Image/Video Inpainting

TMac: Parallel matrix factorization for low-rank tensor completion

Yangyang XU
(https://xu-yangyang.github.io)

June 6, 2023

Inverse Problems and Imaging doi:10.3934/ipi.2015.9.601
Volume 9, No. 2, 2015, 601–624

PARALLEL MATRIX FACTORIZATION FOR LOW-RANK TENSOR COMPLETION

Yangyang Xu
Department of Computational and Applied Mathematics
Rice University
Houston, TX 77005, USA

Ruru Hao
School of Mathematical Sciences
Dalian University of Technology, Dalian, China

Wotao Yin
Department of Mathematics, University of California
Los Angeles, CA 90095, USA

Zhixun Su
School of Mathematical Sciences
Dalian University of Technology, Dalian, China

(Communicated by Weihong Guo)

Abstract. Higher-order low-rank tensors naturally arise in many applications including hyperspectral data recovery, video inpainting, seismic data reconstruction, and so on. We propose a new model to recover a low-rank tensor by simultaneously performing low-rank matrix factorizations to the all-mode matricizations of the underlying tensor. An alternating minimization algorithm is applied to solve the model, along with two adaptive rank-adjusting strategies when the exact rank is not known.

Phase transition plots reveal that our algorithm can recover a variety of synthetic low-rank tensors from significantly fewer samples than the compared methods, which include a matrix completion method applied to tensor recovery and two state-of-the-art tensor completion methods. Further tests on real-world data show similar advantages. Although our model is non-convex, our algorithm performs consistently throughout the tests and gives better results than the compared methods, some of which are based on convex models. In addition, subsequence convergence of our algorithm can be established in the sense that any limit point of the iterates satisfies the KKT condtions.

Low-rank tensor completion

Low-rank tensor completion (LRTC)[1]:

- **[Significance]** Tensor can better exploit the structural information of higher-order data, such as 3^{rd} order image and 4^{th} order video.
- **[General idea]** Global low-rankness (rank minimization or low-rank approximation), local similarity, and nonlocal self-similarity.
- **[Fact]** Tensor rank is not unique and optimize rank is NP-hard!
- **[Technical goal]** Develop reliable models and efficient algorithms for recovering as much information as possible from the given data.
- **[Model assumption]** Mathematically, the LRTC problem can be written as

$$\underset{\mathcal{X}}{\text{minimize}}\ R(\mathcal{X}) \quad \text{s.t. } \mathcal{P}_{\Omega}(\mathcal{X}) = \mathcal{T}, \tag{1}$$

$$\underset{\mathcal{X}}{\text{minimize}}\ L(\hat{\mathcal{X}}, \mathcal{T}) + \lambda R(\mathcal{X}). \tag{2}$$

[1] recovery of higher-order tensors that are (exactly or approximately) low-rank and have missing entries. https://zhaoxile.github.io.

Related works

Matrix completion (MC)

TMac can be regarded as an extension from MC to TC

$$\underset{\mathbf{U},\mathbf{V}}{\text{minimize}} \ \frac{1}{2}\|\mathbf{U}\mathbf{V}-\mathbf{Z}\|_F^2, \ \text{s.t.}, \ \mathcal{P}_\Omega(\mathbf{Z}) = \mathbf{T} \tag{3}$$

which is solved by LMaFit [a]

[a] Wen et. al. 2012, https://doi.org/10.1007/s12532-012-0044-1

Nuclear norm minimization

Matrix nuclear norm is the convex envelope of matrix rank function

$$\underset{\mathcal{X}}{\text{minimize}} \ \sum_{n=1}^{N} \alpha_n \left\|\mathbf{X}_{(n)}\right\|_*, \text{s.t.}, \ \mathcal{P}_\Omega(\mathcal{X}) = \mathcal{T} \tag{4}$$

which is solved by BCD, PAM, and ADMM [a] (convex model).

[a] HaLRTC, Liu et. al. 2013

Motivations

Recovering the missing entries from a under-sampled tensor has presented various theoretical and computational challenges.

- Matrix-based methods utilize only one mode low-rankness and destroy the multi-dimensional structure of the underlying tensor.
- Mode-n unfolding methods (NNM) involve the singular value decomposition (SVD) with high computational complexity.

How to do?

TMac applies low-rank matrix factorization to each mode unfolding of the tensor in order to enforce low-rankness and update the matrix factors alternatively, which is computationally much cheaper than NNM.

