Tensor

Reporter: Morin Wang

Supervisor: Zenglin Xu

Contents

- Notations and Operations
- Tensor Decomposition and Algorithm
- Application

What is tensor？

- A tensor is something that transforms like a tensor!(Physics focuses more on Tensor Field.)

$$A' = TAT^{-1}$$

- A tensor is a multidimensional or N-way array.(More easily)

Why tensor？

- Curse of dimensionality

  large-scale, multi-modal and multi-relational

- Tensor decompositions, Tensor networks

  -As emerging tools for dimensionality reduction and large scale optimization problems.

  -Tensor networks offer a theoretical and computational framework

  for the analysis of computationally prohibitive large volumes of data.

  - Super-compression of datasets as large as 10^50 entries, down to the affordable levels of 10^7 or even less entries.

Some Useful Notations

- A N-order(N-way or N-Mode) Tensor:

$$\mathcal{X} \in R^{I_1 \times I_2 \cdots I_N}$$

- Its $(i_1,\ i_2 \cdots i_N)$ element：

$$x_{i_1 i_2 \cdots i_N}$$

- Outer product of vector $a_n \in R^{I_n}$:

$$\mathcal{X} = a_1 \circ a_2 \cdots a_N \ \in R^{I_1 \times I_2 \cdots I_N}$$

$$x_{i_1 i_2 \cdots i_N} = (a_1)_{i_1}\ (a_2)_{i_2} \cdots (a_n)_{i_n}$$

- N-Mode product (Tensor with Matrix)

$$A \in R^{I_n \times R}$$

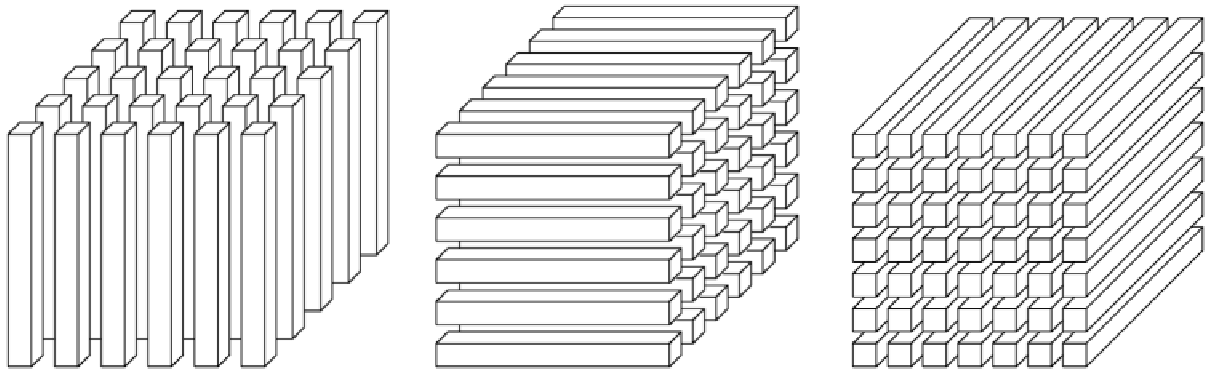
$$(\mathcal{X} \times_n A)_{i_1,i_2 \cdots i_{n-1} r i_{n+1} \cdots i_N} = \sum_{i_n}^{I_n} x_{i_1,i_2 \cdots i_n \cdots i_N} a_{i_n,r}$$

- Mode-$\binom{n}{m}$ product (Tensor and Tensor)

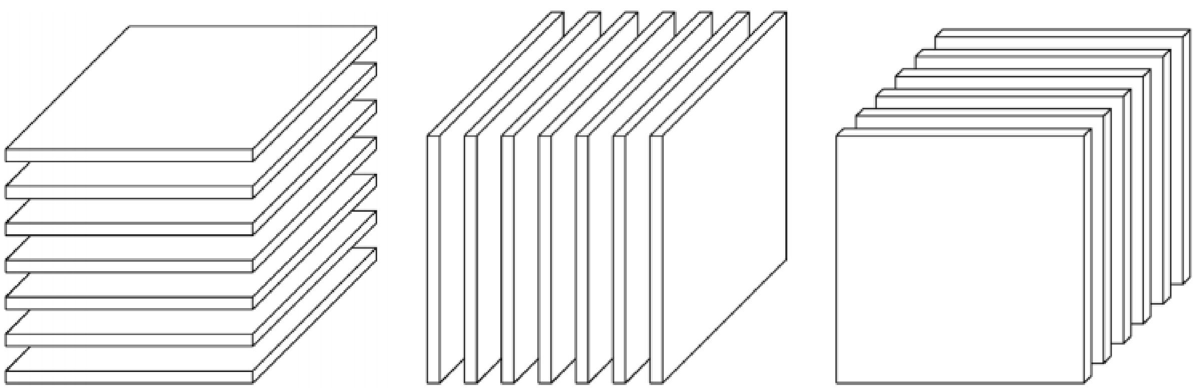
$$\mathcal{X} \in R^{I_1 \times I_2 \cdots I_N} \qquad \mathcal{C} \in R^{J_1 \times J_2 \cdots J_M} (I_n = J_m)$$

$$(\mathcal{X} \times_m^n \mathcal{C})_{i_1,i_2 \cdots i_{n-1} i_{n+1} \cdots i_N, j_1, j_2 \cdots j_{m-1} j_{m+1} \cdots i_M}$$

$$= \sum_{i_n}^{I_n} x_{i_1,i_2 \cdots i_n \cdots i_N} c_{j_1,j_2,\cdots i_n,\cdots j_M}$$

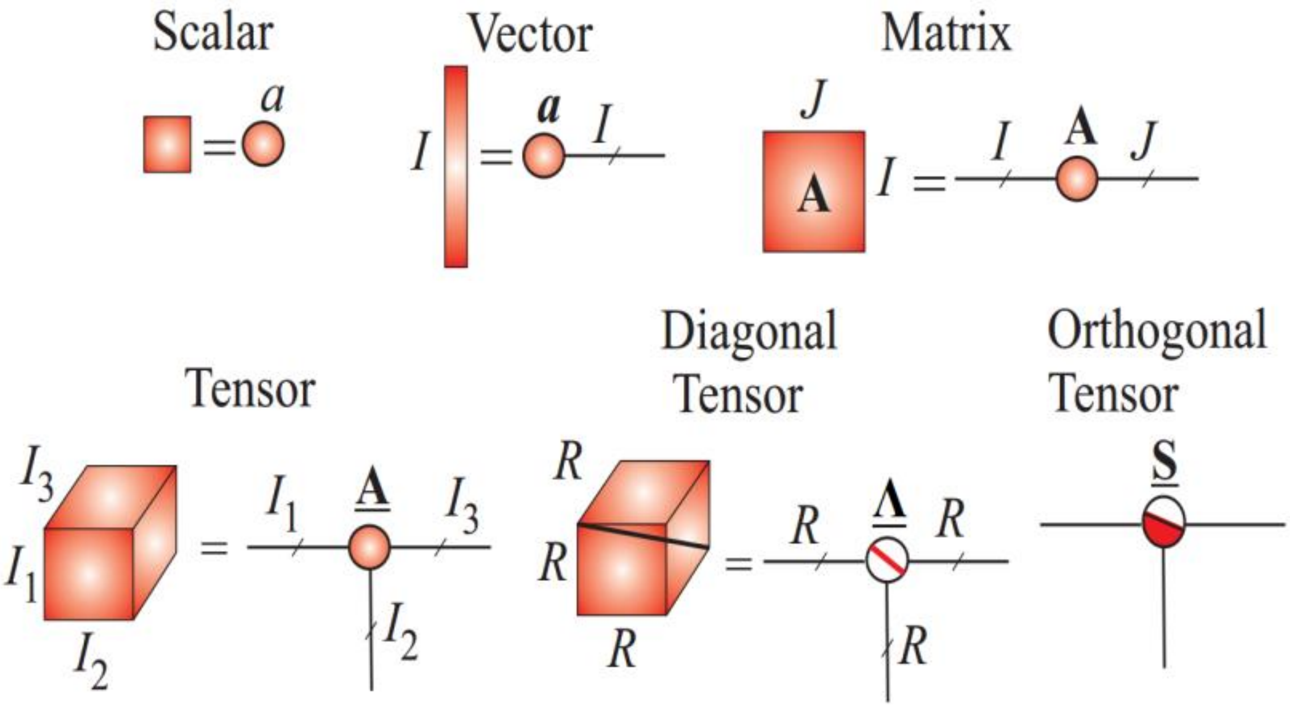


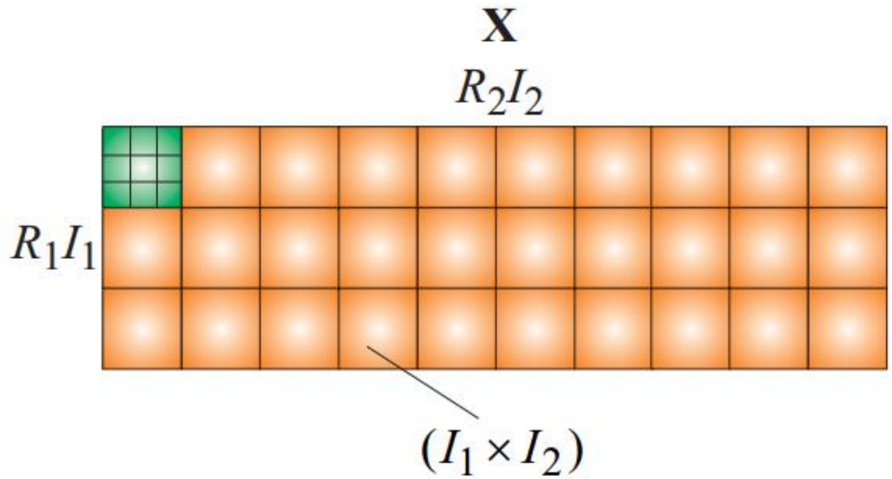
(a) Mode-1 (column) fibers: $\mathbf{x}_{:jk}$ (b) Mode-2 (row) fibers: $\mathbf{x}_{i:k}$ (c) Mode-3 (tube) fibers: $\mathbf{x}_{ij:}$



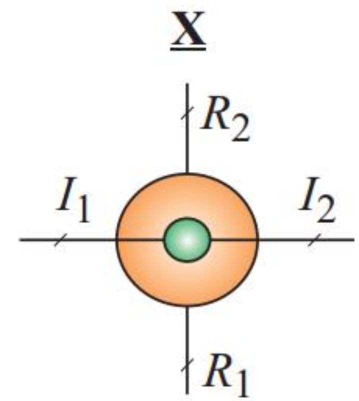
(a) Horizontal slices: $\mathbf{X}_{i::}$ (b) Lateral slices: $\mathbf{X}_{:j:}$ (c) Frontal slices: $\mathbf{X}_{::k}$ (or $\mathbf{X}_k$)

Tensor Network

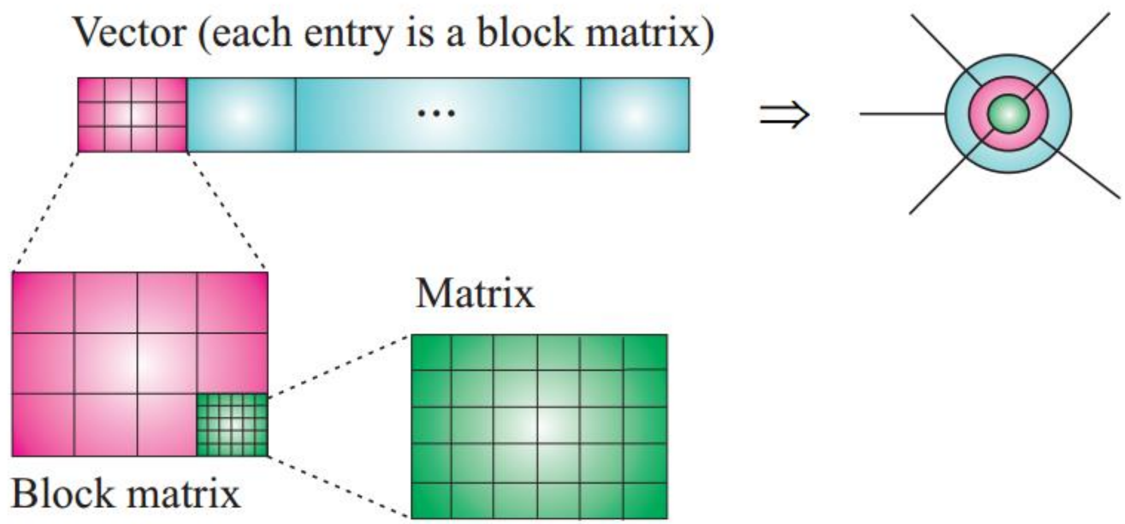




$\mathbf{X}$
R_2I_2
R_1I_1
$(I_1 \times I_2)$

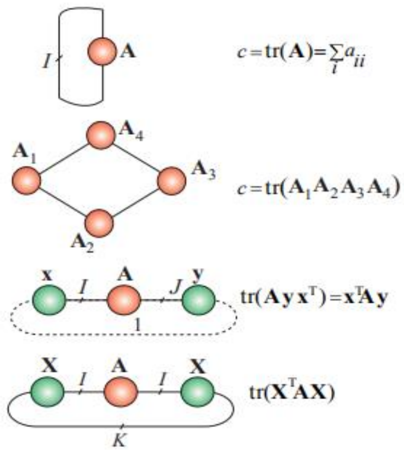


$\underline{\mathbf{X}}$
R_2
I_1
I_2
R_1

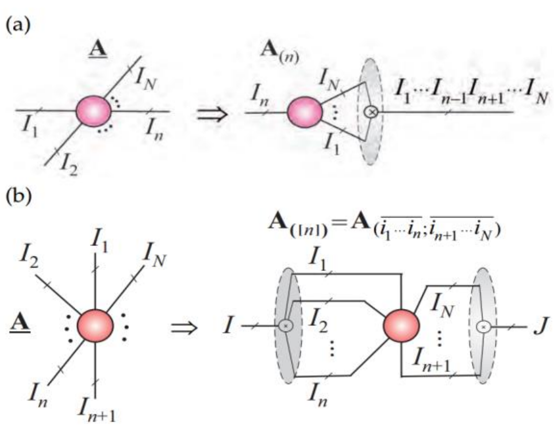


Vector (each entry is a block matrix)
⇒
Matrix
Block matrix

- Matrix trace



- Matricization



1) The little-endian convention

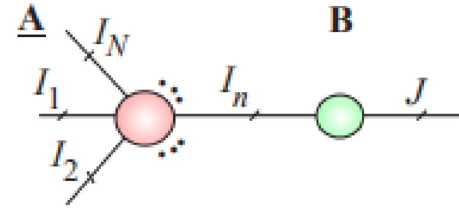
$$\overline{i_1, i_2, \ldots, i_N} = i_1 + (i_2 - 1)I_1 + (i_3 - 1)I_1 I_2 \cdots + (i_N - 1)I_1 \cdots I_{N-1}. \quad (1)$$

