

ENGG5781 Matrix Analysis and Computations

Lecture 10: Non-Negative Matrix Factorization and Tensor Decomposition

Wing-Kin (Ken) Ma

2022-23 First Term

Department of Electronic Engineering
The Chinese University of Hong Kong

Topic 1: Nonnegative Matrix Factorization

Nonnegative Matrix Factorization

Consider again the low-rank factorization problem

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times r}, \mathbf{B} \in \mathbb{R}^{r \times n}} \|\mathbf{Y} - \mathbf{AB}\|_F^2.$$

The solution is not unique: if $(\mathbf{A}^*, \mathbf{B}^*)$ is a solution to the above problem, then $(\mathbf{A}^* \mathbf{Q}^T, \mathbf{QB}^*)$ for any orthogonal \mathbf{Q} is also a solution.

Nonnegative Matrix Factorization (NMF):

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times r}, \mathbf{B} \in \mathbb{R}^{r \times n}} \|\mathbf{Y} - \mathbf{AB}\|_F^2 \quad \text{s.t. } \mathbf{A} \geq \mathbf{0}, \mathbf{B} \geq \mathbf{0}$$

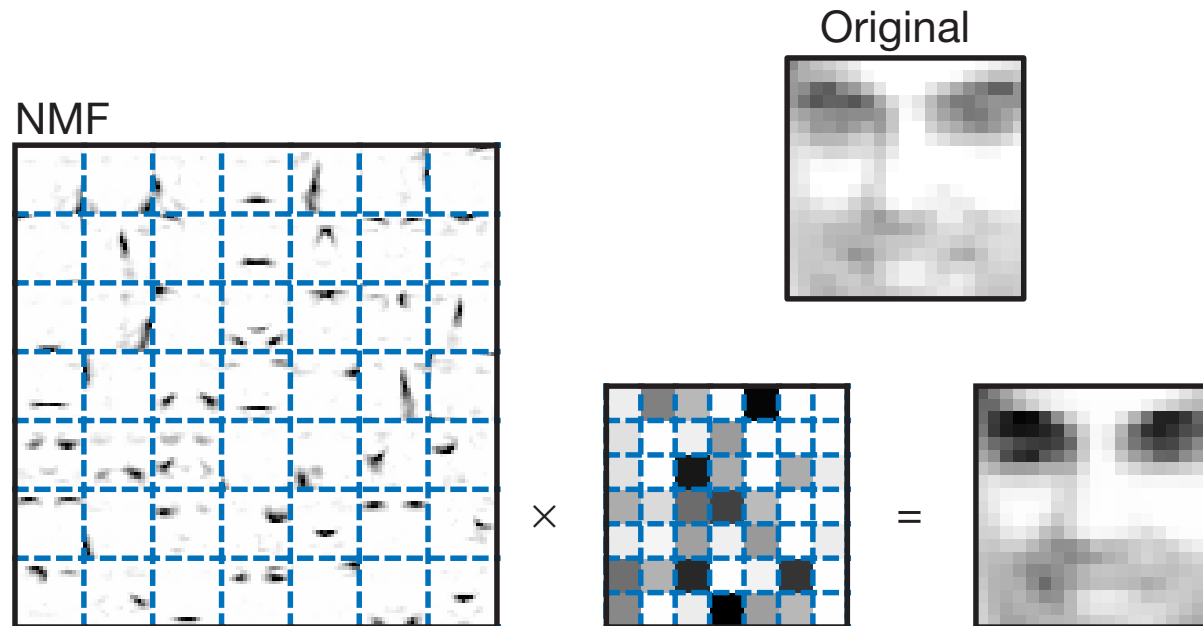
where $\mathbf{X} \geq \mathbf{0}$ means that \mathbf{X} is elementwise non-negative.

- found to be able to extract meaningful features (by empirical studies)
- under some conditions, the NMF solution is provably unique
- numerous applications, e.g., in machine learning, signal processing, remote sensing

