CSE 392: Matrix and Tensor Algorithms for Data

Instructor: Shashanka Ubaru

University of Texas, Austin
Spring 2024

Lecture 21: Tube-fiber product, t-product.

Outline

Algebraic Context - Semantics

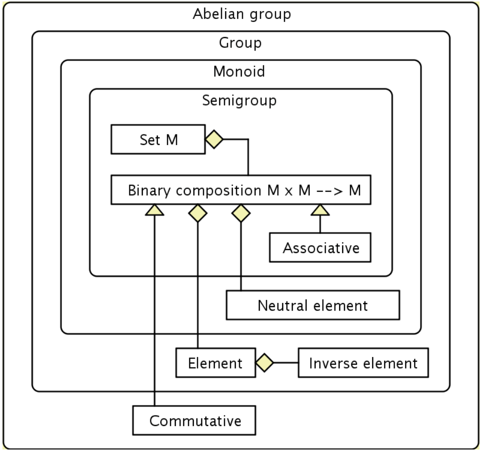
A **binary composition** $*$ on a set $\mathcal{S}$ is a function $\mathcal{S} \times \mathcal{S} \ni (x, y) \mapsto x * y \in \mathcal{S}$

Semigroup: A set $\mathcal{S}$ with a binary composition $*$ that is associative

Monoid: A semigroup which has a unit element

Group: A monoid where each element has an inverse

Abelian group: A group whose binary composition is commutative



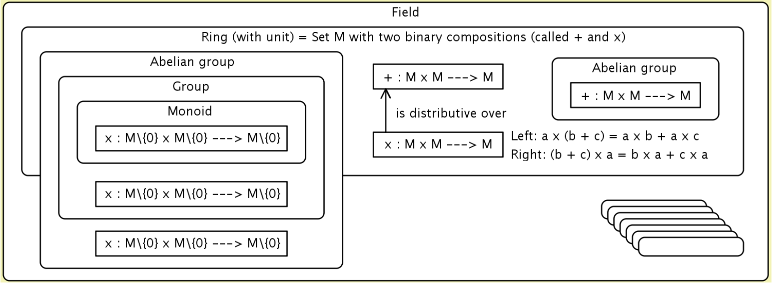
Algebraic Context - Ring and Field

Ring: A set $\mathcal{R}$ with two binary compositions, called addition (denoted by $+$) and multiplication such that $(\mathcal{R}, +)$ is an abelian group, $(\mathcal{R}, \cdot)$ is a semigroup, and multiplication is distributive over addition

Ring with unit: A ring $\mathcal{R}$ where $(\mathcal{R}, \cdot)$ is a monoid

Commutative ring: A ring with commutative multiplication

Field: A commutative ring with unit where each nnz element has a multiplicative inverse



Algebraic Context - Module and Vector Space

Module: An abelian group $(\mathcal{M}, \oplus)$ is called a module over the ring $(\mathcal{R}, +, \cdot)$, or an $\mathcal{R}$ - module, if there is a function $\mathcal{R} \times \mathcal{M} \ni (r, m) \mapsto rm \in \mathcal{M}$, such that

(i) : $0m = 0$ where 0 is the additive unit of $\mathcal{R}$

(ii) : $1m = m$ if $\mathcal{R}$ has a multiplicative unit 1

(iii) : $(r + r')m = (rm) \oplus (r'm)$

(iv) : $r(m \oplus m') = (rm) \oplus (rm')$

(v) : $(r \cdot r')m = r(r'm)$

Vector space: A module over a field

Algebraic Context - Algebra

Algebra - A structure comprising of addition, multiplication, and scalar multiplication (may also include additional assumptions as associativity, commutativity, etc.)

Going forward we shall employ some of these constructs to define new tensorial algebras...

