CSE 494 CSE/CBS 598 (Fall 2007): Numerical Linear Algebra for Data Exploration— Tensor Decomposition
Instructor: Jieping Ye

1 Introduction

- So far we focus on vectors and matrices. These can be thought of as one-dimensional and two-dimensional arrays of data, respectively. For instance, in a term-document matrix, each element is associated with one term and one document.
- In many applications it is common that data are organized according to more than two categories. The corresponding mathematical objects are usually referred to as **tensors**, and the area of mathematics dealing with tensors is **multi-linear algebra**.
  - Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A Multilinear Singular Value Decomposition, SIAM Journal on Matrix Analysis and Applications, 2000.
- In this course, we focus on three-dimensional arrays (third-order tensor).

1.1 Examples

- In the classification of handwritten digits, the training set is a collection of images, manually classified into 10 classes. Each such class is a set of digits of one kind, which can be considered as a tensor. If each digit is represented as a 16-by-16 matrix of numbers representing grey-scale, then a set of n digits can be organized as a tensor $\mathcal{A} \in \mathbb{R}^{16\times 16\times n}$.
- A video can be represented as a tensor $\mathcal{A} \in \mathbb{R}^{r\times c\times n}$, where r and c are the number of rows and columns of each image in the video, respectively and n is the length of the image sequences in the video.
- In computer graphics, the appearance of rendered surfaces is determined by a complex interaction of multiple factors related to scene geometry, illumination, and imaging.
  - M.A.O. Vasilescu and D. Terzopoulos. TensorTextures: Multilinear Image-Based Rendering, SIGGRAPH 2004.
  - H. Wang and et al. Out-of-Core Tensor Approximation of Multi-Dimensional Matrices of Visual Data, SIGGRAPH 2005.

2 Basic Tensor Concepts

- We refer to a tensor $A \in \mathbb{R}^{l\times m\times n}$ as a 3-mode array, i.e., the different dimensions of the array are called modes. The dimensions of a tensor $A \in \mathbb{R}^{l\times m\times n}$ are l, m, and n. In this terminology, a matrix is a two-mode array.
- The inner product of two tensors A and B is defined as

$$< A, B >= \sum_{i,j,k} a_{ijk} b_{ijk}.$$

- The corresponding norm is

$$||A||_F = < A, A >^{1/2} = \left(\sum_{i,j,k} a_{ijk}^2 \right)^{1/2}.$$

- Define i-mode multiplication of a tensor by a matrix.
  - The 1-mode product of a tensor $A \in \mathbb{R}^{l \times m \times n}$ by a matrix $U \in \mathbb{R}^{l_0 \times l}$, denoted by $A \times_1 U$, is an $l_0 \times m \times n$ tensor in which the entries are given by

$$(A \times_1 U)(j, i_2, i_3) = \sum_{k=1}^{l} u_{j,k} \; a_{k,i_2,i_3}.$$

  - For comparison, consider the matrix multiplication $A \times_1 U = UA$, where

$$(UA)(i,j) = \sum_{k=1}^{l} u_{i,k} \; a_{k,j}.$$

  - Recall that matrix multiplication is equivalent to multiplying each column in A by the matrix U. The corresponding is true for tensor-matrix multiplication. In the 1-mode product all column vectors in the 3-mode array are multiplied by the matrix U.
  - Similarly, 2-mode multiplication of a tensor by a matrix V is given by

$$(A \times_2 V)(i_1, j, i_3) = \sum_{k=1}^{m} v_{j,k} a_{i_1,k,i_3}.$$

    Note that 2-mode multiplication of a matrix by V is equivalent to matrix multiplication by V^T from the right, $A \times_2 V = AV^T$.

- It is common to **unfold** a tensor into a matrix.
  - The unfolding along mode i (the resulting matrix is called $A_{(i)}$) makes that mode the first mode of the matrix $A_{(i)}$, and the other modes are handled cyclically.
  - For instance, row i of $A_{(j)}$ contains all the elements of A, which have the j-th index equal to i.
  - Example: Let $B \in \mathbb{R}^{3 \times 3 \times 3}$ be a tensor, defined in MATLAB as

$$B(:,:,1) = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad B(:,:,2) = \begin{pmatrix} 11 & 12 & 13 \\ 14 & 15 & 16 \\ 17 & 18 & 19 \end{pmatrix}, \quad B(:,:,3) = \begin{pmatrix} 21 & 22 & 23 \\ 24 & 25 & 26 \\ 27 & 28 & 29 \end{pmatrix}.$$

    Then unfolding along the third mode gives

$$B_{(3)} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 \\ 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 \end{pmatrix}.$$

- The inverse of the unfolding operation is written

$$\text{fold}_i(\text{unfold}_i(A)) = A.$$

- Using the unfolding-folding operations, we can now formulate a matrix multiplication equivalent of i-mode tensor multiplication:

$$A \times_i U = \text{fold}_i(U \text{unfold}_i(A)) = \text{fold}_i(U A_{(i)}).$$

  - It follows immediately from the definition that i-mode and j-mode multiplication commute if $i \neq j$:
$$(A \times_i F) \times_j G = (A \times_j G) \times_i F = A \times_i F \times_j G.$$

  - Two i-mode multiplications satisfy the identity

$$(A \times_i F) \times_i G = A \times_i (GF).$$

3 Higher Order SVD (HOSVD)

- The matrix singular value decomposition can be generalized to tensors in different ways. We will present one such generalization, that is analogous to an approximate principal component analysis. It is often referred to as the Higher Order SVD (HOSVD).

