ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 3, No. 4, 2008, pp. 269-280

Some properties of matrix product and its applications in nonnegative tensor decomposition

Hongli Yang [1, 2 +], Guoping He [1]

[1] Information Science and Engineering College of Shandong University of Science and Technology
[2] Science College of Shandong University of Science and Technology

(*Received April 14, 2008, accepted August 14, 2008*)

Abstract. Some prosperities of matrix product are presented in the paper, Kronecker product, Khatri-Rao product, Hadamard product and outer product are involved. And we get some results that a multilinear tensor can be represented by the product of matrix product for a three order tensor. For higher tensor, we conjure that the same results also hold. By the representation of matrix, we give an iterative algorithm for nonnegative tensor decomposition method which has good convergence performance. Numerical experiments show that our algorithm is effective.

Keywords: matrix product nonnegative tensor decomposition, algorithm

1. Introduction of Matrix Product and Some Properties

1.1. The Definitions of Matrices Product

(1) Kronecker Product of Matrix

Given a matrix $A \in R^{I\times J}, B \in R^{K\times L}$, we denote the Kronecker product of matrix as "$\otimes$", it's a matrix of IK rows and JL columns,

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{I1}B & a_{I2}B & \cdots & a_{IJ}B \end{pmatrix},$$

(2) Khatri-Rao Product of Matrix

Given a matrix $A \in R^{I\times K}, B \in R^{J\times K}$, we denote the Khatri-Rao product of matrix as "$\odot$", it's a matrix with IJ rows and K columns, it can be rewritten by the Kronecker product as the following,

$$A \odot B = (a_1 \otimes b_1, a_2 \otimes b_2, \cdots, a_K \otimes b_K),$$

where $a_1, a_2, \cdots, a_K, b_1, b_2, \cdots, b_K$ are the columns of matrixes A and B.

(3) Hadamard Product of Matrix

Supposed matrix $A \in R^{I\times K}, B \in R^{I\times K}$, then we denote the Hadamard product of matrix with "$\oplus$",

The result of product is

$$A \oplus B = (a_{i,j} b_{i,j})_{I\times K}.$$

(4) Outer Product of Matrix

If matrix $X \in R^m, Y \in R^n$, we denote it as "$\circ$", it is a matrix with $m\times n$ blocks and the entries at the

+ Corresponding author. Tel.: +86-0532-86057603.
E-mail address: ylmath@yahoo.com.cn.
This work is supported by the NSF grant 10571109.

place of (i, j) is $a_{ij} = x_i y_j$,

$$X \circ Y = A_{m\times n} = (a_{ij})_{m\times n} .$$

1.2. Some Properties of Matrix Product

For matrix product, some basic properties are list here and it's obviously predictable,

(1) Associative

$$A\otimes(B\otimes C) = (A\otimes B)\otimes C, A\odot(B\odot C) = (A\odot B)\odot C ,$$

(2) Distributive

$$(A+B)\otimes C = A\otimes C + B\otimes C, (A+B)\odot C = A\odot C + B\odot C ,$$

(3) Transposition of matrix

If matrix A, B are symmetric matrix, they have the following basic properties

$$\left(A\otimes B\right)^T = A^T \otimes B^T .$$

For more properties of matrix product, one can refer [2] [5], in the following, we will present some obvious properties in the case of matrix are column matrix.

If matrix $X \in R^m, Y \in R^n$, then we have the following prosperities,

(1) $X \circ Y = Y^T \otimes X = X \otimes Y^T$;

(2) $X \otimes Y = X \odot Y$;

(3) $\left(X \otimes Y\right)^T = X^T \odot Y^T$.

Proof: If $X = \left(x_1, x_2, \cdots, x_m\right)^T, Y = \left(y_1, y_2, \cdots, y_n\right)^T$, then $X \circ Y = \left(x_i y_j\right)_{mn}$, and on the other hand,

$$Y^T \otimes X = \left(y_1, y_2, \cdots, y_n\right)\otimes\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} = \left(y_1 X \quad , y_2 X, \quad \cdots \quad , y_n X\right) = (x_i y_j)_{mn}, \text{ and}$$

$$X \otimes Y^T = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}\otimes\left(y_1 \quad y_2 \quad \cdots \quad y_n\right) = \begin{pmatrix} x_1 Y \\ x_2 Y \\ \vdots \\ x_m Y \end{pmatrix} = \left(x_i y_j\right)_{mn} ,$$

The entries in the place $(i, j)(i = 1, 2, \cdots, m; j = 1, 2, \cdots, n)$ are equal, and then conclusion (1) holds.

(2) It's obviously that (2) holds for the reason that

$$X \otimes Y = \begin{pmatrix} x_1 Y \\ x_2 Y \\ \vdots \\ x_m Y \end{pmatrix} = X \odot Y ;$$

(3) If X, Y are the vectors above, by the definition of matrix product, we have the conclusion that

$$\begin{aligned}(X \otimes Y)^T &= \left(x_1 Y^T, \quad x_2 Y^T, \quad \cdots \quad , x_m Y^T\right)^T \\ &= \left(x_1 y_1, x_1 y_2, \cdots x_1 y_n, x_2 y_1, x_2 y_2, \cdots, x_2 y_n, \cdots, x_m y_1, x_m y_2, \cdots, x_m y_n\right)'\end{aligned}$$

On the other hand,

$$X^T \odot Y^T = (x_1, x_2, \cdots, x_m) \odot (y_1, y_2, \cdots, y_n)$$
$$= (x_1 y_1, x_1 y_2, \cdots x_1 y_n, x_2 y_1, x_2 y_2, \cdots, x_2 y_n, \cdots, x_m y_1, x_m y_2, \cdots, x_m y_n),$$

Then we complete our proofs.

