Two Factors Update for Canonical Polyadic Decomposition

Anastasia Sozykina[1], Valentin Leplat[2], Igor Vorona[1], Salman Ahmadi-Asl[1] & Anh-Huy Phan[1]

[1] Center for Artificial Intelligence Technology, Skoltech, Moscow, Russia
[2] Innopolis University, Innopolis, Russia

1. Introduction

The current Canonical Polyadic (CP) tensor decomposition optimization algorithms face challenges like redundant steps and robustness issues with linearly dependent matrices, leading to slow progress.

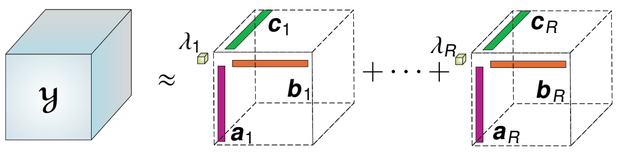

Figure 1: CPD model.

The ALS is the simplest and widely used algorithm to compute the CPD, however it suffers from two main drawbacks:

- ALS-type algorithm is high redundancy in computing the update rules.
- The condition numbers of linear systems in ALS steps are high, when several factor matrices have collinear loading components. Then the optimization process becomes inefficient.

2. Contribution

We develop algorithms to address these challenges. In a nutshell we propose:

- Unfolding of the CP model along two arbitrary modes.
- Updating two factor matrices instead of just one.
- Using property of two mode undolding to derive new update rules able to jointly update two factor matrices at once.

For example, consider a tensor $\mathcal{Y} = [\![\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\!]$, it two mode undolding is $\mathcal{Y}_{(1,2)} = [\![\mathbf{B}, \mathbf{A}_3]\!]$, where $\mathbf{B} = \mathbf{A}_2 \odot \mathbf{A}_1$. Then $\mathbf{A}_1$ and $\mathbf{A}_2$ can be retrieved through the best rank-1 approximation of the reshaped form of the columns of the matrix $\mathbf{B}$ as $\boldsymbol{b}_r = vec(\boldsymbol{a}_{1r}\boldsymbol{a}_{2r}^T)$.

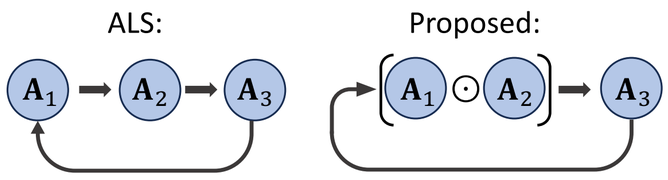

Figure 2: Scheme for updating factors in ALS and in our proposed algorithm.

Example 1. High collinear factor matrices

We considered order-3 and 4 tensors of size $I \times \cdots \times I$ with $I = \{10, 50\}$, and rank $R = \{10, 20\}$, whose all factors $\mathbf{A}_n$ have highly collinear, 97%-99%, loading components.
The proposed algorithm requires less number of update cycles than ALS, in order to obtain accurate solutions.

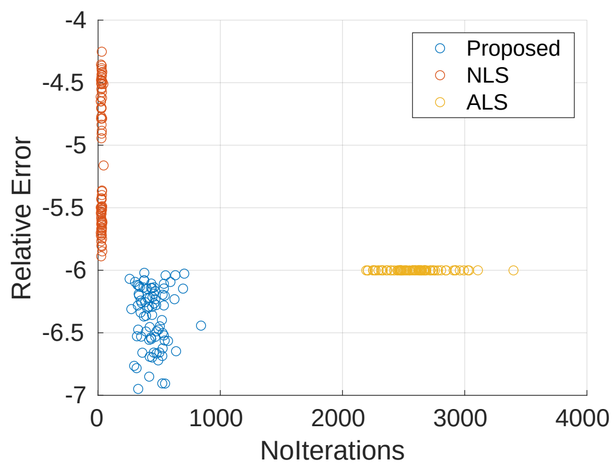

Figure 3: Comparison of relative errors vs number of iterations of algorithms.