NMF Examples

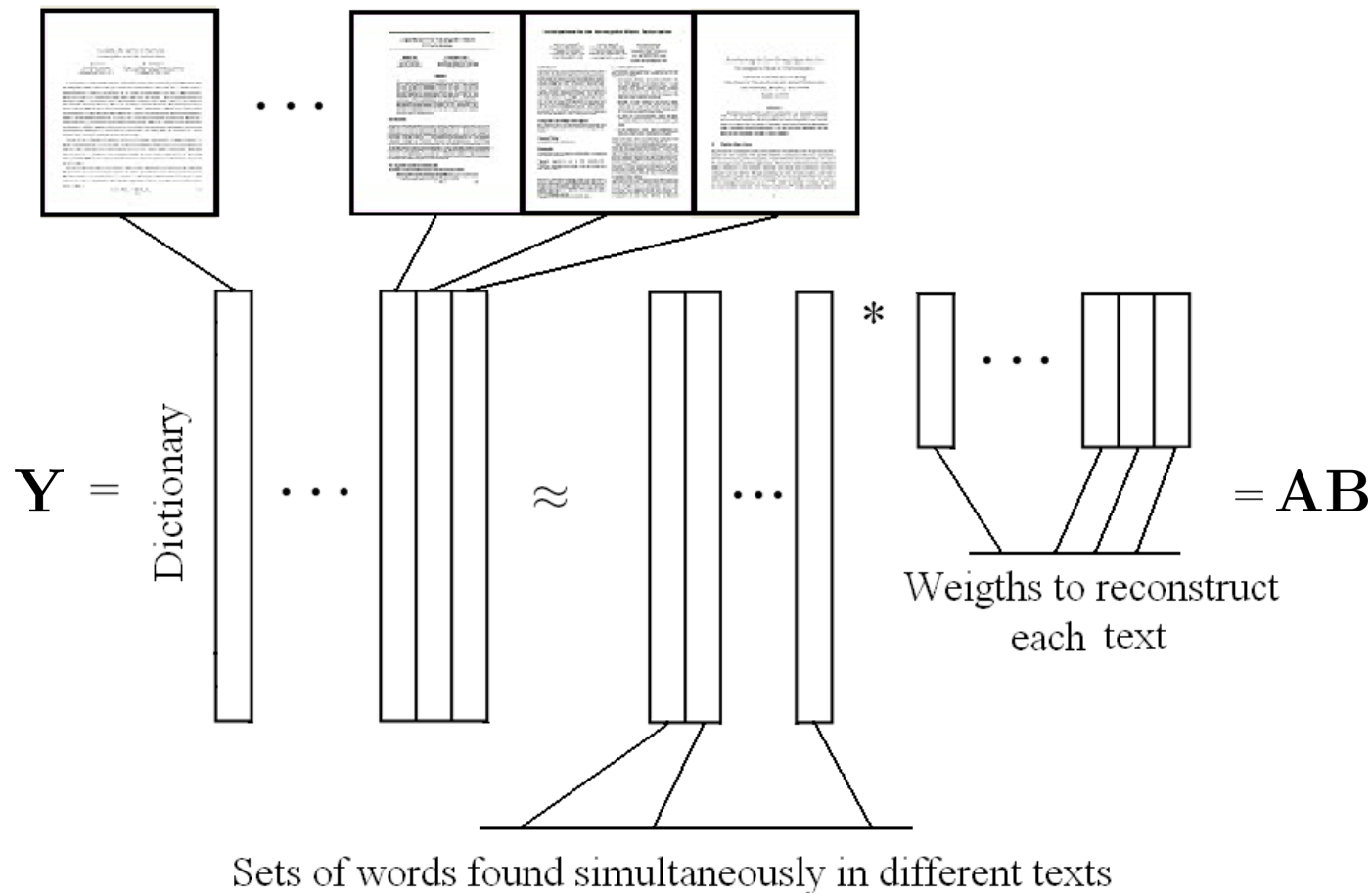
- **Image Processing:**

- $\mathbf{A} \geq \mathbf{0}$ constraints the basis elements to be nonnegative.
- $\mathbf{B} \geq \mathbf{0}$ imposes an additive reconstruction.



- the basis elements extract facial features such as eyes, nose and lips. Source: [\[Lee-Seung1999\]](#)

• Text Mining



- basis elements allow us to recover different topics;
- weights allow us to assign each text to its corresponding topics.

NMF

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times r}, \mathbf{B} \in \mathbb{R}^{r \times n}} \|\mathbf{Y} - \mathbf{AB}\|_F^2 \quad \text{s.t. } \mathbf{A} \geq \mathbf{0}, \mathbf{B} \geq \mathbf{0}.$$

