}^M$ is the vector of observations, and $\Omega : \mathbb{R}^{n_1 \times \cdots \times n_K} \to \mathbb{R}^M$ is a linear operator that reshapes the prespecified (possibly overlapping) elements of the input tensor into an M dimensional vector; M is the number of observations.

Since the estimation procedure (1) is essentially an estimation of a low-rank matrix $\boldsymbol{X}_{(k)}$, we know that $O(\tilde{n}_k^{6/5} r_k \log(\tilde{n}_k))$ samples are enough to perfectly recover the unknown true tensor $\boldsymbol{\mathcal{X}}^*$, where $r_k = \mathrm{rank}_k(\boldsymbol{\mathcal{X}}^*)$ and $\tilde{n}_k = \max(n_k, \prod_{k' \neq k} n_{k'})$, if the rank r_k is not too high [2].

Note that when we can estimate the mode-k unfolding of $\boldsymbol{\mathcal{X}}^*$ perfectly, we can also recover the whole $\boldsymbol{\mathcal{X}}^*$ perfectly, including the ranks of the modes we did not use during the estimation.

However, the success of the above procedure is conditioned on the choice of the mode to unfold the tensor. If we choose a mode with a large rank, even if there are other modes with smaller ranks, we cannot hope to recover the tensor from a small number of samples.

B. Constrained optimization of low rank tensors

In order to exploit the rank deficiency of more than one mode, it is natural to consider the following extension of the estimation procedure (1)

$$\underset{\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times \cdots \times n_K}}{\text{minimize}} \qquad \frac{1}{2\lambda} \|\Omega(\boldsymbol{\mathcal{X}}) - \boldsymbol{y}\|^2 + \sum_{k=1}^{K} \gamma_k \|\boldsymbol{X}_{(k)}\|_*.$$

This is a convex optimization problem, because it can be reformulated as follows:

$$\underset{\boldsymbol{x}, \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_K}{\text{minimize}} \qquad \frac{1}{2\lambda} \|\boldsymbol{\Omega}\boldsymbol{x} - \boldsymbol{y}\|^2 + \sum_{k=1}^{K} \gamma_k \|\boldsymbol{Z}_k\|_*, \tag{2}$$

$$\text{subject to} \qquad \boldsymbol{P}_k \boldsymbol{x} = \boldsymbol{z}_k \quad (k = 1, \ldots, K), \tag{3}$$

where $\boldsymbol{x} \in \mathbb{R}^N$ is the vectorization of $\boldsymbol{\mathcal{X}}$ ($N = \prod_{k=1}^{K} n_k$), $\boldsymbol{P}_k$ is the matrix representation of mode-k unfolding (note that $\boldsymbol{P}_k$ is a permutation matrix; thus ${\boldsymbol{P}_k}^\top \boldsymbol{P}_k = \boldsymbol{I}_N$), $\boldsymbol{Z}_k \in \mathbb{R}^{n_k \times N/n_k}$ is a matrix of the same size as the mode-k unfolding of $\boldsymbol{\mathcal{X}}$, and $\boldsymbol{z}_k$ is the vectorization of $\boldsymbol{Z}_k$. With a slight abuse of notation $\boldsymbol{\Omega} \in \mathbb{R}^{M \times N}$ denotes the observation operator as a matrix.

This approach was considered earlier in [5], but they relaxed the constraints (3) into penalty terms, which is more similar to the approach we discuss in the next subsection.

C. Mixture of low-rank tensors

The optimization problem (2) considers every mode of the tensor $\boldsymbol{\mathcal{X}}$ to be *jointly* low-rank, which might be too strict to be satisfied in practice. Thus we propose to predict instead with a mixture of K tensors; each mixture component is regularized by the trace norm to be low-rank in each mode. More specifically, we solve the following minimization problem:

$$\underset{\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_K}{\text{minimize}} \qquad \frac{1}{2\lambda} \left\| \boldsymbol{\Omega} \left(\sum\nolimits_{k=1}^{K} {\boldsymbol{P}_k}^\top \boldsymbol{z}_k \right) - \boldsymbol{y} \right\|^2 + \sum_{k=1}^{K} \gamma_k \|\boldsymbol{Z}_k\|_*. \tag{4}$$

Note that when $\boldsymbol{x} = \frac{1}{K} {\boldsymbol{P}_k}^\top \boldsymbol{z}_k$ for all $k = 1, \ldots, K$, the problem (4) reduces to the problem (2) with $\gamma'_k = \gamma_k K$.