Step Back to the Matrix SVD

Traditional workhorse, dim reduction/feature extraction: matrix singular value decomposition (SVD)

- PCA - directions of most variability; projections in 'dominant' directions allows for dim reduction/relative comparison
- Compression (reduce near redundancies) via truncated SVD expansion is optimal (Eckart-Young)

For $m \times n$ $\mathbf{A}$, and $\mathbf{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^\top = \sum_{i=1}^{r} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)})$,

$$\mathbf{B} = \sum_{i=1}^{p} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)}) \qquad \text{solves}$$

$$\min \|\mathbf{A} - \mathbf{B}\|_F \qquad \text{s.t. } \mathbf{B} \text{ has rank } p \leq r$$

Step Back to the Matrix SVD

Traditional workhorse, dim reduction/feature extraction: matrix singular value decomposition (SVD)

- PCA - directions of most variability; projections in 'dominant' directions allows for dim reduction/relative comparison
- Compression (reduce near redundancies) via truncated SVD expansion is optimal (Eckart-Young)

For $m \times n$ $\mathbf{A}$, and $\mathbf{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^\top = \sum_{i=1}^{r} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)})$,

$$\mathbf{B} = \sum_{i=1}^{p} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)}) \qquad \text{solves}$$

$$\min \|\mathbf{A} - \mathbf{B}\|_F \qquad \text{s.t. } \mathbf{B} \text{ has rank } p \leq r$$

Implicit storage: only $O(p(n+m))$ numbers stored, vs mn.

Step Back to the Matrix SVD

Traditional workhorse, dim reduction/feature extraction: matrix singular value decomposition (SVD)

- PCA - directions of most variability; projections in 'dominant' directions allows for dim reduction/relative comparison
- Compression (reduce near redundancies) via truncated SVD expansion is optimal (Eckart-Young)

For $m \times n$ $\mathbf{A}$, and $\mathbf{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^\top = \sum_{i=1}^{r} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)})$,

$$\mathbf{B} = \sum_{i=1}^{p} \sigma_i(\mathbf{u}^{(i)} \circ \mathbf{v}^{(i)}) \qquad \text{solves}$$

$$\min \|\mathbf{A} - \mathbf{B}\|_F \qquad \text{s.t. } \mathbf{B} \text{ has rank } p \leq r$$

Extension to higher dimensions?

An Ideal Tensor Algebra

Compressible — Powerful representation

Efficient — Simple implementation

Optimal — Provable superiority

Mimetic — Matrix property preservation

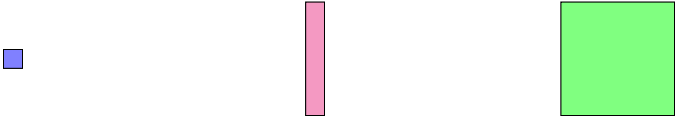
References

Main Reference: Kilmer, Horesh, Avron, Newman: “Tensor-tensor Algebra for Optimal Representation and Compression of Multiway Data,” *PNAS*, 2021.

Originated with work:

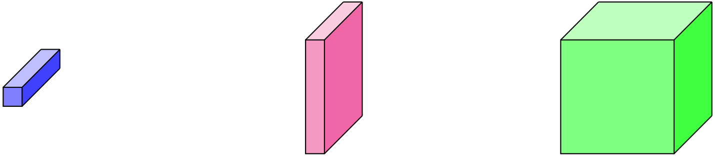
- Kilmer and Martin: “Factorization strategies for third-order tensors,” *LAA*, 2011
- Kernfeld, Kilmer and Aeron, “Tensor-tensor products with invertible linear transforms,” *LAA*, 2015

Mimeticity | yticitemiM



scalar a	vector $\boldsymbol{a}$	matrix $\boldsymbol{A}$
1×1	$\ell \times 1$	$\ell \times m$

