Tensor Completion with Side Information: A Riemannian Manifold Approach

Tengfei Zhou, Hui Qian*, Zebang Shen, Chao Zhang, Congfu Xu
College of Computer Science and Technology, Zhejiang University, China
{zhoutengfei,qianhui,shenzebang,zczju,xucongfu}@zju.edu.cn

Abstract

By restricting the iterate on a nonlinear manifold, the recently proposed Riemannian optimization methods prove to be both efficient and effective in low rank tensor completion problems. However, existing methods fail to exploit the easily accessible side information, due to their format mismatch. Consequently, there is still room for improvement. To fill the gap, in this paper, a novel Riemannian model is proposed to tightly integrate the original model and the side information by overcoming their inconsistency. For this model, an efficient Riemannian conjugate gradient descent solver is devised based on a new metric that captures the curvature of the objective. Numerical experiments suggest that our method is more accurate than the state-of-the-art without compromising the efficiency.

1 Introduction

Low Rank Tensor Completion (LRTC) problem, which aims to recover a tensor from its linear measurements, arises naturally in many artificial intelligence applications. In hyperspectral image inpainting, LRTC is applied to interpolate the unknown pixels based on the partial observation [Xu et al., 2015]. In recommendation tasks, LRTC helps users find interesting items [Liu et al., 2015]. In computational phenotyping, one adopts LRTC to discovery phenotypes in heterogeneous electronic health records [Wang et al., 2015].

Euclidean Models: LRTC can be formulated as a variety of optimization models over the Euclidean space. Amongst them, convex models that encapsulate LRTC as a regression problem penalized by a tensor nuclear norm are the most popular and well-understood [Romera-Paredes and Pontil, 2013; Zhang et al., 2014]. Though most of them have sound theoretical guarantees [Zhang and Aeron, 2016; Chen et al., 2013; Yuan and Zhang, 2015], in general, their solvers are ill-suited for large tensors because these procedures involve Singular Value Decomposition (SVD) of huge matrices per iteration [Liu et al., 2013]. Another class of Euclidean models is formulated as the decomposition problem that factorizes a low rank tensor into small factors [Jain and Oh, 2014; Filipović and Jukić, 2015; Xu et al., 2015]. Many solvers for such decomposition based model have been proposed, and low per-iteration computational cost is illustrated [Beutel et al., 2014; Liu et al., 2014; Smith et al., 2016].

Riemannian Models: LRTC can also be modeled by optimization constrained on Riemannian manifolds [Kressner et al., 2014; Kasai and Mishra, 2016], which is easily handled by many manifold based solvers [Absil et al., 2009]. Empirical comparison has shown that Riemannian solvers use significantly less CPU time to recover the underlying tensor in contrast to the Euclidean solvers [Kasai and Mishra, 2016]. The main reason resides in that they avoid SVD of huge matrices by explicitly exploiting the geometrical structure of LRTC, which makes them more suitable for massive problem.

Of all the Riemannian models, two search spaces, fix multilinear rank manifold [Kressner et al., 2014] and Tucker manifold [Kasai and Mishra, 2016], are usually employed. The former is a sub-manifold of Euclidean space, and the latter is a quotient manifold induced by the Tucker decomposition. Generally, quotient manifold based solvers have higher convergence rates because it is usually easier to design a pre-conditioner for them [Kasai and Mishra, 2016; Mishra and Sepulchre, 2016].

Side Information: In the Euclidean models of LRTC, side information has been proved to be helpful in improving the accuracy [Narita et al., 2011; Acar et al., 2011; Beutel et al., 2014]. One common form of the side information is the feature matrix, which measures the statistical properties of tensor modes [Kolda and Bader, 2009]. For example, in Netflix tasks, feature matrix can be built from the demography of users [Bell and Koren, 2007]. Another form is the similarity matrix, which quantifies the resemblance between two entities of a tensor mode. For instance, the social network generates the similarity matrix by utilizing the correspondence between users [Rai et al., 2015]. In practice, these two matrices can be transformed to each other, and we only consider the feature matrix case.