Example 2. Ranks exceed dimensions

Third-order random tensors of size $10 \times 10 \times 10$ and rank $R = 25$ are randomly generated. Parameters are initialized by random numbers. For this scenario, the proposed algorithm attains a (nearly) perfect success ratio at the relative approximation error of 10^{-6}, while ALS succeeds in less than 30% of its runs. Both ALS and NLS get stuck in false local minima with relative errors greater than 10^{-2}.

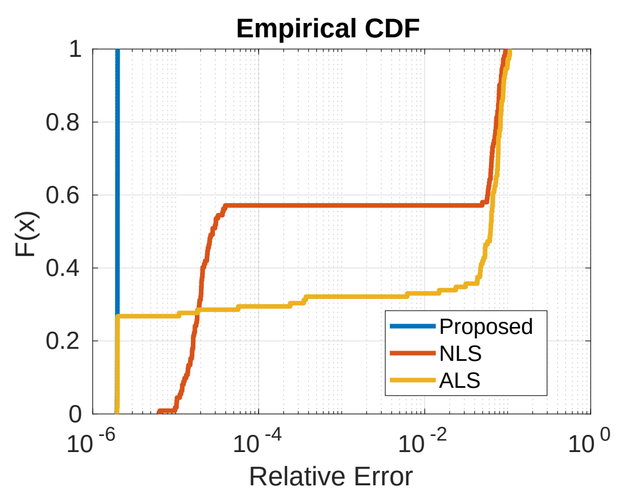

Figure 4: The probability distribution of achieving a specific relative error.

3. Linear Regression with Khatri-Rao structured matrix

The constrained linear regression problem is formulated as follows:

$$\min_{\mathbf{X}} \; f(\mathbf{X}) = \frac{1}{2}\|\mathbf{Y} - \mathbf{\Phi}\,\mathbf{X}^T\|_F^2 + \frac{\mu}{2}\|\mathbf{X}\|_F^2, \quad \text{s.t. } \mathbf{X} = \mathbf{V} \odot \mathbf{U}, \tag{1}$$

An indicator function, $i_{\mathcal{D}}(.)$ is defined for the set of Khatri-Rao structured matrices, $\mathcal{D} = \{\mathbf{X} | \mathbf{X} = \mathbf{U} \odot \mathbf{V}\}$ to simplify the optimization task. By introducing an additional variable $\mathbf{Z}$, the problem is formulated as

$$\min \quad f(\mathbf{Z}) + i_{\mathcal{D}}(\mathbf{X}), \quad \text{s.t.} \quad \mathbf{X} = \mathbf{Z}. \tag{2}$$

We solve the above optimization by Alternating Direction Method of Multipliers framework. The augmented Lagrangian function associated with the problem is given as

$$\mathcal{L}_\gamma(\mathbf{X}, \mathbf{Z}, \mathbf{T}) = f(\mathbf{Z}) + i_D(\mathbf{X}) + \frac{1}{2\gamma}\left(\|\mathbf{Z} - \mathbf{X} - \mathbf{T}\|_F^2 - \|\mathbf{T}\|_F^2\right).$$

The iterative updates for the primal variables $\mathbf{Z}$ and $\mathbf{X}$, and the dual variable, $\mathbf{T}$, are given by

$$\mathbf{Z}^{(k+1)} = \arg\min_{\mathbf{Z}} \; f(\mathbf{Z}) + \frac{1}{2\gamma}\|\mathbf{Z} - \mathbf{X}^{(k)} - \mathbf{T}^{(k)}\|_F^2, \tag{3}$$

$$\mathbf{X}^{(k+1)} = \Pi_{\mathcal{D}}(\mathbf{Z}^{(k+1)} - \mathbf{T}^{(k)}), \tag{4}$$

$$\mathbf{T}^{(k+1)} = \mathbf{T}^{(k)} + \mathbf{X}^{(k+1)} - \mathbf{Z}^{(k+1)}. \tag{5}$$