2. Brief Introduction to Multi-linear Algebra

Tensor algebra is a research branch of multi-linear algebra. In recent years, it has achieved much improvements for the widely applications in the fields of signal processing, computer vision and graph, independent component analysis, Web data mining, and EEG (electroencephalogram) data analysis [7][8][9]. For more reference about application, one can refer [20] for more information. An N order tensor A is an array whose entries can be expressed by N indices,

$$A = \left(a_{i_1 i_2 \cdots i_N}\right)_{I_1 \times I_2 \times \cdots \times I_N},$$

where $i_1 \in I_1, i_2 \in I_2, \cdots, i_N \in I_N$ is a permutation of number group $1, 2, \cdots, N$. If $a_{i_1 i_2 \cdots i_N} \in R$, tensor A is a real tensor and if $a_{i_1 i_2 \cdots i_N} \in C$, then tensor A is a complex tensor. If all the entries of a tensor are nonnegative, then we call the tensor a nonnegative tensor.

With the growth of dimensions of a tensor, the storage volume increase from $I_1 \times I_2$ to $I_1 \times I_2 \times I_3 \times \cdots \times I_N$, and the computation costs also increase rapidly. Then how to find the information we want from the large volume datum is the vital problem. Many techniques have been available for solving the problem. CP model and Tucker model are the popular models. We present an algorithm for the decomposition of CP model in this paper.

It is convenient to be able to represent tensors as matrixes, there are multiple ways to order the columns, and in the following, we first give a definition of tensor matricization which was firstly used in [3]. For a given tensor $A \in R^{I_1 \times I_2 \times \cdots \times I_N}$, we use $A_{(1)}, A_{(2)}, \cdots, A_{(N)}$ to express the matricization matrix, and $A_{(n)} (n = 1, 2, \cdots, N)$ is a $I_n \times (I_1 I_2 \cdots I_{n-1} I_{n+1} I_{n+2} \cdots I_N)$ block matrix, the entries of tensor A in the place of $(i_1, i_2, \cdots, i_N)$ map to the place of (i_n, j) in $A_{(n)} (n = 1, 2, \cdots, N)$, where

$$(i_{n+1} - 1) I_{n+2} I_{n+3} \cdots I_N I_1 I_2 \cdots I_{n-1} + (i_{n+2} - 1) I_{n+3} I_{n+4} \cdots I_N I_1 I_2 \cdots I_{n-1} + \cdots +$$
$$(i_N - 1) I_1 I_2 \cdots I_{n-1} + (i_1 - 1) I_2 I_3 \cdots I_{n-1} + (i_2 - 1) I_3 I_4 \cdots I_{n-1} + \cdots + i_{n-1},$$

for more details of tensor matricization, one can refer [3]. We illustrate with tensor $A \in R^{3 \times 4 \times 2}$, we use "slice" matrix to express the tensor in the following

$$A_1 = \begin{pmatrix} 5 & 7 & 3 & 1 \\ 2 & 9 & 2 & 6 \\ 1 & 3 & 5 & 7 \end{pmatrix}, A_2 = \begin{pmatrix} 4 & 2 & 1 & 3 \\ 7 & 3 & 2 & 1 \\ 7 & 5 & 6 & 1 \end{pmatrix},$$

and the matricization matrices are

$$A_{(1)} = \begin{pmatrix} 5 & 4 & 7 & 2 & 3 & 1 & 1 & 3 \\ 2 & 7 & 9 & 3 & 2 & 2 & 6 & 1 \\ 1 & 7 & 3 & 5 & 5 & 6 & 7 & 1 \end{pmatrix}, A_{(2)} = \begin{pmatrix} 5 & 2 & 1 & 4 & 7 & 7 \\ 7 & 9 & 3 & 2 & 3 & 5 \\ 3 & 2 & 5 & 1 & 2 & 6 \\ 1 & 6 & 7 & 3 & 1 & 1 \end{pmatrix},$$

$$A_{(3)} = \begin{pmatrix} 5 & 7 & 3 & 1 & 2 & 9 & 2 & 6 & 1 & 3 & 5 & 7 \\ 4 & 2 & 1 & 3 & 7 & 3 & 2 & 1 & 7 & 5 & 6 & 1 \end{pmatrix}.$$

3. Tensor Decomposition Model and the Main Results

The method of lower rank decomposition of a high dimension data is widely used in the field of signal processing, computer vision and graph, and independent component analysis. The CP (Canonical PARAFAC) decomposition model is a popular model. For a given N order tensor $A \in R^{I_1 \times I_2 \times \cdots \times I_N}$, the CP decomposition model is to the following form decomposition,

$$A = \sum_{r=1}^{R} A_{:,r}^{(1)} \circ A_{:,r}^{(2)} \circ \cdots \circ A_{:,r}^{(N)},$$

where $A_{:,r}^{(n)} (n = 1, 2, \cdots, N, r = 1, 2, \cdots, R)$ denote the r th column of the n th factorization matrix. Find the tensor decomposition is equivalent to the following optimization problem,

$$\min \left\| a_{i_1 i_2 \cdots i_N} - \sum_{r=1}^{R} A_{i_1,r}^{(1)} \circ A_{i_2,r}^{(2)} \circ \cdots \circ A_{i_N,r}^{(N)} \right\|_F^2, \tag{1}$$

Also we denote the Frobinus norm of tensor X with $\|X\|_F = \sqrt{\sum_{i_1, i_2, \cdots, i_N} x_{i_1 i_2 \cdots i_N}^2}$.

There are several tensor decompositions methods available, including Higher-Order Singular Value Decomposition (HOSVD) [3], Higher-Order Orthogonal Iteration (HOOI) [4], and Slice Projection (SP) [18]. For symmetric tensor, there are also some algorithms available such as [13], and for nonnegative tensor decomposition, [7] [8] [9] [15] present some effective algorithms.