Mimeticity | yticitemiM

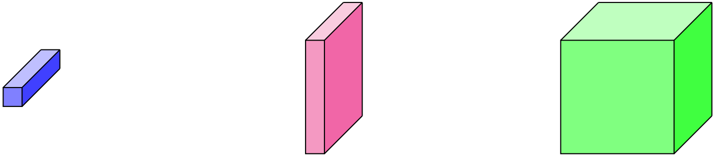


tube $\boldsymbol{a}$ — $1 \times 1 \times n$

lateral slice $\vec{\boldsymbol{\mathcal{A}}}$ — $\ell \times 1 \times n$

tensor $\boldsymbol{\mathcal{A}}$ — $\ell \times m \times n$

Mimeticity | yticitemiM



tube $\boldsymbol{a}$ $1 \times 1 \times n$

lateral slice $\vec{\boldsymbol{\mathcal{A}}}$ $\ell \times 1 \times n$

tensor $\boldsymbol{\mathcal{A}}$ $\ell \times m \times n$

Matrices are linear operators $\leftrightarrow$ Tensors are t-linear operators

Orientation Dependence

CP, Tucker were orientation independent. For the remainder, we fix the orientation, think of a tensor as a matrix of tube fibers; matrix of lateral slices.

Tensor $\mathcal{A}$ Frontal slice $\boldsymbol{A}^{(k)}$ Lateral slice $\vec{\mathcal{A}}_j$ Tube fibers $\boldsymbol{a}_{ij}$

Using MATLAB-like notation: e.g. $\vec{\mathcal{A}}_j = \mathcal{A}_{:,j,:}$.

Find a way to express a tensor that leads to the possibility for optimally compressed representation that maintains important features of the original tensor.

Towards an Elemental Operation

We first consider a simple case - an option for the product between two tube fibers!

Circulant Matrices and Convolution

Let $\boldsymbol{v} \in \mathbb{R}^n$. Then the $n \times n$ circulant matrix generated by $\boldsymbol{v}$ is

$$\mathbf{C} = \mathrm{circ}(\boldsymbol{v}) = \begin{bmatrix} v_1 & v_n & v_{n-1} & \cdots & v_2 \\ v_2 & v_1 & v_n & \cdots & v_3 \\ \vdots & \ddots & \ddots & \ddots & v_4 \\ v_n & v_{n-1} & v_{n-2} & \cdots & v_1 \end{bmatrix}$$

Well known to be diagonalized by the (unitary) DFT matrix[1]:

$$\mathbf{C} = \mathbf{F}^H \Lambda \mathbf{F}$$

where the eigenvalues can be computed from the fast-Fourier transform[2] (FFT) of $\boldsymbol{v}$.

[1] $\mathbf{F}$ is a Vandermonde matrix formed from the nth roots of unity. To compute this in Matlab use `F = 1/sqrt(n) * fft(eye(n))` .

[2] $\hat{\boldsymbol{v}} = \texttt{fft}(\boldsymbol{v}) \equiv \sqrt{n}\mathbf{F}$

Tensor Product for Tube Fibers

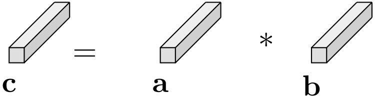
Discrete convolution between $\mathbf{a}, \mathbf{b} \in \mathbb{C}^n$:

$\mathbf{a} * \mathbf{b} := \text{circ}(\mathbf{a})\mathbf{b} = \mathbf{F}^H \text{diag}(\hat{\mathbf{a}})\mathbf{F}\mathbf{b} = \frac{1}{\sqrt{n}}\mathbf{F}^H(\text{diag}(\hat{\mathbf{a}})\hat{\mathbf{b}}) = \frac{1}{\sqrt{n}}\mathbf{F}^H(\hat{\mathbf{a}} \odot \hat{\mathbf{b}}) = \texttt{ifft}(\hat{\mathbf{c}})$, where $\hat{\mathbf{c}} = \hat{\mathbf{a}} \odot \hat{\mathbf{b}}$.

Discrete Convolution between two vectors is commutative (Exercise: show this.)

1-1 correspondence between $1 \times 1 \times n$ tube fiber (element in $\mathbb{K}_n$) and a vector in $\mathbb{C}^n$.

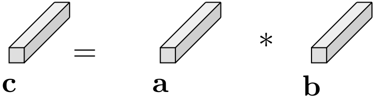
Let's define the product of 2, length-n tube fibers to be convolution!