Theorem 3.1 *The tensor $A \in \mathbb{R}^{l \times m \times n}$ can be written A*

$$A = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}.$$

where $U^{(1)} \in \mathbb{R}^{l \times l}$, $U^{(2)} \in \mathbb{R}^{m \times m}$, and $U^{(3)} \in \mathbb{R}^{n \times n}$ are orthogonal matrices. S is a tensor of the same dimensions as A; it has the property of all-orthogonality: any two slices of S are orthogonal in the sense of the scalar product:

$$\langle S(i,:,:), S(j,:,:)\rangle = \langle S(:,i,:), S(:,j,:)\rangle = \langle S(:,:,i), S(:,:,j)\rangle = 0,$$

for $i \neq j$. The 1-mode singular values are defined $\sigma_j^{(1)} = ||S(j,:,:)||_F$, $j = 1, \cdots, l$, and they are ordered, i.e., $\sigma_1^{(1)} \geq \sigma_2^{(1)} \geq \cdots \geq \sigma_l^{(1)}$. The singular values in other modes and their ordering are analogous.

- The orthogonal factors and the tensor S can be computed as follows:
  - Compute the SVDs $A_{(i)} = U^{(i)} \Sigma^{(i)} (V^{(i)}) T$, $i = 1, 2, 3$; and
  - Put $S = A \times_1 \left(U^{(1)}\right)^T \times_2 \left(U^{(2)}\right)^T \times_3 \left(U^{(3)}\right)^T$.
  - It can be shown that the slices of S are orthogonal and that the i-mode singular values are decreasingly ordered.
  - The all-orthogonal tensor S is usually referred to as the core tensor.

- The expression above can also be written as

$$A_{ijk} = \sum_{p=1}^{l} \sum_{q=1}^{m} \sum_{s=1}^{n} u_{ip}^{(1)} u_{jq}^{(2)} u_{ks}^{(3)} S_{pqs}.$$

4 Rank-$(R_1, \cdots, R_N)$ Tensor Factorization

- Given an Nth-order tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, rank-$(R_1, \cdots, R_N)$ factorization of A is formulated as finding a lower-rank tensor $\tilde{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ with $\text{rank}_n(\tilde{A}) = R_n \leq \text{rank}_n(A)$, for all n, such that the following least-squares cost function is minimized:

$$\tilde{A} = \text{argmin}_{\hat{A}} \left\| A - \hat{A} \right\|. \tag{1}$$

  More specifically, $\tilde{A}$ can be expressed as follows:

$$\tilde{A} = \mathcal{C} \times_1 U^{(1)} \times_2 U^{(2)} \times \cdots \times_N U^{(N)}, \tag{2}$$

  where $U^{(n)} \in \mathbb{R}^{I_n \times R_n}$ has orthonormal columns for $n = 1, \cdots, N$.

- When R_n is much smaller than I_n for all n, the core tensor $\mathcal{C}$ and the basis matrices $\{U^{(n)}\}_{n=1}^N$ give a compact representation of the original tensor A, resulting in data compression.

- Given the basis matrices $\{U^{(n)}\}_{n=1}^N$, the core tensor $\mathcal{C}$ can be readily computed as $\mathcal{C} = A \times_1 (U^{(1)})^T \cdots \times_N (U^{(N)})^T$. Thus, the optimization problem focuses on the computation of the basis matrices only.

$$||A - \tilde{A}||^2 = ||A||^2 - 2 < A, \tilde{A} > + ||\tilde{A}||^2.$$

  Based on the definition of the inner product, we have

$$\begin{aligned} < A, \tilde{A} > &= < A, \mathcal{C} \times_1 U^{(1)} \times_2 U^{(2)} \times \cdots \times_N U^{(N)} > \\ &= < A \times_1 (U^{(1)})^T \times_2 (U^{(2)})^T \times \cdots \times_N (U^{(N)})^T, \mathcal{C} > \\ &= ||\mathcal{C}||^2. \end{aligned}$$

- Since $U^(n)$ for all n have orthonormal columns, they don't affect the norm

$$||\tilde{A}||^2 = ||\mathcal{C}||^2.$$

  Thus,

$$||A - \tilde{A}||^2 = ||A||^2 - ||\mathcal{C}||^2.$$

- The best Rank-$(R_1, \cdots, R_N)$ tensor approximation can be computed by maximizing

$$||A \times_1 (U^{(1)})^T \cdots \times_N (U^{(N)})^T||.$$

- An iterative approach can be applied for the computation. Each iterative step optimizes only one of the basis matrices, while keeping the other $N-1$ basis matrices fixed. With $U^{(1)}, \cdots, U^{(n-1)}, U^{(n+1)}, \cdots, U^{(N)}$ fixed, we first project A onto the $(R_1, \cdots, R_{n-1}, R_{n+1}, \cdots, R_N)$-dimensional space as follows:

$$V^n = A \times_1 (U^{(1)})^T \cdots \times_{n-1} (U^{(n-1)})^T \times_{n+1} (U^{(n+1)})^T \times \cdots \times_N (U^{(N)})^T.$$

- Then, $U^{(n)}$ is given by the first R_n columns of the left singular matrix of $V^n_{(n)}$, which consists of all n-th mode vectors of V^n. The least-squares cost in Eq. (1) decreases monotonically during the iteration.
  - In HOSVD, $U^{(n)}$ is given by the first R_n columns of the left singular matrix of $A_{(n)}$.