**Matricization: Transforming a Tensor into a Matrix.**

Matricization, also known as unfolding or flattening, is the process of reordering the elements of an N-way array into a matrix. For instance, a 2×3×4 tensor can be arranged as a 6×4 matrix or a 3 ×8 matrix, and so on. In this review, we consider only the special case of mode-n matricization because it is the only form relevant to our discussion. A more general treatment of matricization can be found in Kolda [1]. The mode-n matricization of a tensor is denoted by and arranges the mode-n fibers to be the columns of the resulting matrix. Though conceptually simple, the formal notation is clunky. Tensor element maps to matrix element , where:

The concept is easier to understand using an example. Let the frontal slices of be

Then the three mode-*n* unfoldings are

Last, we note that it is also possible to vectorize a tensor. Once again the ordering

of the elements is not important so long as it is consistent. In the example above, the

vectorized version is

**Tensor Multiplication: The n-Mode Product***.*

The *n*-Mode Product, i.e., multiplying a tensor by a matrix (or a vector) in mode n.The n-mode (matrix) product of a tensor with a matrix is denoted by and is of size. Elementwise, we have

**Matrix Kronecker,Khatri-Rap, and Hadamard Products.**

In the next section we used the product of some matrices, here we give a brief description.

***Kronecker Product***

The Kronecker product of matrices***A***∈ and ***B***∈ is denoted by ***A****⨂****B****.* The result is a matrix of size(*IK*)×(*JL*) and defined by

As an example of the utility of the Kronecker product, consider the following.

Let and for all . Then, for any , we have The Khatri-Rao product

***Hadamard product***

Suppose matrices , then the Hadamard product is denoted by

**TUCKER3 Factorization**

We assume, The TUCKER3 factorization aims to find that solves the following optimization problem:

Then we transform this equation into tensor and matrix form:

Here, are the factor matrices (which are usually orthogonal) and can be thought of as the principal components in each mode, are the number of components (i.e., columns,) in the factor matrices *.* The tensor is called the core tensor and its entries show the level of interaction between the different components. The last equality uses the shorthand introduced in Kolda[1].

Next, we give an iterative derivation formula for the feature matrices and core tensor G.

**Update of the feature matrix:**

Considering that the derivation forms of are similar, here we only derive the iterative formula of.

The objective function in equation (5) can be rewritten as a matrixed form of X along the first dimension.

where，.

Assuming that the optimal solution satisfies all the constraints in equation (7), then

where

Therefore, Equation 9 can be regarded as a non-negative matrix factorization (NMF) form in the literature [2-3]. Then, use the NMF update method to solve :

**Core tensor update:**

We fixed the feature matrix , and the objective function in equation (5) can be converted to:

The following linear equation can be obtained from equation (12):

Suppose Equation (13) can also be transformed into an NMF model to update the core tensor according to the method in [3] :

By formula(14)

Reference

[1] T. G. Kolda, *Multilinear Operators for Higher-Order Decompositions*, Tech. Report SAND2006-2081, Sandia National Laboratories, Albuquerque, NM, Livermore, CA, 2006.

[2] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization. Nature 401, 788 (1999).

[3] Lee D D，Seung H S． Algorithms for nonnegative matrix factorization ［C］/ /Proceedings of Advances in Neural Information Processing Systems，2001: 556 － 562