From Matrix to Tensor:
The Transition to Numerical Multilinear Algebra

Lecture 3. Transpositions, Kronecker Products, and Contractions

Charles F. Van Loan

Cornell University

The Gene Golub SIAM Summer School 2010
Selva di Fasano, Brindisi, Italy

Where We Are

What is this Lecture About?

Preview

Now we turn our attention to the engagement of these multidimensional data objects in two elementary operations.

Transposition. While there is only one way transpose a matrix there are an exponential number of ways to transpose an order-d tensor. This will require (surprise surprise) more facility with multi-indexing.

Contraction. The contraction of two tensors can be regarded as a generalization of matrix-matrix multiplication. Analogous to how the serious business of matrix decompositions builds on matrix multiplication, we will find that tensor contractions have a foundational role to play in tensor computations.

What is this Lecture About?

More "Bridging the Gap"

The Kronecker product helps bridge the gap between matrix computations and tensor computations. For example, the contraction between two tensors can sometimes be "reshaped" into a matrix computation that involves Kronecker products.

So in advance of our introduction to tensor contractions, we will get familiar with this all-important matrix operation and some of its nearby "cousins."

Tensor Transposition

Example: $A(1{:}4, 1{:}5, 1{:}6) \;\rightarrow\; B(1{:}5, 1{:}6, 1{:}4)$

```
A = randn([ 4 5 6]);
B = zeros([5 6 4]);
for i1=1:4
   for i2 = 1:5
      for i3 = 1:6
         B(i2,i3,i1) = A(i1,i2,i3);
      end
   end
end
```

How to say this: $\mathcal{B} \;=\; \mathcal{A}^{<\,[\,2\;3\;1\,]\,>}$.

Tensor Transposition: The 3rd Order Case

Six possibilities...

If $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, then there are $6 = 3!$ possible transpositions identified by the notation $\mathcal{A}^{< [i\ j\ k] >}$ where $[i\ j\ k]$ is a permutation of $[1\ 2\ 3]$:

$$\mathcal{B} = \left\{ \begin{array}{c} \mathcal{A}^{< [1\ 2\ 3] >} \\ \mathcal{A}^{< [1\ 3\ 2] >} \\ \mathcal{A}^{< [2\ 1\ 3] >} \\ \mathcal{A}^{< [2\ 3\ 1] >} \\ \mathcal{A}^{< [3\ 1\ 2] >} \\ \mathcal{A}^{< [3\ 2\ 1] >} \end{array} \right\} \quad \Longrightarrow \quad \left\{ \begin{array}{c} b_{ijk} \\ b_{ikj} \\ b_{jik} \\ b_{jki} \\ b_{kij} \\ b_{kji} \end{array} \right\} = a_{ijk}$$

for $i = 1{:}n_1,\ j = 1{:}n_2,\ k = 1{:}n_3$.

Tensor Transposition: General Case

Definition

If $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ and $\mathbf{p} = [p_1, \ldots, p_d]$ is a permutation of the index vector $1{:}d$, then $\mathcal{A}^{<\mathbf{p}>} \in \mathbb{R}^{n_{p_1} \times \cdots \times n_{p_d}}$ denotes the **p**-transpose of $\mathcal{A}$ and is defined by

$$\mathcal{A}^{<\mathbf{p}>}(j_{p_1}, \ldots, j_{p_d}) \;=\; \mathcal{A}(j_1, \ldots, j_d)$$

where $1 \leq j_k \leq n_k$ for $k = 1{:}d$.

Matlab Tensor Toolbox: **Transposition**

```
% If A is an order-d tensor and
% p is a permutation of 1:d, then
% the following assigns the p-transpose
% of A to B...

B = permute(A,p)
```

Problem 3.1. Any square matrix can be written as the sum of a symmetric matrix and a skew-symmetric matrix:

$$A = \left(\frac{A + A^T}{2}\right) + \left(\frac{A - A^T}{2}\right)$$

Formulate a generalization of this result for order-3 tensors and write a Matlab function that embodies your idea.