For a three order tensor, we have the following observations.

Theorem 1 For a three order tensor $A \in R^{I_1 \times I_2 \times I_3}$, $A_{(n)} (n = 1, 2, 3)$, is the matricization matrix and $A^{(1)}, A^{(2)}, A^{(3)}$ is the matrix in the CP model, the following conclusions hold,

$$A_{(1)} = A^{(1)} (A^{(2)} \odot A^{(3)})^T,$$
$$A_{(2)} = A^{(2)} (A^{(3)} \odot A^{(1)})^T,$$
$$A_{(3)} = A^{(3)} (A^{(1)} \odot A^{(2)})^T.$$

Proof: According to the definition of tensor matricization and the definition of matrix Khatri-Rao product, the matrix $A_{(1)}$ is a matrix of $I_1 \times I_2 I_3$ block, the entries in the place of (i_1, j) is the entries $a_{i_1 i_2 i_3}$ in place of (i_1, i_2, i_3) of tensor A, where

$$j = (i_{n+1} - 1) I_{n+2} I_{n+3} \cdots I_N I_1 I_2 \cdots I_{n-1} + (i_{n+2} - 1) I_{n+3} I_{n+4} \cdots I_N I_1 I_2 \cdots I_{n-1}$$
$$+ \cdots + (i_N - 1) I_1 I_2 \cdots I_{n-1} + (i_1 - 1) I_2 I_3 \cdots I_{n-1} + (i_2 - 1) I_3 I_4 \cdots I_{n-1} + \cdots + i_{n-1},$$

In order to prove the conclusions, we only need proving that the entries in the place of (i_1, j) of matrix $A^{(1)} (A^{(2)} \odot A^{(3)})^T$ is the entries $a_{i_1 i_2 i_3}$ in tensor A, $A^{(2)}$ is a matrix of the form $I_2 \times R$, and $A^{(3)}$ is a matrix of form $I_3 \times R$, $A^{(2)} \odot A^{(3)}$ is a matrix of form $I_2 I_3 \times R$, and matrix $A^{(1)} (A^{(2)} \odot A^{(3)})^T$ is a matrix of form $I_1 \times I_2 I_3$, it's the same as that of $A_{(1)}$, by the properties of matrix product, we can rewrite $A^{(1)} (A^{(2)} \odot A^{(3)})^T$ as the following,

$$\left(A^{(2)} \odot A^{(3)} \right)^T = \left(A_{:,1}^{(2)} \otimes A_{:,1}^{(3)}, A_{:,2}^{(2)} \otimes A_{:,2}^{(3)}, \cdots, A_{:,R}^{(2)} \otimes A_{:,R}^{(3)} \right)^T,$$

According to the properties we have derivate in section 1, we have the following equality

$$= \begin{pmatrix} \left(A_{:,1}^{(2)} \otimes A_{:,1}^{(3)} \right)^T \\ \left(A_{:,2}^{(2)} \otimes A_{:,2}^{(3)} \right)^T \\ \vdots \\ \left(A_{:,R}^{(2)} \otimes A_{:,R}^{(3)} \right)^T \end{pmatrix} = \begin{pmatrix} (A_{:,1}^{(2)})^T \odot (A_{:,1}^{(3)})^T \\ (A_{:,2}^{(2)})^T \odot (A_{:,2}^{(3)})^T \\ \vdots \\ (A_{:,R}^{(2)})^T \odot (A_{:,R}^{(3)})^T \end{pmatrix},$$

$$\begin{pmatrix} A^{(2)}_{1,1}A^{(3)}_{1,1} & A^{(2)}_{1,1}A^{(3)}_{2,1} & \cdots & A^{(2)}_{1,1}A^{(3)}_{I_3,1} & A^{(2)}_{2,1}A^{(3)}_{1,1} & A^{(2)}_{2,1}A^{(3)}_{2,1} & \cdots & A^{(2)}_{2,1}A^{(3)}_{I_3,1} & \cdots & A^{(2)}_{I_2,1}A^{(3)}_{1,1} & A^{(2)}_{I_2,1}A^{(3)}_{2,1} & \cdots & A^{(2)}_{I_2,1}A^{(3)}_{I_3,1} \\ A^{(2)}_{1,2}A^{(3)}_{1,2} & A^{(2)}_{1,2}A^{(3)}_{2,1} & \cdots & A^{(2)}_{1,2}A^{(3)}_{I_3,2} & A^{(2)}_{2,2}A^{(3)}_{1,2} & A^{(2)}_{2,2}A^{(3)}_{2,2} & \cdots & A^{(2)}_{2,2}A^{(3)}_{I_3,2} & \cdots & A^{(2)}_{I_2,2}A^{(3)}_{1,2} & A^{(2)}_{I_2,2}A^{(3)}_{1,2} & \cdots & A^{(2)}_{I_2,2}A^{(3)}_{I_3,2} \\ & & & & & & \vdots & & & & & & \\ A^{(2)}_{1,R}A^{(3)}_{1,R} & A^{(2)}_{1,R}A^{(3)}_{2,R} & \cdots & A^{(2)}_{1,R}A^{(3)}_{I_3,R} & A^{(2)}_{2,R}A^{(3)}_{1,R} & A^{(2)}_{2,R}A^{(3)}_{2,R} & \cdots & A^{(2)}_{2,R}A^{(3)}_{I_3,R} & \cdots & A^{(2)}_{I_2,R}A^{(3)}_{1,R} & A^{(2)}_{I_2,R}A^{(3)}_{2,R} & \cdots & A^{(2)}_{I_2,R}A^{(3)}_{I_3,R} \end{pmatrix}$$