However, as far as we know, side information has not been incorporated in any Riemannian model for LRTC. The first difficulty lies in the model design. Fusing the side information into the Riemannian model inevitably compromises the integrity of the low rank tensor due to the compactness of the manifold. The second difficulty results from the solver design. Incorporating the side information may aggravate the
ill-conditioning of LRTC problem and degenerates the convergence significantly.

**Contributions**: To address these difficulties, a novel Riemannian LRTC method is proposed from the perspective of both model and solver designs. By exploring the relation between the subspace spanned by the tensor fibers and the column space of the feature matrix, we explicitly integrate the side information in a compact way. Meanwhile, a first order solver is devised under the manifold optimization framework. To ease the ill-conditioning, we design a novel metric based on the manifold. Empirical studies illustrate that our method achieves much

\[
\mathcal{F}_r = \{ \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \mid \text{rank vec} (\mathbf{X}) = r \}. 
\]

To define the Tucker manifold, we first define a total space

\[
\mathcal{M}_r = \mathbb{R}^{r_1 \times r_2 \times r_3} \times S(r_1, n_1) \times S(r_2, n_2) \times S(r_3, n_3), \quad (1)
\]

in which \(S(r_i, n_i)\) is the Stiefel manifold of \(n_i \times r_i\) matrices with orthogonal columns. Then, we can depict the Tucker manifold of multi-linear rank \(r\) as follows.

\[
\mathcal{M}_r / \sim = \{ [\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3] \mid [\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3] \in \mathcal{M}_r \}. \quad (2)
\]

The Tucker manifold is a quotient manifold of the total space (1). We use the abstract quotient manifold, rather than the concrete total space, as search space because the non-uniqueness of the Tucker decomposition is undesirable for optimization. Note that such non-uniqueness will introduce more local optima into the minimization. The relation of manifold \(\mathcal{F}_r\) and \(\mathcal{M}_r / \sim\) is characterized as follows.

**Proposition 1.** The quotient manifold \(\mathcal{M}_r / \sim\) is diffeomorphic to the fix multi-linear rank manifold \(\mathcal{F}_r\), with diffeomorphism \(\rho()\) from \(\mathcal{F}_r\) to \(\mathcal{M}_r / \sim\) defined by \(\rho(\mathbf{X}) = [\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3]\) where \([\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3]\) is the Tucker representation of \(\mathbf{X}\).

This proposition says that each tensor \(\mathbf{X} \in \mathcal{F}_{r}\) can be represented by a unique equivalent class \([\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3] \in \mathcal{M}_{r}/\sim\) and vice-versa.

### 2.1 Search Space of Riemannian Models

The Tucker manifold that we used in our Riemannian model is a quotient manifold induced by the Tucker decomposition. In order to lay the ground for Tucker manifold, we first describe its counterpart, the fix multi-rank manifold, which will be helpful in understanding the whole derivation. A fixed multi-linear rank manifold \(\mathcal{F}_r\) consists of tensors with the same fixed multi-linear rank. Specifically

\[
\mathcal{F}_r = \{ \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \mid \text{rank vec} (\mathbf{X}) = r \}.
\]

In the search space of Riemannian models, \(\mathcal{F}_r\) and \(\mathcal{M}_r / \sim\) can be represented by a unique equivalent class \([\mathbf{G}, \{ \mathbf{U}_i \}_{i=1}^3] \in \mathcal{M}_{r}/\sim\) and vice-versa.