Tensor Notation: Subscript Vectors (Review)

Reference

If $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ and $\mathbf{i} = (i_1, \ldots, i_d)$ with $1 \leq i_k \leq n_k$ for $k = 1{:}d$, then

$$\mathcal{A}(\mathbf{i}) \;\equiv\; \mathcal{A}(i_1, \ldots, i_k)$$

We say that $\mathbf{i}$ is a subscript vector. Bold font will be used designate subscript vectors.

Bounds

If $\mathbf{L}$ and $\mathbf{R}$ are subscript vectors having the same dimension, then $\mathbf{L} \leq \mathbf{R}$ means that $L_k \leq R_k$ for all k.

Special Cases

A subscript vector of all ones is denoted by $\mathbf{1}$. (Dimension clear from context.) If N is an integer, then $\mathbf{N} = N \cdot \mathbf{1}$.

Tensor Transposition: General Case

Definition

If $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ and $\mathbf{p} = [p_1, \ldots, p_d]$ is a permutation of the index vector $1{:}d$, then $\mathcal{A}^{<\mathbf{p}>} \in \mathbb{R}^{n_{p_1} \times \cdots \times n_{p_d}}$ denotes the **p**-transpose of $\mathcal{A}$ and is defined by

$$\mathcal{A}^{<\mathbf{p}>}(\mathbf{j}(\mathbf{p})) = \mathcal{A}(\mathbf{j}) \qquad \mathbf{1} \leq \mathbf{j} \leq \mathbf{n}.$$

Problem 3.2. Verify that if **p** and **q** are both permutations of $1{:}d$, then

$$\left(\mathcal{A}^{<\mathbf{p}>}\right)^{<\mathbf{q}>} \;=\; \mathcal{A}^{<\mathbf{p(q)}>}$$

where $\mathcal{A} \in \mathbb{R}^{n_1\times\cdots\times n_d}$.

Problem 3.3. Suppose $\mathcal{A} \in \mathbb{R}^{n_1\times\cdots\times n_d}$ and that $\mathcal{B} = \mathcal{A}^{[\,\mathbf{p}\,]}$ where **p** is a permutation of $1{:}d$. Specify a permutation matrix P so that $\mathcal{B}_{(k)} \;=\; \mathcal{A}_{(p(k))}P$.

What is a Kronecker Product?

Definition

$B \otimes C$ is a *block matrix* whose ij-th block is $b_{ij} C$.

An Example...

$$\left[\begin{array}{cc} b_{11} & b_{12} \\ b_{21} & b_{22} \end{array}\right] \otimes C \;=\; \left[\begin{array}{c|c} b_{11}C & b_{12}C \\ \hline b_{21}C & b_{22}C \end{array}\right]$$

Replicated Block Structure

What Is a Kronecker Product?

There are three ways to regard $A = B \otimes C$...

If

$$A = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mn} \end{bmatrix} \otimes \begin{bmatrix} c_{11} & \cdots & c_{1q} \\ \vdots & \ddots & \vdots \\ c_{p1} & \cdots & c_{pq} \end{bmatrix}$$

then

(i). A is an m-by-n block matrix with p-by-q blocks.

(ii). A is an mp-by-nq matrix of scalars.

(iii). A is an unfolded 4-th order tensor $\mathcal{A} \in \mathbb{R}^{p \times q \times m \times n}$:

$$\mathcal{A}(i_1, i_2, i_3, i_4) = B(i_3, i_4)C(i_1, i_2)$$

All Possible Entry-Entry Products

$4 \times 9 = 36$

$$\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \otimes \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