The i_1 th row entries of matrix $A^{(1)}$ is

$$A^{(1)}_{i_1,:} = \left(A^{(1)}_{i_1,1}, A^{(1)}_{i_1,2}, \cdots, A^{(1)}_{i_1,R} \right)_{1\times R},$$

The j th column entries of matrix $(A^{(2)} \odot A^{(3)})^T$ is

$$\left(A^{(2)}_{i_3,1}A^{(3)}_{i_2,1}, A^{(2)}_{i_3,2}A^{(3)}_{i_2,2}, \cdots, A^{(2)}_{i_3,R}A^{(3)}_{i_2,R} \right)^T,$$

where $j = (i_2 - 1)I_3 + i_3$, hence the entries of matrix $A^{(1)}(A^{(2)} \odot A^{(3)})^T$ in the place of (i_1, j) is

$$\sum_{r=1}^{R} A^{(1)}_{i_1,r}A^{(3)}_{i_3,r}A^{(2)}_{i_2,r},$$

This entries equals the entries of tensor A in the place of (i_1, i_2, i_3), so we completed our proof, carry out the same procedure ,we can complete our proofs of conclusion (2) and (3).

Theorem 2 Given a three order symmetric tensor $A \in R^{I_1\times I_2\times I_3}$, $A_{(n)} (n = 1,2,3)$ is the matricization matrix and $A^{(1)}, A^{(2)}, A^{(3)}$ is the matrix in the CP model, the following conclusions hold,

$$A_{(1)} = A^{(1)}(A^{(2)} \odot A^{(3)})^T = CV(C \odot C)^T,$$
$$A_{(2)} = A^{(2)}(A^{(3)} \odot A^{(1)})^T = CV(C \odot C)^T,$$
$$A_{(3)} = A^{(3)}(A^{(1)} \odot A^{(2)})^T = CV(C \odot C)^T.$$

where V is a diagonal matrix.

Proof: Because tensor A is a symmetric tensor, then $I_1 = I_2 = I_3$, we have that $a_{\mu(i_1,i_2,i_3)} = a_{\sigma(i_1,i_2,i_3)}$, where $\sigma(i_1,i_2,i_3), \mu(i_1,i_2,i_3)$ is any permutation of number group i_1, i_2, i_3, for example, we have

$$a_{1,2,3} = a_{1,3,2} = a_{2,1,3} = a_{2,3,1} = a_{3,2,1} = a_{3,1,2}.$$

For the reason that $a_{i_1,i_2,i_3} = \sum_{r=1}^{R} A^{(1)}_{i_1,r}A^{(2)}_{i_2,r}A^{(3)}_{i_3,r}$, $a_{\mu(i_1,i_2,i_3)} = a_{\sigma(i_1,i_2,i_3)}$, the entries in the same place of matrix

$A^{(1)}, A^{(2)}, A^{(3)}$ equals, First we consider the case of $R = 1$, so we have the following form of decomposition

$$A = \begin{pmatrix} a^{(1)}_1 \\ a^{(1)}_2 \\ \vdots \\ a^{(1)}_{I_1} \end{pmatrix} \circ \begin{pmatrix} a^{(2)}_1 \\ a^{(2)}_2 \\ \vdots \\ a^{(2)}_{I_2} \end{pmatrix} \circ \begin{pmatrix} a^{(3)}_1 \\ a^{(3)}_2 \\ \vdots \\ a^{(3)}_{I_3} \end{pmatrix}$$

Because tensor A is a symmetric tensor, so we have $a_{\mu(i_1,i_2,i_3)} = a_{\sigma(i_1,i_2,i_3)}$, we fixed the first indices I_1 and we get that $a^{(1)}_{i_1}a^{(2)}_{i_2}a^{(3)}_{i_3} = a_{i_1i_2i_3} = a_{i_1i_3i_2} = a^{(1)}_{i_1}a^{(2)}_{i_3}a^{(3)}_{i_2}$ for any i_1, i_2, i_3, so we have the equality

$$a^{(2)}_{i_2}a^{(3)}_{i_3} = a^{(2)}_{i_3}a^{(3)}_{i_2},$$

And we change it into the following equality

$$\frac{a_{i_2}^{(2)}}{a_{i_2}^{(3)}} = \frac{a_{i_3}^{(2)}}{a_{i_3}^{(3)}},$$

which means that the entries in the second vector are proportion to the corresponding entries in the third vector, we can extract the coefficient λ ,so tensor A can be rewritten as

$$A = \lambda \begin{pmatrix} a_1^{(1)} \\ a_2^{(1)} \\ \vdots \\ a_{I_1}^{(1)} \end{pmatrix} \circ \begin{pmatrix} c_1^{(2)} \\ c_2^{(2)} \\ \vdots \\ c_{I_2}^{(2)} \end{pmatrix} \circ \begin{pmatrix} c_1^{(3)} \\ c_2^{(3)} \\ \vdots \\ c_{I_3}^{(3)} \end{pmatrix},$$

Carry out the same procedure, we fix the second indices i_2, we can get the entries in the first vector are proportion to the corresponding entries in the third vector, so we also extract the coefficient, we ca get the following equality

$$A = \lambda\mu \begin{pmatrix} c_1^{(1)} \\ c_2^{(1)} \\ \vdots \\ c_{I_1}^{(1)} \end{pmatrix} \circ \begin{pmatrix} c_1^{(2)} \\ c_2^{(2)} \\ \vdots \\ c_{I_2}^{(2)} \end{pmatrix} \circ \begin{pmatrix} c_1^{(3)} \\ c_2^{(3)} \\ \vdots \\ c_{I_3}^{(3)} \end{pmatrix},$$

So we can draw the conclusion that

$$A_{(1)} = k_1 C^{(1)} \left(C^{(1)} \odot C^{(1)} \right)^T$$
$$A_{(2)} = k_1 C^{(1)} \left(C^{(1)} \odot C^{(1)} \right)^T,$$
$$A_{(3)} = k_1 C^{(1)} \left(C^{(1)} \odot C^{(1)} \right)^T$$