TMac[2]

Problem formulation: LRTC aim at recovering a low-rank tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ from partial observations $\mathcal{T} = \mathcal{P}_\Omega(\mathcal{X})$, where Ω is the index set of observed entries, and $\mathcal{P}_\Omega$ keeps the entries in Ω.

- TMac applies low-rank matrix factorization to $\mathbf{X}_{(n)}$:

$$\mathbf{U}_n \in \mathbb{R}^{I_n \times r_n}, \mathbf{V}_n \in \mathbb{R}^{r_n \times \Pi_{j \neq n} I_j}, \text{ s.t., } \mathbf{X}_{(n)} \approx \mathbf{U}_n \mathbf{V}_n, \tag{5}$$

  where r_n is the estimated rank, either fixed or adaptively updated.

- Introducing auxiliary variable $\mathcal{Z}$ to solve $\mathcal{X}$

$$\underset{\mathbf{U},\mathbf{V},\mathcal{Z}}{\text{minimize}} \sum_{n=1}^{N} \frac{\alpha_n}{2} \left\| \mathbf{U}_n \mathbf{V}_n - \mathbf{Z}_{(n)} \right\|_F^2, \text{ subject to } \mathcal{P}_\Omega(\mathcal{Z}) = \mathcal{T}, \tag{6}$$

  where $\mathbf{U} = (\mathbf{U}_1, \dots, \mathbf{U}_N)$ and $\mathbf{V} = (\mathbf{V}_1, \dots, \mathbf{V}_N)$, $\sum_n \alpha_n = 1$.

[2] Xu et. al. 2015 10.3934/ipi.2015.9.601

Contributions

All-mode matricizations

Utilizing all mode low-ranknesses of the tensor gives much better performance [a] and is computationally much cheaper than SVD [b].

[a] HaLRTC, Liu et. al., 2013
[b] Numerical results

Cyclic block minimization

TMac is non-convex, subsequence convergence can be established that any limit point of the iterates satisfies the KKT condtions.

Rank adaptive adjustment

Given the overestimated ranks [a] and then decreases them by checking the singular values [b] of the factor matrices in each mode.

[a] or underestimated ranks with increment
[b] consecutive large gap

Algorithm

TMac applies the alternating least squares method to (6).

- Model (6) is convex with respect to each block of the variables.

Algorithm 1: Low-rank **T**ensor Completion by Parallel **M**atrix **Fac**torization (TMac)

Input: Ω, $\boldsymbol{\mathcal{B}} = \mathcal{P}_\Omega(\boldsymbol{\mathcal{M}})$, and $\alpha_n \geq 0, n = 1, \ldots, N$ with $\sum_{n=1}^N \alpha_n = 1$.
Parameters: $r_n, \Delta r_n, r_n^{\max}, \xi_n, n = 1, \ldots, N$.
Initialization: $(\mathbf{X}^0, \mathbf{Y}^0, \boldsymbol{\mathcal{Z}}^0)$ with $\mathcal{P}_\Omega(\boldsymbol{\mathcal{Z}}^0) = \boldsymbol{\mathcal{B}}$.
for $k = 0, 1, \ldots$ **do**
  $\mathbf{X}^{k+1} \leftarrow$ (11), $\mathbf{Y}^{k+1} \leftarrow$ (10b), and $\boldsymbol{\mathcal{Z}}^{k+1} \leftarrow$ (10c).
  if *stopping criterion is satisfied* **then**
    Output $(\mathbf{X}^{k+1}, \mathbf{Y}^{k+1}, \boldsymbol{\mathcal{Z}}^{k+1})$.
  for $n = 1, \ldots, N$ **do**
    if $\xi_n = -1$ **then**
      Apply rank-decreasing scheme to $\mathbf{X}_n^{k+1}$ and $\mathbf{Y}_n^{k+1}$ in section 3.2.1.
    else if $\xi_n = 1$ **then**
      Apply rank-increasing scheme to $\mathbf{X}_n^{k+1}$ and $\mathbf{Y}_n^{k+1}$ in section 3.2.2.

- TMac uses $\sum_{n=1}^N \alpha_n \textbf{fold}_n\left(\mathbf{U}_n \mathbf{V}_n\right)$ to estimate the tensor $\boldsymbol{\mathcal{X}}$, which is usually better than $\mathcal{Z}$ when the underlying $\boldsymbol{\mathcal{X}}$ is only approximately low-rank or the observations are contaminated by noise, or both.