2) The big-endian

$$\overline{i_1, i_2, \ldots, i_N} = i_N + (i_{N-1} - 1)I_N + + (i_{N-2} - 1)I_N I_{N-1} + \cdots + (i_1 - 1)I_2 \cdots I_N. \quad (2)$$

- N-Mode product (Tensor with Matrix)

$$A \in R^{I_n \times R}$$



$\underline{\mathbf{C}} = \underline{\mathbf{A}} \times_n \mathbf{B}$

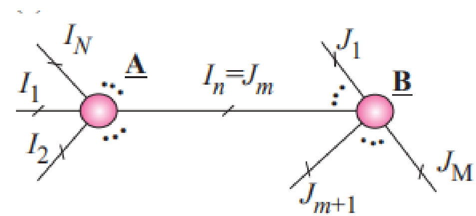
$$(\mathcal{X} \times_n A)_{i_1, i_2 \cdots i_{n-1} r i_{n+1} \cdots i_N} = \sum_{i_n}^{I_n} x_{i_1, i_2 \cdots i_n \cdots i_N} a_{i_n, r}$$

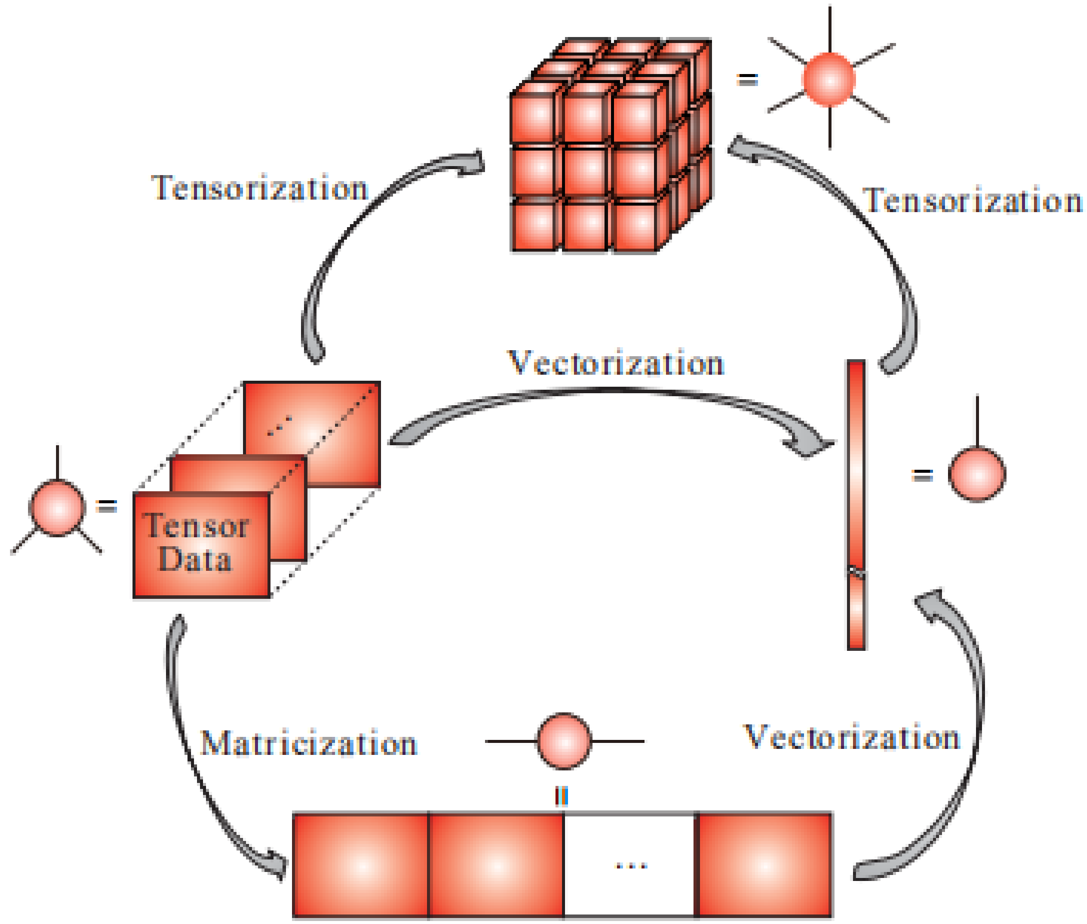
- Mode-$\binom{n}{m}$ product (Tensor and Tensor)

$$\mathcal{X} \in R^{I_1 \times I_2 \cdots I_N} \qquad \mathcal{C} \in R^{J_1 \times J_2 \cdots J_M} (I_n = J_m)$$

$$(\mathcal{X} \times_m^n \mathcal{C})_{i_1, i_2 \cdots i_{n-1} i_{n+1} \cdots i_N, j_1, j_2 \cdots j_{m-1} j_{m+1} \cdots i_M}$$

$$= \sum_{i_n}^{I_n} x_{i_1, i_2 \cdots i_n \cdots i_N} c_{j_1, j_2, \cdots i_n, \cdots j_M}$$




Tensorization
Tensorization
Vectorization
Tensor Data
Matricization
Vectorization

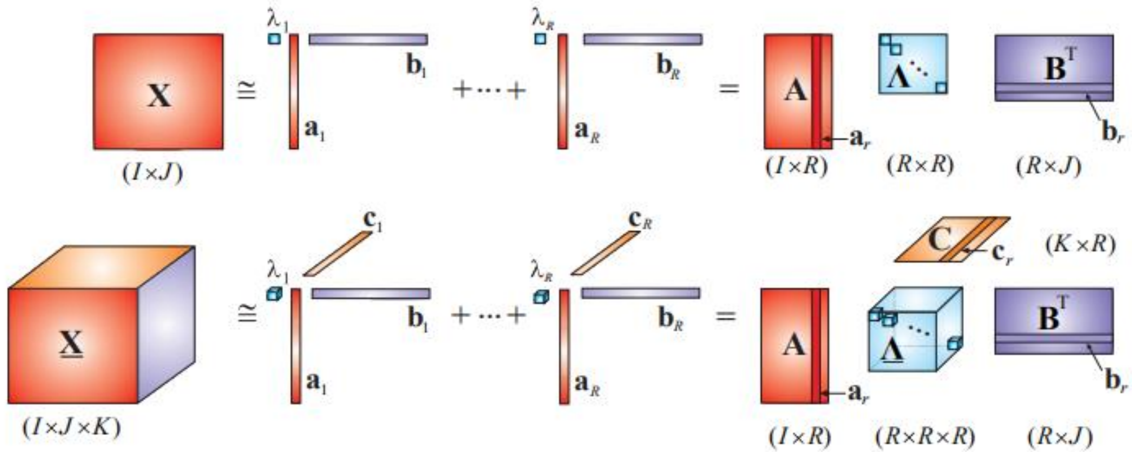
About Tensor Network

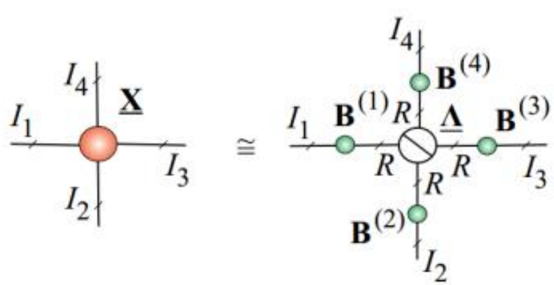
A tensor network aims to decompose a higher-order tensor into a set of lower-order tensors (typically, 2nd (matrices) and 3rd-order tensors called cores or components) which are sparsely interconnected.

Tensor Decomposition

- CP
- Tucker
- Tensor Train & Ring
- Others

CP Formats





$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} = \sum_{r=1}^{R} \sigma_r \, \mathbf{u}_r \circ \mathbf{v}_r = \sum_{r=1}^{R} \sigma_r \, \mathbf{u}_r \mathbf{v}_r^{\mathrm{T}},$$

CP ALS

Optimization:

$$min\|\mathcal{X} - \widehat{\mathcal{X}}\|_F$$

$$s.t.\, \widehat{\mathcal{X}} = \sum_{r}^{R} \lambda_r a_{1_r} \circ a_{2_r} \cdots a_{N_r} = [\![\Lambda; A_1, A_2 \dots A_N]\!]$$

$$\widehat{\mathcal{X}}_{(n)} = A_n \Lambda (A_1 \odot \cdots A_{n-1} \odot A_{n+1} \cdots \odot A_N)^{\mathrm{T}}$$

Algorithm:

Input: $\mathcal{X}$, R

Initialize: $\forall n$, A_n

Repeat:

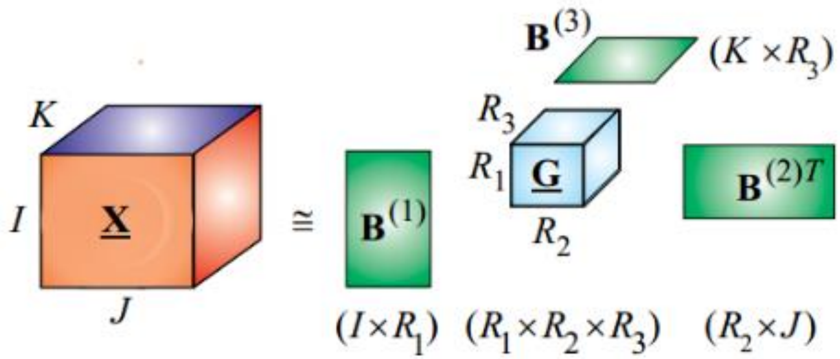
$$\mathrm{V} \longleftarrow \mathrm{A}_1^{\mathrm{T}}\mathrm{A}_1 \circledast \cdots \circledast \mathrm{A}_{n-1}^{\mathrm{T}}\mathrm{A}_{n-1} \circledast \mathrm{A}_{n+1}^{\mathrm{T}}\mathrm{A}_{n+1} \cdots \circledast \mathrm{A}_{\mathrm{N}}^{\mathrm{T}}\mathrm{A}_{\mathrm{N}}$$

$$\mathrm{A}_{\mathrm{n}} \longleftarrow \mathrm{X}_{(\mathrm{n})}(\mathrm{A}_{\mathrm{N}} \odot \cdots \odot \mathrm{A}_{\mathrm{n}+1} \odot \mathrm{A}_{\mathrm{n}-1} \cdots \odot \mathrm{A}_1)\mathrm{V}^{\dagger}$$

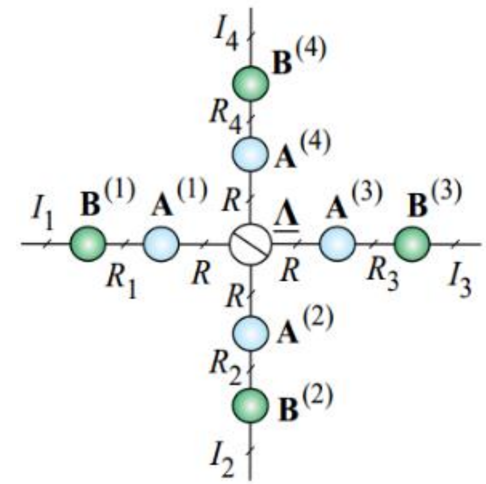
$$\lambda_{\mathrm{n}} \longleftarrow \|\mathrm{A}_{\mathrm{n}}\|$$

$$\mathrm{A}_{\mathrm{n}} \longleftarrow \frac{\mathrm{A}_{\mathrm{n}}}{\lambda_{\mathrm{n}}}$$

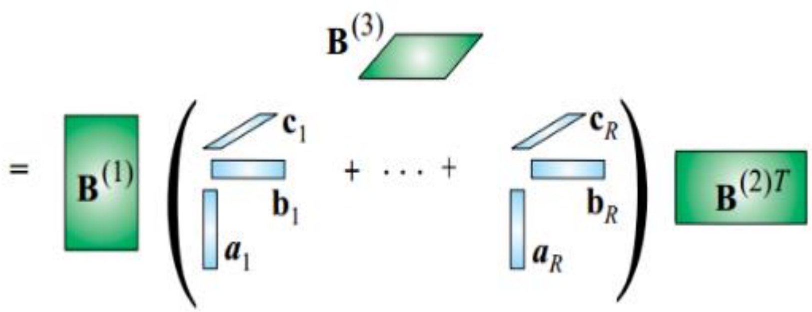
Tucker



$\mathbf{B}^{(3)}$
$(K \times R_3)$
K
I
$\underline{\mathbf{X}}$
J
$\cong$
$\mathbf{B}^{(1)}$
R_3
R_1
$\underline{\mathbf{G}}$
R_2
$\mathbf{B}^{(2)T}$
$(I \times R_1)$
$(R_1 \times R_2 \times R_3)$
$(R_2 \times J)$



I_4
$\mathbf{B}^{(4)}$
R_4
$\mathbf{A}^{(4)}$
I_1
$\mathbf{B}^{(1)}$
$\mathbf{A}^{(1)}$
R
$\underline{\mathbf{\Lambda}}$
$\mathbf{A}^{(3)}$
$\mathbf{B}^{(3)}$
R_1
R
R
R_3
I_3
R
$\mathbf{A}^{(2)}$
R_2
$\mathbf{B}^{(2)}$
I_2