For the case when $R = 2$, we denote another $\widehat{A} = A - k_1 C \left(C \odot C \right)^T$,which is also a symmetric tensor, we also can get the rank-1 decomposition of tensor $\widehat{A}$, then we have the equality

$$A = k_1 C^{(1)} \circ C^{(1)} \circ C^{(1)} + k_2 C^{(2)} \circ C^{(2)} \circ C^{(2)},$$

carry out the same procedure above, we have

$$A = k_1 C^{(1)} \circ C^{(1)} \circ C^{(1)} + k_2 C^{(2)} \circ C^{(2)} \circ C^{(2)} + \cdots + k_R C^{(R)} \circ C^{(R)} \circ C^{(R)},$$

and

$$A = \sum_{r=1}^{R} k_r C_{:,r} \circ C_{:,r} \circ C_{:,r},$$

Which has the same form as that of a tensor in [19], and we have

$$A_{(1)} = CV \left(C \odot C \right)^T$$
$$A_{(2)} = CV \left(C \odot C \right)^T$$
$$A_{(3)} = CV \left(C \odot C \right)^T$$

Where matrix V is a diagonal matrix with the diagonal elements are the factors. Therefore we have the conclusion that

$$A^{(1)} = A^{(2)} = A^{(3)} = C.$$

For a general higher order tensor and high order symmetric tensor, we have the following conjures:

4. Algorithm for Tensor Decomposition and Convergence Analysis

4.1. Model and Algorithm

For a nonnegative tensor, the nonnegative decomposition of tensor is to find the nonnegative matrix $A^{(1)}, A^{(2)}, A^{(3)}$, such that $A = \sum_{r=1}^{R} A^{(1)}_{\cdot,r} \circ A^{(2)}_{\cdot,r} \circ A^{(3)}_{\cdot,r}$, we have the decomposition model that

$$\begin{aligned} &\min \left\| a_{i_1 i_2 i_3} - \sum_{r=1}^{R} A^{(1)}_{i_1,r} \circ A^{(2)}_{i_2,r} \circ A^{(3)}_{i_3,r} \right\|_F^2 \\ &s.t. \qquad\qquad\qquad\qquad\qquad\qquad , \\ &A^{(n)} \geq 0, n = 1,2,3 \end{aligned} \tag{2}$$

In the real world models such as computer vision and graph, EEG data processing, the entries in the entries are all nonnegative. So it's a meaningful work to find the nonnegative tensor decomposition. According to the conclusions we achieved above, model (2) is equivalent to the following optimization problem

$$\begin{aligned} &\min \left\| A_{(n)} - A^{(n)} B^{(n)} \right\|_F^2 \\ &s.t. \\ &A^{(n)} \geq 0, B^{(n)} \geq 0; \\ &n = 1,2,3 \end{aligned} \tag{3}$$

where $B^{(n)} = A^{(n+1)} \odot A^{(n+2)} \odot \cdots \odot A^{(N)} \odot A^{(1)} \odot \cdots \odot A^{(n-1)}$.

Because of the widely applications in pattern recognition [7] [15], nonnegative tensor decomposition techniques have gained more achievements, there are a few effective algorithms available. In this paper, we present a algorithm for three order nonnegative tensor decomposition based on the nonnegative matrix factorization technique. The following can be diffused to the case of higher order tensor.

Algorithm

Step1 Given the value of R, error ε and the maximum iterates times k, initialize matrix $A^{(n)}, n = 1,2,3$ with the schemes we discuss the following;

Step2 Let $\widehat{A}^{(k)} = \left(A^{(1)}\right)^{(k)} \circ \left(A^{(2)}\right)^{(k)} \circ \left(A^{(3)}\right)^{(k)}$, if $\left\| A - \widehat{A}^{(k)} \right\|_F^2 < \varepsilon$, stop, otherwise, go to step3,

Step3 Iterate step 4 to step 5;

Step4 For $n = 1$, we have the following iteration

$$\left(B^{(1)}_{i,j}\right)^{(k+1)} \leftarrow \left(B^{(1)}_{i,j}\right)^{(k)} \frac{\left(\left(A^{(1)}\right)^T A_{(1)}\right)^{(k)}_{i,j}}{\left(\left(A^{(1)}\right)^T A^{(1)} B^{(1)}\right)^{(k)}_{i,j}}, \left(A^{(1)}_{i,j}\right)^{(k+1)} \leftarrow \left(A^{(1)}_{i,j}\right)^{(k)} \frac{\left(A_{(1)} \left(B^{(1)}\right)^T\right)^{(k)}_{i,j}}{\left(A^{(1)} B^{(1)} \left(B^{(1)}\right)^T\right)^{(k)}_{i,j}},$$

Step5 For $n = 2$, we have the following iteration, and matrix $A^{(1)}$ is the matrix we get in step4

$$\left(B^{(2)}_{i,j}\right)^{(k+1)} \leftarrow \left(B^{(2)}_{i,j}\right)^{(k)} \frac{\left(\left(A^{(2)}\right)^T A_{(2)}\right)^{(k)}_{i,j}}{\left(\left(A^{(2)}\right)^T A^{(2)} B^{(2)}\right)^{(k)}_{i,j}}, \left(A^{(2)}_{i,j}\right)^{(k+1)} \leftarrow \left(A^{(2)}_{i,j}\right)^{(k)} \frac{\left(A_{(2)} \left(B^{(2)}\right)^T\right)^{(k)}_{i,j}}{\left(A^{(2)} B^{(2)} \left(B^{(2)}\right)^T\right)^{(k)}_{i,j}},$$

Step6 For $n = 3$, we have the following iteration ,and matrix $A^{(1)}, A^{(2)}$ are the matrixes we get from step4 and step5;

$$\left(A_{i,j}^{(3)}\right)^{(k+1)} \leftarrow \left(A_{i,j}^{(3)}\right)^{(k)} \frac{\left(A_{(3)}\left(B^{(3)}\right)^{T}\right)_{i,j}^{(k)}}{\left(A^{(3)}B^{(3)}\left(B^{(3)}\right)^{T}\right)_{i,j}^{(k)}},$$

The termination criterion we choose is $\left\|\widehat{A}^{(k+1)} - \widehat{A}^{(k)}\right\| < \varepsilon$, and ε is the given error.