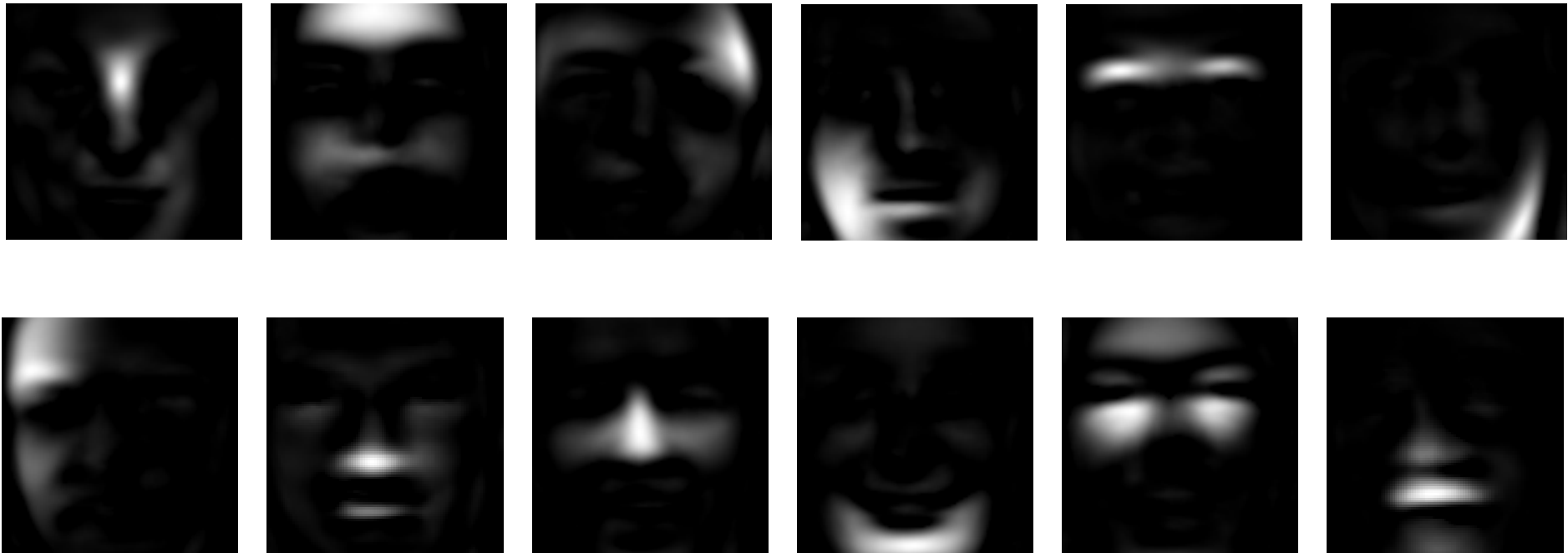
- NP-hard in general
- a practical way to go: alternating optimization w.r.t. \mathbf{A}, \mathbf{B}
 - given \mathbf{B} , minimizing $\|\mathbf{Y} - \mathbf{AB}\|_F^2$ over $\mathbf{A} \geq \mathbf{0}$ is convex; given \mathbf{A} , minimizing $\|\mathbf{Y} - \mathbf{AB}\|_F^2$ over $\mathbf{B} \geq \mathbf{0}$ is also convex
 - despite that, there is no closed-form solution with $\min_{\mathbf{A} \geq \mathbf{0}} \|\mathbf{Y} - \mathbf{AB}\|_F^2$ or $\min_{\mathbf{B} \geq \mathbf{0}} \|\mathbf{Y} - \mathbf{AB}\|_F^2$; it takes time to solve them when \mathbf{Y} is big
 - state of the art, such as the Lee-Seung multiplicative update, applies *inexact* alternating optimization

Toy Demonstration of NMF



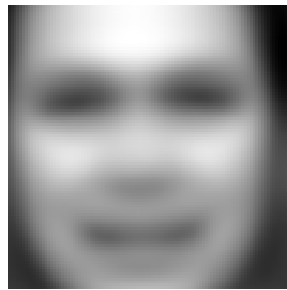
A face image dataset. Image size = 101×101 , number of face images = 13232. Each \mathbf{x}_n is the vectorization of one face image, leading to $m = 101 \times 101 = 10201$, $N = 13232$.

Toy Demonstration of NMF: NMF-Extracted Features



NMF settings: $r = 49$, Lee-Seung multiplicative update with 5000 iterations.

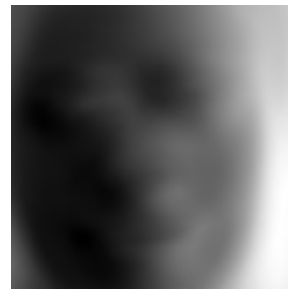
Toy Demonstration of NMF: Comparison with PCA



