Structured Matrix Computations from Structured Tensors

Lecture 1. Matrix-Tensor Connections

Charles F. Van Loan

Cornell University

CIME-EMS Summer School
June 22-26, 2015
Cetraro, Italy

Using examples, let us first take a look at what we might mean by "hidden structure" in a matrix.

Hidden Matrix Structure: Five Motivating Examples

The Discrete Fourier Transform

Definition

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega_4 & \omega_4^2 & \omega_4^3 \\ 1 & \omega_4^2 & \omega_4^4 & \omega_4^6 \\ 1 & \omega_4^3 & \omega_4^6 & \omega_4^9 \end{bmatrix} \qquad \omega_n = \cos\left(\frac{2\pi}{n}\right) - i\sin\left(\frac{2\pi}{n}\right)$$

Hidden Structure

$$F_{2m}\Pi_{2,m} = \begin{bmatrix} F_m & \Omega_m F_m \\ F_m & -\Omega_m F_m \end{bmatrix}$$

$\Pi_{2,m} =$ perfect shuffle

$\Omega_m =$ diagonal

Recursive Block Structure

The DFT Matrix is Data Sparse

The DFT matrix is dense, but can be factored into a product of sparse matrices:

$$F_{1024} \;=\; A_{10} \cdots A_2 A_1 P^T$$

The A_k have the form $I \otimes \left[\begin{array}{cc} I & D \\ I & -D \end{array} \right]$, $D =$ diagonal.

That is what makes the FFT possible:

$y = x$
for $k = 1{:}10$
$\quad y = A_k y$

An N-by-N matrix is data sparse if it can be represented with many fewer than N^2 numbers. F_N is data sparse: $O(N \log N)$ vs $O(N^2)$.

Hidden Matrix Structure: Five Motivating Examples

2. Hamiltonian Matrices

Definition

$$M = \begin{bmatrix} A & F \\ G & -A^T \end{bmatrix} \qquad F = F^T, \; G = G^T$$

Hidden Structure

$$M \begin{bmatrix} y \\ z \end{bmatrix} = \lambda \begin{bmatrix} y \\ z \end{bmatrix} \quad \Rightarrow \quad M^T \begin{bmatrix} z \\ -y \end{bmatrix} = -\lambda \begin{bmatrix} z \\ -y \end{bmatrix}$$

Eigenvalues come in plus-minus pairs.

Transformations that Preserve Structure

Equivalent Definition

$$J_{2n}^T M J_{2n} = -M^T \qquad J_{2n} = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}$$

Structured Schur Decomposition

$$\begin{bmatrix} Q_1 & Q_2 \\ -Q_2 & Q_1 \end{bmatrix}^T M \begin{bmatrix} Q_1 & Q_2 \\ -Q_2 & Q_1 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ 0 & -T_{11}^T \end{bmatrix}$$

Q is orthogonal and symplectic ($J_{2n}^T Q J_{2n} = Q^{-T}$)

Consequence: Efficient methods for Ricatti equations and various "nearness" problems.

Hidden Matrix Structure: Five Motivating Examples

3. Cauchy Matrices

Definition

$$A = (a_{kj}) = \left(\frac{1}{\omega_k - \lambda_j}\right) = \begin{bmatrix} \frac{1}{\omega_1-\lambda_1} & \frac{1}{\omega_1-\lambda_2} & \frac{1}{\omega_1-\lambda_3} & \frac{1}{\omega_1-\lambda_4} \\ \frac{1}{\omega_2-\lambda_1} & \frac{1}{\omega_2-\lambda_2} & \frac{1}{\omega_2-\lambda_3} & \frac{1}{\omega_2-\lambda_4} \\ \frac{1}{\omega_3-\lambda_1} & \frac{1}{\omega_3-\lambda_2} & \frac{1}{\omega_3-\lambda_3} & \frac{1}{\omega_3-\lambda_4} \\ \frac{1}{\omega_4-\lambda_1} & \frac{1}{\omega_4-\lambda_2} & \frac{1}{\omega_4-\lambda_3} & \frac{1}{\omega_4-\lambda_4} \end{bmatrix}$$

Hidden Structure

$$\Omega A - A\Lambda = \text{Rank-1} \qquad \Omega = \text{diag}(\omega_i),\ \Lambda = \text{diag}(\lambda_i),$$

With respect to Ω and Λ, A has displacement rank equal to one.

Hidden Matrix Structure: Five Motivating Examples

Fast LU

First Step:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \ell_{21} & 1 & 0 & 0 \\ \ell_{31} & 0 & 1 & 0 \\ \ell_{41} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & b_{22} & b_{23} & b_{24} \\ 0 & b_{32} & b_{33} & b_{34} \\ 0 & b_{42} & b_{43} & b_{44} \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Do not compute B as the usual rank-1 update of $A(2{:}n, 2{:}n)$. That would be $O(n^2)$

Instead, use the fact that B has unit displacement rank.