### 2.2 Vanilla Riemannian Tensor Completion

The purest incarnation of Riemannian tensor completion model is the Riemannian model over the fix multi-linear rank manifold. Let \(\mathcal{R} \in \mathbb{R}_{r_1 \times r_2 \times r_3}^{n_1 \times n_2 \times n_3}\) be a partially observed tensor. Let \(\Omega\) be the set which contains the indices of observed entries. The model can be expressed as:

\[
\min_{\mathbf{X}} \frac{1}{2} \| \mathcal{P}_\Omega(\mathbf{X} - \mathcal{R}) \|_F^2 \quad \text{s.t.} \quad \mathbf{X} \in \mathcal{F}_r, \qquad (3)
\]

with \(\mathcal{P}_\Omega\) maps \(\mathcal{X}\) to the sparsified tensor \(\mathcal{P}_\Omega(\mathbf{X})\), where \(\mathcal{P}_\Omega(\mathbf{X})(i_1, i_2, i_3) = \mathbf{X}(i_1, i_2, i_3)\) if \((i_1, i_2, i_3) \in \Omega\), and \(\mathcal{P}_\Omega(\mathbf{X})(i_1, i_2, i_3) = 0\) otherwise.

Another popular model, Tucker model, is based on the quotient manifold \(\mathcal{M}_r / \sim\), which can be expressed as:

\[
\min_{\mathbf{X}} \frac{1}{2} \| \mathcal{P}_\Omega(\rho^{-1}(\mathbf{X}) - \mathcal{R}) \|_F^2 \quad \text{s.t.} \quad \mathbf{X} \in \mathcal{M}_r / \sim, \quad (4)
\]

with \(\rho\) defined in Prop. 1.

Note that since the dawn of Riemannian framework for LRTC, a quandary exists: on one hand, sparse measurement limits the capacity of the solution; on the other hand, rich side information can not be incorporated into this framework. In many artificial intelligence applications, demands for high accuracy further exacerbates such dilemma.
3 Riemannian Model with Side Information

We focus on the case that the side information is encoded in feature matrices \( P_f \in \mathbb{R}^{n_f \times k_i} \). Suppose \( R \in \mathcal{F}_r \) has Tucker factors \( \{ G_i \}_{i=1}^3 \). Without loss of generality, we assume that \( k_i \geq r_i \) and \( P_f \) has orthogonal columns.

In the ideal case, we assume that \( \text{span}(U_i) \subset \text{span}(P_f) \). (5)

Such relation means that the feature matrices contain all the information in the latent space of the underlying tensor. Equivalently, there exists a matrix \( W_i \) such that \( U_i = P_f W_i \). However, in practice, due to the existence of noise, one can only expect such relation to hold approximately, i.e., \( U_i \approx P_f W_i \). Incorporating such relation to a tensor completion model via penalization, we have the following formulation

\[
\min_{\mathcal{G}, \{ U_i \}_{i=1}^3} L(\mathcal{G}, \{ U_i \}_{i=1}^3) + \sum_{i=1}^3 \frac{\alpha_i}{2} \| U_i - P_f W_i \|_F^2,
\]

s.t. \( \{ G_i \}_{i=1}^3 \in \mathcal{M}_r \),

where \( L(\mathcal{G}, \{ U_i \}_{i=1}^3) = \| P_\Omega (G \times_3 U_i - R) \|_F^2/2 \). Fixing \( \mathcal{G} \) and \( U_i \), with respect to \( W_i \), (6) has a close form solution

\[
W_i = (P_f^T P_f)^{-1} P_f^T U_i.
\]

Since \( \min_{x,y} l(x,y) = \min_{x} l(x, y(x)) \) where \( y(x) = \arg \min_{x} l(x, y) \), one can substitute (7) into the above problem and obtain the following equivalence

\[
\min_{\mathcal{G}, \{ U_i \}_{i=1}^3} L(\mathcal{G}, \{ U_i \}_{i=1}^3) + \sum_{i=1}^3 \frac{\alpha_i}{2} \text{trace}(U_i^T (I - P_f P_f^T) U_i)
\]

\[
\leq f(\mathcal{G}, \{ U_i \}_{i=1}^3)
\]

s.t. \( \{ G_i \}_{i=1}^3 \in \mathcal{M}_r \),

(8)