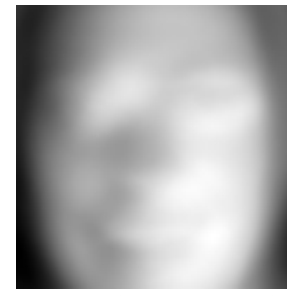
Mean face



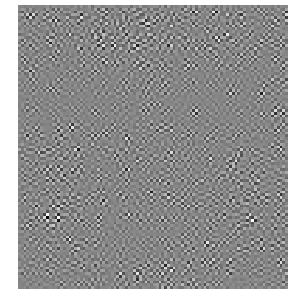
1st principal left
singular vector



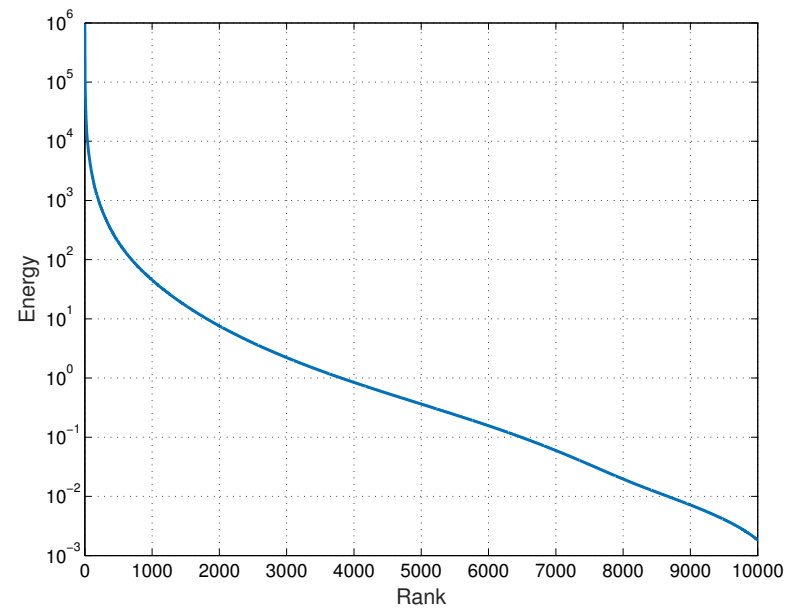
2nd principal left
singular vector



3th principal left
singular vector



last principal left
singular vector



Energy Concentration

Suggested Further Readings for NMF

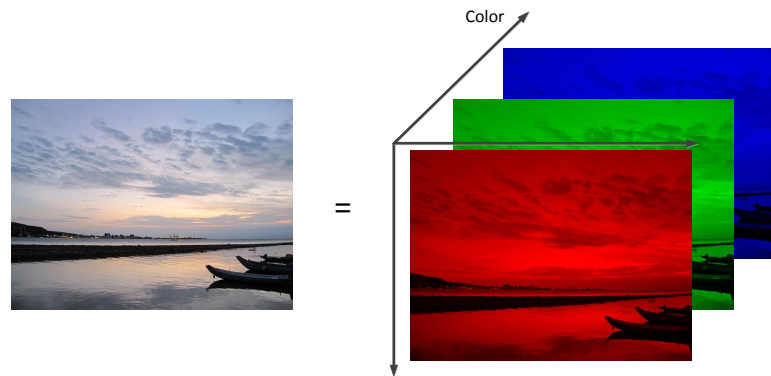
- **[Gillis2014]** for a review of some classical NMF algorithms and [separable NMF](#)
 - separable NMF is an NMF subclass that features provably polynomial-time or very simple algorithms for solving NMF under certain assumptions
- **[Fu-Huang-Sidiropoulos-Ma2018]** for an overview of classic and most recent developments of NMF identifiability
 - note volume minimization NMF, which has provably better identifiability than classic NMF

Topic 2: Tensor Decomposition

Tensor

A tensor is a multi-way numerical array. An N -way tensor is denoted by $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ and its entries by x_{i_1, i_2, \dots, i_N} .

- natural extension of matrices (which are two-way tensors)
- example: a color picture is a 3-way tensor, video 4-way



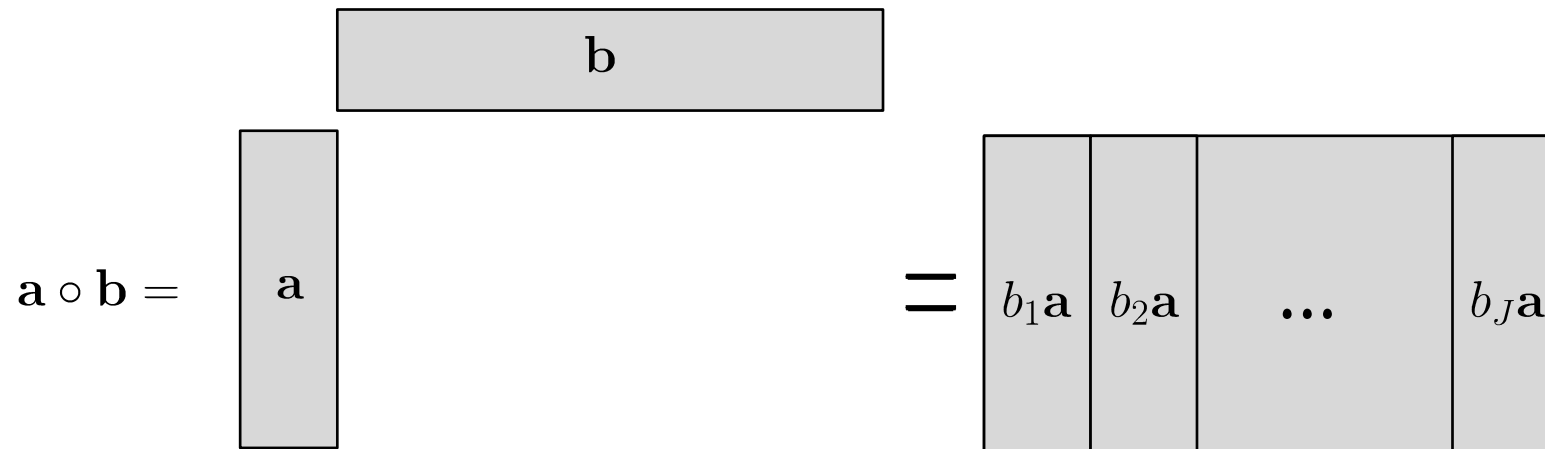
- applications: blind signal separation, chemometrics, data mining, ...
- **focus:** decomposition for 3-way tensors $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ (sufficiently complicated)

Outer Product

The outer product of $\mathbf{a} \in \mathbb{R}^I$ and $\mathbf{b} \in \mathbb{R}^J$ is an $I \times J$ matrix