Update of Z. Solving the optimization problem in (3) amounts to minimize a quadratic function without constraints, the solution can be computed in closed-form expression as follows

$$\begin{aligned}\mathbf{Z}^{(k+1)} &= \arg\min_{\mathbf{Z}} \; \frac{1}{2}\|\mathbf{Y} - \mathbf{\Phi}\mathbf{Z}^T\|_F^2 + \frac{\mu}{2}\|\mathbf{Z}\|_F^2 + \frac{1}{2\gamma}\|\mathbf{Z} - \mathbf{X}^{(k)} - \mathbf{T}^{(k)}\|_F^2 \\ &= (\mathbf{Y}^T\mathbf{\Phi} + \frac{1}{\gamma}(\mathbf{X}^{(k)} + \mathbf{T}^{(k)}))(\mathbf{\Phi}^T\mathbf{\Phi} + (\mu + \frac{1}{\gamma})\mathbf{I})^{-1}.\end{aligned}$$

Algorithm 1 Linear Regression with Khatri-Rao structured Regressor

Input: Data matrix $\mathbf{Y}$: $K \times (IJ)$, $\mathbf{\Phi}$ of size $K \times R$, an initialization for $(\mathbf{X}, \mathbf{Z})$, a maximum number of iterations $k_{\max}$, and a threshold ϵ
Output: $\mathbf{X}$ is Khatri-Rao product such that it minimizes $\|\mathbf{Y} - \mathbf{\Phi}\mathbf{X}^T\|_F^2$
begin
  Initiate $\mathbf{T} = \mathbf{X} = 0$
  Precompute $\mathbf{W} = \mathbf{Y}^T\mathbf{\Phi}$, $\mathbf{Q} = \mathbf{\Phi}^T\mathbf{\Phi}$, $\tilde{\mathbf{Q}} = (\mathbf{Q} + (\mu + \frac{1}{\gamma})\mathbf{I})^{-1}$
  $k \leftarrow 1$
  while $k \leq k_{max}$ *and* $\frac{\|\mathbf{X}-\mathbf{Z}\|_F}{\|\mathbf{Z}\|_F} > \epsilon$ **do**
    $\mathbf{Z} \leftarrow (\mathbf{W} + \frac{1}{\gamma}(\mathbf{X} + \mathbf{T}))\tilde{\mathbf{Q}}$ `// Update` Z
    for $r = 1, \ldots, R$ **do** `// Update each column of` X
      $\mathbf{H}_r \leftarrow \texttt{reshape}(\boldsymbol{z}_r - \boldsymbol{t}_r, [\texttt{I} \times \texttt{J}])$
      $\mathbf{X}_r \leftarrow \text{vec}(\boldsymbol{u}_r s_r \boldsymbol{v}_r^T)$ where $[\boldsymbol{u}_r, s_r, \boldsymbol{v}_r] = \texttt{svds}(\mathbf{H}_r, 1)$
    $\mathbf{T} \leftarrow \mathbf{T} + \mathbf{X} - \mathbf{Z}$ `// Dual ascent step updates` T
    $k \leftarrow k + 1$

Update of X. By reshaping the vectors $\boldsymbol{z}_r^{(k+1)} - \boldsymbol{t}_r^{(k)}$ to matrices $\mathbf{H}_r$ of size $I \times J$, where $r = 1, 2, \ldots, R$. From (4) and by definition of the Khatri-Rao product each column of the matrix, $\mathbf{X}^{(k+1)}$, corresponds to vectorization of the best rank-1 approximation of the matrices $\mathbf{H}_r \approx \boldsymbol{u}_r\boldsymbol{v}_r^T$. This approximation has a unique optimal solution which can be computed in closed-form by using the truncated SVD.

Theorem 1. *For a sufficiently large $\beta = 1/\gamma$, the sequence $(\mathbf{X}^{(k)}, \mathbf{Z}^{(k)}, \mathbf{T}^{(k)})$ generated by Algorithm 1 applied to Problem (2) converges globally, that is regardless of where the initial point is, to the unique limit point $(\mathbf{X}^{(*)}, \mathbf{Z}^{(*)}, \mathbf{T}^{(*)})$, which is a stationary point of the augmented Lagrangian function, $\mathcal{L}_\gamma$, and $\mathbf{X}^{(*)}$ is a stationary point of Problem (2).*

4. Proposed algorithm for computing the CPD

The computation of the CPD of a tensor $\mathcal{Y}$ of order-N is achieved by minimizing the Frobenius norm of the error tensor

$$\min_{\{\mathbf{A}_n\}} \|\mathcal{Y} - [\![\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_N]\!]\|_F^2.$$