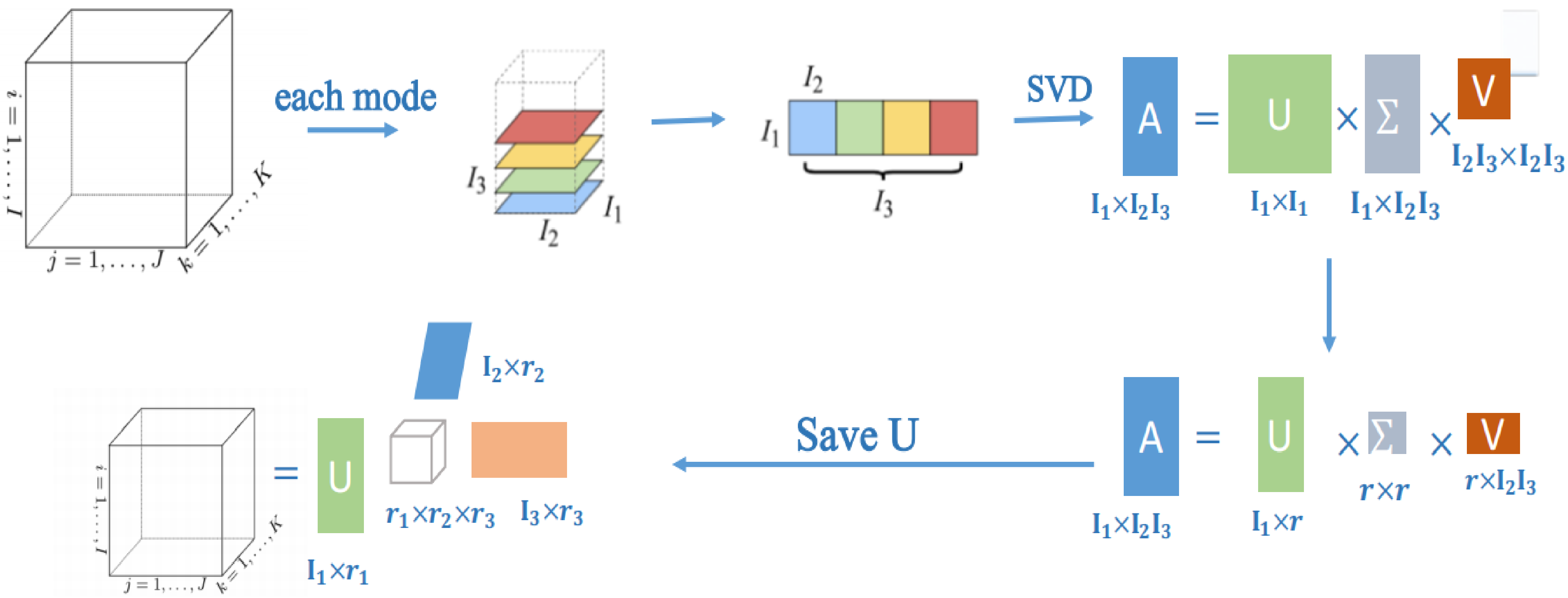
$\mathbf{B}^{(3)}$
$=$
$\mathbf{B}^{(1)}$
$\mathbf{c}_1$
$\mathbf{b}_1$
$\boldsymbol{a}_1$
$+ \cdots +$
$\mathbf{c}_R$
$\mathbf{b}_R$
$\boldsymbol{a}_R$
$\mathbf{B}^{(2)T}$

HOSVD

Optimization:

$$\min \|\mathcal{X} - \hat{\mathcal{X}}\|_F$$
$$s.t.\ \hat{\mathcal{X}} = [\![G; A_1, A_2 \dots A_N]\!]$$

$$X_{(n)} = U^{(n)} G_{(n)} \left(U^{(1)} \otimes \cdots \otimes U^{(n-1)} \otimes U^{(n+1)} \otimes \cdots \otimes U^{(N)} \right)^T$$



HOOI

Input: Nth-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ (usually in Tucker/HOSVD format)

Output: Improved Tucker approximation using ALS approach, with orthogonal factor matrices $\mathbf{U}^{(n)}$

Initialization via the standard HOSVD (see Algorithm 2)
repeat
  for $n = 1$ to N **do**
    $\underline{\mathbf{Z}} \leftarrow \underline{\mathbf{X}} \times_{p \neq n} \{\mathbf{U}^{(p)\,\mathrm{T}}\}$
    $\mathbf{C} \leftarrow \mathbf{Z}_{(n)} \mathbf{Z}_{(n)}^{\mathrm{T}} \in \mathbb{R}^{R \times R}$
    $\mathbf{U}^{(n)} \leftarrow$ leading R_n eigenvectors of $\mathbf{C}$
  end for
  $\underline{\mathbf{G}} \leftarrow \underline{\mathbf{Z}} \times_N \mathbf{U}^{(N)\,\mathrm{T}}$
until the cost function $(\|\underline{\mathbf{X}}\|_F^2 - \|\underline{\mathbf{G}}\|_F^2)$ ceases to decrease
return $[\![\underline{\mathbf{G}}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}]\!]$

Input: Nth-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and multilinear rank $\{R_1, R_2, \ldots, R_N\}$

Output: Approximative representation of a tensor in Tucker format, with orthogonal factor matrices $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}$

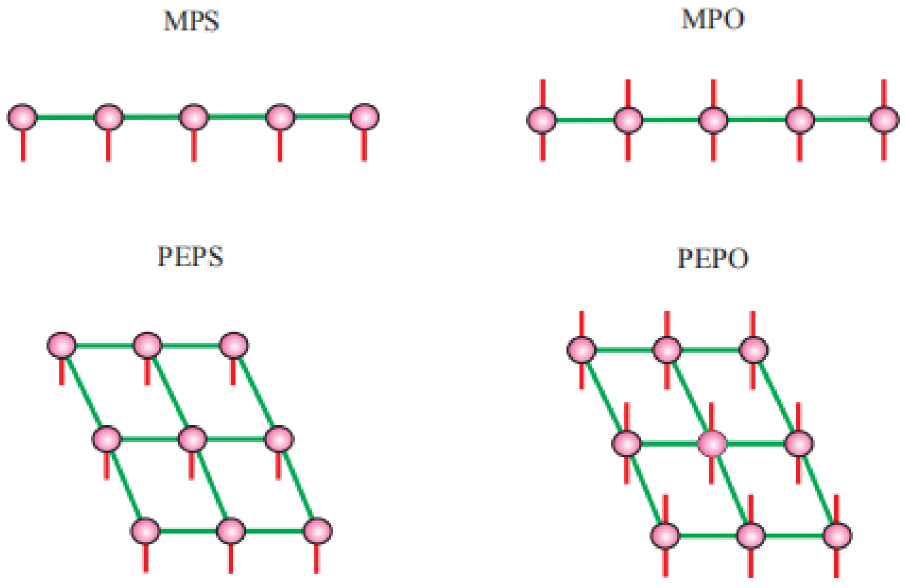
1: Initialize factor matrices $\mathbf{U}^{(n)}$ as random Gaussian matrices
  Repeat steps (2)-(6) only two times:
2: **for** $n = 1$ to N **do**
3:   $\underline{\mathbf{Z}} = \underline{\mathbf{X}} \times_{p \neq n} \{\mathbf{U}^{(p)\,\mathrm{T}}\}$
4:   Compute $\tilde{\mathbf{Z}}^{(n)} = \mathbf{Z}_{(n)} \mathbf{\Omega}^{(n)} \in \mathbb{R}^{I_n \times R_n}$, where $\mathbf{\Omega}^{(n)} \in \mathbb{R}^{\prod_{p \neq n} R_p \times R_n}$ is a random matrix drawn from Gaussian distribution
5:   Compute $\mathbf{U}^{(n)}$ as an orthonormal basis of $\tilde{\mathbf{Z}}^{(n)}$, e.g., by using QR decomposition
6: **end for**
7: Construct the core tensor as $\underline{\mathbf{G}} = \underline{\mathbf{X}} \times_1 \mathbf{U}^{(1)\,\mathrm{T}} \times_2 \mathbf{U}^{(2)\,\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\,\mathrm{T}}$
8: **return** $\underline{\mathbf{X}} \cong [\![\underline{\mathbf{G}}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \underline{\mathbf{U}}^{(N)}]\!]$

CP	Tucker
Scalar product	
$x_{i_1,\ldots,i_N} = \sum_{r=1}^{R} \lambda_r \, b_{i_1,r}^{(1)} \cdots b_{i_N,r}^{(N)}$	$x_{i_1,\ldots,i_N} = \sum_{r_1=1}^{R_1} \cdots \sum_{r_N=1}^{R_N} g_{r_1,\ldots,r_N} \, b_{i_1,r_1}^{(1)} \cdots b_{i_N,r_N}^{(N)}$
Outer product	
$\underline{\mathbf{X}} = \sum_{r=1}^{R} \lambda_r \, \mathbf{b}_r^{(1)} \circ \cdots \circ \mathbf{b}_r^{(N)}$	$\underline{\mathbf{X}} = \sum_{r_1=1}^{R_1} \cdots \sum_{r_N=1}^{R_N} g_{r_1,\ldots,r_N} \, \mathbf{b}_{r_1}^{(1)} \circ \cdots \circ \mathbf{b}_{r_N}^{(N)}$
Multilinear product	
$\underline{\mathbf{X}} = \underline{\boldsymbol{\Lambda}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \cdots \times_N \mathbf{B}^{(N)}$	$\underline{\mathbf{X}} = \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \cdots \times_N \mathbf{B}^{(N)}$
$\underline{\mathbf{X}} = [\![\underline{\boldsymbol{\Lambda}}; \mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \ldots, \mathbf{B}^{(N)}]\!]$	$\underline{\mathbf{X}} = [\![\underline{\mathbf{G}}; \mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \ldots, \mathbf{B}^{(N)}]\!]$
Vectorization	
$\mathrm{vec}(\underline{\mathbf{X}}) = \left(\bigodot_{n=N}^{1} \mathbf{B}^{(n)} \right) \boldsymbol{\lambda}$	$\mathrm{vec}(\underline{\mathbf{X}}) = \left(\bigotimes_{n=N}^{1} \mathbf{B}^{(n)} \right) \mathrm{vec}(\underline{\mathbf{G}})$

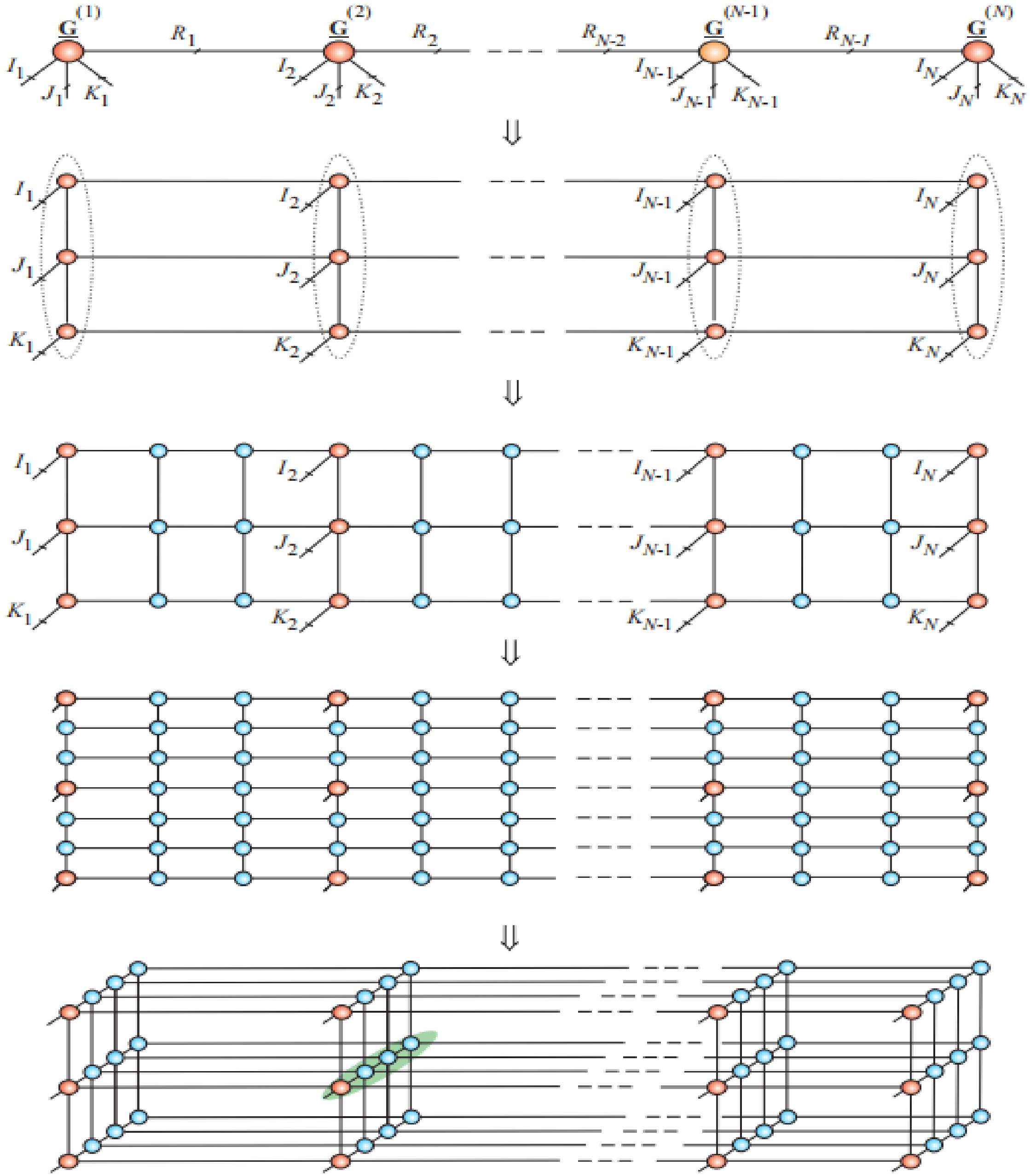
Different Constraints

Cost Function	Constraints
Multilinear (sparse) PCA (MPCA) $\max_{\mathbf{u}_r^{(n)}} \underline{\mathbf{X}} \bar{\times}_1 \mathbf{u}_r^{(1)} \bar{\times}_2 \mathbf{u}_r^{(2)} \cdots \bar{\times}_N \mathbf{u}_r^{(N)} + \gamma \sum_{n=1}^{N} \lVert \mathbf{u}_r^{(n)} \rVert_1$	$\mathbf{u}_r^{(n)\mathrm{T}} \mathbf{u}_r^{(n)} = 1, \ \forall (n, r)$ $\mathbf{u}_r^{(n)\mathrm{T}} \mathbf{u}_q^{(n)} = 0 \ \text{ for } \ r \neq q$
HOSVD/HOOI $\min_{\mathbf{U}^{(n)}} \lVert \underline{\mathbf{X}} - \underline{\mathbf{G}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} \rVert_F^2$	$\mathbf{U}^{(n)\mathrm{T}} \mathbf{U}^{(n)} = \mathbf{I}_{R_n}, \ \forall n$
Multilinear ICA $\min_{\mathbf{B}^{(n)}} \lVert \underline{\mathbf{X}} - \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \cdots \times_N \mathbf{B}^{(N)} \rVert_F^2$	Vectors of $\mathbf{B}^{(n)}$ statistically as independent as possible
Nonnegative CP/Tucker decomposition (NTF/NTD) (Cichocki *et al.*, 2009) $\min_{\mathbf{B}^{(n)}} \lVert \underline{\mathbf{X}} - \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \cdots \times_N \mathbf{B}^{(N)} \rVert_F^2$ $+\gamma \sum_{n=1}^{N} \sum_{r_n=1}^{R_n} \lVert \mathbf{b}_{r_n}^{(n)} \rVert_1$	Entries of $\underline{\mathbf{G}}$ and $\mathbf{B}^{(n)}$, $\forall n$ are nonnegative
Sparse CP/Tucker decomposition $\min_{\mathbf{B}^{(n)}} \lVert \underline{\mathbf{X}} - \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \cdots \times_N \mathbf{B}^{(N)} \rVert_F^2$ $+\gamma \sum_{n=1}^{N} \sum_{r_n=1}^{R_n} \lVert \mathbf{b}_{r_n}^{(n)} \rVert_1$	Sparsity constraints imposed on $\mathbf{B}^{(n)}$
Smooth CP/Tucker decomposition (SmCP/SmTD) (Yokota *et al.*, 2016) $\min_{\mathbf{B}^{(n)}} \lVert \underline{\mathbf{X}} - \underline{\mathbf{\Lambda}} \times_1 \mathbf{B}^{(1)} \cdots \times_N \mathbf{B}^{(N)} \rVert_F^2$ $+\gamma \sum_{n=1}^{N} \sum_{r=1}^{R} \lVert \mathbf{L} \mathbf{b}_r^{(n)} \rVert_2$	Smoothness imposed on vectors $\mathbf{b}_r^{(n)}$ of $\mathbf{B}^{(n)} \in \mathbb{R}^{I_n \times R}, \ \forall n$ via a difference operator $\mathbf{L}$