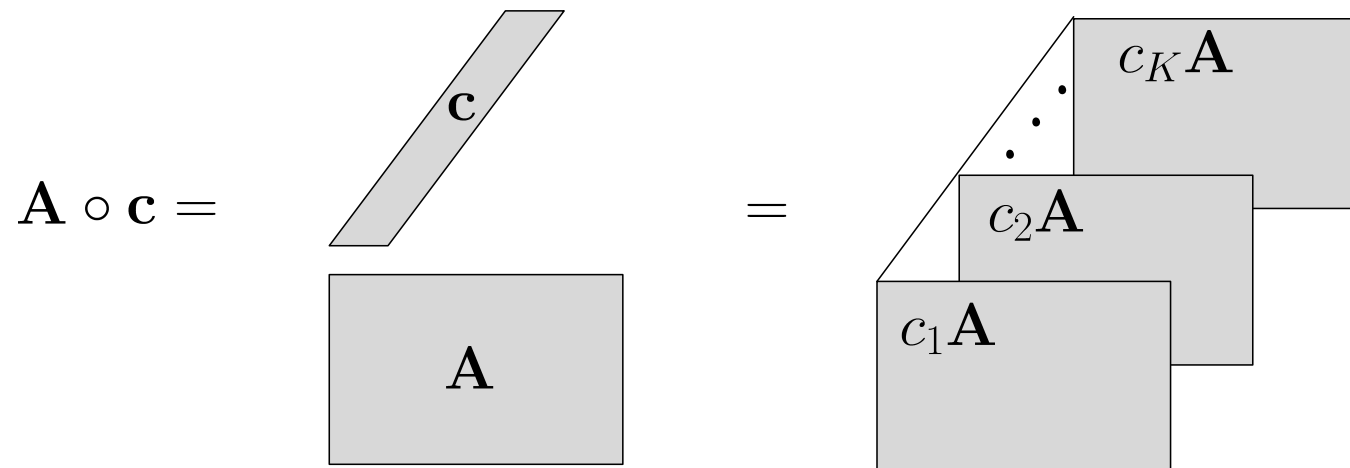
$$\mathbf{a} \circ \mathbf{b} = \mathbf{a}\mathbf{b}^T = [b_1\mathbf{a}, b_2\mathbf{a}, \dots, b_J\mathbf{a}].$$

Here, “ \circ ” is used to denote the outer product operator.



Outer Product

The outer product of a matrix $\mathbf{A} \in \mathbb{R}^{I \times J}$ and a vector $\mathbf{c} \in \mathbb{R}^K$, denoted by $\mathbf{A} \circ \mathbf{c}$, is a three-way $I \times J \times K$ tensor that takes the following form:

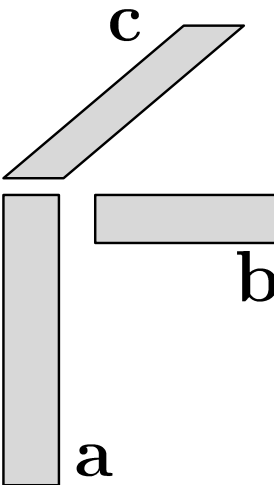


Specifically, if we let $\mathcal{X} = \mathbf{A} \circ \mathbf{c}$, then

$$\mathcal{X}(:, :, k) = c_k \mathbf{A}, \quad k = 1, \dots, K.$$

Outer Product

Tensors in the form of $\mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$ are called rank-1 tensors.

$$\mathbf{a} \circ \mathbf{b} \circ \mathbf{c} = \begin{array}{c} \text{c} \\ \text{a} \\ \text{b} \end{array}$$


Tensor Decomposition

Problem: decompose $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ as

$$x_{i,j,k} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr},$$

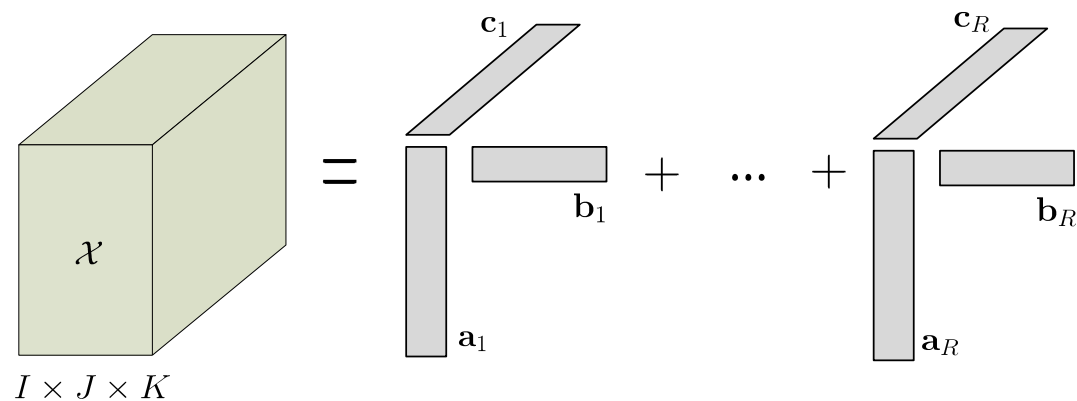
for some a_{ir} , b_{jr} , and c_{kr} , $i = 1, \dots, I$, $j = 1, \dots, J$, $k = 1, \dots, K$, $r = 1, \dots, R$, or equivalently,

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r,$$

where $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_R] \in \mathbb{R}^{I \times R}$, $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_R] \in \mathbb{R}^{J \times R}$, $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R] \in \mathbb{R}^{K \times R}$.