The displacement rank representation of B costs $O(n)$

By working with a clever representations it is sometimes possible to dramatically improve efficiency.

4. Matrices with Orthonormal Columns

Definition

$$Q \;=\; \left[\begin{array}{c} Q_1 \\ Q_2 \end{array}\right] \qquad Q_1^T Q_1 + Q_2^T Q_2 \;=\; I$$

Hidden Structure

$$\left[\begin{array}{cc} U_1 & 0 \\ 0 & U_2 \end{array}\right]^T \left[\begin{array}{c} Q_1 \\ Q_2 \end{array}\right] V \;=\; \left[\begin{array}{c} \mathrm{diag}(c_i) \\ \mathrm{diag}(s_i) \end{array}\right] \qquad c_i^2 + s_i^2 = 1$$

$$U_1,\; U_2,\; V = \text{orthogonal}$$

Q_1 and Q_2 have related SVDs. This is the CS Decomposition.

Hidden Matrix Structure: Five Motivating Examples

Simultaneous Diagonalization of A_1 and A_2

1. QR factorization: $\begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} R$

2. CS decomposition: $\begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix}^T \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} V = \begin{bmatrix} \text{diag}(c_i) \\ \text{diag}(s_i) \end{bmatrix}$

3. Setting $X = R^T V$ gives the generalized singular value decomposition:

$$A_1 = U_1 \cdot \text{diag}(c_i) \cdot X^T \qquad A_2 = U_2 \cdot \text{diag}(s_i) \cdot X^T$$

An example where exploiting the hidden structure of Q_1 and Q_2 ensures numerical stability.

Hidden Matrix Structure: Five Motivating Examples

5. Block Matrices

Definition

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & A_{22} & \cdots & A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{M1} & A_{M2} & \cdots & A_{MN} \end{bmatrix}.$$

Each A_{ij} a matrix.

Hidden Structure

The data in each A_{ij} is contiguous in memory.

Not a hidden "math" structure but a "man-made" hidden data structure.

Hidden Matrix Structure: Five Motivating Examples

Respect Data Layout to Minimize Memory Traffic

$$A \leftarrow \begin{bmatrix} A_{11}^T & A_{12}^T & \cdots & A_{1N}^T \\ A_{21}^T & A_{22}^T & \cdots & A_{2N}^T \\ \vdots & \vdots & \ddots & \vdots \\ A_{M1}^T & A_{M2}^T & \cdots & A_{MN}^T \end{bmatrix}.$$

Overwrite A_{ij} with A_{ij}^T.

$$A \leftarrow \begin{bmatrix} A_{11}^T & A_{21}^T & \cdots & A_{M1}^T \\ A_{12}^T & A_{22}^T & \cdots & A_{M2}^T \\ \vdots & \vdots & \ddots & \vdots \\ A_{1N}^T & A_{2N}^T & \cdots & A_{MN}^T \end{bmatrix}.$$

Swap A_{ij}^T with A_{ji}^T

A 2-pass transpose that exploits the “hidden” data structure.

Hidden Structure in Matrices

Each of these examples has a connection to our agenda:

Monday

Lecture 1. Matrix-tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

The Plan

Monday

Lecture 1. Matrix-Tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

Data Sparsity

The Plan

Monday

Lecture 1. Matrix-Tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

Structured Permutation Similarity

The Plan

Monday

Lecture 1. Matrix-Tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

A Higher-Order CS Decompositions

The Plan

Monday

Lecture 1. Matrix-Tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

Blocking for Data Locality

The Plan

Monday

Lecture 1. Matrix-Tensor Connections

Lecture 2. Tensor Symmetries and Rank

Tuesday

Lecture 3. The Tucker and Tensor Train Representations

Lecture 4. The CP and KSVD Representations

Thursday

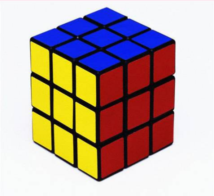
Lecture 5. Unfolding a Tensor with Multiple Symmetries

Lecture 6. A Higher-Order GSVD

Clever Representations

Let us Begin!

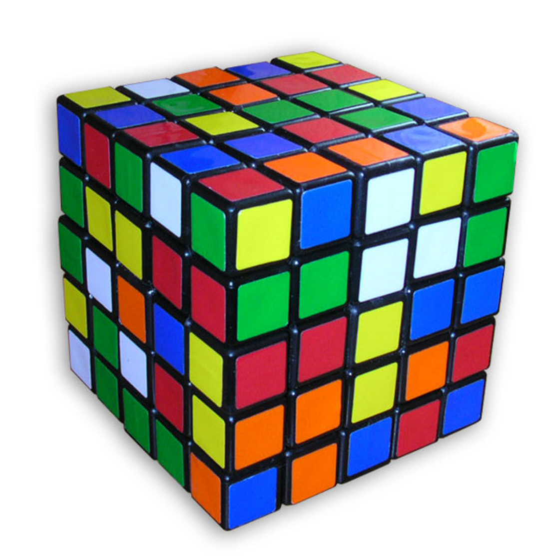
Matrix-Tensor Connections

U^T  V $=$

8	3	5	4	1	6	9	2	7
2	9	6	8	5	7	4	3	1
4	1	7	2	9	3	6	5	8
5	6	9	1	3	4	7	8	2
1	2	3	6	7	8	5	4	9
7	4	8	5	2	9	1	6	3
6	5	2	7	8	1	3	9	4
9	8	1	3	4	5	2	7	6
3	7	4	9	6	2	8	1	5

Much of the discussion will revolve around SVD-like operations.