$$=$$

$$\left[\begin{array}{ccc|ccc} b_{11}c_{11} & b_{11}c_{12} & b_{11}c_{13} & b_{12}c_{11} & b_{12}c_{12} & b_{12}c_{13} \\ b_{11}c_{21} & b_{11}c_{22} & b_{11}c_{23} & b_{12}c_{21} & b_{12}c_{22} & b_{12}c_{23} \\ b_{11}c_{31} & b_{11}c_{32} & b_{11}c_{33} & b_{12}c_{31} & b_{12}c_{32} & b_{12}c_{33} \\ \hline b_{21}c_{11} & b_{21}c_{12} & b_{21}c_{13} & b_{22}c_{11} & b_{22}c_{12} & b_{22}c_{13} \\ b_{21}c_{21} & b_{21}c_{22} & b_{21}c_{23} & b_{22}c_{21} & b_{22}c_{22} & b_{22}c_{23} \\ b_{21}c_{31} & b_{21}c_{32} & b_{21}c_{33} & b_{22}c_{31} & b_{22}c_{32} & b_{22}c_{33} \end{array}\right]$$

Hierarchical

Kronecker Products of Kronecker Products

$$B \otimes C \otimes D$$

$$=$$

$$\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \otimes \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix} \otimes \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix}$$

A 2-by-2 block matrix whose entries are 4-by-4 block matrices whose entries are 3-by-3 matrices.

Kronecker Product Properties

Some Basic Facts...

$$\begin{aligned} (B \otimes C)^T &= B^T \otimes C^T \\ (B \otimes C)^{-1} &= B^{-1} \otimes C^{-1} \\ (B \otimes C)(D \otimes F) &= BD \otimes CF \\ B \otimes (C \otimes D) &= (B \otimes C) \otimes D \end{aligned}$$

Note that $B \otimes C \neq C \otimes B$. *However, it can be shown that*

$$C \otimes B = P^T (B \otimes C) Q$$

where P *and* Q *are perfect shuffle permutations. More later.*

Kronecker Products and Structure

Structured Factors Produce a Structured Product...

B and C	$B \otimes C$
Nonsingular	Nonsingular
Lower(Upper) triangular	Lower(Upper) Triangular
Banded	Block Banded
Symmetric	Symmetric
Positive Definite	Positive Definite
Stochastic	Stochastic
Toeplitz	Block Toeplitz
Permutation	Permutation
Orthogonal	Orthogonal

Kronecker Products and Data Sparsity

Example 1.

If $B, C \in \mathbb{R}^{m \times m}$ and

$$A \;\approx\; B \otimes C$$

then we are using $O(m^2)$ numbers (the b_{ij} and c_{ij}) to represent/approximate $O(m^4)$ data (the a_{ij}.)

Example 2.

If $B_{ij} \in \mathbb{R}^{m \times m}$ and

$$A \approx \sum_{k=1}^{r} B_{k,1} \otimes B_{k,2} \otimes \cdots \otimes B_{k,d}$$

then we are using rdm^2 numbers to represent/approximate a matrix with m^{2d} entries.

Kronecker Products and Fast Matrix-Vector Multiplication

A Basic Reshaping Result...

$$Y = CXB^T \quad \Leftrightarrow \quad \mathrm{vec}(Y) = (B \otimes C)\mathrm{vec}(X)$$

In Practice...

```
function y = KronTimesVector(B,C,y)
% y = kron(B,C)*x
[mb,nb] = size(B);
[mc,nc] = size(C);
Y = C*reshape(x,nc,nb)*B';
y = reshape(Y,mc*mb,1);
```

If both matrices are n-by-n, then work is $O(n^3)$. Ordinarily, an n^2-by-n^2 matrix-vector product would be $O(n^4)$.

Problem 3.4. How would you compute the matrix-vector product $y = (I_p \otimes A \otimes I_r)x$ where $A \in \mathbb{R}^{q \times q}$ and $x \in \mathbb{R}^{pqr}$? Assess the amount of work.