IV. Optimization

In this section, we describe the optimization algorithms based on the alternating direction method of multipliers [6] (also known as the split Bregman iteration [7]) for the problems (1), (2), and (4).

A. Optimization of problem (1)

We consider the following constrained reformulation of the problem (1)

$$\underset{\boldsymbol{x},\boldsymbol{Z}}{\text{minimize}} \quad \frac{1}{2\lambda}\|\boldsymbol{\Omega x}-\boldsymbol{y}\|^2+\|\boldsymbol{Z}\|_*, \quad \text{subject to} \quad \boldsymbol{P}_k\boldsymbol{x}=\boldsymbol{z}. \tag{5}$$

The augmented Lagrangian (AL) function of the above constrained minimization problem can be written as follows:

$$L_\eta(\boldsymbol{x},\boldsymbol{Z},\boldsymbol{A})=\frac{1}{2\lambda}\|\boldsymbol{\Omega x}-\boldsymbol{y}\|^2+\|\boldsymbol{Z}\|_*+\boldsymbol{\alpha}^\top(\boldsymbol{P}_k\boldsymbol{x}-\boldsymbol{z})+\frac{\eta}{2}\|\boldsymbol{P}_k\boldsymbol{x}-\boldsymbol{z}\|^2,$$

where $\boldsymbol{x}\in\mathbb{R}^N$ is a vectorization of $\boldsymbol{\mathcal{X}}$, $\boldsymbol{Z}\in\mathbb{R}^{d_k\times N/d_k}$ is an auxiliary variable that corresponds to the mode-k unfolding of $\boldsymbol{\mathcal{X}}$, and $\boldsymbol{z}\in\mathbb{R}^N$ is the vectorization of $\boldsymbol{Z}$; $\boldsymbol{\alpha}\in\mathbb{R}^N$ is the Lagrangian multiplier vector that corresponds to the constraint $\boldsymbol{P}_k\boldsymbol{x}=\boldsymbol{z}$. Note that the AL function reduces to the ordinary Lagrangian if $\eta=0$.

Starting from an initial point $(\boldsymbol{x}^0,\boldsymbol{Z}^0,\boldsymbol{\alpha}^0)$, the alternating direction method of multipliers for the problem (5) performs the following steps:

$$\boldsymbol{x}^{t+1}=\underset{\boldsymbol{x}}{\operatorname{argmin}}\, L_\eta(\boldsymbol{x},\boldsymbol{Z}^t,\boldsymbol{\alpha}^t), \tag{6}$$

$$\boldsymbol{Z}^{t+1}=\underset{\boldsymbol{Z}}{\operatorname{argmin}}\, L_\eta(\boldsymbol{x}^{t+1},\boldsymbol{Z},\boldsymbol{\alpha}^t), \tag{7}$$

$$\boldsymbol{\alpha}^{t+1}=\boldsymbol{\alpha}^t+\eta(\boldsymbol{P}_k\boldsymbol{x}^{t+1}-\boldsymbol{z}^{t+1}). \tag{8}$$