We have a **commutative ring**. There is an identity element $\mathbf{e}$ (Exercise: what is it?)

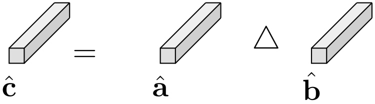
Tube-fiber Product

Elemental action: Compute



The work can be done by n independent scalar multiplications in the Fourier domain as follows.

Tube-fiber Product Computation



where $\triangle$ denotes face-wise scalar multiplication, analogous to the $\odot$ notation between two vectors.

Then $\mathbf{c}$ is obtained by applying the inverse transform.

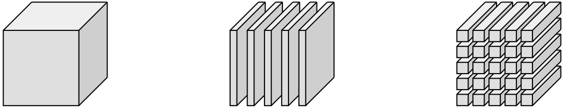
The algorithmic cost of the operation: cost to compute $\hat{\mathbf{a}}, \hat{\mathbf{b}}$ $O(n \lg n)$ flops, the cost for the n scalar products at $O(n)$ flops, and the cost of the inverse transform on $\hat{\mathbf{c}}$ $O(n \lg n)$ flops.

Exercise

Show that the t-product of two tube fibers can be expressed using the mode-wise notation as follows:

- Apply a DFT along the tube fibers $\mathbf{a}, \mathbf{b}$:
  - Compute $\mathbf{a} \times_3 \mathbf{F}$, and $\mathbf{b} \times_3 \mathbf{F}$
- Pointwise multiply the entries, call the result $\hat{\mathbf{c}}$
- Apply inverse DFT to $\hat{\mathbf{c}}$: $\hat{\mathbf{c}} \times_3 \mathbf{F}^{-1}$.

Algebraic Context - Algebra

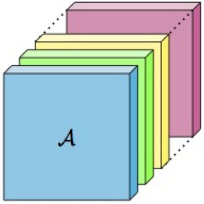


Tensor $\mathcal{A}$ Lateral slice $\vec{\mathcal{A}}_j$ Tube fibers $\mathbf{a}_{ij}$

Next we define an entirely different tensor decomposition based on **circulant algebra**

In this factorization, a tensor in $\mathbb{R}^{n_1 \times n_2 \times n_3}$ is viewed as a $n_1 \times n_2$ **matrix of "tubes"** also known as **elements** of the **ring** $\mathbb{K}_{n_3}$ where addition is defined as vector addition and multiplication as **circular convolution**.

This "matrix-of-tubes" formalism leads to definitions of a **new multiplication for tensors** ("tubal multiplication"), a **new rank** for tensors ("tubal rank"), and a new notion of a **SVD for tensors** ("tubal SVD")



$$\xrightarrow{\texttt{circ}(\mathcal{A})} \begin{pmatrix} A^{(1)} & A^{(n)} & \dots & A^{(3)} & A^{(2)} \\ A^{(2)} & A^{(1)} & A^{(n)} & \dots & A^{(3)} \\ A^{(3)} & A^{(2)} & A^{(1)} & \ddots & \dots \\ \vdots & \ddots & \ddots & \ddots & A^{(n)} \\ A^{(n)} & \dots & A^{(3)} & A^{(2)} & A^{(1)} \end{pmatrix}$$

The t-product is defined as:

$$\mathcal{A} * \mathcal{B} = \texttt{fold}(\texttt{circ}(\mathcal{A}) \cdot \texttt{unfold}(\mathcal{B})).$$

It is obvious that if $\mathcal{A}$ is $m \times p \times n$, need $\mathcal{B}$ to be $p \times k \times n$, and the result is $m \times k \times n$.