Tensor Decomposition

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \quad (*)$$

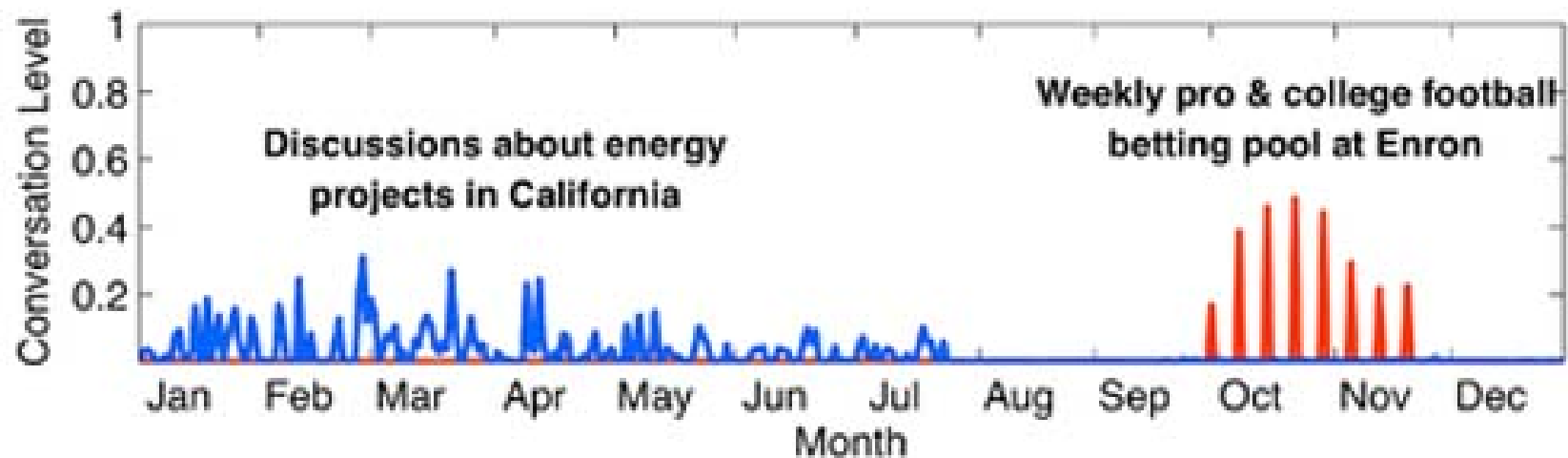
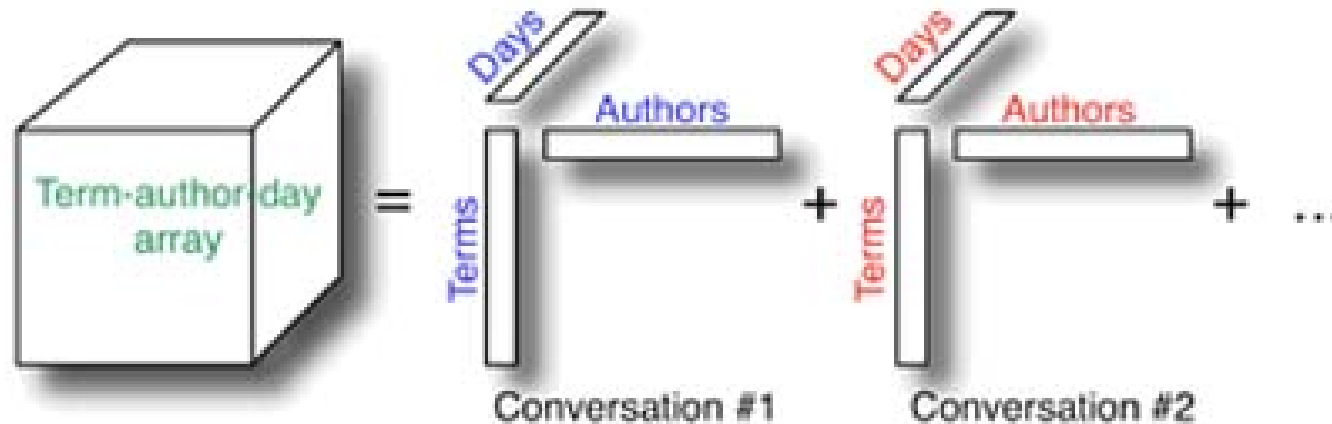


- a sum of rank-1 tensors
- the smallest R that satisfies (*) is called the rank of the tensor
- many names: tensor rank decomposition, canonical polyadic decomposition (CPD), parallel factor analysis (PARARFAC), CANDECOMP

Application Example: Enron Data

- About Enron
 - once ranked the 6th largest energy company in the world
 - shares were worth \$90.75 at their peak in Aug 2000 and dropped to \$0.67 in Jan 2002
 - most top executives were tried for fraud after it was revealed in Nov 2001 that Enron's earning has been overstated by several hundred million dollars
- Enron email database
 - a large database of over 600,000 emails generated by 158 employees of the Enron Corporation and acquired by the Federal Energy Regulatory Commission during its investigation after the company's collapse, according to wiki

Application Example: Enron Data



Source: <http://www.math.uwaterloo.ca/~hdesterc/websiteW/Data/presentations/pres2012/Valencia.pptx.pdf>

Uniqueness of Tensor Decomposition

The low-rank matrix factorization problem $\mathbf{X} = \mathbf{A}\mathbf{B}$ is not unique.