What is a Tensor?



What is a Tensor?

Definition

An order-d tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ is a real d-dimensional array

$$\mathcal{A}(1{:}n_1, \ldots, 1{:}n_d)$$

where the index range in the k-th mode is from 1 to n_k.

Low-Order Tensors

A scalar is an order-0 tensor.

A vector is an order-1 tensor.

A matrix is an order-2 tensor.

We use calligraphic font to designate tensors that have order 3 or greater e.g., $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, etc.

Fibers

A fiber of a tensor $\mathcal{A}$ is a vector obtained by fixing all but one $\mathcal{A}$'s indices. For example, if $\mathcal{A} = \mathcal{A}(1{:}3, 1{:}5, 1{:}4, 1{:}7)$, then

$$\mathcal{A}(2, :, 4, 6) \;=\; \mathcal{A}(2, 1{:}5, 4, 6) \;=\; \left[\begin{array}{c} \mathcal{A}(2, 1, 4, 6) \\ \mathcal{A}(2, 2, 4, 6) \\ \mathcal{A}(2, 3, 4, 6) \\ \mathcal{A}(2, 4, 4, 6) \\ \mathcal{A}(2, 5, 4, 6) \end{array}\right]$$

is a fiber.

Parts of a Tensor

Slices

A slice of a tensor $\mathcal{A}$ is a matrix obtained by fixing all but two of $\mathcal{A}$'s indices. For example, if $\mathcal{A} = \mathcal{A}(1{:}3, 1{:}5, 1{:}4, 1{:}7)$, then

$$\mathcal{A}(:,3,:,6) = \begin{bmatrix} \mathcal{A}(1,3,1,6) & \mathcal{A}(1,3,2,6) & \mathcal{A}(1,3,3,6) & \mathcal{A}(1,3,4,6) \\ \mathcal{A}(2,3,1,6) & \mathcal{A}(2,3,2,6) & \mathcal{A}(2,3,3,6) & \mathcal{A}(2,3,4,6) \\ \mathcal{A}(3,3,1,6) & \mathcal{A}(3,3,2,6) & \mathcal{A}(3,3,3,6) & \mathcal{A}(3,3,4,6) \end{bmatrix}$$

is a slice.

Where Might They Come From?

Discretization

$\mathcal{A}(i,j,k,\ell)$ might house the value of $f(w,x,y,z)$ at $(w,x,y,z) = (w_i, x_j, y_k, z_\ell)$.

Multiway Analysis

$\mathcal{A}(i,j,k,\ell)$ is a value that captures an interaction between four variables/factors.

You Have Seen them Before

Block Matrices (With Uniformly-Sized Blocks)

$$A = \left[\begin{array}{cc|cc|cc} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \\ \hline a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} \\ a_{41} & a_{42} & a_{43} & a_{44} & \boxed{\mathbf{a}_{45}} & a_{46} \\ \hline a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} \end{array}\right]$$

Matrix entry a_{45} is the (2,1) entry of the (2,3) block:

$$a_{45} \quad \Leftrightarrow \quad \mathcal{A}(2,3,2,1)$$

You Have Seen Them Before

Kronecker Products (At the Scalar Level)

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

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

Matrix A is an **unfolding** of tensor $\mathcal{A}$ where $\mathcal{A}(p,q,r,s) = b_{pq}c_{rs}$.

You Have Seen Them Before

Kronecker Products (At the Block Level)

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

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

Matrix A is a block matrix whose ij block is $b_{ij}C$.

You Have Seen Them Before

Matrix: $A = B \otimes C \otimes D$

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

Hierarchy: A is a 2-by-2 block matrix whose entries are 4-by-4 block matrices whose entries are 3-by-3 matrices.

Tensor: $\mathcal{A} = \mathcal{D} \circ \mathcal{C} \circ \mathcal{B}$

$$\mathcal{A}(i_1, i_2, i_3, i_4, i_5, i_6) = \mathcal{D}(i_1, i_2)\mathcal{C}(i_3, i_4)\mathcal{B}(i_5, i_6)$$

A First Look at Tensor Symmetry

Let's look at the connection between Kronecker products and tensors when symmetry is present.