Problem 3.5. How would you compute the matrix-vector product $y = (A_1 \otimes A_2 \otimes A_3)x$ where $A_i \in \mathbb{R}^{n \times n}$ and $x \in \mathbb{R}^{n^3}$?. Hint:

$$(A_1 \otimes A_2 \otimes A_3) \;=\; (A_1 \otimes I_{n^2})(I_n \otimes A_2 \otimes I_n)(I_{n^2} \otimes A_3)$$

Assess the amount of work.

Problem 3.6. How would you compute the matrix-vector product $y = (A_1 \otimes \cdots \otimes A_d)x$ where $A_i \in \mathbb{R}^{n \times n}$ and $x \in \mathbb{R}^{n^3}$?. Assess the amount of work.

Kronecker Products and Matrix Factorizations

Just take the Kronecker Product of the Factors...

LU: $$(P_B \otimes P_C)(B \otimes C) = (L_B \otimes L_C)(U_B \otimes U_C)$$

Cholesky: $$B \otimes C = (L_B \otimes L_C)(L_B \otimes L_C)^T$$

Schur: $$(Q_B \otimes Q_C)^T(B \otimes C)(Q_B \otimes Q_C) = T_B \otimes T_C$$

QR: $$B \otimes C = (Q_B \otimes Q_C)(R_B \otimes R_C)$$

SVD: $$B \otimes C = (U_B \otimes U_C)(\Sigma_B \otimes \Sigma_C)(V_B \otimes V_C)^T$$

There are some annoying details if the matrices are rectangular, e.g., $(\Sigma_B \otimes \Sigma_C)$ *needs to be permuted to obtain a true diagonal form.*

Problem 3.7. How would you minimize $\| (A_1 \otimes \cdots \otimes A_d)x - b \|_2$ given that each $A_1, \ldots, A_d \in \mathbb{R}^{m \times n}$ has rank $r_i < n$?

Perfect Shuffle Permutations

Transposition is a Permutation

This:

$$\begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}^T$$

Is the same as this:

$$\begin{bmatrix} b_{11} \\ b_{21} \\ b_{12} \\ b_{22} \\ b_{13} \\ b_{23} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{12} \\ a_{22} \\ a_{32} \end{bmatrix}$$

Perfect Shuffle Permutations

Formal Definition

If $n = pr$ then the mod-p perfect shuffle permutation $S_{p,r}$ is defined by

$$S_{p,r}x \;=\; \left[\begin{array}{c} x(1{:}r{:}n) \\ x(2{:}r{:}n) \\ \vdots \\ x(r{:}r{:}n) \end{array}\right]$$

where $x \in \mathbb{R}^n$.

Implication for High Performance Computing: *Matrix transpose involves non-unit stride access to matrix elements.*

Problem 3.8. (a) Think about the perfect shuffling of a 52-card deck and explain why we use the term “perfect shuffle” when referring to $S_{p,r}$. (b) Show that $S_{r,p} = S_{p,r}^T$. (c) Explain why $S_{p,r}$ is aptly called the “mod-p sort permutation.”

Perfect Shuffle Permutations

The Connection to Matrix Transpose...

If $X \in \mathbb{R}^{p \times r}$ and $Y = X^T$, then

$$\text{vec}(Y) \;=\; S_{p,r} \cdot \text{vec}(X)$$

The Connection to Kronecker Products...

If $A_1 \in \mathbb{R}^{m_1 \times n_1}$ and $A_2 \in \mathbb{R}^{m_2 \times n_2}$ then

$$A_2 \otimes A_1 \;=\; S_{m_1,m_2}^T \cdot (A_1 \otimes A_2) \cdot S_{n_1,n_2}$$