Tensor Train&Ring

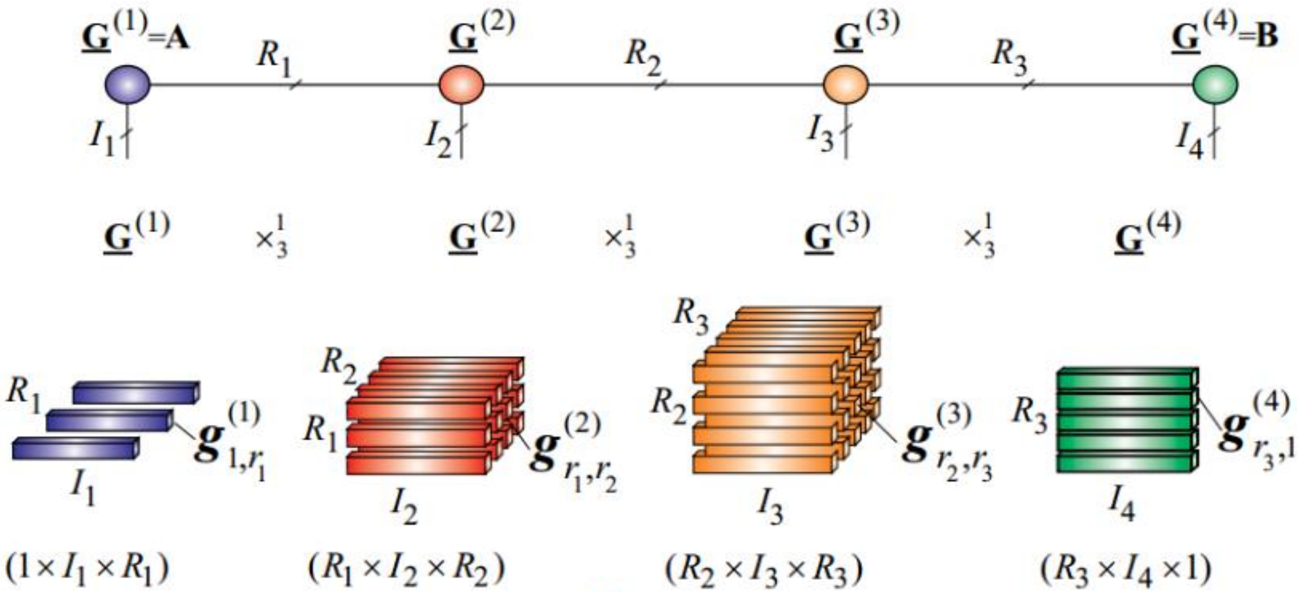


- Periodic Boundary Conditions(PBC)
- Open Boundary Conditions (OBC)
- Matrix Product State(MPS)
- Matrix Product Operator(MPO)
- Projected Entangled-Pair State(PEPS)
- Projected Entangled-Pair Operator(PEPO)



G(1)
G(2)
G(N-1)
G(N)
R1
R2
RN-2
RN-1
I1
J1
K1
I2
J2
K2
IN-1
JN-1
KN-1
IN
JN
KN

Tensor Train(MPS with OBC)



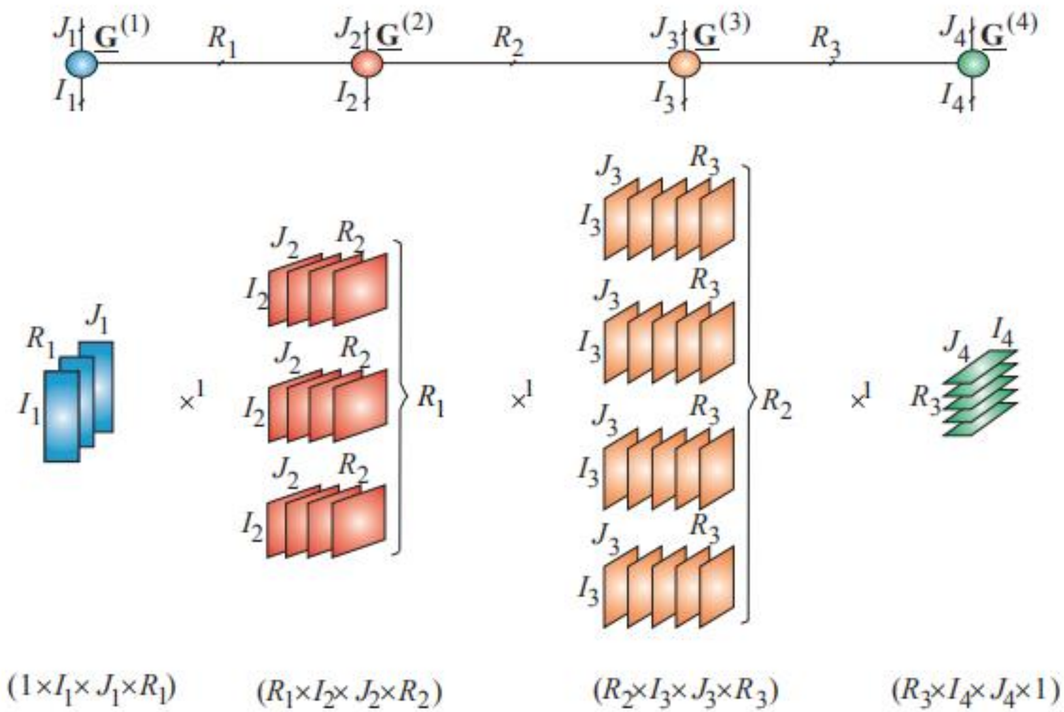
$$x_{i_1 i_2 \cdots, i_N} = \sum_{r_1, r_2 \cdots r_{N-1}}^{R_1, R_2 \cdots R_{N-1}} g^{(1)}_{1,i_1,r_1} g^{(2)}_{r_1,i_2,r_2} \cdots g^{(N)}_{r_{(N-1)}, i_{(N)}, 1}$$

$$\mathcal{X} = G_{(1)} \times_1 G_{(2)} \times_1 \cdots \times_1 G_{(N)} \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_N}$$

$$G_{(n)} \in \mathbb{R}^{R_{n-1} \times I_n \times R_n} \ \ \forall n$$

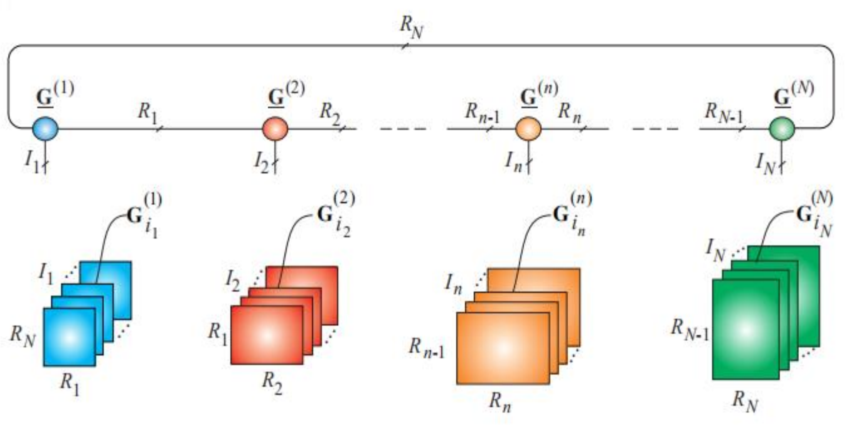
$$R_0 = 1, R_N = 1$$

Tensor Train(MPO with OBC)



$$x_{i_1 j_1 i_2 j_2 \cdots, i_N j_N} = \sum_{r_1, r_2 \cdots r_{N-1}}^{R_1, R_2 \cdots R_{N-1}} g^{(1)}_{1, i_1, j_1, r_1} g^{(2)}_{r_1, i_2, j_2, r_2} \cdots g^{(N)}_{r_{(N-1)}, i_{(N)}, j_N, 1}$$

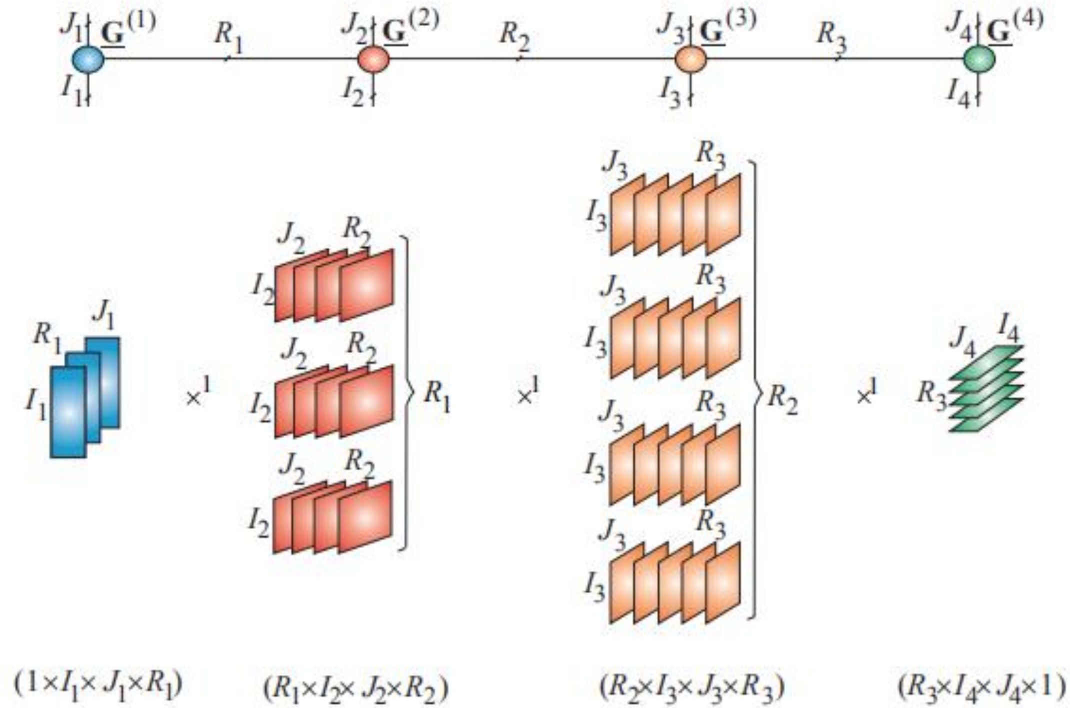
Tensor Ring (MPS with PBC)



$$x_{\mathrm{i_1 i_2 \cdots, i_N}} = \sum_{\mathrm{r_1, r_2 \cdots r_{N-1} r_N}}^{\mathrm{R_1, R_2 \cdots R_{N-1} R_N}} \mathrm{g}^{(1)}_{\mathrm{r_N, i_1, r_1}} \mathrm{g}^{(2)}_{\mathrm{r_1, i_2, r_2}} \cdots \mathrm{g}^{(\mathrm{N})}_{\mathrm{r_{(N-1)}, i_{(N)}, r_N}}$$

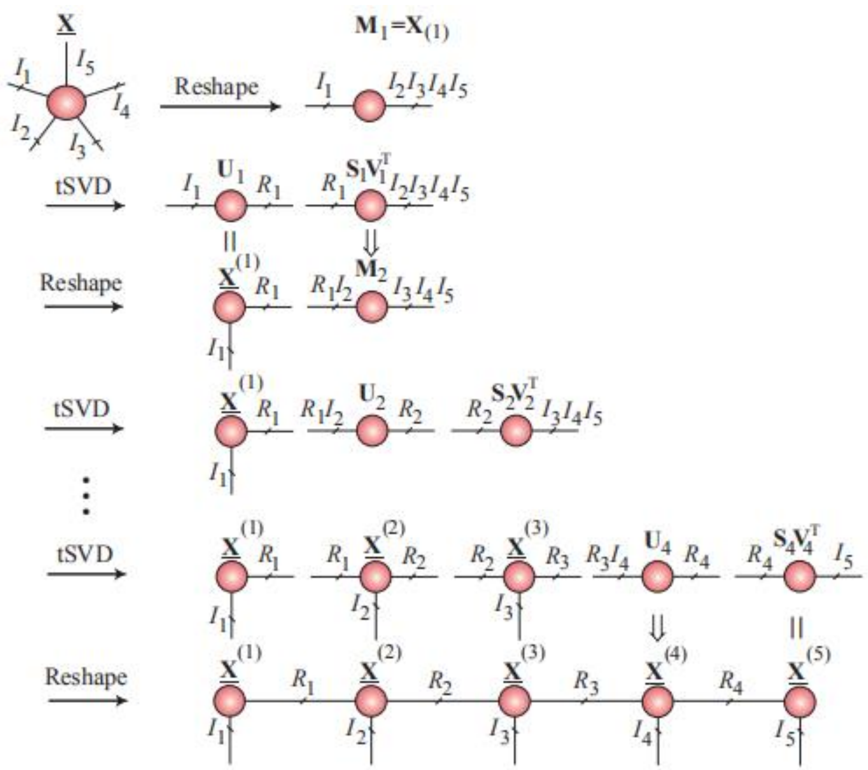
$$\mathcal{X} = \mathrm{G}_{(1)} \times_1 \mathrm{G}_{(2)} \times_1 \cdots \times_1 \mathrm{G}_{(\mathrm{N})} \in \mathbb{R}^{\mathrm{I_1 \times I_2 \cdots \times I_N}}$$
$$\mathrm{G}_{(\mathrm{n})} \in \mathbb{R}^{\mathrm{R_{n-1} \times I_n \times R_n}} \ \ \forall \mathrm{n}$$