$A = B \otimes C$ with Symmetric B and C

$$A = B \otimes C = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix} \otimes \begin{bmatrix} 11 & 12 & 13 \\ 12 & 14 & 15 \\ 13 & 15 & 16 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 11 & 12 & 13 & 22 & 24 & 26 & 33 & 36 & 39 \\ 12 & 14 & 15 & 24 & 28 & 30 & 36 & 42 & 45 \\ 13 & 15 & 16 & 26 & 30 & 32 & 39 & 45 & 48 \\ \hline 22 & 24 & 26 & 44 & 48 & 52 & 55 & 60 & 65 \\ 24 & 28 & 30 & 48 & 56 & 60 & 60 & 70 & 75 \\ 26 & 30 & 32 & 52 & 60 & 64 & 65 & 75 & 80 \\ \hline 33 & 36 & 39 & 55 & 60 & 65 & 66 & 72 & 78 \\ 36 & 42 & 45 & 60 & 70 & 75 & 72 & 84 & 90 \\ 39 & 45 & 48 & 65 & 75 & 80 & 78 & 90 & 96 \end{array}\right]$$

$A = B \otimes C$ with Symmetric B and C

$$A = B \otimes C = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix} \otimes \begin{bmatrix} 11 & 12 & 13 \\ 12 & 14 & 15 \\ 13 & 15 & 16 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 11 & 12 & 13 & 22 & 24 & 26 & 33 & 36 & 39 \\ 12 & 14 & 15 & 24 & 28 & 30 & 36 & 42 & 45 \\ 13 & 15 & 16 & 26 & 30 & 32 & 39 & 45 & 48 \\ \hline 22 & 24 & 26 & 44 & 48 & 52 & \color{red}{55} & \color{red}{60} & \color{red}{65} \\ 24 & 28 & 30 & 48 & 56 & 60 & \color{red}{60} & \color{red}{70} & \color{red}{75} \\ 26 & 30 & 32 & 52 & 60 & 64 & \color{red}{65} & \color{red}{75} & \color{red}{80} \\ \hline 33 & 36 & 39 & 55 & 60 & 65 & 66 & 72 & 78 \\ 36 & 42 & 45 & 60 & 70 & 75 & 72 & 84 & 90 \\ 39 & 45 & 48 & 65 & 75 & 80 & 78 & 90 & 96 \end{array}\right]$$

Each block is symmetric.

$A = B \otimes C$ with Symmetric B and C

$$A = B \otimes C = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix} \otimes \begin{bmatrix} 11 & 12 & 13 \\ 12 & 14 & 15 \\ 13 & 15 & 16 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 11 & 12 & 13 & 22 & 24 & 26 & 33 & 36 & 39 \\ 12 & 14 & 15 & 24 & 28 & 30 & 36 & 42 & 45 \\ 13 & 15 & 16 & 26 & 30 & 32 & 39 & 45 & 48 \\ \hline 22 & 24 & 26 & 44 & 48 & 52 & \color{red}{55} & \color{red}{60} & \color{red}{65} \\ 24 & 28 & 30 & 48 & 56 & 60 & \color{red}{60} & \color{red}{70} & \color{red}{75} \\ 26 & 30 & 32 & 52 & 60 & 64 & \color{red}{65} & \color{red}{75} & \color{red}{80} \\ \hline 33 & 36 & 39 & \color{red}{55} & \color{red}{60} & \color{red}{65} & 66 & 72 & 78 \\ 36 & 42 & 45 & \color{red}{60} & \color{red}{70} & \color{red}{75} & 72 & 84 & 90 \\ 39 & 45 & 48 & \color{red}{65} & \color{red}{75} & \color{red}{80} & 78 & 90 & 96 \end{array}\right]$$

Block (i, j) equals Block (j, i)

$A = B \otimes C$ with Symmetric B and C

$$A = B \otimes C = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix} \otimes \begin{bmatrix} 11 & 12 & 13 \\ 12 & 14 & 15 \\ 13 & 15 & 16 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 11 & 12 & 13 & 22 & 24 & 26 & 33 & 36 & 39 \\ 12 & 14 & 15 & 24 & 28 & 30 & 36 & 42 & 45 \\ 13 & 15 & 16 & 26 & 30 & 32 & 39 & 45 & 48 \\ \hline 22 & 24 & 26 & 44 & 48 & 52 & 55 & 60 & 65 \\ 24 & 28 & 30 & 48 & 56 & 60 & 60 & 70 & 75 \\ 26 & 30 & 32 & 52 & 60 & 64 & 65 & 75 & 80 \\ \hline 33 & 36 & 39 & 55 & 60 & 65 & 66 & 72 & 78 \\ 36 & 42 & 45 & 60 & 70 & 75 & 72 & 84 & 90 \\ 39 & 45 & 48 & 65 & 75 & 80 & 78 & 90 & 96 \end{array}\right]$$

If $\mathcal{A}(p,q,r,s) = b_{pq}c_{rs}$ then $\mathcal{A}(p,q,r,s) = \begin{cases} \mathcal{A}(q,p,r,s) \\ \mathcal{A}(p,q,s,r) \end{cases}$