Although the cost function is already smooth over the total space \( \mathcal{M}_r \), due to its invariance over the equivalent class \( \mathcal{G}, \{ U_i \}_{i=1}^3 \), there can be infinite local optima, which is extremely undesirable. Indeed, if \( \{ G_i \}_{i=1}^3 \) is a local optimal of the objective, then so is every point in the infinite set \( \{ G_i \}_{i=1}^3 \). One way to reduce the number of local optima is to mathematically treated the entire set \( \{ G_i \}_{i=1}^3 \) as a point. Consequently, we redefine the cost by \( \tilde{f}(\mathcal{G}, \{ U_i \}_{i=1}^3) = f(\mathcal{G}, \{ U_i \}_{i=1}^3) \) and obtain the following Riemannian optimization problem over the quotient manifold \( \mathcal{M}_r/\sim \):

\[
\min_{\mathcal{X} \in \mathcal{F}_r} \frac{1}{2} \| P_\Omega (\mathcal{X} - R) \|_F^2
\]

\[
+ \sum_{i=1}^3 \frac{\alpha_i}{2} \text{dist}^2(\text{span}(\mathcal{X}_{(i)}), \text{span}(P_f))
\]

where \( \text{dist}(\cdot, \cdot) \) is the Chodale distance [Ye and Lim, 2014] between two subspaces. And vice versa.

4 Riemannian Conjugate Gradient Descent

We depict the optimization framework for quotient manifolds in Fig. 1. Under this framework, we solve the proposed problem (9) by Riemannian Conjugate Gradient descent (CG).

Figure 1: Optimization Framework for Quotient Manifold: most Riemannian solvers are based on the iteration formula: \([x^+] \leftarrow R_{\mathcal{F}}(\eta(x))\), where \( t > 0 \) is the stepsize, \( \eta(x) \) is the search direction picked from current tangent space \( T_x \mathcal{M} / \sim \), and \( R_{\mathcal{F}}(\cdot) \) is the retraction, i.e. a map from current tangent space to \( \mathcal{M} / \sim \). Due to the abstractness of quotient manifold, such iteration is often lifted (in represented) in the total space as \( x = R_{\mathcal{F}}(\eta(x)) \) where \( x \in [x] \), \( \eta_x \) is the horizontal lift of \( \eta(x) \), and \( R_{\mathcal{F}}(\cdot) \) is the lifted retraction. Such representation is possible only if \( \mathcal{M} / \sim \) has the structure of Riemannian quotient, that is the total space is endowed with an invariant Riemannian metric.

Remark 2. Since each \( [\mathcal{X}] \in \mathcal{M}_r / \sim \) has a unique tensor representation \( \mathcal{X} \in \mathcal{F}_r \), we show that the abstract model (9) can be represented as a concrete model over the manifold \( \mathcal{F}_r \). Specifically, the following Proposition interprets the proposed model as an optimization problem with a regularizer that encourages the mode-i space of the estimated tensor close to \( \text{span}(P_f) \).

Proposition 2. If \( [\mathcal{X}] \) is a critical point of problem (9) then its tensor representation \( \mathcal{X} \) is a critical point of the following problem.

\[
\min_{\mathcal{X} \in \mathcal{F}_r} \frac{1}{2} \| P_\Omega (\mathcal{X} - R) \|_F^2
\]

\[
+ \sum_{i=1}^3 \frac{\alpha_i}{2} \text{dist}^2(\text{span}(\mathcal{X}_{(i)}), \text{span}(P_f))
\]

where \( \text{dist}(\cdot, \cdot) \) is the Chodale distance [Ye and Lim, 2014] between two subspaces. And vice versa.