We also have many matrix decompositions: SVD, QR, ...

Theorem 10.1. Let $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ be a factor for $\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$. If

$$\text{krank}(\mathbf{A}) + \text{krank}(\mathbf{B}) + \text{krank}(\mathbf{C}) \geq 2R + 2,$$

then $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ is the unique tensor decomposition factor for \mathcal{X} up to a common column permutation and scaling.

- **Implication:** under some mild conditions with $\mathbf{A}, \mathbf{B}, \mathbf{C}$, low-rank tensor decomposition is essentially unique
- the above theorem is just among one of the known sufficient conditions for unique tensor decomposition; some other results suggest much more relaxed conditions for unique tensor decomposition

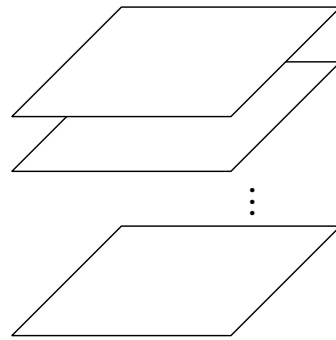
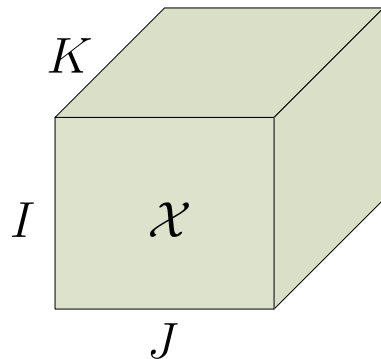
Slabs

A slab of a tensor is a matrix obtained by fixing one index of the tensor.

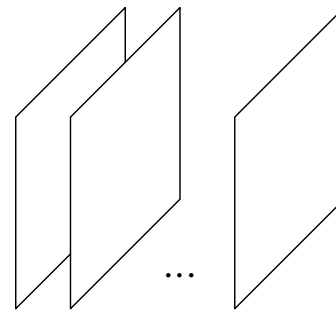
Horizontal slabs: $\{\mathbf{X}_i^{(1)} = \mathcal{X}(i, :, :)\}_{i=1}^I$

Lateral slabs: $\{\mathbf{X}_j^{(2)} = \mathcal{X}(:, j, :)\}_{j=1}^J$

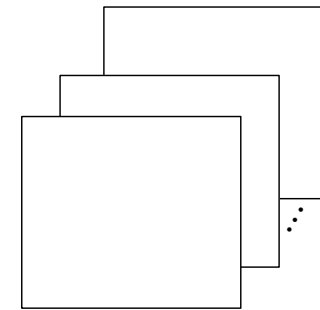
Frontal slabs: $\{\mathbf{X}_k^{(3)} = \mathcal{X}(:, :, k)\}_{k=1}^K$



Horizontal slabs
 $\mathbf{X}_i^{(1)} = \mathcal{X}(i, :, :)$



Lateral slabs
 $\mathbf{X}_j^{(2)} = \mathcal{X}(:, j, :)$

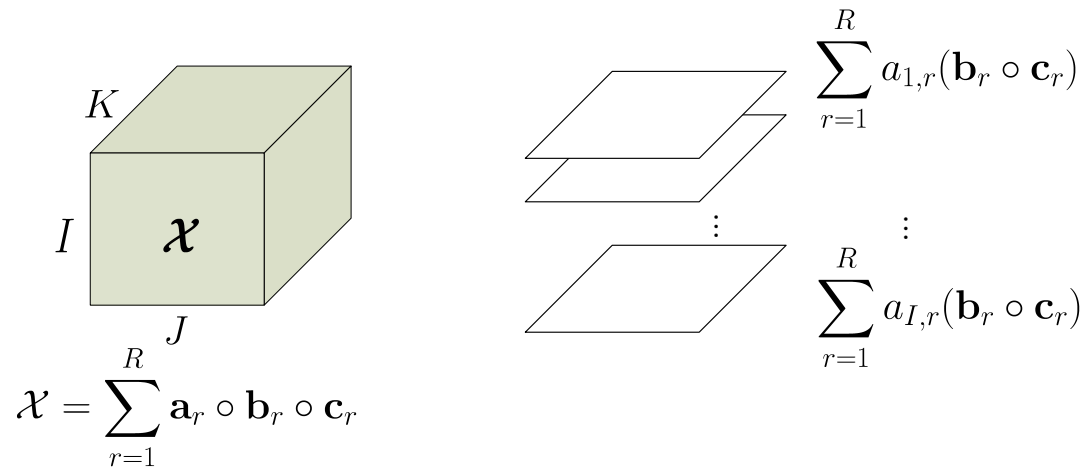


Frontal slabs
 $\mathbf{X}_k^{(3)} = \mathcal{X}(:, :, k)$

PARAFAC Formulation

Consider the horizontal slabs as an example.

$$\mathbf{X}_i^{(1)} = \sum_{r=1}^R a_{i,r} (\mathbf{b}_r \circ \mathbf{c}_r) = \sum_{r=1}^R a_{i,r} \mathbf{b}_r \mathbf{c}_r^T$$