$A = B \otimes B$ with Symmetric B

$$A = B \otimes B = \begin{bmatrix} 4 & 5 & 6 \\ 5 & 7 & 8 \\ 6 & 8 & 9 \end{bmatrix} \otimes \begin{bmatrix} 4 & 5 & 6 \\ 5 & 7 & 8 \\ 6 & 8 & 9 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 16 & 20 & 24 & 20 & 25 & 30 & 24 & 30 & 36 \\ 20 & 28 & \boxed{\mathbf{32}} & 25 & 35 & \boxed{\mathbf{40}} & 30 & 42 & \boxed{\mathbf{48}} \\ 24 & 32 & 36 & 30 & 40 & 45 & 36 & 48 & 54 \\ \hline 20 & 25 & 30 & 28 & 35 & 42 & 32 & 40 & 48 \\ 25 & 35 & \boxed{\mathbf{40}} & 35 & 49 & \boxed{\mathbf{56}} & 40 & 56 & \boxed{\mathbf{64}} \\ 30 & 40 & 45 & 42 & 56 & 63 & 48 & 64 & 72 \\ \hline 24 & 30 & 36 & 32 & 40 & 48 & 36 & 45 & 54 \\ 30 & 42 & \boxed{\mathbf{48}} & 40 & 56 & \boxed{\mathbf{64}} & 45 & 63 & \boxed{\mathbf{72}} \\ 36 & 48 & 54 & 48 & 64 & 72 & 54 & 72 & 81 \end{array}\right]$$

$\text{Block}(i,j) = A(i{:}n{:}n^2, j{:}n{:}n^2)$ $\qquad$ $\text{Block}(2,3) = A(2{:}3{:}9, 3{:}3{:}9)$

$A = B \otimes B$ with Symmetric B

$$A \;=\; B \otimes B \;=\; \begin{bmatrix} 4 & 5 & 6 \\ 5 & 7 & 8 \\ 6 & 8 & 9 \end{bmatrix} \otimes \begin{bmatrix} 4 & 5 & 6 \\ 5 & 7 & 8 \\ 6 & 8 & 9 \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc|ccc} 16 & 20 & 24 & 20 & 25 & 30 & 24 & 30 & 36 \\ 20 & 28 & 32 & 25 & 35 & 40 & 30 & 42 & 48 \\ 24 & 32 & 36 & 30 & 40 & 45 & 36 & 48 & 54 \\ \hline 20 & 25 & 30 & 28 & 35 & 42 & 32 & 40 & 48 \\ 25 & 35 & 40 & 35 & 49 & 56 & 40 & 56 & 64 \\ 30 & 40 & 45 & 42 & 56 & 63 & 48 & 64 & 72 \\ \hline 24 & 30 & 36 & 32 & 40 & 48 & 36 & 45 & 54 \\ 30 & 42 & 48 & 40 & 56 & 64 & 45 & 63 & 72 \\ 36 & 48 & 54 & 48 & 64 & 72 & 54 & 72 & 81 \end{array}\right]$$

If $\mathcal{A}(p,q,r,s) = b_{pq} b_{rs}$ then $\mathcal{A}(p,q,r,s) \;=\; \begin{cases} \mathcal{A}(q,p,r,s) \\ \mathcal{A}(p,q,s,r) \\ \mathcal{A}(r,s,p,q) \end{cases}$

A First Look at Tensor Symmetry

For a matrix, there is only one type of symmetry:

$$A(p,q) = A(q,p)$$

For an order-d tensor, there are $d! - 1$ possibilities:

$$\mathcal{A}(p,q,r,s) \;=\; \left\{ \begin{array}{c} \mathcal{A}(q,p,r,s) \\ \mathcal{A}(r,q,p,r) \\ \mathcal{A}(s,q,r,p) \\ \vdots \end{array} \right.$$

Next, let's look at the connection between Kronecker products and tensors in the rank-1 setting.

Rank-1 Reshaping

If u and v are vectors, then $A = uv^T$ is a Rank-1 Matrix

$$A = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}^T = \begin{bmatrix} u_1 v_1 & u_1 v_2 \\ u_2 v_1 & u_2 v_2 \\ u_3 v_1 & u_3 v_2 \end{bmatrix}$$

A is a rank-1 matrix

$A = uv^T \Rightarrow \text{vec}(A) = v \otimes u$

$$\begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{12} \\ a_{22} \\ a_{32} \end{bmatrix} = \begin{bmatrix} u_1 v_1 \\ u_2 v_1 \\ u_3 v_1 \\ u_1 v_2 \\ u_2 v_2 \\ u_3 v_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \otimes \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