All the above steps can be implemented in closed forms. First, minimization with respect to $\boldsymbol{x}$ yields,

$$\boldsymbol{x}^{t+1}=\left(\boldsymbol{\Omega}^\top\boldsymbol{y}+\lambda{\boldsymbol{P}_k}^\top(\eta\boldsymbol{z}^t-\boldsymbol{\alpha}^t)\right)./(\mathbf{1}_\Omega+\lambda\eta\mathbf{1}_N),$$

where $\mathbf{1}_\Omega$ is an N-dimensional vector that has one for observed elements and zero otherwise; $\mathbf{1}_N$ is an N-dimensional vector filled with ones; $./$ denotes element-wise division. Note that when $\lambda\to 0$ (zero training error), the above expression reduces to

$$x_i^{t+1}=\begin{cases}(\boldsymbol{\Omega}^\top\boldsymbol{y})_i, & i\in\Omega,\\ ({\boldsymbol{P}_k}^\top(\boldsymbol{z}^t-\boldsymbol{\alpha}^t/\eta))_i, & i\notin\Omega\end{cases}\qquad(i=1,\ldots,N). \tag{9}$$

Next, the minimization with respect to $\boldsymbol{Z}$ yields,

$$\boldsymbol{Z}^{t+1}=\operatorname{prox}^{\mathrm{tr}}_{1/\eta}\left(\boldsymbol{P}_j\boldsymbol{x}^{t+1}+\boldsymbol{\alpha}^t/\eta\right),$$

where $\operatorname{prox}^{\mathrm{tr}}_{1/\eta}$ is the proximity operator with respect to the trace norm and is defined as follows (see [8]):

$$\operatorname{prox}^{\mathrm{tr}}_{\lambda}(\boldsymbol{z}):=\boldsymbol{U}\max(\boldsymbol{S}-\lambda,0)\boldsymbol{V}^\top,$$

where $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ is the SVD of the matrix $\boldsymbol{Z}$ obtained by appropriately rearranging the elements of $\boldsymbol{z}$.

The KKT condition for the problem (5) can be written as follows:

$$\frac{1}{\lambda}\boldsymbol{\Omega}^\top(\boldsymbol{\Omega}\boldsymbol{x}-\boldsymbol{y}) + \boldsymbol{P}_k{}^\top\boldsymbol{\alpha} = 0, \tag{10}$$

$$\boldsymbol{P}_k\boldsymbol{x} - \boldsymbol{z} = 0, \tag{11}$$

$$\boldsymbol{\alpha} \in \partial\|\boldsymbol{Z}\|_*. \tag{12}$$

Note that from the update equations (7) and (8), the third condition (12) is automatically satisfied. Therefore, we monitor the first two conditions (10)-(11) and stop the algorithm when the norm $\|\boldsymbol{\Omega}^\top(\boldsymbol{\Omega}\boldsymbol{x}^{t+1} - \boldsymbol{y})/\lambda + \boldsymbol{P}_k{}^\top\boldsymbol{\alpha}^{t+1}\|$ and $\|\boldsymbol{P}_k\boldsymbol{x}^{t+1} - \boldsymbol{z}^{t+1}\|$ both fall below some tolerance, say 10^{-3}.

B. Optimization of problem (2)-(3)

The AL function of the constrained minimization problem (2)-(3) can be written as follows:

$$L_\eta(\boldsymbol{x}, \{\boldsymbol{Z}_k\}_{k=1}^K, \{\boldsymbol{\alpha}_k\}_{k=1}^K) = \frac{1}{2\lambda}\|\boldsymbol{\Omega}\boldsymbol{x}-\boldsymbol{y}\|^2 + \sum_{k=1}^K\left(\gamma_k\|\boldsymbol{Z}_k\|_* + \boldsymbol{\alpha}_k{}^\top(\boldsymbol{P}_k\boldsymbol{x}-\boldsymbol{z}_k) + \frac{\eta}{2}\|\boldsymbol{P}_k\boldsymbol{x}-\boldsymbol{z}_k\|^2\right).$$