Tensor Ring (MPO with PBC)



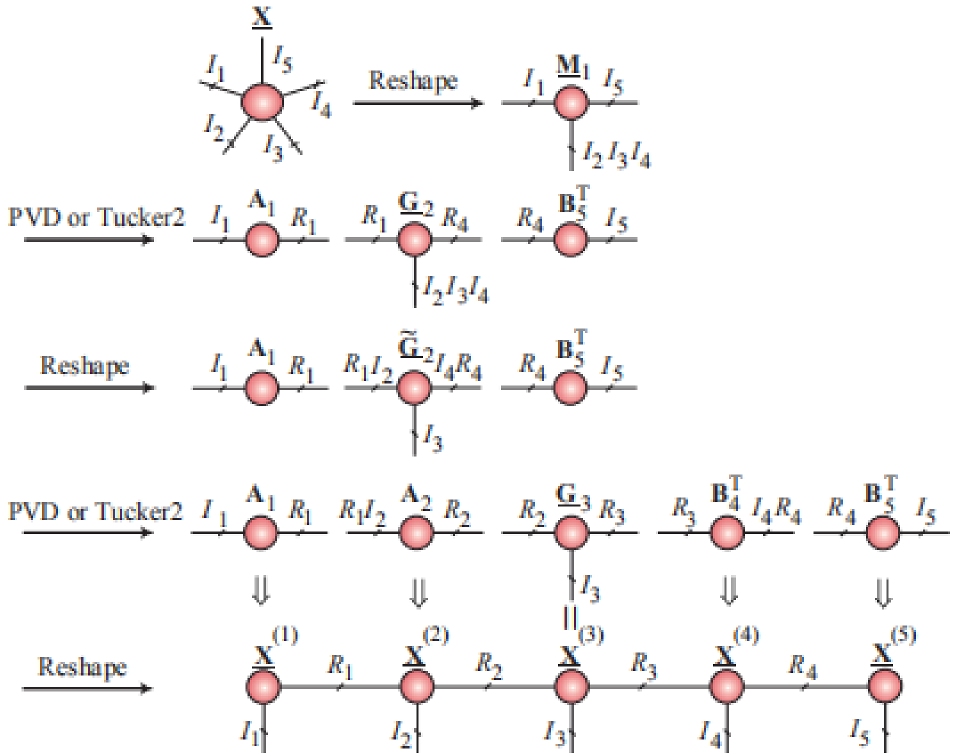
$$x_{i_1 j_1 i_2 j_2 \cdots, i_N j_N} = \sum_{r_1, r_2 \cdots r_{N-1} r_N}^{R_1, R_2 \cdots R_{N-1} R_N} g^{(1)}_{r_N, i_1, j_1, r_1} g^{(2)}_{r_1, i_2, j_2, r_2} \cdots g^{(N)}_{r_{(N-1)}, i_{(N)}, j_N, r_N}$$

TT SVD



Reshape
$\mathbf{M}_1 = \mathbf{X}_{(1)}$
tSVD
$\mathbf{U}_1$
$\mathbf{S}_1\mathbf{V}_1^T$
Reshape
$\mathbf{M}_2$
tSVD
$\mathbf{U}_2$
$\mathbf{S}_2\mathbf{V}_2^T$
tSVD
$\mathbf{U}_4$
$\mathbf{S}_4\mathbf{V}_4^T$
Reshape

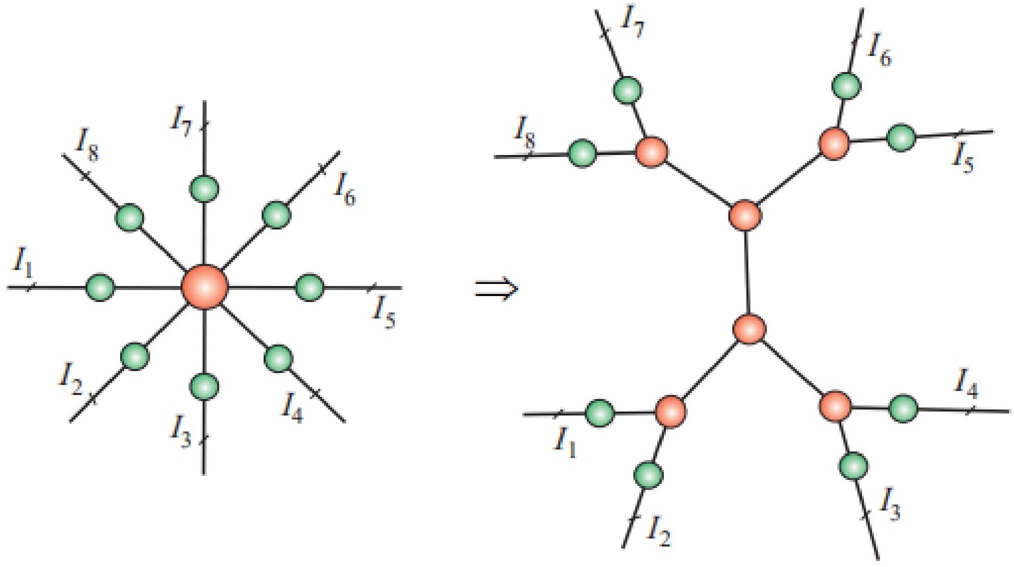
TT Rounding

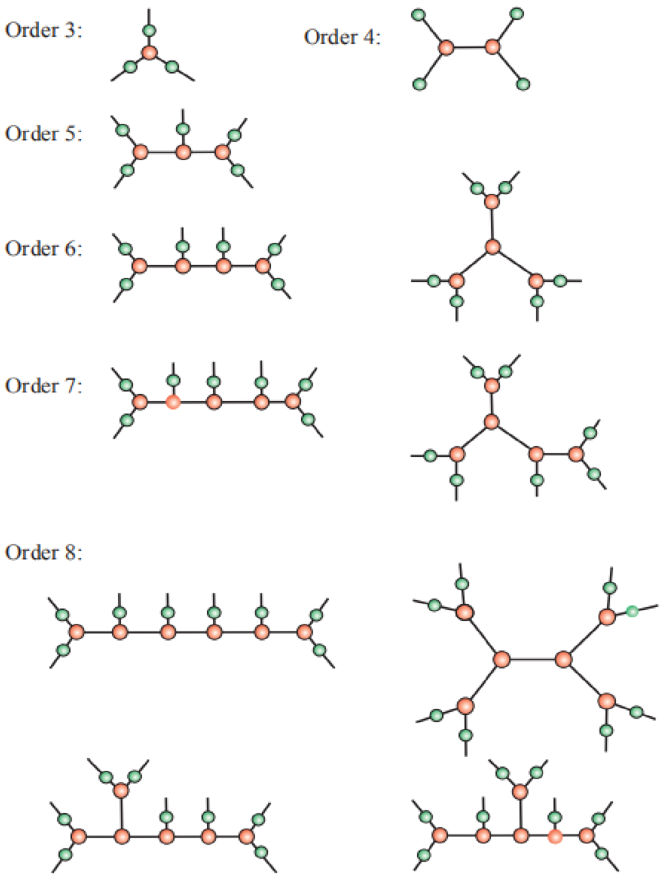


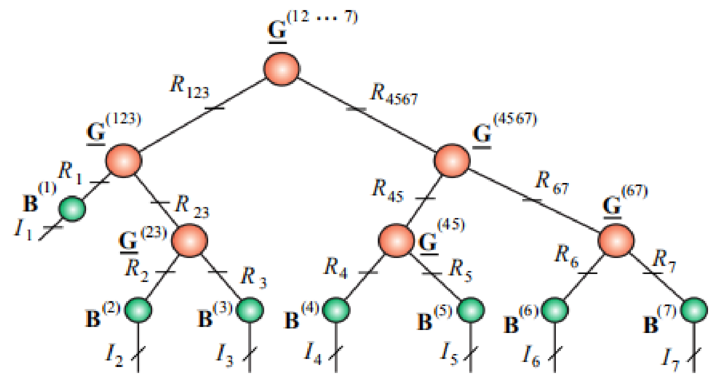
Why TR?

- The constraint on TT-ranks, leads to the limited representation ability and flexibility;
- TT-ranks are small in the border cores and large in the middle cores, which might not be optimal for a given data tensor;
- TT representations and TT-ranks are sensitive to the order of tensor dimensions

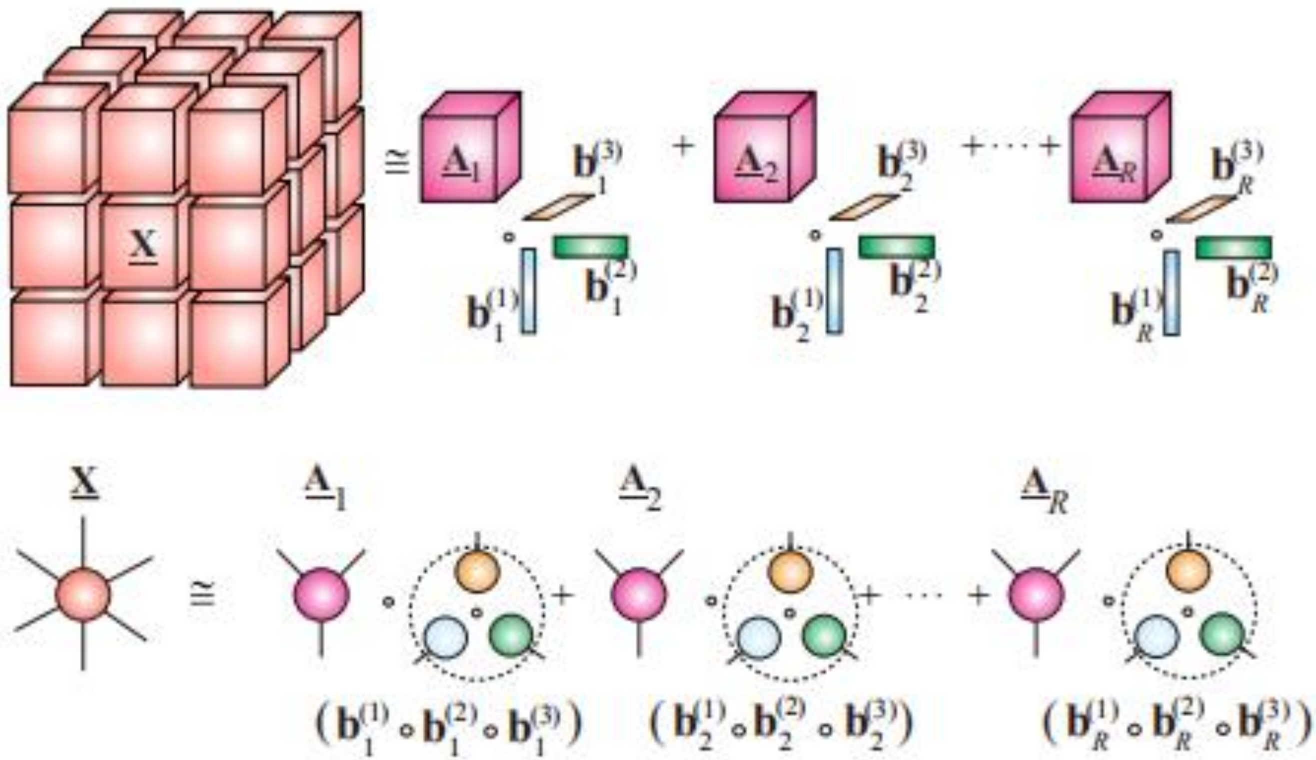
Hierarchical Tucker



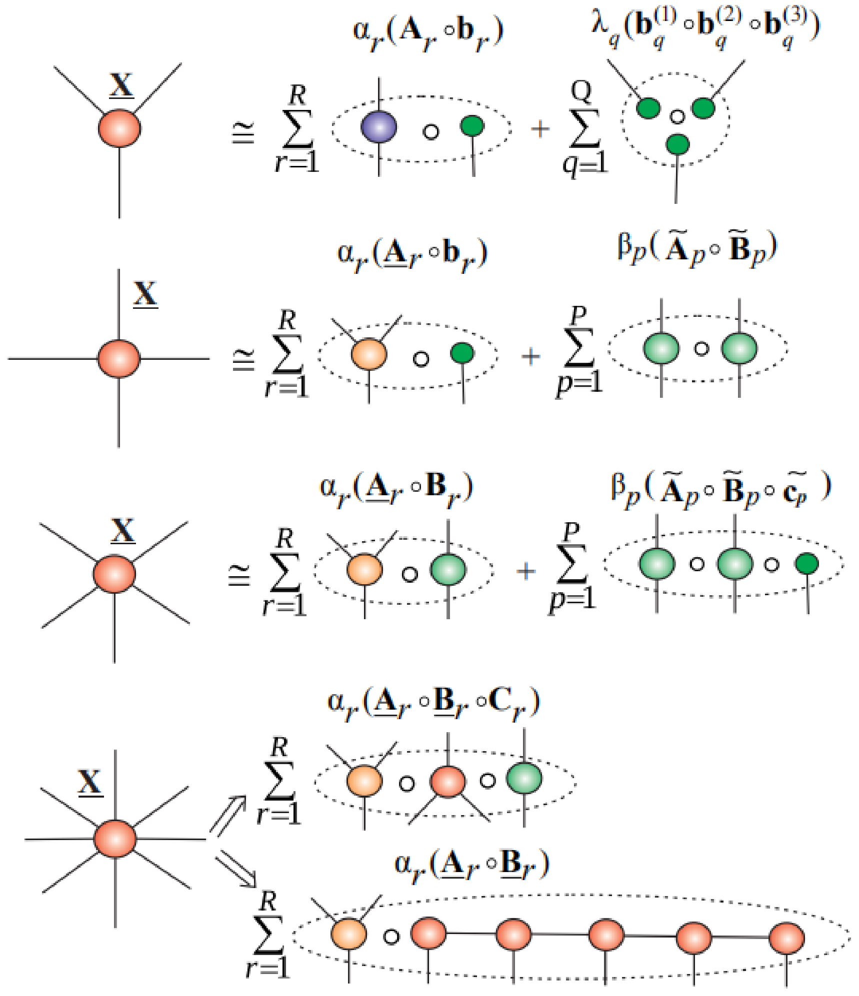

Order 3:
Order 4:
Order 5:
Order 6:
Order 7:
Order 8:


$\underline{\mathbf{G}}^{(12\cdots7)}$
R_{123}
R_{4567}
$\underline{\mathbf{G}}^{(123)}$
$\underline{\mathbf{G}}^{(4567)}$
R_1
R_{23}
R_{45}
R_{67}
$\mathbf{B}^{(1)}$
I_1
$\underline{\mathbf{G}}^{(23)}$
$\underline{\mathbf{G}}^{(45)}$
$\underline{\mathbf{G}}^{(67)}$
R_2
R_3
R_4
R_5
R_6
R_7
$\mathbf{B}^{(2)}$
$\mathbf{B}^{(3)}$
$\mathbf{B}^{(4)}$
$\mathbf{B}^{(5)}$
$\mathbf{B}^{(6)}$
$\mathbf{B}^{(7)}$
I_2
I_3
I_4
I_5
I_6
I_7

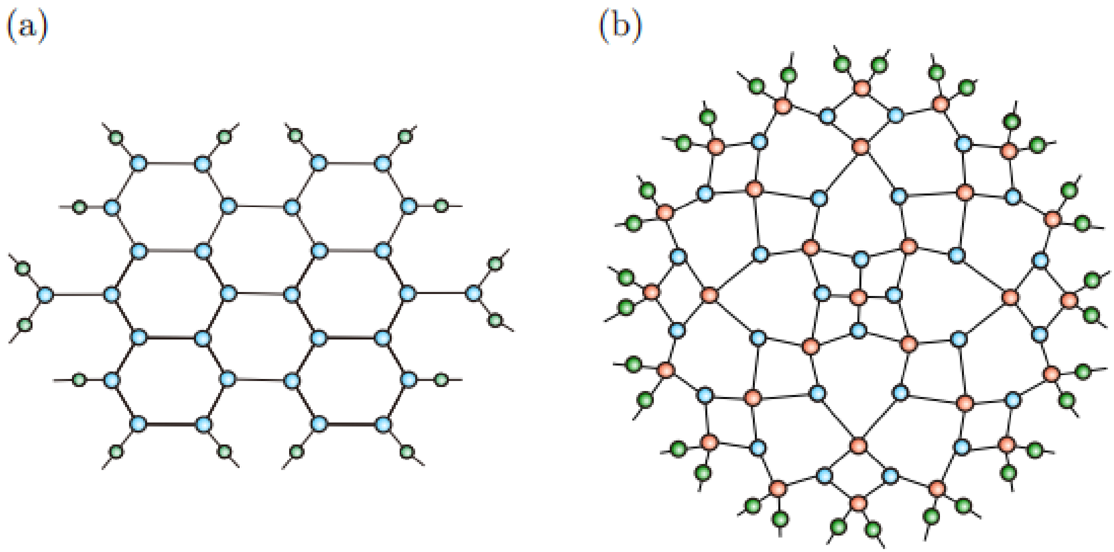
Block Term Decomposition



Hierarchical Outer Product Tensor Approximation (HOPTA)



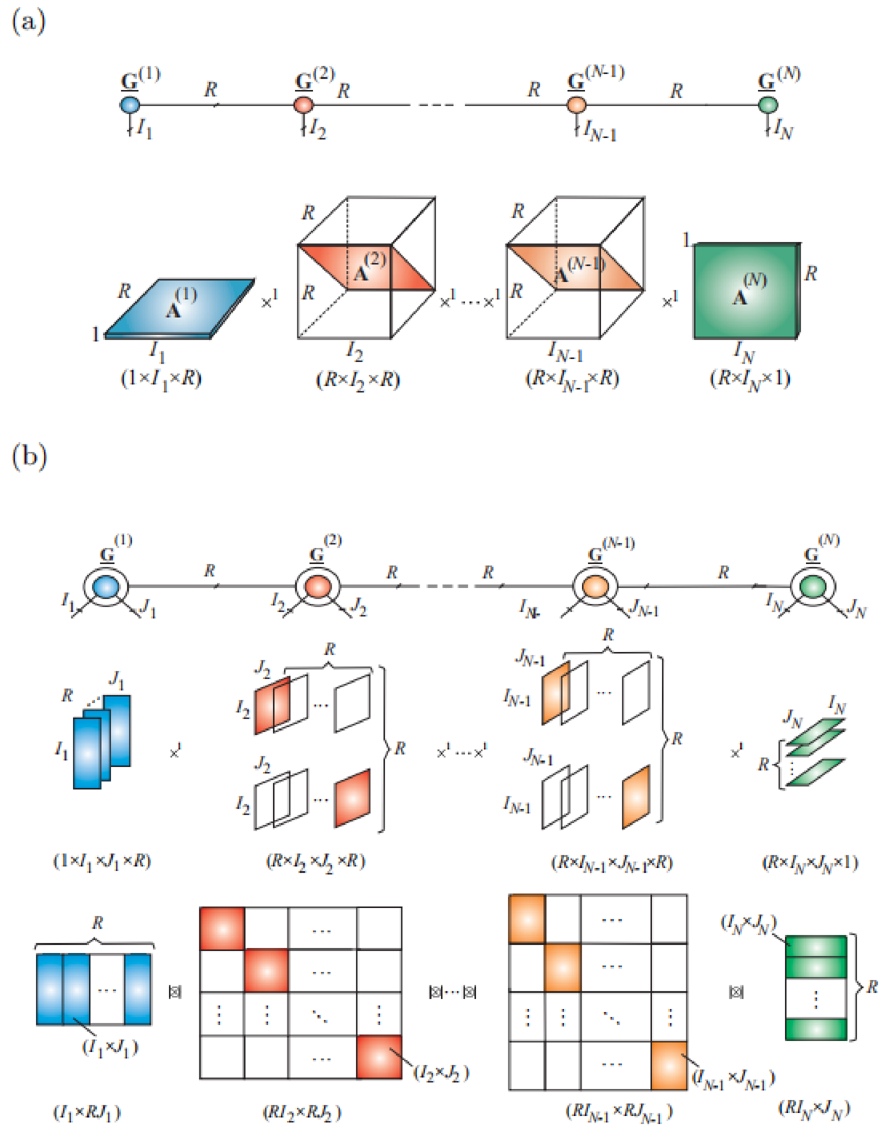
Honey-Comb Lattice



Some Discussions

Tensor Networks	Neural Networks and Graphical Models in ML/Statistics
TT/MPS	Hidden Markov Models (HMM)
HT/TTNS	Deep Learning Neural Networks, Gaussian Mixture Model (GMM)
PEPS	Markov Random Field (MRF), Conditional Random Field (CRF)
MERA	Wavelets, Deep Belief Networks (DBN)
ALS, DMRG/MALS Algorithms	Forward-Backward Algorithms, Block Non-linear Gauss-Seidel Methods

Some Discussions



- Tensor decomposition may be considered as a multilinear extension of PCA.(Especially CP and Tucker)

- It was recently shown that Tensor decomposition can also perform simultaneous subspace selection (data compression) and clustering.

Some Applications

- SVM
- Recommendation
- Image
- Biology
- Neural Network

SVM To STM

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} J(\mathbf{w},b,\boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{m=1}^{M}\xi_m,$$

$$\text{s.t.} \quad y_m(\mathbf{w}^{\mathrm{T}}\mathbf{x}_m + b) \geqslant 1 - \xi_m, \quad \boldsymbol{\xi} \geqslant 0, \quad m = 1,\ldots,M,$$

$$\min_{\mathbf{W},b,\boldsymbol{\xi}} \frac{1}{2}\mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{W}) + C\sum_{m=1}^{M}\xi_m$$

$$\text{s.t.} \quad y_m(\mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{X}_m) + b) \geqslant 1 - \xi_m, \boldsymbol{\xi} \geqslant 0, \quad m = 1,\ldots,M.$$

$$\min_{\mathbf{w}_n,b,\boldsymbol{\xi}} J(\mathbf{w}_n,b,\boldsymbol{\xi}) = \frac{1}{2}\left\|\otimes_{n=1}^{N}\mathbf{w}_n\right\|^2 + C\sum_{m=1}^{M}\xi_m$$

$$\text{s.t.} \quad y_m\left(\underline{\mathbf{X}}_m \bar{\times}_1 \mathbf{w}_1 \cdots \bar{\times}_N \mathbf{w}_N + b\right) \geqslant 1 - \xi_m, \quad \boldsymbol{\xi} \geqslant 0,$$
$$m = 1,\ldots,M.$$

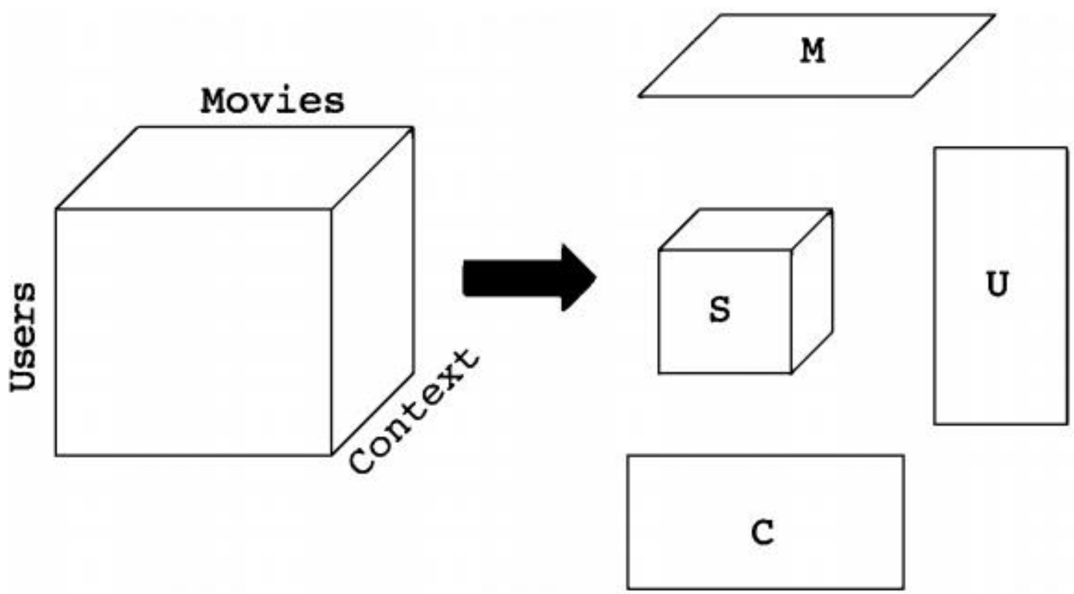
$$\phi(\underline{\mathbf{X}}_m) = \begin{bmatrix} \varphi(\mathbf{z}_{m1}) \\ \varphi(\mathbf{z}_{m2}) \\ \vdots \\ \varphi(\mathbf{z}_{mI_1}) \end{bmatrix}.$$

Methods	ADNI	HIV	ADHD
SVM	0.49 ± 0.02	0.70 ± 0.01	0.58 ± 0.00
SVM+PCA	0.50 ± 0.02	0.73 ± 0.03	0.63 ± 0.01
K_{3rd}	0.55 ± 0.01	0.75 ± 0.02	0.55 ± 0.00
sKL	0.51 ± 0.03	0.65 ± 0.02	0.50 ± 0.04
FK	0.51 ± 0.02	0.70 ± 0.01	0.50 ± 0.00
STuM	0.52 ± 0.01	0.66 ± 0.01	0.54 ± 0.03
DuSK	0.75 ± 0.02	0.74 ± 0.00	0.65 ± 0.01
MMK_{best}	0.81 ± 0.01	0.79 ± 0.01	0.70 ± 0.01
MMK_{cov}	0.69 ± 0.01	0.72 ± 0.02	0.66 ± 0.02
3D CNN	0.52 ± 0.03	0.75 ± 0.02	0.68 ± 0.02
KSTM	$\mathbf{0.84 \pm 0.03}$	$\mathbf{0.82 \pm 0.02}$	$\mathbf{0.74 \pm 0.02}$

CVPR(2017)

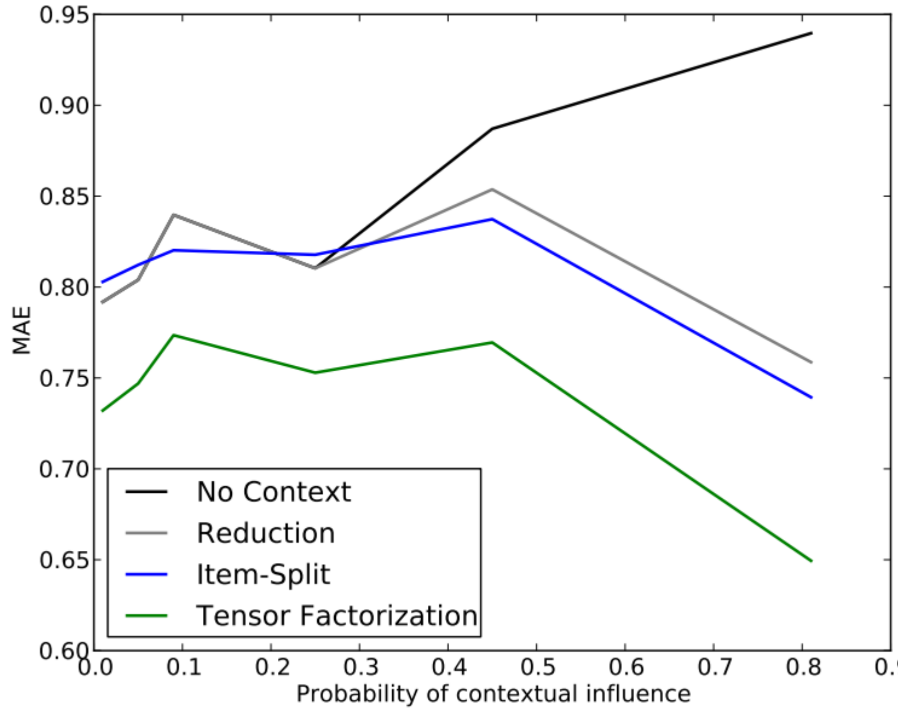
Multi-way Multi-level Kernel Modeling for Neuroimaging Classification