Rank schemes

Rank-decreasing scheme for exactly low-rank tensors

TMac calculates the eigenvalues of $\mathbf{U}_n^\top \mathbf{U}_n$ after each iteration, and sorted as $\lambda_1^n \geq \lambda_2^n \geq \ldots \geq \lambda_{r_n}^n$. Then compute the quotients $\bar{\lambda}_i^n = \lambda_i^n / \lambda_{i+1}^n, i = 1, \ldots, r_n - 1$. Suppose $\hat{r}_n = \underset{1 \leq i \leq r_n - 1}{\operatorname{argmax}} \bar{\lambda}_i^n$, if $\operatorname{gap}_n = \frac{(r_n - 1)\bar{\lambda}_{\hat{r}_n}^n}{\sum_{i \neq \hat{r}_n} \bar{\lambda}_i^n} \geq 10$, which means a "big" gap between $\lambda_{\hat{r}_n}^n$ and $\lambda_{\hat{r}_n+1}^n$, then reduces r_n to $\hat{r}_n$. The details have not been provided yet.

Rank-increasing scheme for general cases

Start an underestimated rank, i.e., $r_n \leq \operatorname{rank}_n(\mathcal{X})$. TMac increase r_n to $\min\left(r_n + \Delta r_n, r_n^{\max}\right)$ at iteration $k+1$ if

$$\left| 1 - \frac{\left\| \mathcal{T} - \mathcal{P}_\Omega \left(\mathbf{fold}_n \left(\mathbf{U}_n^{k+1} \mathbf{V}_n^{k+1} \right) \right) \right\|_F}{\left\| \mathcal{T} - \mathcal{P}_\Omega \left(\mathbf{fold}_n \left(\mathbf{U}_n^{k} \mathbf{V}_n^{k} \right) \right) \right\|_F} \right| \leq 10^{-2} \tag{7}$$

which means "slow" progress in the r_n dimensional space along the mode-n.

Convergence analysis

Theorem 1

Suppose $\left\{\left(\mathbf{U}^k, \mathbf{V}^k, \mathcal{Z}^k\right)\right\}$ is a sequence generated by Algorithm 1 with fixed r_n 's and fixed positive α_n 's. Then any limit point of Algorithm 1 satisfies the KKT conditions.

Numerical experiments

- Parameters setting and stopping rules

Parameters setting

Let $\mathbf{fit}_n\left(\mathbf{U}_n\mathbf{V}_n\right) = \left\|\mathcal{P}_\Omega\left(\mathbf{fold}_n\left(\mathbf{U}_n\mathbf{V}_n\right) - \mathcal{T}\right)\right\|_F$, TMac sets
$\alpha_n^k = \frac{\left[\mathbf{fit}_n\left(\mathbf{U}_n^k\mathbf{V}_n^k\right)\right]^{-1}}{\sum_{i=1}^N\left[\mathbf{fit}_i\left(\mathbf{U}_i^k\mathbf{V}_i^k\right)\right]^{-1}}, n = 1, \ldots, N$

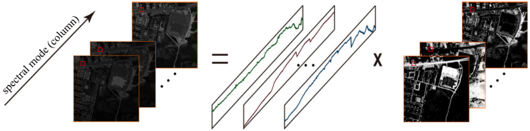
  - Dynamic updating α_n 's can improve the recovery quality for tensors that have better low-rankness in one mode than others.
  - The relative change of the overall fitting and verifies the weighted fitting is good enough.

- Numerical results: Synthetic data, MRI data, Hyperspectral data, and Video inpainting.

Extension: SMF-LRTC[3]

SMF-LRTC considers the low-rankness and the spatial piecewise smoothness priors of the underlying tensor.

- Hyperspectral unmixing



- Model formula

$$\underset{\mathcal{X},\mathbf{U},\mathbf{V}}{\text{minimize}} \sum_{n=1}^{N} \frac{\alpha_n}{2} \left\| \mathbf{X}_{(n)} - \mathbf{U}_n \mathbf{V}_n \right\|_F^2 + \lambda_1 \left\| \mathbf{W} \mathbf{V}_3^T \right\|_{1,1} + \lambda_2 \left\| \nabla_y \mathbf{U}_3 \right\|_{1,1}, \tag{8}$$

where λ_1 and λ_2 are regularization parameters, $\mathbf{W}$ denotes the framelet transformation matrix, ∇_y indicates the vertical derivative operator.

[3]Zhao et. al. 2019, https://doi.org/10.1016/j.apm.2019.02.001