4.1 Metric Tuning

Riemannian metric \( \langle \cdot, \cdot \rangle_{\mathcal{F}} \) of \( \mathcal{M}_r \) is an inner product defined over each tangent space \( T_{\mathcal{F}} \mathcal{M}_r \). A high-quality Riemannian solver for a quotient manifold should be equipped with
Algorithm 1 CGSI: a Riemannian CG method

Require: Initialize $\mathcal{X}^{(0)} = (\mathcal{G}^{(0)}, \{U_i^{(0)}\}_{i=1}^3)$ and tolerance $\epsilon$
1: $k = 0$;
2: $\eta^{(-1)} = (0, \{0\}_{i=1}^3)$;
3: repeat
4: compute current Riemannian gradient $\xi^{(k)} = \nabla f(\mathcal{X}^{(k)})$;
5: compose CG direction $\eta^{(k)} = -\xi^{(k)} + \beta^{(k)} T_k(\eta^{(k-1)})$;
6: choose a step size $t_k > 0$;
7: update by retraction $\mathcal{X}^{(k+1)} = R_{\mathcal{X}^{(k)}}(t_k \eta^{(k)})$;
8: $k = k + 1$;
9: until $(\xi^{(k-1)}, \xi^{(k-1)})_{\mathcal{X}^{(k-1)}} \leq \epsilon$;
10: return $\mathcal{X}^{(k)}$.

a well-tuned metric, because (1) the metric determines the differential structure of the quotient manifold, and more importantly (2) it implicitly endows the solver with a preconditioner, which heavily affects the convergent rate [Mishra and Sepulchre, 2014; Mishra, 2014].

From the perspective of preconditioning, it seems that the best candidate is the Newton metric $\langle \eta, \xi \rangle_{\mathcal{X}} = D^2 f(\mathcal{X})|_{\eta, \xi} = T_{\mathcal{X}}^2 \mathcal{M}_r$ where $D^2 f(\mathcal{X})$ is the second order differential of the cost function. However, under such metric, computing the search direction involves solving a large system of linear equations, which precludes the Newton metric from the approach to huge datasets. Therefore, we propose to use the following alternative:

$$
\langle \eta, \xi \rangle_{\mathcal{X}} = D^2 f(\mathcal{X})|_{\eta, \xi} = \sum_{i=1}^3 \langle \eta_i, \xi_i \rangle_{\mathcal{G}_i(\mathcal{G}_i^T)} + \langle \eta_\varphi, \xi_\varphi \rangle + \eta_\varphi \nabla f(\mathcal{X})
$$

(11)

$$
+ \sum_{i=1}^3 N^2 \alpha^2 \max \{k' \in \{1, 2, 3\}, k \neq i \} \eta_{k'} f_{k'}^T(\mathcal{X}) \xi_{k'}
$$

where $g(\mathcal{X})$ is a scaled approximation to the original cost function, and $D^2 g(\mathcal{X})$ is the block approximation of its second derivative, specifically $g(\mathcal{X}) = \frac{1}{2} \| \mathcal{G} \times_1 U_i - \mathcal{R}_i \|_F^2 + \sum_{i=1}^3 \alpha^2 \text{trace}(U_i^T (I_i - P_iP_i^T) U_i)$ with $N = n_1n_2n_3$.

Our metric is more scalable than Newton metric. The following Proposition indicates that the scale gradient induced by this metric can be computed with $O(\sum_{i=1}^3 n_i k_i r_i + r_i^3)$ additional operations.

**Proposition 3.** Suppose that the cost function $f(\cdot)$ has Euclidean gradient $\nabla f(\mathcal{X}) = \langle \nabla f, \{ \nabla U_i f \}_{i=1}^3 \rangle$. Then its scaled gradient $\nabla f(\mathcal{X})$ under the metric (11) can be computed by:

$$
\nabla f(\mathcal{X}) = \nabla f(\mathcal{X})
$$

$$
\nabla U_i f(\mathcal{X}) = E_i G_i^{-1} + F_i G_i + N \alpha^2 I_i
$$

where $E_i = P_i P_i^T \nabla U_i f$, $F_i = \nabla U_i f - E_i$, and $G_i = \mathcal{G}_i(\mathcal{G}_i^T)$. $G_i$.

The final proposition suggests that the proposed metric makes the representation of solvers in the total space possible.