Starting from an initial point $(\boldsymbol{x}^0, \{\boldsymbol{Z}_k^0\}_{k=1}^K, \{\boldsymbol{\alpha}_k^0\}_{k=1}^K)$, we take exactly the same steps as in (6)-(8) except that the last two steps are performed for all $k = 1, \ldots, K$. That is,

$$\boldsymbol{x}^{t+1} = \left(\boldsymbol{\Omega}^\top\boldsymbol{y} + \lambda\sum\nolimits_{k=1}^K \boldsymbol{P}_k{}^\top(\eta\boldsymbol{z}_k^t - \boldsymbol{\alpha}_k^t)\right)./(\boldsymbol{1}_\Omega + \lambda\eta K\boldsymbol{1}_N), \tag{13}$$

$$\boldsymbol{Z}_k^{t+1} = \mathrm{prox}^{\mathrm{tr}}_{\gamma_k/\eta}\left(\boldsymbol{P}_k\boldsymbol{x}^{t+1} + \boldsymbol{\alpha}_k^t/\eta\right) \qquad (k = 1, \ldots, K), \tag{14}$$

$$\boldsymbol{\alpha}_k^{t+1} = \boldsymbol{\alpha}_k^t + \eta(\boldsymbol{P}_k\boldsymbol{x}^{t+1} - \boldsymbol{z}^{t+1}) \qquad (k = 1, \ldots, K). \tag{15}$$

The KKT condition for the problem (2)-(3) can be written as follows:

$$\frac{1}{\lambda}\boldsymbol{\Omega}^\top(\boldsymbol{\Omega}\boldsymbol{x}-\boldsymbol{y}) + \sum\nolimits_{k=1}^K \boldsymbol{P}_k{}^\top\boldsymbol{\alpha}_k = 0, \tag{16}$$

$$\boldsymbol{P}_k\boldsymbol{x} - \boldsymbol{z}_k = 0 \qquad (\forall k = 1, \ldots, K), \tag{17}$$

$$\boldsymbol{\alpha}_k \in \gamma_k\partial\|\boldsymbol{Z}_k\|_* \qquad (\forall k = 1, \ldots, K). \tag{18}$$

Note again that the third condition (18) is automatically satisfied. Therefore, we stop the algorithm when the errors $\|\boldsymbol{\Omega}^\top(\boldsymbol{\Omega}\boldsymbol{x}^{t+1} - \boldsymbol{y})/\lambda + \sum_{k=1}^K \boldsymbol{P}_k{}^\top\boldsymbol{\alpha}_k^{t+1}\|$ and $\max_k \|\boldsymbol{P}_k\boldsymbol{x}^{t+1} - \boldsymbol{z}_k^{t+1}\|$ both fall below some tolerance.

C. Optimization for problem (4)

We consider the following dual problem of (4):

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^N,\boldsymbol{W}_k\in\mathbb{R}^{n_k\times N/n_k}}{\text{minimize}} \quad \frac{\lambda}{2}\|\boldsymbol{\alpha}\|^2-\boldsymbol{\alpha}^\top\boldsymbol{y}+\sum_{k=1}^{K}\delta_{\gamma_k}(\boldsymbol{W}_k), \quad \text{subject to} \quad \boldsymbol{w}_k=\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha} \quad (k=1,\ldots,K), \tag{19}$$

where $\boldsymbol{\alpha}\in\mathbb{R}^N$ is a dual vector; $\boldsymbol{W}_k\in\mathbb{R}^{n_k\times N/n_k}$ is an auxiliary variable that corresponds to the mode-k unfolding of $\boldsymbol{\Omega}^\top\boldsymbol{\alpha}$, and $\boldsymbol{w}_k\in\mathbb{R}^N$ is the vectorization of $\boldsymbol{W}_k$; the indicator function δ_λ is defined as $\delta_\lambda(\boldsymbol{W})=0$, if $\|\boldsymbol{W}\|\leq\lambda$, and $\delta_\lambda(\boldsymbol{W})=+\infty$, otherwise, where $\|\cdot\|$ is the spectral norm (maximum singular-value of a matrix).