Recommendation

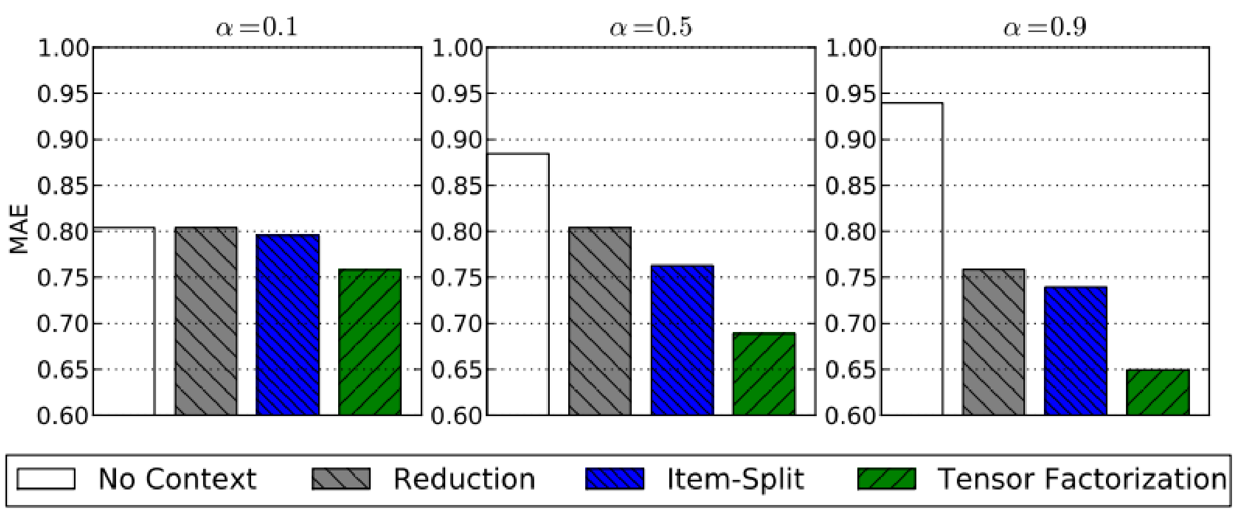


Acm Conference on Recommender Systems(2010)

Multiverse Recommendation: N-dimensional TensorFactorization for Context-aware Collaborative Filteri



MAE
Probability of contextual influence
No Context
Reduction
Item-Split
Tensor Factorization



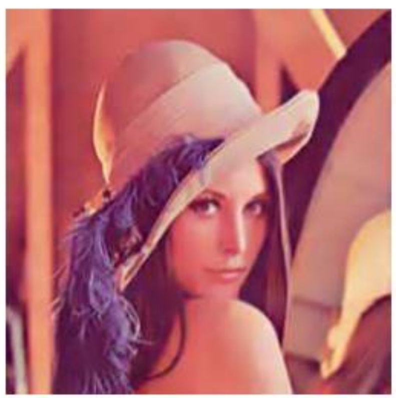
$\alpha = 0.1$
$\alpha = 0.5$
$\alpha = 0.9$
MAE
No Context
Reduction
Item-Split
Tensor Factorization

Image

(a) Noisy image

(b) TT-ASCU, MSE = **27.37** dB

(c) TT-SVD, MSE = 35.11 dB



(d) K-SVD, MSE = 34.76 dB

(a) Noisy patch

(b) TT-ASCU, MSE = **31.47** dB

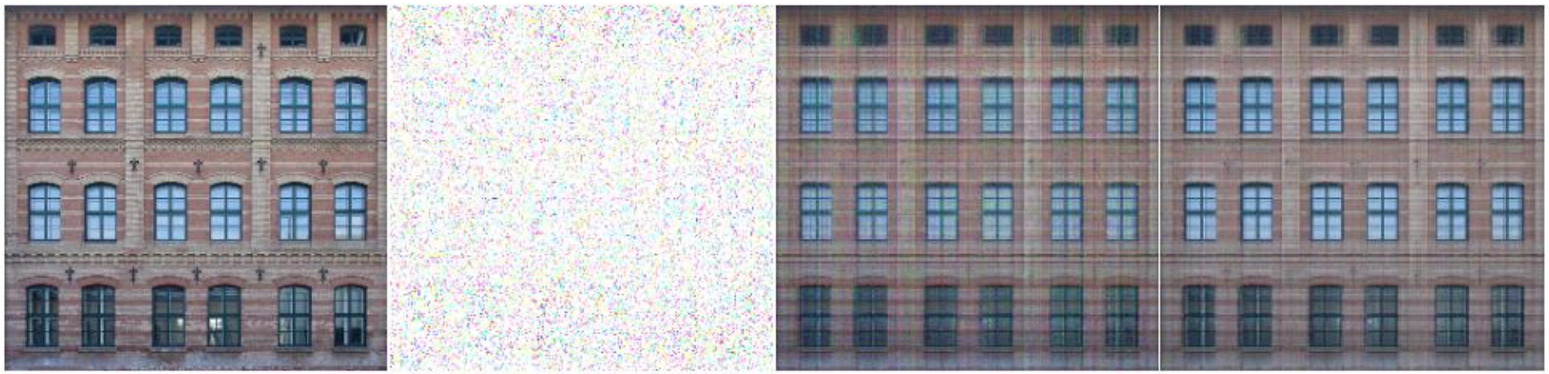
(c) TT-SVD, MSE = 40.40 dB

(d) K-SVD, MSE = 35.74 dB

arXiv (2016)

Tensor Networks for Latent Variable Analysis. Part I: Algorithms for Tensor Train Decomposition

Samples	MC1	MC2	MC3	TC
20%	670	781	862	**17**
50%	27	36	75	**0**



(a) Original (b) With 90% missing entries (c) S-MC: RSE=0.1889 (d) TC: RSE=0.1445

IEEE International Conference on Computer Vision (2009)

Tensor completion for estimating missing values in visual data.

Paper: PLTD: Patch-Based Low-Rank Tensor Decomposition for Hyperspectral Images

	Proposed	MPCA	MPCA2	PCA	JPEG	PCA+JPEG	KSVD
Rate	0.0202	0.0203	0.0291	0.0202	0.0206	0.0220	0.0625
PSNR	49.5481	45.9795	47.7080	47.9091	42.0732	45.5686	43.1158
SSIM	0.9908	0.9850	0.9884	0.9876	0.9717	0.9821	0.9795
FSIM	0.9983	0.9939	0.9935	0.9937	0.9645	0.9850	0.9710
ERGAS	16.6267	25.3740	19.8433	26.4368	40.6427	28.7114	33.9282
MSAM	0.0312	0.0335	0.0352	0.0416	0.0583	0.0590	0.0346
Time	36.31 s	30.66 s	3.75 s	0.71 s	510.39 s	1000.61 s	11650.27 s

IEEE TRANSACTIONS ON MULTIMEDIA(2017)

PLTD: Patch-Based Low-Rank Tensor Decomposition for Hyperspectral Images

Tensor Networks for Image Compression

Author: Alex Trujillo Boque

JPEG (Image coding standard) Theses. Issue Date: Jan-2016.

Method: Fine-grained and coarse-grained

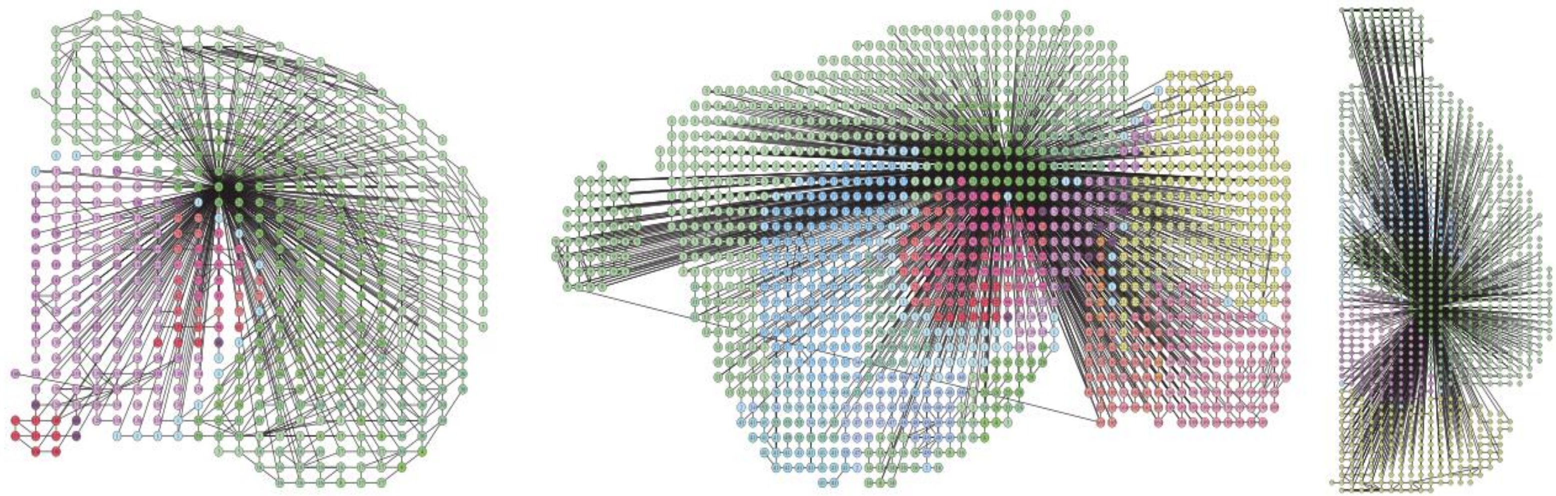
Not a competitive alternative

	Lat. size	512	256	512	256	256
	SSIM	0.8311	0.8122	0.9014	0.8910	0,9400
MPS	χ_{trunc}	2	2	3	3	4
	DCR	17.97	19.60	7.64	7.93	3.82
JPEG	Q	10	11	24	32	70
	DCR	33.14	29.06	20.06	16.58	8.90

From Internet

Tensor Networks for Image Compression

Biology



$3012 \times 67 \times 41 \times 58$

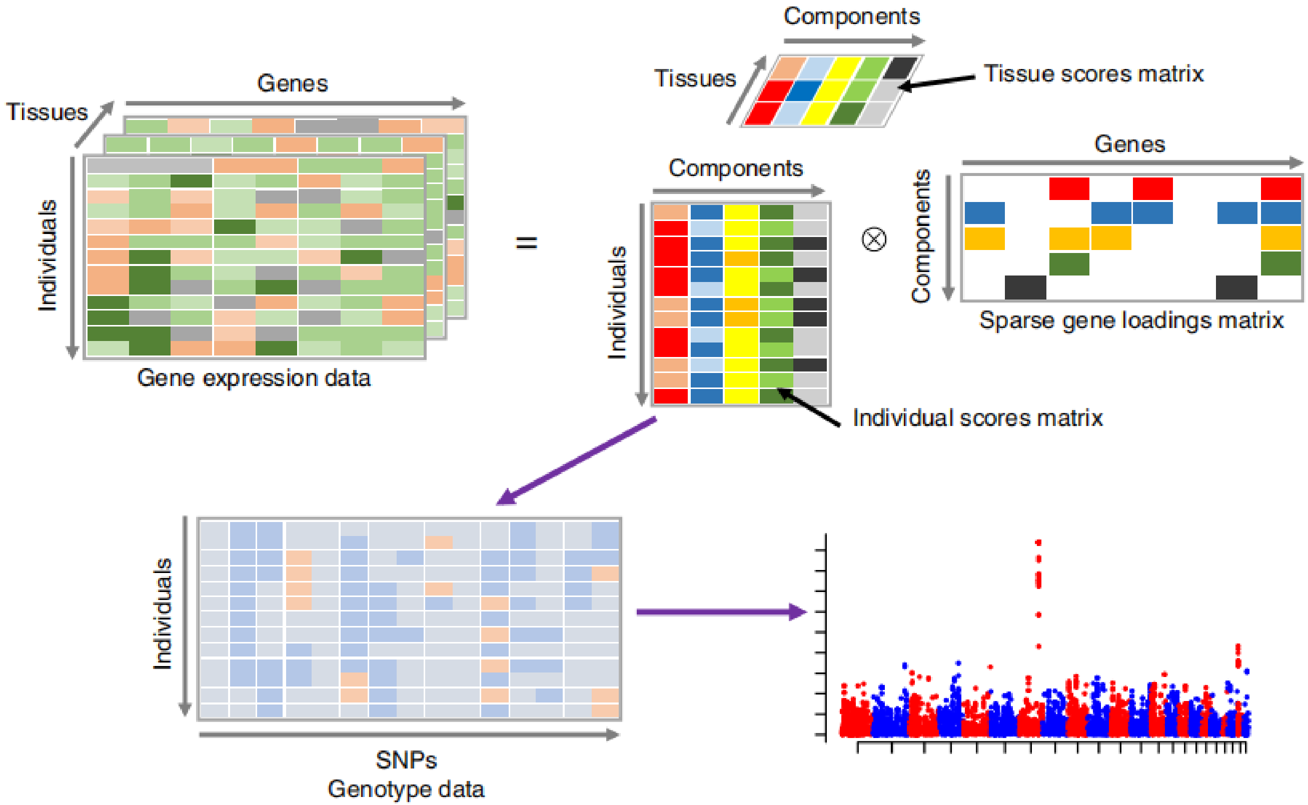
$179 \times 67 \times 41 \times 33$

$3021 \times 22 \times 13 \times 19$

BIOINFORMATICS(2011)

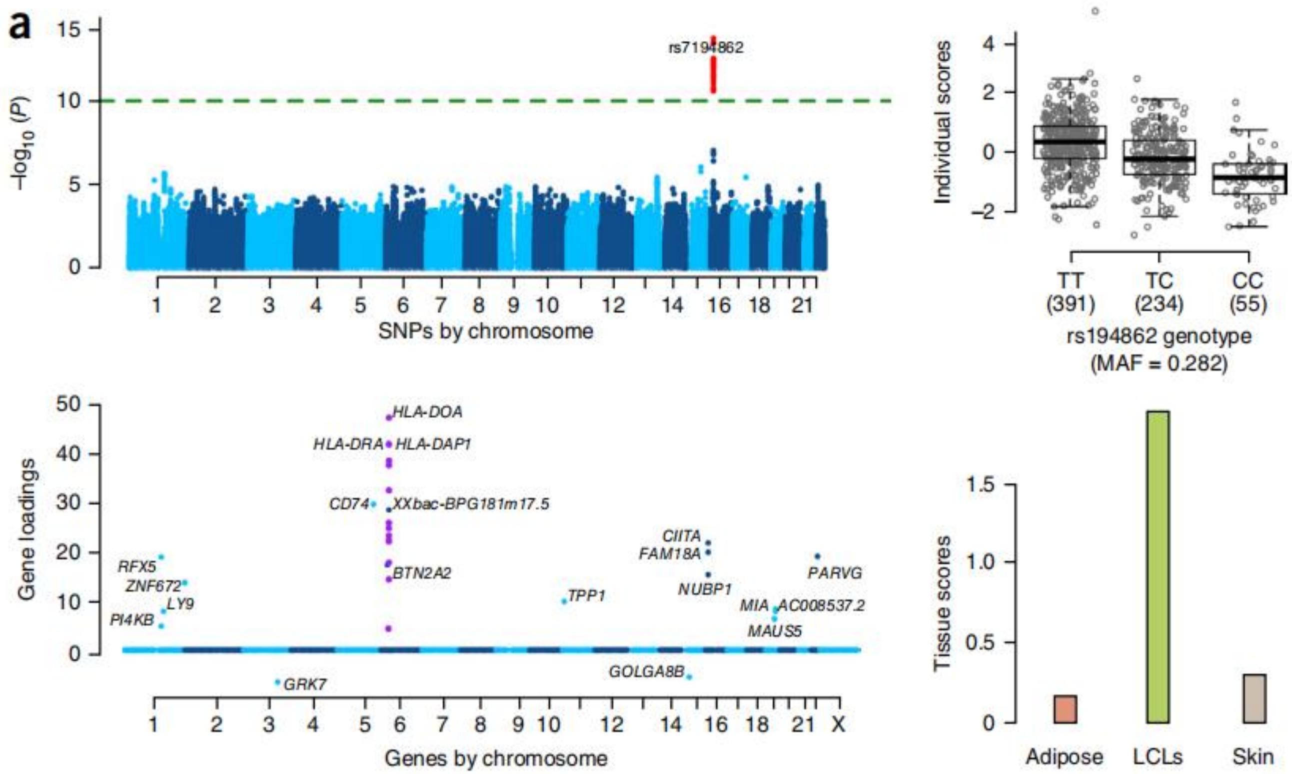
Computational network analysis of the anatomical and genetic organizations in the mouse brain