**Proposition 4.** The quotient manifold $\mathcal{M}_r / \sim$ admits a structure of Riemannian quotient manifold, if $\mathcal{M}_r$ is endowed with the Riemannian metric defined in (11).

### 4.2 Other Optimization Items

**Projectors:** To derive the optimization related items, two orthogonal projectors, $\Psi(\cdot)$ and $\Pi(\cdot)$, are required. The former projects a vector onto the tangent space $T_{\mathcal{X}}^2 \mathcal{M}_r$, and the latter is a projector from the tangent space onto the horizontal space $\mathcal{H}(\mathcal{X})$. The orthogonality of both projectors is measured by the metric (11). For lack of space, the mathematical derivation is deferred to a long version of this paper.

**Riemannian Gradient:** According to [Absil et al., 2009], the Riemannian gradient can be computed by projecting the scaled gradient onto tangent space, specifically

$$
\nabla f(\mathcal{X}) = \Psi(\nabla f(\mathcal{X})).
$$

(12)

**Retraction:** We use the retraction defined by

$$
R_{\mathcal{X}}(\eta_{\mathcal{X}}) = \mathcal{G} + \eta_{\mathcal{X}} \{ \eta f(U_i + \eta_i) \}_{i=1}^3.
$$

(13)

where $f(\cdot)$ extracts the orthogonal component from a matrix. Such retraction is proposed by [Kasai and Mishra, 2016]. In the long version of this paper, we give rigorous analysis to prove that the above retraction is compatible with the proposed metric.

### 5 Experiments

We validate the effectiveness of the proposed solver CGSI by comparing it with the state-of-the-art. The baseline can be partitioned into three classes. The first class contains Riemannian solvers including GeomCG [Kressner et al., 2014], FTC [Kasai and Mishra, 2016], and gHOI [Liu et al., 2016]. The second class consists of Euclidean solvers that take no account of the side information, including AlMIn [Romera-Paredes et al., 2013] and HalRTC [Liu et al., 2013]. The third class comprises of two methods that incorporate side information, including RUBIK [Wang et al., 2015] and TFAI
Table 2: Performance of the compared methods on hyperspectral images.

<table>
<thead>
<tr>
<th>Data</th>
<th>AltMin</th>
<th>FTC</th>
<th>GeoCG</th>
<th>gHOI</th>
<th>HalRTC</th>
<th>RUBIK</th>
<th>TFAI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Time</td>
<td>Time</td>
<td>Time</td>
<td>Time</td>
<td>Time</td>
<td>Time</td>
</tr>
<tr>
<td>S1</td>
<td>1.013</td>
<td>0.15</td>
<td>0.077</td>
<td>0.092</td>
<td>0.077</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td></td>
<td>0.135</td>
<td>0.115</td>
<td>0.115</td>
<td>0.124</td>
<td>0.085</td>
<td>0.132</td>
<td>0.132</td>
</tr>
<tr>
<td></td>
<td>1.013</td>
<td>0.15</td>
<td>0.077</td>
<td>0.092</td>
<td>0.077</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td></td>
<td>0.135</td>
<td>0.115</td>
<td>0.115</td>
<td>0.124</td>
<td>0.085</td>
<td>0.132</td>
<td>0.132</td>
</tr>
<tr>
<td>S2</td>
<td>1.013</td>
<td>0.15</td>
<td>0.077</td>
<td>0.092</td>
<td>0.077</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td></td>
<td>0.135</td>
<td>0.115</td>
<td>0.115</td>
<td>0.124</td>
<td>0.085</td>
<td>0.132</td>
<td>0.132</td>
</tr>
<tr>
<td></td>
<td>1.013</td>
<td>0.15</td>
<td>0.077</td>
<td>0.092</td>
<td>0.077</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td></td>
<td>0.135</td>
<td>0.115</td>
<td>0.115</td>
<td>0.124</td>
<td>0.085</td>
<td>0.132</td>
<td>0.132</td>
</tr>
</tbody>
</table>