The AL function for the problem (19) can be written as follows:

$$L_\eta(\boldsymbol{\alpha},\{\boldsymbol{W}_k\}_{k=1}^K,\{\boldsymbol{z}_k\}_{k=1}^K)=\frac{\lambda}{2}\|\boldsymbol{\alpha}\|^2-\boldsymbol{\alpha}^\top\boldsymbol{y}+\sum_{k=1}^{K}\left(\delta_{\gamma_k}(\boldsymbol{W}_k)+\boldsymbol{z}_k{}^\top(\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha}-\boldsymbol{w}_k)+\frac{\eta}{2}\|\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha}-\boldsymbol{w}_k\|^2\right)$$

Similar to the previous two algorithms, we start from an initial point $(\boldsymbol{\alpha}^0,\{\boldsymbol{W}_k^0\}_{k=1}^K,\{\boldsymbol{z}_k^0\}_{k=1}^K)$, and compute the following steps:

$$\begin{aligned}
\boldsymbol{\alpha}^{t+1}&=\underset{\boldsymbol{\alpha}}{\operatorname{argmin}}\, L_\eta(\boldsymbol{\alpha},\{\boldsymbol{W}_k^t\}_{k=1}^K,\{\boldsymbol{z}_k^t\}_{k=1}^K)\\
\boldsymbol{W}_k^{t+1}&=\underset{\boldsymbol{W}_k}{\operatorname{argmin}}\, L_\eta(\boldsymbol{\alpha}^{t+1},\{\boldsymbol{W}_k\}_{k=1}^K,\{\boldsymbol{z}_k^t\}_{k=1}^K)\\
\boldsymbol{z}_k^{t+1}&=\boldsymbol{z}_k^t+\eta(\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha}^{t+1}-\boldsymbol{w}_k^{t+1}).
\end{aligned} \tag{20}$$

The above steps can be computed in closed forms. In fact,

$$\boldsymbol{\alpha}^{t+1}=\frac{1}{\lambda+\eta K}\left(\left(\boldsymbol{y}-\boldsymbol{\Omega}\textstyle\sum_{k=1}^K\boldsymbol{P}_k\boldsymbol{z}_k^t\right)+\eta\boldsymbol{\Omega}\textstyle\sum_{k=1}^K\boldsymbol{P}_k{}^\top\boldsymbol{w}_k^t\right), \tag{21}$$

$$\boldsymbol{W}_k^{t+1}=\operatorname{prox}_{\gamma_k}^{\mathrm{tr}^*}(\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha}^{t+1}+\boldsymbol{z}_k^t/\eta), \tag{22}$$

where the proximity operator $\operatorname{prox}_\lambda^{\mathrm{tr}^*}$ is the projection onto a radius λ-spectral-norm ball, as follows:

$$\operatorname{prox}_\lambda^{\mathrm{tr}^*}(\boldsymbol{w}):=\boldsymbol{U}\min(\boldsymbol{S},\lambda)\boldsymbol{V}^\top,$$

where $\boldsymbol{W}=\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ is the SVD of the matricization of the input vector $\boldsymbol{w}$. Moreover, combining the two steps (22) and (20), we have (see [8])

$$\boldsymbol{z}_k^{t+1}=\operatorname{prox}_{\gamma_k\eta}^{\mathrm{tr}}\left(\boldsymbol{z}_k^t+\eta\boldsymbol{P}_k\boldsymbol{\Omega}^\top\boldsymbol{\alpha}^{t+1}\right). \tag{23}$$

Therefore, we can simply iterate steps (21) and (23).

The KKT condition for the problem (19) can be written as follows:

$$\lambda\boldsymbol{\alpha} - \boldsymbol{y} + \boldsymbol{\Omega}\sum_{k=1}^{K} \boldsymbol{P}_k{}^{\top}\boldsymbol{z}_k = 0, \qquad \boldsymbol{P}_k\boldsymbol{\Omega}^{\top}\boldsymbol{\alpha} - \boldsymbol{w}_k = 0 \qquad (\forall k = 1, \ldots, K),$$

$$\boldsymbol{z}_k \in \partial\delta_{\gamma_k}(\boldsymbol{W}_k) \qquad (\forall k = 1, \ldots, K).$$

Again the last condition is automatically satisfied by step (23); thus we check the first two conditions and terminate the algorithm when they are both below some tolerance.