Perfect Shuffle Permutations

The Connection to Tensor Transpose

If $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, then

$$\mathcal{B} = \left\{ \begin{array}{c} \mathcal{A}^{<\,[1\ 2\ 3]\,>} \\ \mathcal{A}^{<\,[1\ 3\ 2]\,>} \\ \mathcal{A}^{<\,[2\ 1\ 3]\,>} \\ \mathcal{A}^{<\,[2\ 3\ 1]\,>} \\ \mathcal{A}^{<\,[3\ 1\ 2]\,>} \\ \mathcal{A}^{<\,[3\ 2\ 1]\,>} \end{array} \right\} \Longrightarrow \text{vec}(\mathcal{B}) = \left\{ \begin{array}{c} I_{n_1 n_2 n_3} \\ S_{n_2 , n_3} \otimes I_{n_1} \\ I_{n_3} \otimes S_{n_1 , n_2} \\ S_{n_1 , n_2 n_3} \\ S_{n_1 n_2 , n_3} \\ (S_{n_1 , n_2} \otimes I_{n_3}) S_{n_1 n_2 , n_3} \end{array} \right\} \cdot \text{vec}(\mathcal{A})$$

Implication for High Performance Computing: *Tensor transpose involves non-unit stride access to matrix elements.*

Problem 3.9. Suppose $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, $N = n_1 \cdots n_d$, and that $\mathbf{p}$ is a permutation of $1{:}d$ that involves swapping a single pair of indices, e.g., [1 4 3 2 5]. Determine a permutation matrix $P \in \mathbb{R}^{N \times N}$ so that if $\mathcal{B} = \mathcal{A}^{< \, \mathbf{p} \, >}$, then

$$\text{vec}(\mathcal{B}) \;=\; P \cdot \text{vec}(\mathcal{A}).$$

Tensor Contractions

Definition

A contraction between two tensors produces a third tensor through a summation process.

Linear Algebra Operations in Tensor Contraction Terms

The Contraction of an Order-1 Tensor with an Order-1 Tensor

$$c = \sum_{k=1}^{n} a(k) \cdot b(k) \qquad \text{(Dot Product)}$$

Linear Algebra Operations in Tensor Contraction Terms

The Contraction of an Order-2 Tensor and an Order-1 Tensor

$$c(i) = \sum_{k=1}^{n} A(i,k) \cdot b(k) \qquad \text{(Matrix-Vector Product)}$$

Linear Algebra Operations in Tensor Contraction Terms

The Contraction of an Order-2 Tensor and an Order-2 Tensor

$$C(i,j) = \sum_{k=1}^{n} A(i,k) \cdot B(k,j) \qquad \text{(Matrix-Matrix Product)}$$

Tensor Contractions: Subscript Level

Example 1. (A Single-Index Contraction)

$$\mathcal{C}(i,j,\alpha_3,\alpha_4,\beta_3,\beta_4,\beta_5) \;=\; \sum_k \mathcal{A}(i,k,\alpha_3,\alpha_4)\cdot\mathcal{B}(k,j,\beta_3,\beta_4,\beta_5)$$

$$\Uparrow \qquad\qquad \Uparrow \qquad\qquad \Uparrow$$

7th Order — 4th Order — 5th Order

$$\begin{aligned} \mathcal{A} &= \mathcal{A}(1{:}n_1, 1{:}n_2, 1{:}n_3, 1{:}n_4) \\ \mathcal{B} &= \mathcal{B}(1{:}m_1, 1{:}m_2, 1{:}m_3, 1{:}m_4, 1{:}m_5) \\ \Rightarrow \mathcal{C} &= \mathcal{C}(1{:}n_1, 1{:}m_2, 1{:}n_3, 1{:}n_4, 1{:}m_3, 1{:}m_4, 1{:}m_5) \end{aligned}$$

$$\text{Order}(\mathcal{C}) \;=\; \text{Order}(\mathcal{A}) + \text{Order}(\mathcal{B}) \text{ - } 2$$

Tensor Contractions: Subscript Level

Example 2. (A Double-Index Contraction)

$$\mathcal{C}(i,j,p,q) \;=\; \sum_k \sum_s \mathcal{A}(i,s,p,k) * \mathcal{B}(s,j,k,q)$$