5.1 Hyperspectral Image Inpainting

A hyperspectral image is a tensor whose observations are photographs of the same scene under different wavelengths. We adopt the dataset provided in [Foster et al., 2006] which contains images about eight different dwarf scenes taken under 33 various wavelengths. To make all methods in our baseline applicable to the completion problem, we resize hyper- spectral images to a small dimension such that \( n_1 = 306, n_2 = 402, \) and \( n_3 = 33 \). Empirically, we treat these graphs as tensors of rank \( r = (30, 30, 6) \). The observed pixels, or the training set, are sampled from the tensors uniformly at random. And the sample size is set to \(|\Omega| = OS \times p \) in which \( OS \) is so-called Over-Sampling ratio and \( p = \sum_{j=1}^{3} (n_j r_j - r_j^2) + r_1 r_2 r_3 \) is the number of free parameters in a size \( n \) tensor with rank \( r \). In the observed entries, the mode-1 feature matrix is constructed by extracting the top-(\( r_1 + 10 \)) singular vectors from a matrix of size \( n_1 \times 10r_1 \) whose columns are sampled from the mode-1 fibers of the hyperspectral graphs. The recovery accuracy is measured by Normalized Root mean Square Error (NRSE) [Kressner et al., 2014]. All the compared methods are terminated when the training NRSE is less than 0.003 or iterate more than 300 epochs. We report the NRSE and CPU time of the compared methods in Tab. 2. From the table, we can see that the proposed method has much higher accuracy than the other solvers in our baseline. The empirical results also indicate that the sparser the observed pixels are the higher CGSI’s improvement is on the recovery accuracy. The visual results of the 27th slices of recovered hyperspectral images of scene 7 are illustrated in Fig. 2.

5.2 Recommender System

In recommendation tasks, two datasets are considered: MovieLens 10M (ML10M) and MovieLens 20M (ML20M). Both datasets contain the rating history of users for items at specific moments. For both datasets, we partition the samples into 731 slices in terms of time stamp. Those slices have the identical time intervals. Accordingly, the completion tasks for the two datasets are of sizes 71567 \( \times 10681 \times 731 \) and 138493 \( \times 26744 \times 731 \) respectively. In addition to the rating history, both datasets contain two extra files: one describes the genres of each movie, and the other contains tags of each
movie. We construct a corpus that contains the text description of all movies from the genres descriptions and all the tags. The feature matrix is extracted from the above corpus by the latent semantic analysis (LSA) method. The processing is efficient since LSA is implemented via randomized SVD.

Various empirical studies are conducted to validate the performance of the proposed method. In the first scenario, we record the CPU time and the Root Mean Square Error (RMSE) outputted by the compared algorithms under different choices of multi-linear rank. In this scenario, for both datasets, 80% samples are chosen as training set, and the rest are left for testing. The results are listed in Tab. 3, which suggests that the proposed method outperforms all other solvers in terms of accuracy. For ML10M, our method uses significantly less CPU time than its competitors. In Fig. 3, we report another scenario, in which the percentage of training samples are varied from 10% to 70% and the rank parameter is fixed to (10, 10, 10). Experimental results in this figure indicate that our method has the lowest RMSE.

To show the impact of parameter $\alpha$ on the performance of our method, we depict the relation between RMSE and $\alpha$ in Fig. 4, where the rank parameter is set to (10, 10, 10), and percentage of training samples is set to 80%. From this Figure we can see that our method has higher accuracy than the vanilla Riemannian model’s solver FTC for a wide range of parameter choices.

### 6 Conclusion

In this paper, we exploit the side information to improve the accuracy of Riemannian tensor completion. A novel Riemannian model is proposed. To solve the model efficiently, we design a new Riemannian metric that implicitly induce an adaptive preconditioner for the solving procedure. Then, we devise a Riemannian conjugate gradient descent method using the well-tuned metric. Empirical results show that our solver outperforms the state-of-the-art.

### Acknowledgments

This work is partially supported by National Natural Science Foundation of China (Grant No: 61472347, 61672376, and 61672449).

### References