Remark (1) For loop 3, model is equivalent to finding the matrix $A^{(3)}$ which satisfies the equation

$$A_{(3)} = XB^{(3)},$$

And $X = A_{(3)}\left(B^{(3)}\right)^{\dagger}$, where $\left(B^{(3)}\right)^{\dagger}$ is the Moore-Perose inverse of matrix B. But the Moore-Perose inverse couldn't guarantee the nonnegative of matrix $A^{(3)}$, the above equation is equivalent to the following optimization problem, where $A^{(3)}, B^{(3)}$ are known in advance.

$$\begin{aligned} &\min \left\|A_{(3)} - XB^{(3)}\right\|_F^2 \\ &s.t. \qquad\qquad , \\ &X \geq 0 \end{aligned}$$

The scale factor in loop 3 is

$$\left(A_{i,j}^{(3)}\right)^{(k+1)} \leftarrow \left(A_{i,j}^{(3)}\right)^{(k)} \frac{\left(A_{(3)}\left(B^{(3)}\right)^{T}\right)_{i,j}^{(k)}}{\left(A^{(3)}B^{(3)}\left(B^{(3)}\right)^{T}\right)_{i,j}^{(k)}},$$

Which is used in [15], and the partial derivatives of the error function in variable $A_{i_j,a}^{(j)}$ is

$$\frac{\partial}{\partial A_{i_j,a}^{(j)}}\left\|A_{(j)} - A^{(j)}\left(B^{(j)}\right)^{T}\right\|_F^2 = 2\left(A^{(j)}\left(B^{(j)}\right)^{T}B^{(j)} - A_{(j)}B^{(j)}\right)_{i_j,a},$$

Because at the minimum of the error function, the derivatives is zero, which implies that the scale factor is identity. Therefore the, the minimum is a fixed point of the update rule. Also, if the gradient is negative, the scale factor is grater than 1, which implies that we move towards the minimum. Reversely, if the gradient is positive, the scale factor is small than 1, which also implies that we move towards the minimum.

(2) There are two iterations for nonnegative matrix factorization [17] [14] [15], we choose iterations in [14] because it's a mature effective technique and has good convergence. Because the optimal point of the loop optimization is not necessarily the optimal point of model (2), so we need not choose the optimal point in every loop. The value R is always less than the minimum value of I_1, I_2, I_3 for the reason that in real world problems factorization of the tensor need less storage space and convenient to transmit.

4.2. Convergence of the Algorithm

Lemma1 The loop 1 and loop 2 algorithms in our algorithm is convergent.

Proof:

In every loop algorithms, model (3) is equivalent to the following nonnegative matrix factorization

$$\begin{aligned} &\min_{W,H} \left\|V - WH\right\|_F^2 \\ &s.t. \qquad\qquad , \\ &W, H \geq 0 \end{aligned}$$

By the convergence of algorithm in [15], the iteration algorithm

$$H_{i,j} \leftarrow H_{i,j}\frac{\left(W^TV\right)_{i,j}}{\left(W^TWH\right)_{i,j}}, W_{i,j} \leftarrow W_{i,j}\frac{\left(VH^T\right)_{i,j}}{\left(WHH^T\right)_{i,j}},$$

is convergent. Hence the loop algorithm in our algorithm is convergent.

Lemma 2 The loop 3 algorithm in our algorithm is convergent.

Proof:

According to the convergence analysis in [15], we can conclude that loop 3 algorithms in our algorithm are convergent.

Lemma 3 The algorithm in model (2) is monotone decrease algorithm.

Proof:

The objective functions in model (2) is the same in the reason that it's the approximate error of the entries between the entries of tensor A and the corresponding entries of the matrix in the loop algorithm, because the loop algorithms are descent algorithms, therefore the algorithm in model is also a descent algorithm.

Theorem 3 The algorithm in our model is global convergent.

Proof: Assume that we start the iteration with $A_k^{(0)}(k=1,2,3)$, where $A_k^{(0)}(k=1,2,3)\neq 0$ and $f(A_1^{(0)},A_2^{(0)},A_3^{(0)})=M$, and the $M-$level set of function f includes 0 in its interior, i.e. $f(0,0,0)=m<M$ .By the lemmas above, we know that the iteration on f is a monotonically decreasing sequence, where $f(A_1^{(0)},A_2^{(0)},A_3^{(0)})=M$ is a maximize value of f over the $m-$level set of f. If a convex quadratic function f is bounded above, its $m-$level set is also bounded. Consider the set $L\left(A_1^{(0)},A_2^{(0)},A_3^{(0)}\right)=\left\{A_i^{(k)}\geq 0, i=1,2,3 \middle| f\left(A_1^{(k)},A_2^{(k)},A_3^{(k)}\right)\leq f\left(A_1^{(0)},A_2^{(0)},A_3^{(0)}\right)\right\}$, representing the intersection of the nonnegative orthant with the $M-$level set of f and chooses $\left(B_1^{(0)},B_2^{(0)},B_3^{(0)}\right)\in L$,

Such that

$$\left\|A_1^{(k)}\right\|_F^2\leq\left\|B_1^{(0)}\right\|_F^2, \left\|A_2^{(k)}\right\|_F^2\leq\left\|B_2^{(0)}\right\|_F^2, \left\|A_3^{(k)}\right\|_F^2\leq\left\|B_3^{(0)}\right\|_F^2,$$

Then, the iteration $\left\{A_1^{(k)},A_2^{(k)},A_3^{(k)}\right\}$ is bounded by 0 and $\left\|B_i^{(0)}\right\|_F^2, i=1,2,3$, hence it has a limit point in $M-$Level set.