⇑ 4th Order ⇑ 4th Order ⇑ 4th Order

$$\text{Order}(\mathcal{C}) \;=\; \text{Order}(\mathcal{A}) + \text{Order}(\mathcal{B}) \text{ - 2 - 2}$$

Tensor Contractions: Summation Notation

Summation

If $\mathbf{L} = (L_1, \ldots, L_k)$ and $\mathbf{R} = (R_1, \ldots, R_k)$ then

$$\sum_{\mathbf{i}=\mathbf{L}}^{\mathbf{R}} \quad \equiv \quad \sum_{i_1=L_1}^{R_1} \sum_{i_2=L_2}^{R_2} \cdots \sum_{i_k=L_k}^{R_k}$$

Example: Frobenius Norm of $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$

$$\| \mathcal{A} \|_F \; = \; \sqrt{\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \mathcal{A}(\mathbf{i})^2}$$

Tensor Contractions: Subscript Level

Example 3. (A 4-Index Contraction)

$$\mathcal{C}(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$$

$$=$$

$$\sum_{\mathbf{i=1}}^{\mathbf{m}} \underset{\substack{\Uparrow \\ \text{4th} \\ \text{Order}}}{\mathcal{A}(i_1, i_2, i_3, i_4)} \cdot \underset{\substack{\Uparrow \\ \text{2nd} \\ \text{Order}}}{W(i_1, \gamma_1)} \cdot \underset{\substack{\Uparrow \\ \text{2nd} \\ \text{Order}}}{X(i_2, \gamma_2)} \cdot \underset{\substack{\Uparrow \\ \text{2nd} \\ \text{Order}}}{Y(i_3, \gamma_3)} \cdot \underset{\substack{\Uparrow \\ \text{2nd} \\ \text{Order}}}{Z(i_4, \gamma_4)}$$

$$\text{Order}(\mathcal{C}) \;=\; \text{Order}(\mathcal{A}) + \text{Order}(\mathcal{W}) + \text{Order}(\mathcal{X}) + \text{Order}(\mathcal{Y}) + \text{Order}(\mathcal{Z}) \text{ - 2 - 2 - 2 - 2}$$

Towards a High-Performance Contraction Framework

In Reality...

Tensor computations are matrix computations. A contraction can be regarded as a generalization of matrix-matrix multiplication.

The Challenge...

Expose the underlying matrix multiplications with good notation.

Two Reasons:

1. To facilitate tensor-level thinking.
2. To facilitate level-3 BLA exploitation

Let us revisit the five examples rephrasing them to expose the underlying matrix products.

Towards a High-Performance Contraction Framework

The Block Vec Unfolding

$$\text{Blockvec}(\mathcal{A}) = \left[\begin{array}{c} \mathcal{A}(:,:,1,1,1) \\ \mathcal{A}(:,:,2,1,1) \\ \mathcal{A}(:,:,1,2,1) \\ \mathcal{A}(:,:,2,2,1) \\ \mathcal{A}(:,:,1,1,2) \\ \mathcal{A}(:,:,2,1,2) \\ \mathcal{A}(:,:,1,2,2) \\ \mathcal{A}(:,:,2,2,2) \end{array}\right] \qquad \mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times 2 \times 2 \times 2}$$

The Block-Vector Product

$$\left[\begin{array}{c} F_1 \\ F_2 \end{array}\right] \circledast \left[\begin{array}{c} G_1 \\ G_2 \\ G_3 \end{array}\right] = \left[\begin{array}{c} F_1 G_1 \\ F_1 G_2 \\ F_1 G_3 \\ F_2 G_1 \\ F_2 G_2 \\ F_2 G_3 \end{array}\right]$$

Tensor Contractions: Matrix Level

Example 1.

$$\mathcal{C}(i,j,\alpha_3,\alpha_4,\beta_3,\beta_4,\beta_5) \;=\; \sum_k \mathcal{A}(i,k,\alpha_3,\alpha_4)\mathcal{B}(k,j,\beta_3,\beta_4,\beta_5)$$