In the Language of Tensor Products

If u and v are vectors then $\mathcal{A} = u \circ v$ is a Rank-1 Tensor

$$\mathcal{A}(i_1, i_2) \;=\; u(i_1)v(i_2)$$

$$\mathcal{A} \;=\; u \circ v \;=\; \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \circ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad \Leftrightarrow \quad \text{vec}(\mathcal{A}) \;=\; \begin{bmatrix} u_1 v_1 \\ u_2 v_1 \\ u_3 v_1 \\ u_1 v_2 \\ u_2 v_2 \\ u_3 v_2 \end{bmatrix}$$

Higher-Order Rank-1 Tensors

If u, v, and w are vectors, then $\mathcal{A} = u \circ v \circ w$ is a Rank-1 Tensor

$$\mathcal{A}(p,q,r) \;=\; u_p v_q w_r$$

$$\mathcal{A} = u \circ v \circ w = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \circ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \circ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad \Rightarrow \quad \text{vec}(\mathcal{A}) = \begin{bmatrix} u_1 v_1 w_1 \\ u_2 v_1 w_1 \\ u_1 v_2 w_1 \\ u_2 v_2 w_1 \\ u_1 v_1 w_2 \\ u_2 v_1 w_2 \\ u_1 v_2 w_2 \\ u_2 v_2 w_2 \end{bmatrix}$$

A tensor product of d vectors produces an order-d rank-1 tensor.

A Notation Detail: u-v-w versus w-v-u

$$\mathrm{vec}(u \circ v \circ w) \equiv \begin{bmatrix} u_1 v_1 w_1 \\ u_2 v_1 w_1 \\ u_1 v_2 w_1 \\ u_2 v_2 w_1 \\ u_1 v_1 w_2 \\ u_2 v_1 w_2 \\ u_1 v_2 w_2 \\ u_2 v_2 w_2 \end{bmatrix} = w \otimes v \otimes u$$

Let's look at how we might compute the the nearest rank-1 tensor to a given tensor.

The Nearest Rank-1 Problem for Matrices

Formulation:

Given $A \in \mathbb{R}^{m \times n}$, find unit-2 norm vectors $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$ and a nonnegative scalar σ that minimizes

$$\phi(\sigma, u, v) = \| A - \sigma u v^T \|_F.$$

SVD Solution:

If $U^T A V \; = \; \Sigma = \text{diag}(\sigma_i)$ where

$$U = [u_1 \mid \cdots \mid u_m] \qquad V = [v_1 \mid \cdots \mid v_n]$$

are orthogonal and $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$, then $\sigma_{\text{opt}} u_{\text{opt}} v_{\text{opt}}^T \; = \; \sigma_1 u_1 v_1^T$.

An Alternating Least Squares Approach

$v =$ unit vector

Repeat Until Happy:

% Fix v and choose σ and u to minimize $\| A - \sigma u v^T \|_F$

$x = Av; \quad \sigma = \| x \|; \; u = x/\sigma$

% Fix u and choose σ and v to minimize $\| A - \sigma u v^T \|_F$

$x = A^T u; \; \sigma = \| x \|; \; v = x/\sigma$

$\sigma_{\text{opt}} = \sigma; \; u_{\text{opt}} = u; \; v_{\text{opt}} = v$

$$\| A - \sigma u v^T \|_F^2 \; = \; trace(A^T A) - 2\sigma u^T A v + \sigma^2$$

The best u is in the direction of Av. The best v is in the direction of $A^T u$.

The Nearest Rank-1 Problem for Matrices

An Alternating Least Squares Approach

v = unit vector

Repeat Until Happy:

% Fix v and choose σ and u to minimize $\| A - \sigma u v^T \|_F$

$x = Av; \quad \sigma = \| x \|; \; u = x/\sigma$

% Fix u and choose σ and v to minimize $\| A - \sigma u v^T \|_F$

$x = A^T u; \; \sigma = \| x \|; \; v = x/\sigma$

$\sigma_{\text{opt}} = \sigma; \; u_{\text{opt}} = u; \; v_{\text{opt}} = v$

This is just the power method applied to $A^T A$:

$$x = (A^T A)v, \; v = x/\| x \|$$

Nearest Rank-1 Problem for Tensors

Formulation

Given $\mathcal{A} \in \mathbb{R}^{m \times n \times p}$, determine unit vectors $u \in \mathbb{R}^m$, $v \in \mathbb{R}^n$, and $w \in \mathbb{R}^p$ and scalar σ so that the following is minimized:

$$\| \mathcal{A} - \sigma \cdot w \circ v \circ u \|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} (a_{ijk} - u_i v_j w_k) \right)^{1/2}$$

$$= \| \text{vec}(\mathcal{A}) - \sigma \cdot w \otimes v \otimes u \|_2$$

Nearest Rank-1 Problem for Tensors

Alternating Least Squares Framework for min $\| \operatorname{vec}(\mathcal{A}) - \sigma \cdot w \otimes v \otimes u \|_2$