Classical algorithms like ALS update each factor sequentially. The ALS update for each factor involves tensor mode unfolding and optimization

$$\min_{\mathbf{A}_n} \|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{\Psi}_n^T\|_F^2,$$

where $\mathbf{\Psi}_n = \mathbf{A}_N \odot \cdots \odot \mathbf{A}_{n+1} \odot \mathbf{A}_{n-1} \odot \cdots \odot \mathbf{A}_1$. The ALS update for $\mathbf{A}_n$ is given by $\mathbf{A}_n = \mathbf{Y}_{(n)}\mathbf{\Psi}_n\,(\mathbf{\Psi}_n^T\mathbf{\Psi}_n)^{-1}$.
Several key insights can be made:

- **High redundancy.** The above update is simple, but each update requires to compute the product $\mathbf{Y}_{(n)}\mathbf{\Psi}_n$, which is the most expensive step in the ALS algorithm.
- **Degeneracy and inaccurate update.** Since $\mathbf{\Psi}_n^T\mathbf{\Psi}_n = (\mathbf{A}_N^T\mathbf{A}_N) \circledast \cdots \circledast (\mathbf{A}_{n+1}^T\mathbf{A}_{n+1}) \circledast (\mathbf{A}_{n-1}^T\mathbf{A}_{n-1}) \circledast \cdots \circledast (\mathbf{A}_1^T\mathbf{A}_1)$ ($\circledast$ is Hadamart product), when the factor matrices consist of highly collinear loading components, the correlation matrix $\mathbf{\Psi}_n^T\mathbf{\Psi}_n$ becomes poorly conditioned, and the computation of its inverse is likely to be inaccurate.

Motivated by these observations, we propose a new algorithm to efficiently tackle redundancy by updating two factor matrices simultaneously. By unfolding the tensor along two arbitrary modes n and $m, (n < m)$, we redefine the optimization problem:

$$\min_{\mathbf{A}_m, \mathbf{A}_n} \|\mathbf{Y}_{(n,m)}^T - \mathbf{\Psi}_{n,m}(\mathbf{A}_m \odot \mathbf{A}_n)^T\|_F^2, \tag{6}$$

where $\mathbf{\Psi}_{n,m} = \mathbf{A}_N \odot \cdots \odot \mathbf{A}_{m+1} \odot \mathbf{A}_{m-1} \odot \cdots \odot \mathbf{A}_{n+1} \odot \mathbf{A}_{n-1} \odot \cdots \odot \mathbf{A}_1$ is the Khatri-Rao product of all but two matrices $\mathbf{A}_n$ and $\mathbf{A}_m$.
The minimization problem in (6) simplifies into a regression problem with a Khatri-Rao structured matrix as in (1). Our algorithm updates two consecutive factor matrices, e.g., $\mathbf{A}_1$ and $\mathbf{A}_2$ using Algorithm 1, then continues with pairs like $\mathbf{A}_j$ and $\mathbf{A}_{j+1}$ until all are updated (Algorithm 2). After updating $\mathbf{A}_1$ and $\mathbf{A}_2$, the algorithm shifts the tensor dimensions by 2 and updates the next pair. After updating all pairs of factor matrices, the algorithm randomly permutes the factor matrices.