As a collection of matrix-matrix products...

$$\mathcal{C}(\,:\,,\,:\,,\alpha_3,\alpha_4,\beta_3,\beta_4,\beta_5) \;=\; \mathcal{A}(\,:\,,\,:\,,\alpha_3,\alpha_4)B(\,:\,,\,:\,,\beta_3,\beta_4,\beta_5)$$

Tensor Contractions: Matrix Level

As a Block Vector Product...

$$\begin{bmatrix} \mathcal{C}(\,:\,,\,:\,,1,1,1,1,1) \\ \mathcal{C}(\,:\,,\,:\,,1,1,2,1,1) \\ \mathcal{C}(\,:\,,\,:\,,1,1,1,2,1) \\ \vdots \\ \mathcal{C}(\,:\,,\,:\,,2,2,1,2,2) \\ \mathcal{C}(\,:\,,\,:\,,2,2,2,2,2) \end{bmatrix} = \begin{bmatrix} \mathcal{A}(\,:\,,\,:\,,1,1) \\ \mathcal{A}(\,:\,,\,:\,,2,1) \\ \mathcal{A}(\,:\,,\,:\,,1,2) \\ \mathcal{A}(\,:\,,\,:\,,2,2) \end{bmatrix} \circledast \begin{bmatrix} \mathcal{B}(\,:\,,\,:\,,1,1,1) \\ \mathcal{B}(\,:\,,\,:\,,2,1,1) \\ \mathcal{B}(\,:\,,\,:\,,1,2,1) \\ \mathcal{B}(\,:\,,\,:\,,2,2,1) \\ \mathcal{B}(\,:\,,\,:\,,1,1,2) \\ \mathcal{B}(\,:\,,\,:\,,2,1,2) \\ \mathcal{B}(\,:\,,\,:\,,1,2,2) \\ \mathcal{B}(\,:\,,\,:\,,2,2,2) \end{bmatrix}$$

32 matrix-matrix products

Problem 3.10. Suppose $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ and $\mathcal{B} \in \mathbb{R}^{m_1 \times \cdots \times m_e}$ with $n_2 = m_1$. If

$$\mathcal{C}(\alpha_1, \beta_2, \alpha_3, \ldots, \alpha_d, \beta_3, \ldots, \beta_e) \;=\; \sum_{k=1}^{n_2} \mathcal{A}(\alpha_1, k, \alpha_3, \ldots, \alpha_d) \cdot \mathcal{B}(k, \beta_2, \ldots, \beta_e)$$

then we refer to $\mathcal{C}$ as the canonical contraction of $\mathcal{A}$ and $\mathcal{B}$. Write a MATLAB function `C = CanonContract(A,B)` that computes this tensor given that $\mathcal{A}$ and $\mathcal{B}$ are tensors.

Towards a High-Performance Contraction Framework

Block Matrix Unfolding for $\mathcal{A} = \mathcal{A}(1{:}n_1, 1{:}n_2, 1{:}n_3, 1{:}n_4)$

$$\text{BlockMat}(\mathcal{A}) = \begin{bmatrix} \mathcal{A}(\,:\,,\,:\,,1,1) & \mathcal{A}(\,:\,,\,:\,,1,2) & \cdots & \mathcal{A}(\,:\,,\,:\,,1,n_4) \\ \mathcal{A}(\,:\,,\,:\,,2,1) & \mathcal{A}(\,:\,,\,:\,,2,2) & \cdots & \mathcal{A}(\,:\,,\,:\,,2,n_4) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}(\,:\,,\,:\,,n_3,1) & \mathcal{A}(\,:\,,\,:\,,n_3,2) & \cdots & \mathcal{A}(\,:\,,\,:\,,n_3,n_4) \end{bmatrix}$$

$\mathcal{A}(i,j,k,\ell)$ is the (i,j) entry of block (k,ℓ)

Note that `tenmat(A,[1 3],[2 4])` produces the BlockMat unfolding.