v and w given unit vectors

Repeat Until Happy

Determine $x \in \mathbb{R}^m$ that minimizes $\| \operatorname{vec}(\mathcal{A}) - w \otimes v \otimes x \|_2$

and set $\sigma = \| x \|$ and $u = x/\sigma$

Determine $y \in \mathbb{R}^n$ that minimizes $\| \operatorname{vec}(\mathcal{A}) - w \otimes y \otimes u \|_2$

and set $\sigma = \| y \|$ and $v = y/\sigma$

Determine $z \in \mathbb{R}^p$ that minimizes $\| \operatorname{vec}(\mathcal{A}) - z \otimes v \otimes u \|_2$

and set $\sigma = \| z \|$ and $w = z/\sigma$

Details in next Lecture. For now, we look at the special structure of these linear least square problems for the case $m = n = p = 2$.

The Nearest Rank-1 Problem for Tensors

The Case $m = n = p = 2$

$$\text{minimize} \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \sigma \cdot w \otimes v \otimes u \right\|_2$$

$$u = \begin{bmatrix} \cos(\theta_1) \\ \sin(\theta_1) \end{bmatrix} = \begin{bmatrix} c_1 \\ s_1 \end{bmatrix} \quad v = \begin{bmatrix} \cos(\theta_2) \\ \sin(\theta_2) \end{bmatrix} = \begin{bmatrix} c_2 \\ s_2 \end{bmatrix} \quad w = \begin{bmatrix} \cos(\theta_3) \\ \sin(\theta_3) \end{bmatrix} = \begin{bmatrix} c_3 \\ s_3 \end{bmatrix}$$

A Highly Structured Nonlinear Optimization Problem

It Depends on Four Parameters...

$$\phi(\sigma, \theta_1, \theta_2, \theta_3) = \left\| a - \sigma \begin{bmatrix} \cos(\theta_3) \\ \sin(\theta_3) \end{bmatrix} \otimes \begin{bmatrix} \cos(\theta_2) \\ \sin(\theta_2) \end{bmatrix} \otimes \begin{bmatrix} \cos(\theta_1) \\ \sin(\theta_1) \end{bmatrix} \right\|_2$$

$$= \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \sigma \cdot \begin{bmatrix} c_3 c_2 c_1 \\ c_3 c_2 s_1 \\ c_3 s_2 c_1 \\ c_3 s_2 s_1 \\ s_3 c_2 c_1 \\ s_3 c_2 s_1 \\ s_3 s_2 c_1 \\ s_3 s_2 s_1 \end{bmatrix} \right\|_2$$

A Highly Structured Nonlinear Optimization Problem

Set $x_1 = \sigma\cos(\theta_1)$ and $y_1 = \sigma\sin(\theta_1)$ and then Reshape...

$$\phi = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \sigma \cdot \begin{bmatrix} c_3c_2c_1 \\ c_3c_2s_1 \\ c_3s_2c_1 \\ c_3s_2s_1 \\ s_3c_2c_1 \\ s_3c_2s_1 \\ s_3s_2c_1 \\ s_3s_2s_1 \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \begin{bmatrix} c_3c_2 & 0 \\ 0 & c_3c_2 \\ c_3s_2 & 0 \\ 0 & c_3s_2 \\ s_3c_2 & 0 \\ 0 & s_3c_2 \\ s_3s_2 & 0 \\ 0 & s_3s_2 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \right\|_2$$

This is an ordinary linear least squares problem for x_1 and y_1 if we "freeze" θ_2 and θ_3. Solve and update σ and u_1 using

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \sigma u_1 \qquad \sigma = \sqrt{x_1^2 + y_1^2}$$

A Highly Structured Nonlinear Optimization Problem

Set $x_2 = \sigma\cos(\theta_2)$ and $y_2 = \sigma\sin(\theta_2)$ and then Reshape...

$$\phi = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \sigma \cdot \begin{bmatrix} c_3c_2c_1 \\ c_3c_2s_1 \\ c_3s_2c_1 \\ c_3s_2s_1 \\ s_3c_2c_1 \\ s_3c_2s_1 \\ s_3s_2c_1 \\ s_3s_2s_1 \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \begin{bmatrix} c_3c_1 & 0 \\ c_3s_1 & 0 \\ 0 & c_3c_1 \\ 0 & c_3s_1 \\ s_3c_1 & 0 \\ s_3s_1 & 0 \\ 0 & s_3c_1 \\ 0 & s_3s_1 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \right\|_2$$

This is an ordinary linear least squares problem for x_2 and y_2 if we "freeze" θ_1 and θ_3. Solve and update σ and u_2 using

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \sigma u_2 \qquad \sigma = \sqrt{x_2^2 + y_2^2}$$

A Highly Structured Nonlinear Optimization Problem

Set $x_3 = \sigma\cos(\theta_3)$ and $y_3 = \sigma\sin(\theta_3)$ and then Reshape...