We can write

$$\mathbf{X}_i^{(1)} = \mathbf{B} \mathbf{D}_{\mathbf{A}(i,:)} \mathbf{C}^T, \quad \mathbf{D}_{\mathbf{A}(i,:)} = \text{Diag}(\mathbf{A}(i, :)).$$

PARAFAC Formulation

Khatri-Rao (KR) product: the KR product of $\mathbf{A} \in \mathbb{R}^{I \times R}$ and $\mathbf{B} \in \mathbb{R}^{J \times R}$ is

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1, \dots, \mathbf{a}_R \otimes \mathbf{b}_R].$$

A key KR property: let $\mathbf{D} = \text{diag}(\mathbf{d})$. We have

$$\text{vec}(\mathbf{A}\mathbf{D}\mathbf{B}^T) = (\mathbf{B} \odot \mathbf{A})\mathbf{d}.$$

Idea: recall the horizontal slabs expression

$$\mathbf{X}_i^{(1)} = \mathbf{B}\mathbf{D}_{\mathbf{A}(i,:)}\mathbf{C}^T, \quad \mathbf{D}_{\mathbf{A}(i,:)} = \text{Diag}(\mathbf{A}(i, :)).$$

It can be reexpressed as

$$\text{vec}(\mathbf{X}_i^{(1)}) = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T(i, :).$$

– roughly speaking, we can do $\mathbf{A}^T(i, :) = (\mathbf{C} \odot \mathbf{B})^\dagger \text{vec}(\mathbf{X}_i^{(1)})$.

PARAFAC Formulation

By the trick above, we can write

$$\text{Horizontal slabs: } \mathbf{X}_i^{(1)} = \mathcal{X}(i, :, :) = \mathbf{B}\mathbf{D}_{\mathbf{A}(i,:)}\mathbf{C}^T$$

$$\text{vec}(\mathbf{X}_i^{(1)}) = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T(i, :)$$

$$\text{Lateral slabs: } \mathbf{X}_j^{(2)} = \mathcal{X}(:, j, :) = \mathbf{C}\mathbf{D}_{\mathbf{B}(j,:)}\mathbf{A}^T$$

$$\text{vec}(\mathbf{X}_j^{(2)}) = (\mathbf{A} \odot \mathbf{C})\mathbf{B}^T(j, :)$$

$$\text{Frontal slabs: } \mathbf{X}_k^{(3)} = \mathcal{X}(:, :, k) = \mathbf{A}\mathbf{D}_{\mathbf{C}(k,:)}\mathbf{B}^T$$

$$\text{vec}(\mathbf{X}_k^{(3)}) = (\mathbf{B} \odot \mathbf{A})\mathbf{C}^T(k, :)$$

Observation:

- fixing \mathbf{B} , \mathbf{C} , solving for \mathbf{A} is a linear system problem
- fixing \mathbf{A} , \mathbf{C} , solving for \mathbf{B} is a linear system problem
- fixing \mathbf{A} , \mathbf{B} , solving for \mathbf{C} is a linear system problem

PARAFAC Formulation

A tensor decomposition formulation: given $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, $R > 0$, solve

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \left\| \mathcal{X} - \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \right\|_F^2,$$

- NP-hard in general
- can be conveniently handled by alternating optimization w.r.t. $\mathbf{A}, \mathbf{B}, \mathbf{C}$
 - e.g., optimization w.r.t. \mathbf{A} and fixing \mathbf{B}, \mathbf{C} is

$$\min_{\mathbf{A}} \sum_{i=1}^I \left\| \mathbf{X}_i^{(1)} - \mathbf{B} \mathbf{D}_{\mathbf{A}(i,:)} \mathbf{C}^T \right\|_F^2 = \min_{\mathbf{A}} \sum_{i=1}^I \left\| \text{vec}(\mathbf{X}_i^{(1)}) - (\mathbf{C} \odot \mathbf{B}) \mathbf{A}^T(i, :) \right\|_2^2$$

and a solution is $(\mathbf{A}^*(i, :))^T = (\mathbf{C} \odot \mathbf{B})^\dagger \text{vec}(\mathbf{X}_i^{(1)})$, $i = 1, \dots, I$.

Suggested Further Readings for Tensor Decomposition

- [Sidiropoulos-De Lathauwer-Fu-Huang-Papalexakis-Faloutsos2017]

References

- [Lee-Seung1999]** D.D. Lee and H.S. Seung. “Learning the parts of objects by non-negative matrix factorization,” *Nature*, 1999.
- [Gillis2014]** N. Gillis, “The Why and how of nonnegative matrix factorization”, in *Regularization, Optimization, Kernels, and Support Vector Machines*, J.A.K. Suykens, M. Signoretto and A. Argyriou (eds), Chapman & Hall/CRC, Machine Learning and Pattern Recognition Series, 2014.
- [Fu-Huang-Sidiropoulos-Ma2018]** X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, “Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications,” *IEEE Signal Process. Mag.*, 2019.
- [Sidiropoulos-De Lathauwer-Fu-Huang-Papalexakis-Faloutsos2017]** N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor decomposition for signal processing and machine learning,” *IEEE Trans. Signal Process.*, 2017.