Tensor Contractions: Matrix Level

Example 2.

$$\mathcal{C}(i,j,p,q) \;=\; \sum_k \sum_s \mathcal{A}(i,s,p,k) \cdot \mathcal{B}(s,j,k,q)$$

As a collection of matrix products and sums...

$$\mathcal{C}(\,:\,,\,:\,,p,q) \;=\; \sum_k \mathcal{A}(\,:\,,\,:\,,p,k) \cdot \mathcal{B}(\,:\,,\,:\,,k,q)$$

As a product of two block matrices...

$$\text{BlockMat}(\mathcal{C}) \;=\; \text{BlockMat}(\mathcal{A}) \cdot \text{BlockMat}(\mathcal{B})$$

Problem 3.11. Write an efficient MATLAB function `C = Eg2(A,B)` that takes two order-4 tensors $\mathcal{A}$ and $\mathcal{B}$ and produces the order-4 tensor defined in Example 2.

Tensor Contractions: Matrix Level

Example 3.

$$\mathcal{C}(m_1, m_2, m_3, m_4)$$
$$=$$
$$\sum_{\mathbf{i=1}}^{\mathbf{m}} \mathcal{A}(i_1, i_2, i_3, i_4) \cdot W(i_1, m_1) \cdot X(i_2, m_2) \cdot Y(i_3, m_3) \cdot Z(i_4, m_4)$$

As a block-matrix product...

$$\text{BlockMat}(\mathcal{C}) \;=\; (W \otimes Y) \cdot \text{BlockMat}(\mathcal{A}) \cdot (X \otimes Z)$$

Problem 3.12. Write an efficient MATLAB function `C = Eg5(A,W,X,Y,Z)` that takes an order-4 tensor $\mathcal{A}$ and four suitably dimensioned matrices W, X, Y, and Z and produces the order-4 tensor $\mathcal{C}$ defined in Example 5.

High Performance Tensor Computations

There are Locality Issues...

If the $\mathcal{B}(\,:\,,\,:\,, p, r)$ and $\mathcal{C}(\,:\,,\,:\,, r, q)$ are stored continguously, then

$$\mathcal{A}(i,j,p,q) \;=\; \sum_k \sum_s \mathcal{B}(i,s,p,k) * \mathcal{C}(s,j,k,q)$$

is nice because the participating “slices” are laid out nicely:

$$\mathcal{A}(\,:\,,\,:\,,p,q) \;=\; \sum_r B(\,:\,,\,:\,,p,r) * C(\,:\,,\,:\,,r,q)$$

Unfriendly:

$$\mathcal{A}(i,j,p,q) \;=\; \sum_k \sum_s \mathcal{B}(i,p,s,k) * \mathcal{C}(k,s,j,q)$$

LAPACK has it easy: BC, BC^T, B^TC , B^TC^T. Tensor BLAS?

High Performance Tensor Computations

There are Data Structure/Layout Issues...

Suppose$\mathcal{A} \;=\; \mathcal{A}(1{:}n_1,\ldots,1{:}n_7)$.

Can regard $\mathcal{A}$ as a 4-th order tensor with 3rd order entries. Store the 3rd order subtensors contiguously.

Can regard $\mathcal{A}$ as an aggregation of “little” 7th order tensors. (Like a block matrix.)

The what-fits-in-cache inequality:

$$m^7 \leq N_{cache}$$

Summary of Lecture 3.

Key Words

- A transposition of an order-d tensor is defined by a permutation $\mathbf{p}$ of $1{:}d$. Element $\mathbf{i}$ is mapped to element $\mathbf{p}(\mathbf{i})$.
- The Kronecker product of two matrices is a highly structured block matrix.
- Perfect shuffle permutations are a handy way to describe tensor transposition.
- A contraction between two tensors is a summation process that produces a new tensor.