Biology

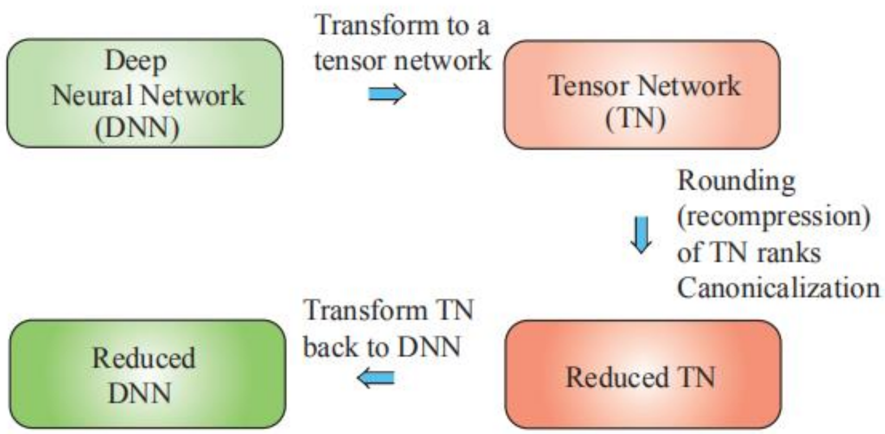


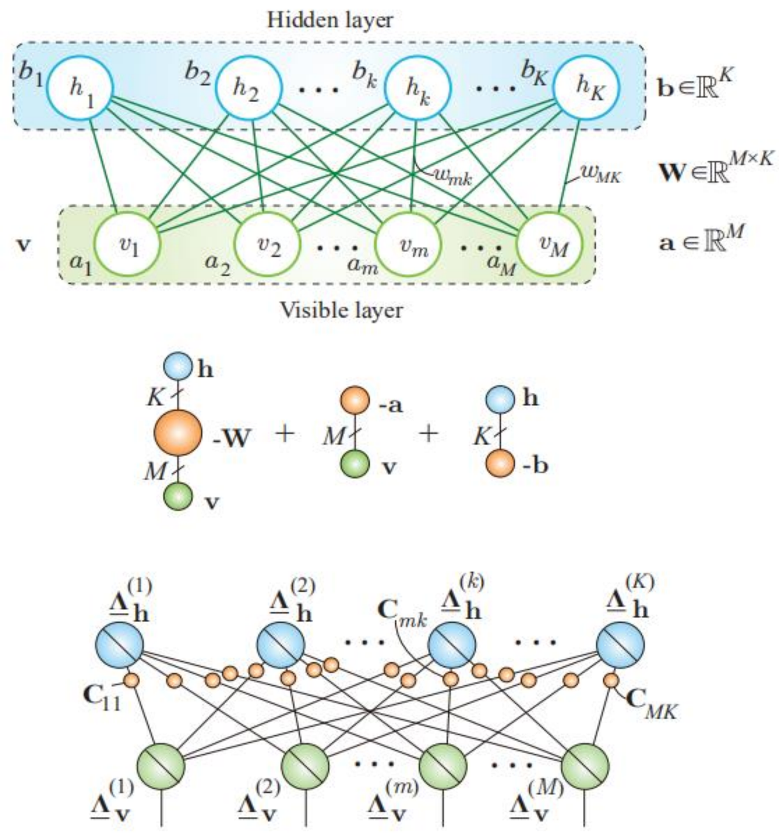
Nature Genes(2016)

Tensor decomposition for multiple-tissue gene expression


a
rs7194862
–log10 (P)
SNPs by chromosome
Individual scores
TT (391)
TC (234)
CC (55)
rs194862 genotype (MAF = 0.282)
Gene loadings
HLA-DOA
HLA-DRA
HLA-DAP1
CD74
XXbac-BPG181m17.5
BTN2A2
RFX5
ZNF672
LY9
PI4KB
GRK7
TPP1
CIITA
FAM18A
NUBP1
GOLGA8B
MIA
AC008537.2
MAUS5
PARVG
Genes by chromosome
Tissue scores
Adipose
LCLs
Skin

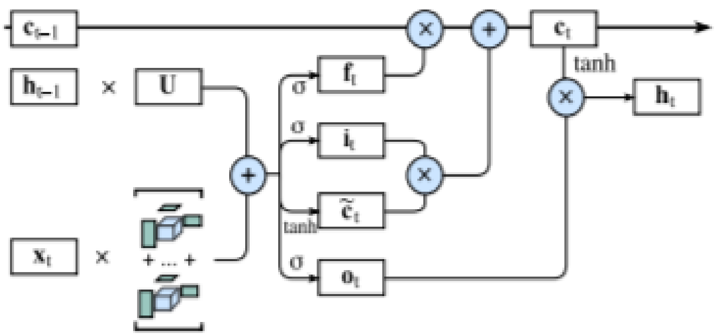
Neural Network

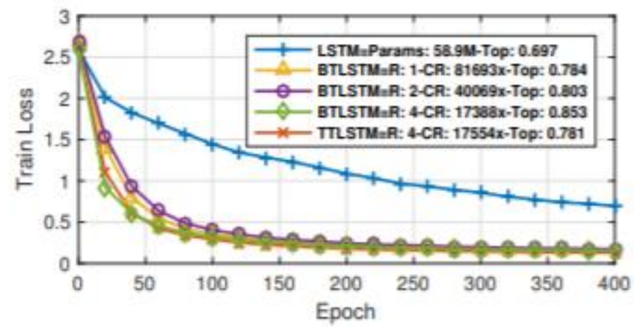




CVPR2018

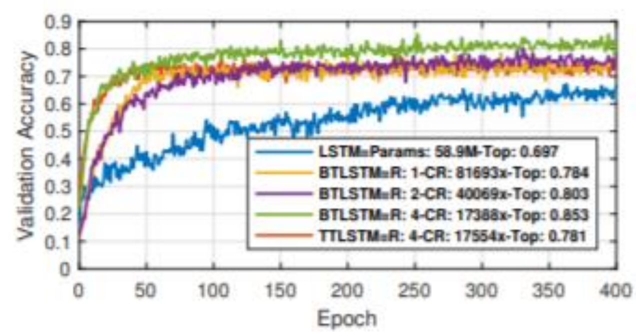
Learning Compact Recurrent Neural Networks with Block-Term Tensor Decomposition



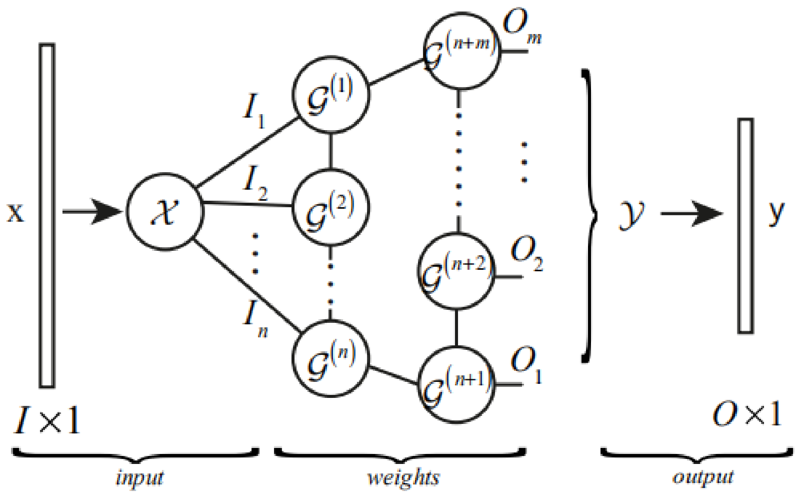


(a) Training loss of baseline LSTM, TT-LSTM and BT-LSTM.

	Method	Accuracy
Orthogonal Approaches	Original [25]	0.712
	Spatial-temporal [24]	0.761
	Visual Attention [32]	0.850
RNN Approaches	LSTM	0.697
	TT-LSTM [47]	0.796
	BT-LSTM	**0.853**



Our work



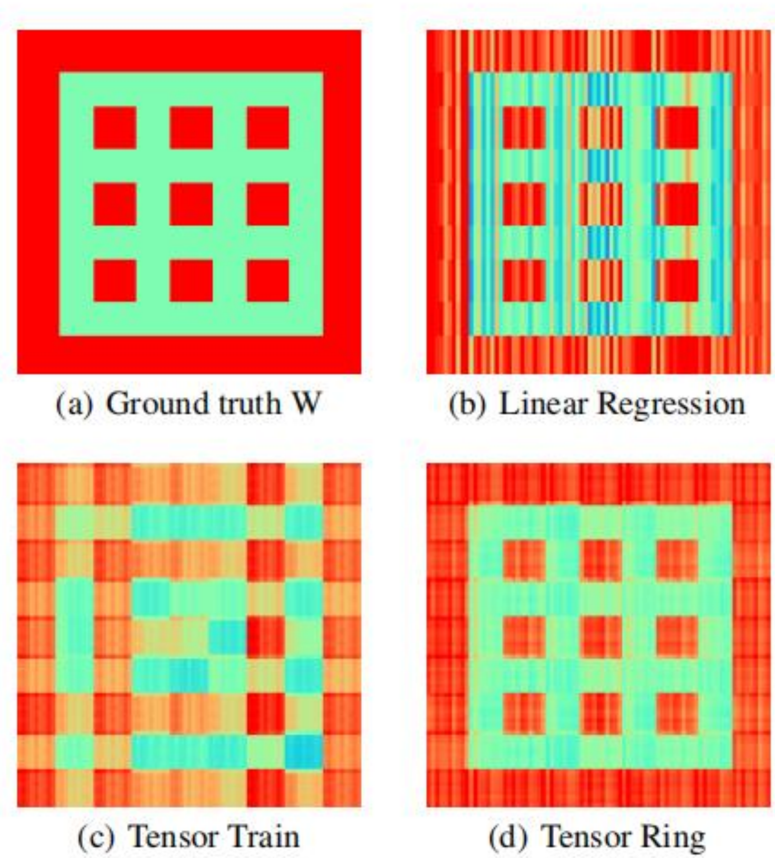
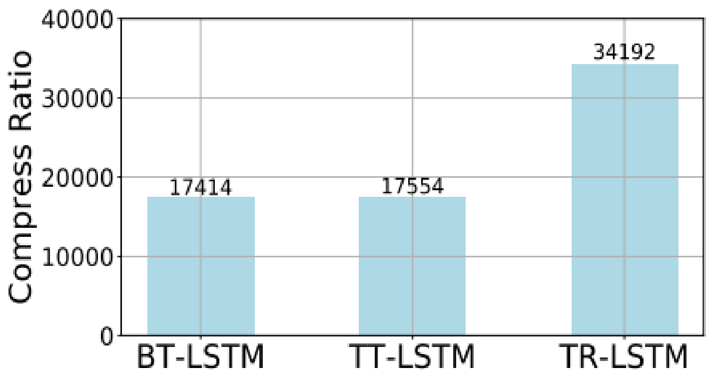


Figure 5: The illustration on the ground truth **W** and the recovered weights from different models. The recovered RMSEs of the linear model, tensor train, and tensor ring, are 0.16, 0.18, and 0.09, respectively.

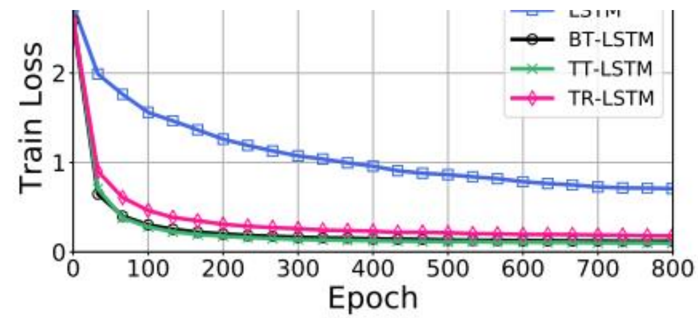


(a) Compression Ratio

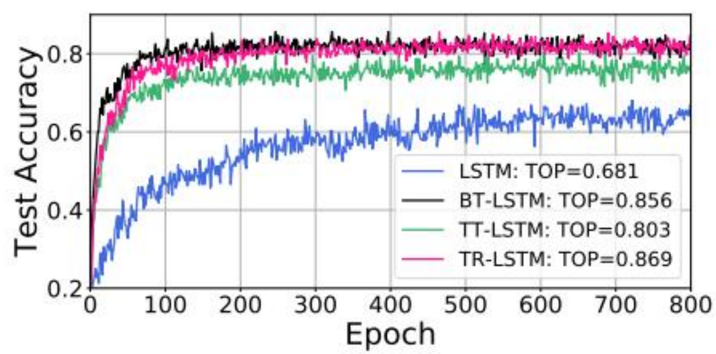
Method	Accuracy
(Hasan and Roy-Chowdhury 2014)	54.5%
(Liu, Luo, and Shah 2009)	71.2%
(Ikizler-Cinbis and Sclaroff 2010)	75.2%
(Liu, Shyu, and Zhao 2013)	76.1%
(Sharma, Kiros, and Salakhutdinov 2015)	85.0%
(Wang et al. 2011)	84.2%
(Sharma, Kiros, and Salakhutdinov 2015)	84.9%
(Cho et al. 2014a)	88.0%
(Gammulle et al. 2017)	**94.6%**
CNN + LSTM	92.3%
CNN + TR-LSTM	93.8%

Table 2: The state-of-the-art performance on UCF11.

Method	#Params	Accuracy
LSTM	59M	0.697
TT-LSTM	3360	0.796
BT-LSTM	3387	0.853
TR-LSTM	1725	**0.869**



(b) Train Loss



(c) Test Accuracy