V. Numerical experiments

We randomly generated a rank-(7,8,9) tensor of dimensions (50,50,20) by drawing the core from the standard normal distribution and multiplying its each mode by an orthonormal factor randomly drawn from the Haar measure. We randomly selected some elements of the true tensor for training and kept the remaining elements for testing. We used the algorithms described in the previous section with the tolerance 10^{-3}. We choose $\gamma_k = 1$ for simplicity in the later two approaches. For the first two approaches, $\lambda \to 0$ (zero training error) was used; see (9). For the last approach, we used $\lambda = 0$. The Tucker decomposition algorithm `tucker` in the N-way toolbox [9] is also included as a baseline, for which we used the correct rank ("exact") and the 20% higher rank ("large"). Note that all proposed approaches can find the rank automatically. The generalization error is defined as the square-root of the sum of squared differences between the true and the estimated tensors over the unobserved elements. For the "As a Matrix" strategy, error for each mode is reported. The experiment was repeated 10 times and averaged.

Figure 1 shows the result of tensor completion using three strategies we proposed above, as well as the Tucker decomposition. The proposed "Constraint" approach shows a sharp threshold behaviour around 35% observation from a poor fit (generalization error> 1) to an almost perfect fit (generalization error$\simeq 10^{-3}$). The "As a Matrix" approach also show similar transition for mode 1 and mode 2 (around 40%), and mode 3 (around 80%), but even the first transition is slower than the "Constraint" approach. The "Mixture" approach shows a transition around 70% slightly faster than the mode 3 in the "As A Matrix" approach. Tucker shows early decrease in the generalization error, but when the rank is missspecified ("large"), the error remains almost constant; even when the correct rank is known ("exact"), the convergence is slower than the proposed "Constraint" approach.

We have further investigated the condition for the threshold behaviour using the proposed "Constraint" approach. In Figure 3, we can see that the fraction of observations required to perfectly recover an unknown tensor is roughly proportional to the sum of the rank of the underlying tensor, where we define the reconstruction to be perfect when the mean generalization error is less than 0.01.

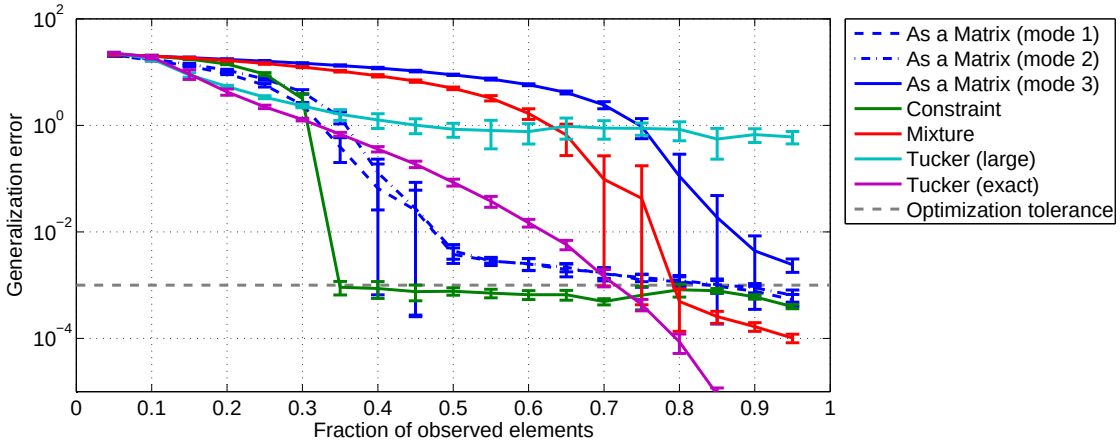

Fig. 1. Comparison of three strategies, tensor as a matrix ("As a Matrix"), constrained optimization ("Constraint"), and mixture of low-rank tensors ("Mixture"). Also the Tucker decomposition with 20% higher rank ("large") and with the correct rank ("exact") implemented in the N-way toolbox [9] are included as baselines. The generalization error is plotted against the fraction of observed elements of the underlying low-rank tensor. Also the tolerance of optimization (10^{-3}) is shown.