Algorithm 2 CPD with Two Factors Update

Input: Data tensor $\mathcal{Y}$: $(I_1 \times I_2 \times \cdots \times I_N)$, and rank R
Output: $\hat{\mathcal{Y}} = [\![\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_N]\!]$ such that it minimizes $\|\mathcal{Y} - \hat{\mathcal{Y}}\|_F^2$
begin
  Initialize $\hat{\mathcal{Y}}$
  while *a stopping criterion is not met* **do**
    $\mathcal{B} \leftarrow \texttt{randompermutation}(\texttt{N})$
    $j \leftarrow 1$
    while $j \leq |\mathcal{B}| - 1$ **do**
      $(n, m) \leftarrow (\mathcal{B}(j), \mathcal{B}(j+1))$
      $\mathbf{Y}_{(n,m)} \leftarrow (n, m)$-unfolding of $\mathcal{Y}$
      $\mathbf{\Psi}_{n,m} \leftarrow \mathbf{A}_N \odot \cdots \odot \mathbf{A}_{m+1} \odot \mathbf{A}_{m-1} \odot \cdots \odot \mathbf{A}_{n+1} \odot \mathbf{A}_{n-1} \odot \cdots \odot \mathbf{A}_1$
      Solve $\min_{\mathbf{X}} \|\mathbf{Y}_{(n,m)}^T - \mathbf{\Psi}_{n,m}\mathbf{X}^T\|$ s.t. $\mathbf{X} = \mathbf{A}_m \odot \mathbf{A}_n$ using Algorithm 1
      $j \leftarrow j + 2$

Example 3. Multiplication Tensor

We decomposed tensors associated with multiplication of two matrices of size $(2 \times 3) \times (3 \times 2)$ and $(3 \times 3) \times (3 \times 3)$. The first tensor is of size $6 \times 6 \times 4$ and rank-11, and the second tensor of size $9 \times 9 \times 9$ and rank-23, both contain only zeros and ones, and obey Finding CPD of these tensors with minimal rank is related to seeking the fastest multiplication of two matrices. The proposed algorithm can explain the tensor with a relative error below 10^{-6} in around 1000 iterations.

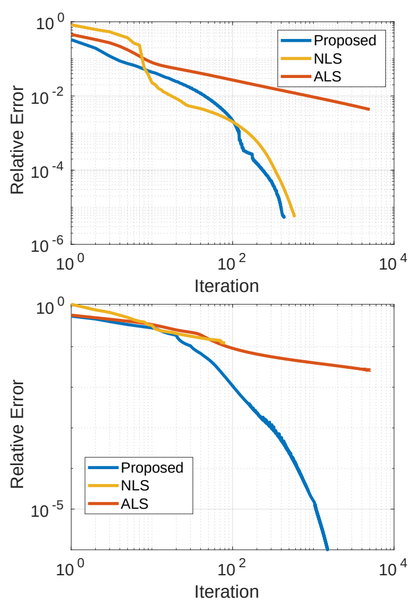

Figure 5: Convergence of algorithms.

Example 4. Compress Conv layers

The proposed algorithm in most of cases achieves better accuracy compared to the ALS. In addition, the proposed algorithm is more resistant to perturbations. These extensive simulations convinced us that the proposed algorithm can be efficiently utilized for compressing convolutional layer of deep neural networks.

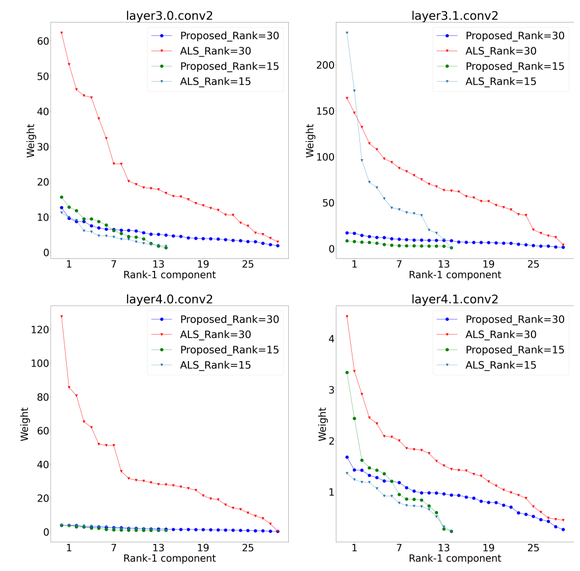

Figure 6: Norms of rank-1 tensors in the approx. of the conv layers weights with ranks 30 and 15.

5. Acknowledgements

This work was partially supported by the joint project Artificial Intelligence for Life (AIfoL) between the University of Sharjah and the Skolkovo Institute of Science and Technology.

6. Conclusions

Novel algorithm for CPD updates two factor matrices simultaneously to address the problem of ALS-type algorithms and instability issues. The extensive simulation results consistently demonstrated the superiority of our algorithm over both the ALS and NLS algorithms across various scenarios.