Kilmer, Martin, **Factorization Strategies for Third-Order Tensors**, LAA, 2011

Block Circulants

- A **block circulant** can be **block-diagonalized** by a (normalized) DFT in the 2^{nd} dimension:

$$(\boldsymbol{F} \otimes \boldsymbol{I})\mathtt{circ}(\mathcal{A})(\boldsymbol{F}^* \otimes \boldsymbol{I}) = \begin{bmatrix} \hat{\boldsymbol{A}}_1 & 0 & \cdots & 0 \\ 0 & \hat{\boldsymbol{A}}_2 & 0 & \cdots \\ 0 & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & \hat{\boldsymbol{A}}_n \end{bmatrix}$$

- Here $\otimes$ is a Kronecker product of matrices
- If $\boldsymbol{F}$ is $n \times n$, and $\boldsymbol{I}$ is $m \times m$, $(\boldsymbol{F} \otimes \boldsymbol{I})$ is the $mn \times mn$ block matrix, of n block rows and columns, each block is $m \times m$, where the ij^{th} block is $f_{i,j}\boldsymbol{I}$
- In practice, one never explicitly implement it this way because an FFT along tube fibers of $\mathcal{A}$ yields a tensor, $\hat{\mathcal{A}}$, whose frontal slices are the $\hat{\boldsymbol{A}}_i$

T-product, General Case

$$\mathcal{A} * \mathcal{B} = \texttt{fold}(\texttt{circ}(\mathcal{A}) \cdot \texttt{unfold}(\mathcal{B})).$$

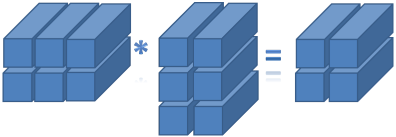
Since the term in the middle can be computed using the previous observation (compare to the case tube-fiber special case), we have:

The t-product can be computed in the Fourier domain with n, independent matrix-matrix products $\hat{\boldsymbol{A}}_i \cdot \hat{\boldsymbol{B}}_i$, $i = 1, \ldots, n$, and then moving back to the spatial domain with an inverse transform of the result.

T-product

Block circulants block-diagonalized via 1D FFTs $\Rightarrow$ The t-product can be computed in-place using FFTs:

- $\widehat{\mathcal{A}} \leftarrow \texttt{fft}(\mathcal{A}, [\,], 3)$
- $\widehat{\mathcal{B}} \leftarrow \texttt{fft}(\mathcal{B}, [\,], 3)$
- $\widehat{\mathcal{C}}_{:,:,i} = \widehat{\mathcal{A}}_{:,:,i} \cdot \widehat{\mathcal{B}}_{:,:,i}, \; i = 1, \ldots, n$
- $\mathcal{C} = \texttt{ifft}(\widehat{\mathcal{C}}, [\,], 3)$



Framework

Now that the t-product is defined, additional linear-algebraic like framework:

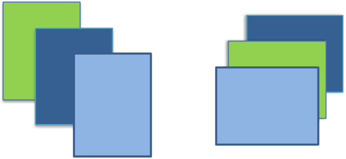
The $\ell \times \ell \times n$ **identity tensor** $\mathcal{I}$ is the tensor whose frontal slice is the $\ell \times \ell$ identity matrix, and whose other frontal slices are all zeros. (Exercise: show this.)

An $\ell \times \ell \times n$ tensor $\mathcal{A}$ has an **inverse** $\mathcal{B}$ provided that

$$\mathcal{A} * \mathcal{B} = \mathcal{I}, \qquad \text{and} \qquad \mathcal{B} * \mathcal{A} = \mathcal{I}.$$

Transpose and Orthogonality

$\mathcal{A} \in \mathbb{R}^{\ell \times m \times n} \Rightarrow \mathcal{A}^\top \in \mathbb{R}^{m \times \ell \times n}$ is obtained by transposing each frontal slice & reversing order of transposed frontal slices 2 through n.



$\mathcal{U} \in \mathbb{R}^{m \times m \times n}$ is **orthogonal** if $\mathcal{U}^\top * \mathcal{U} = \mathcal{I} = \mathcal{U} * \mathcal{U}^\top$.

Can show **Frobenius norm invariance**: $\|\mathcal{U} * \mathcal{A}\|_F = \|\mathcal{A}\|_F$.
Exercise: show $(\mathcal{A} * \mathcal{B})^\top = \mathcal{B}^\top * \mathcal{A}^\top$

Questions?