$$\phi = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \sigma \cdot \begin{bmatrix} c_3 c_2 c_1 \\ c_3 c_2 s_1 \\ c_3 s_2 c_1 \\ c_3 s_2 s_1 \\ s_3 c_2 c_1 \\ s_3 c_2 s_1 \\ s_3 s_2 c_1 \\ s_3 s_2 s_1 \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} a_{111} \\ a_{211} \\ a_{121} \\ a_{221} \\ a_{112} \\ a_{212} \\ a_{122} \\ a_{222} \end{bmatrix} - \begin{bmatrix} c_2 c_1 & 0 \\ c_2 s_1 & 0 \\ s_2 c_1 & 0 \\ s_2 s_1 & 0 \\ 0 & c_2 s_1 \\ 0 & c_2 s_1 \\ 0 & s_2 c_1 \\ 0 & s_2 s_1 \end{bmatrix} \begin{bmatrix} x_3 \\ y_3 \end{bmatrix} \right\|_2$$

This is an ordinary linear least squares problem for x_3 and y_3 if we "freeze" θ_1 and θ_2. Solve and update σ and u_3 using

$$\begin{bmatrix} x_3 \\ y_3 \end{bmatrix} = \sigma u_3 \qquad \sigma = \sqrt{x_3^2 + y_3^2}$$

Componentwise Optimization

A Common Framework for Tensor-Related Optimization

- Choose a subset of the unknowns such that if they are (temporarily) fixed, then we are presented with some standard matrix problem in the remaining unknowns.
- By choosing different subsets, cycle through all the unknowns.
- Repeat until converged.

In tensor computations, the "standard matrix problem" that we end up solving is usually the linear least squares problem. In that case, the overall solution process is referred to as alternating least squares.

Optional "Fun" Problems

Problem E1. Consider the the three linear least (LS) squares problems that arise when the alternating least squares framework is applied to the 2-by-2-by-2 problem. Outline a solution approach when these linear LS problems are solved using the method of normal equations. (Recall that the method of normal equations for the LS problem min $\| Mu - b \|_2$ involves solving the symmetric positive definite linear system $M^T Mu = M^T b$.)

Problem A1. Repeat E1 but when $\mathcal{A} \in \mathbb{R}^{2\times 2\times \cdots \times 2}$ is an order-d tensor.

Closing Remarks

Where Do We Go From Here?

To sums of rank-1's...

$$\text{vec}(\mathcal{A}) = \sum_{k=1}^{r} \sigma_k w_k \otimes v_k \otimes u_k$$

To more general unfoldings...

$$\mathcal{A} \in \mathbb{R}^{4\times 2\times 3} \;\Rightarrow\; \begin{bmatrix} a_{111} & a_{121} & a_{131} & a_{112} & a_{122} & a_{132} \\ a_{211} & a_{221} & a_{231} & a_{212} & a_{222} & a_{232} \\ a_{311} & a_{321} & a_{331} & a_{312} & a_{322} & a_{332} \\ a_{411} & a_{421} & a_{431} & a_{412} & a_{422} & a_{432} \end{bmatrix}$$

To more complicated multilinear optimizations...

$$\min_{\substack{U,V,W \in \mathbb{R}^{n\times n} \text{ orthogonal} \\ s \in \mathbb{R}^{n^3}}} \| \text{vec}(\mathcal{A}) - (W \otimes V \otimes U)s \|_2$$

How Will the Structured Matrix Computations Show Up?

Tensor computations are typically disguised matrix computations and that is because of

Kronecker Products

$$A = A_1 \otimes A_2 \otimes A_3 \qquad \text{an order 6 tensor}$$

Tensor Unfoldings

$$\text{Rubik Cube} \quad \longrightarrow \quad 3 \times 9 \quad \text{matrix}$$

Alternating Least Squares

Multilinear optimization via component-wise linear optimization

These are the three ways that structured tensor computations will lead to structured matrix computations.

Context

Preparation for the Next Big Thing...

Scalar-Level Thinking

1960's $\Downarrow$ $\Leftarrow$ The factorization paradigm: LU, LDL^T, QR, $U\Sigma V^T$, etc.

Matrix-Level Thinking

1980's $\Downarrow$ $\Leftarrow$ Cache utilization, parallel computing, LAPACK, etc.

Block Matrix-Level Thinking

2000's $\Downarrow$ $\Leftarrow$ New applications, factorizations, data structures, nonlinear analysis, optimization strategies, etc.

Tensor-Level Thinking

More Context

A Changing Definition of "Big"

In Matrix Computations, to say that $A \in \mathbb{R}^{n_1 \times n_2}$ is "big" is to say that both n_1 and n_2 are big.

In Tensor Computations, to say that $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ is "big" is to say that $n_1 n_2 \cdots n_d$ is big and this need not require big n_k. E.g. $n_1 = n_2 = \cdots = n_{1000} = 2$.

Algorithms that scale with d will induce a transition...

Matrix-Based Scientific Computation

$\Downarrow$

Tensor-Based Scientific Computation