5. Numerical Experiments

Example 1 We consider the tensor $A\in R^{3\times4\times2}$ in example before.

$$A_1=\begin{pmatrix}5&7&3&1\\2&9&2&6\\1&3&5&7\end{pmatrix}, A_2=\begin{pmatrix}4&2&1&3\\7&3&2&1\\7&5&6&1\end{pmatrix},$$

$$A_{(1)}=\begin{pmatrix}5&4&7&2&3&1&1&3\\2&7&9&3&2&2&6&1\\1&7&3&5&5&6&7&1\end{pmatrix}, A_{(2)}=\begin{pmatrix}5&2&1&4&7&7\\7&9&3&2&3&5\\3&2&5&1&2&6\\1&6&7&3&1&1\end{pmatrix},$$

$$A_{(3)}=\begin{pmatrix}5&7&3&1&2&9&2&6&1&3&5&7\\4&2&1&3&7&3&2&1&7&5&6&1\end{pmatrix}$$

The results are the following

When $R=2$

$$A^{(1)} = 1.0e{+}014 \begin{pmatrix} 0.0000 & 0.0000 \\ 1.2248 & 0.0000 \\ 4.8690 & 0.0000 \end{pmatrix}, A^{(2)} = \begin{pmatrix} 0.0006 & 427.0966 \\ 0.0005 & 657.2316 \\ 0.0007 & 180.9674 \\ 0.0005 & 301.4445 \end{pmatrix},$$

$$A^{(3)} = 1.0e-007 \begin{pmatrix} 0.0001 & 0.1016 \\ 0.0002 & 0.0528 \end{pmatrix}$$

When $R = 3$, we have the following results

$$A^{(1)} = \begin{pmatrix} 3.2012 & 0.2826 & 0.0001 \\ 2.4735 & 0.4330 & 0.7333 \\ 0.0000 & 0.5896 & 1.1701 \end{pmatrix}, A^{(2)} = \begin{pmatrix} 1.4898 & 6.6202 & 0.0000 \\ 3.0350 & 3.8197 & 0.0923 \\ 0.6169 & 3.7624 & 0.1032 \\ 0.5133 & 1.4750 & 0.2056 \end{pmatrix},$$

$$A^{(3)} = \begin{pmatrix} 0.7751 & 0.1594 & 29.6650 \\ 0.0000 & 2.0371 & 0.0000 \end{pmatrix}.$$

Example 2 For a 3-order symmetric tensor A,

$$a_{111} = 0.2883, a_{112} = 0.0031, a_{113} = 0.1973, a_{122} = 0.2485, a_{123} = 0.2939$$
$$a_{133} = 0.3847, a_{222} = 0.2972, a_{223} = 0.1862, a_{233} = 0.0919, a_{333} = 0.3619,$$

$$A_1 = \begin{pmatrix} 0.2883 & 0.0031 & 0.1973 \\ 0.0031 & 0.2485 & 0.2939 \\ 0.1973 & 0.2939 & 0.3847 \end{pmatrix}, A_2 = \begin{pmatrix} 0.0031 & 0.2485 & 0.2939 \\ 0.2485 & 0.2972 & 0.1862 \\ 0.2939 & 0.1862 & 0.0919 \end{pmatrix},$$

$$A_3 = \begin{pmatrix} 0.1973 & 0.2939 & 0.3847 \\ 0.2939 & 0.1862 & 0.0919 \\ 0.3847 & 0.0919 & 0.3619 \end{pmatrix},$$

$$A_{(1)} = \begin{pmatrix} 0.2883 & 0.0031 & 0.1793 & 0.0031 & 0.2485 & 0.2939 & 0.1973 & 0.2939 & 0.3847 \\ 0.0031 & 0.2485 & 0.2939 & 0.2485 & 0.2972 & 0.1862 & 0.2939 & 0.1862 & 0.0919 \\ 0.1973 & 0.2939 & 0.3487 & 0.2939 & 0.1862 & 0.0919 & 0.3847 & 0.0919 & 0.3619 \end{pmatrix},$$

$$A_{(2)} = \begin{pmatrix} 0.2883 & 0.0031 & 0.1793 & 0.0031 & 0.2485 & 0.2939 & 0.1973 & 0.2939 & 0.3847 \\ 0.0031 & 0.2485 & 0.2939 & 0.2485 & 0.2972 & 0.1862 & 0.2939 & 0.1862 & 0.0919 \\ 0.1973 & 0.2939 & 0.3487 & 0.2939 & 0.1862 & 0.0919 & 0.2939 & 0.0919 & 0.3619 \end{pmatrix};$$

$$A_{(3)} = \begin{pmatrix} 0.2883 & 0.0031 & 0.1793 & 0.0031 & 0.2485 & 0.2939 & 0.1973 & 0.2939 & 0.3847 \\ 0.0031 & 0.2485 & 0.2939 & 0.2485 & 0.2972 & 0.1862 & 0.2939 & 0.1862 & 0.0919 \\ 0.1973 & 0.2939 & 0.3487 & 0.2939 & 0.1862 & 0.0919 & 0.3847 & 0.0919 & 0.3619 \end{pmatrix}.$$