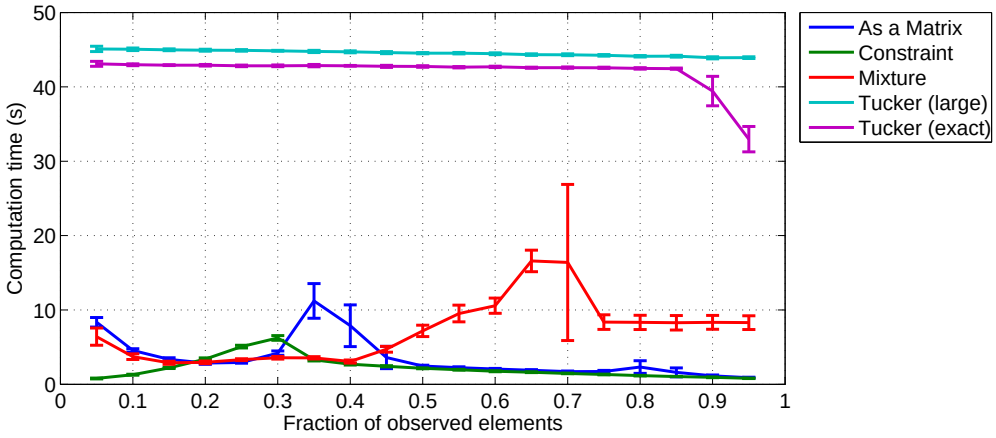

Fig. 2. Comparison of computation times.

VI. Summary

In this paper we have proposed three strategies to extend the trace norm to tensor rank minimization, and we have compared them on a simulated tensor completion problem. We have found that tensor completion using the "Constraint" approach shows nearly perfect reconstruction from only 35% observations. There is no need to specify the rank of the decomposition as in the conventional Tucker decomposition approach.

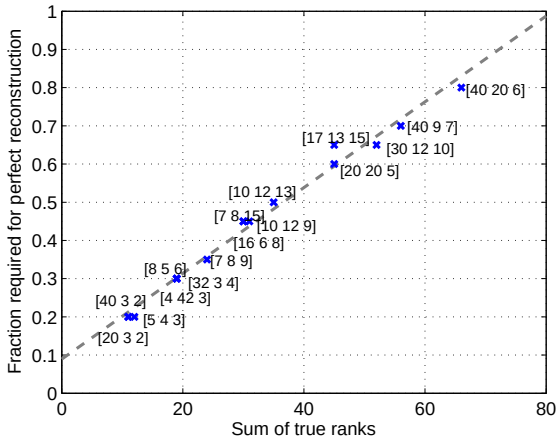

Fig. 3. Fraction of observations at the threshold plotted against the sum of true ranks. Numbers in the brackets denote the k-rank of the underlying tensor. The dimension of the tensor is (50,50,20).

The proposed approach shows a sharp threshold behaviour and we have found that the fraction of samples at the threshold is roughly proportional to the sum of ranks of the underlying tensor. Further analysis is necessary to explain the threshold behaviour.

REFERENCES

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

[2] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[3] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[4] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[5] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," in *Prof. ICCV*, 2009.

[6] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Comput. Math. Appl.*, vol. 2, no. 1, pp. 17–40, 1976.

[7] T. Goldstein and S. Osher, "The split Bregman method for L1 regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[8] R. Tomioka, T. Suzuki, and M. Sugiyama, "Augmented lagrangian methods for learning, selecting, and combining features," in *Optimization for Machine Learning*, S. Sra, S. Nowozin, and S. J. Wright, Eds. MIT Press, 2010.

[9] C. A. Andersson and R. Bro, "The n-way toolbox for matlab," *Chemometrics & Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1–4, 2000, http://www.models.life.ku.dk/source/nwaytoolbox/.