When $R = 2$

$$A^{(1)} = \begin{pmatrix} 1.6995 & 0.0823 \\ 2.5091 & 0.0607 \\ 2.8466 & 0.0807 \end{pmatrix}, A^{(2)} = \begin{pmatrix} 0.0001 & 0.0000 \\ 2.6896 & 0.4956 \\ 0.0000 & 0.0000 \end{pmatrix}, A^{(3)} = \begin{pmatrix} 0.0315 & 0.0000 \\ 0.0156 & 3.7669 \\ 0.0000 & 5.1010 \end{pmatrix},$$

When $R=3$

$$A^{(1)}=\begin{pmatrix}4.1477 & 6.8607 & 0.5751\\ 2.1319 & 9.9551 & 0.6956\\ 5.8198 & 12.1083 & 0.0345\end{pmatrix}, A^{(2)}=\begin{pmatrix}0.0000 & 0.0000 & 0.0000\\ 0.7356 & 1.4206 & 0.7847\\ 0.0000 & 0.0000 & 0.0000\end{pmatrix},$$

$$A^{(3)}=\begin{pmatrix}0.0000 & 0.0146 & 0.0000\\ 0.0108 & 0.0077 & 0.3126\\ 0.0250 & 0.0000 & 0.3540\end{pmatrix}$$

6. Conclusions and Future Work

In this paper, we get a tensor representation of tensor unfolded matrix by the matrix product properties we get, and combined the representation of tensor and nonnegative matrix factorization technique [13] [14], we present an algorithm for nonnegative tensor decomposition. The algorithm has a simple frame and is convenient to implement and has a good performance. Numerical experiments show that our algorithm is effective and it can be used to solve the engineering problems.

Future works:

(1) The loop algorithm in the paper is the steepest gradient algorithm and it has slow convergence,

can we utilize other methods such as conjugate gradients algorithm in the vicinity of the optimal point for the good convergence of the conjugate gradients algorithm?

(2) We haven't consider the sparseness of the CP factorization matrix in our algorithm, and for the nonnegative constraints of the model, there must be some zeroes in the factorization matrix especially for the higher order tensor ,then how to utilize the sparseness to save the storage and the computation is an interesting work.

(3) How to optimize our algorithm by the latest technique of nonnegative matrix factorization is a future work.

(4) Use our algorithm to solve the real world problems in other fields is also a challenging work.

7. References

[1] Charles F. Van Loan. The ubiquitous Kronecker product. *Journal of Computational and applied mathematics.* 2000, **123**(1-2): 85-100.

[2] Cheng Daizhan. *Semi-tensor product theory and its applications*. Beijing, Science Publisher，2007.

[3] L. De. Lathauwer. A multilinear singular value decomposition. *SIAM. J. Matrix analysis and application.* 2000, **21**(4): 1253-1287.

[4] L. De. Lathauwer, B. D. Moor, J. Vandewalle. On the best rank-1 and rank-($R_1, R_2, \cdots, R_N$) approximation of higher-order tensor. *SIAM. J. Matrix analysis and application.* 2000, **21**(5): 1324-1342.

[5] Zhan Xianda. *Matrix Analysis and Applications*. Beijing, Tsinghua University Publisher, Springer Publisher, 2004.

[6] Xu Junyi. Matrix product and applications. *Journal of Computer-Aided Design & Computer Graphics.* 2003, **15**(4): 377-388.

[7] T. Hanzan, S. Polak, and A. Shashua. Sparse image coding using a 3d non-negative tensor factorization. *Proceedings of international conference on computer vision*, Beijing, PRC, 2005.

[8] M. Morup, L.K. Hansen, and S.M. Arnfred. Sparse higher order nonnegative tensor factorization. Technical report, 2006, http://www.mortenmorup.dk.

[9] A. Shashua,T. Hazan. Nonnegative tensor factorization with applications to statistics and computer vision. *Proceedings of international conference on machine learning*, Bonn, Germany, 2005.

[10] RA. Harshman. Foundations of PARAFAC procedure: models and conditions for an “explanatory” multimode factor analysis. *UCLA working papers phonetics*. 1970, **116**: 1-84.

[11] RA. Harshman. Determination and proof of minimum uniqueness conditions for PARAFAC. *UCLA working papers phonetics*. 1972, **22**: 111-117.

[12] J.D. Carroll and J.J. Chang. Analysis of individual differences in multidimensional scaling via an N-way

generalization of "Eckart-Young" decomposition. *Psychometrika.* 1970, **35**: 283-319.

[13] D.D. Lee, H.S. Seung. Learning the parts of objects nonnegative matrix factorization. *Nature.* 1999, **401**: 788-791.

[14] D.D. Lee, H.S. Seung. Algorithms for nonnegative matrix factorization. *Advances in Neural information processing systems.* 2001, **13**: 556-562.

[15] Max Welling, Markus Weber. Positive tensor factorization. *Pattern Recognition Letters.* 2001,**22**: 1255-1261.

[16] Liu Weixiang, Zheng Nanning, You Qubo. Nonnegative matrix factorization and its applications in pattern recognition. *Chinese Science Bulletin.* 2006, **51**(1): 7-18.

[17] Russell Albright, et al. Algorithms, Initializations, and Convergence for the Nonnegative Matrix Factorization. *NCSU Technical Report Math* 81706, 2006.

[18] H. Wang and N. Ahuja. Rank-R approximation of tensors: Using image as matrix representation, *Proceedings of 2005 IEEE computer Society Conference on Computer Vision and Pattern Recognition*, 2, pp.346-353, 2005.

[19] G.Y. Ni, Y.J. Wang. On the best rank-1 approximation to higher-order symmetric tensors. *Mathematical and Computer Modeling.* 2007, **46**(9/10): 1345–1352.

[20] L.Q. Qi, W.Y. Sun, Y.J. Wang. Numerical multilinear algebra and its applications. *Front Math*. China, 2007, **2**(4): 501-526.