# Tensor methods: An introduction

Lieven De Lathauwer

STADIUS Seminar
Leuven, 23 February 2026















© Copyright KU Leuven. Without written permission of the author it is forbidden to reproduce or adapt in any form or by any means any part of this presentation. Requests for obtaining the right to reproduce or utilize parts of this presentation should be addressed to Lieven De Lathauwer via email at Lieven.DeLathauwer@kuleuven.be.
L. De Lathauwer, "Tensor methods: An introduction," slides presented at a STADIUS colloquium seminar, Leuven, February 23, 2026 (71 p.)

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

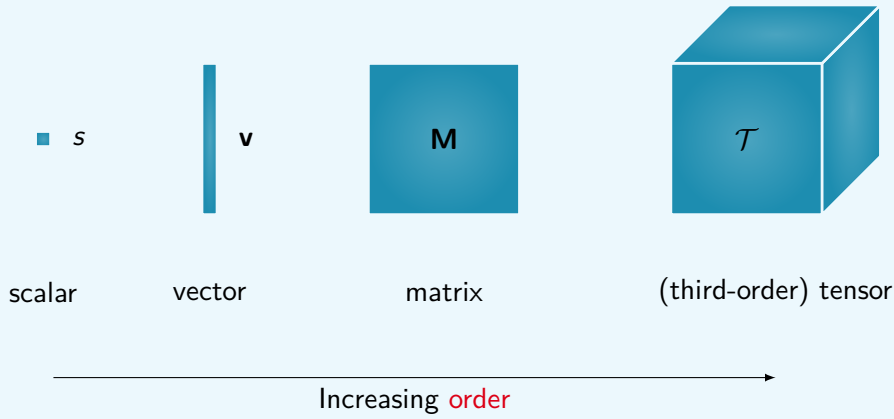
Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

Tensorlab and Tensorlab+

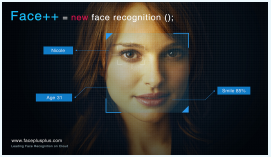
Higher-order tensors are a generalization of scalars, vectors, and matrices.



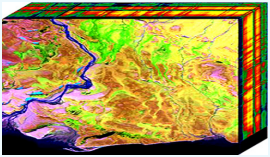
# Examples of tensors



- Color images, video, ...
- Hyperspectral images
- Text mining: term $\times$ document $\times$ author
- (Social) networks: score $\times$ object $\times$ referee $\times$ criterion
- Event Related Potential (ERP): subject $\times$ time $\times$ frequency $\times$ channel $\times$ ...
- Face recognition: people $\times$ pose $\times$ illumination $\times$ angle
- Multi-way data acquisition (intentional)





# From matrix to tensor: tensorization

- Set of data matrices, recorded at different $t$, under different conditions...
- Sets of covariance matrices
- Higher-order statistics
- Deliberate tensorization of data matrix
  time-frequency, wavelet, Hankel, Löwner, derivatives, decimation/segmentation...
- ...



## Exponentials as rank-1 matrices and tensors

- Consider an exponential signal $f(k) = az^k$, sampled uniformly:

$$\begin{bmatrix} a & az & az^2 & az^3 & \cdots \end{bmatrix}$$

- Let's arrange it in a Hankel matrix $\mathbf{H}$ — which has rank 1!

$$\mathbf{H} = \begin{bmatrix} a & az & az^2 & \cdots \\ az & az^2 & az^3 & \cdots \\ az^2 & az^3 & az^4 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} = a \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \end{bmatrix} \begin{bmatrix} 1 & z & z^2 & \cdots \end{bmatrix}$$

- Row $\begin{bmatrix} 1 & z & z^2 & \cdots \end{bmatrix}$ can by itself be mapped to Hankel matrix. Also column can be mapped to Hankel matrix. Etc.
- Leads to ... a symmetric $(2 \times 2 \times \ldots \times 2)$ tensor $\mathcal{H}$ that has rank 1!

$$\mathcal{H} = a \begin{bmatrix} 1 \\ z \end{bmatrix} \otimes \begin{bmatrix} 1 \\ z \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ z \end{bmatrix}$$

$a$ : amplitude
$z$ : base

- Tensor recognition

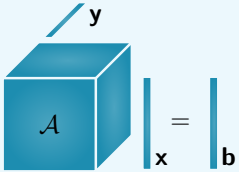
# Tensor recognition: example (1)

## Linear transformation



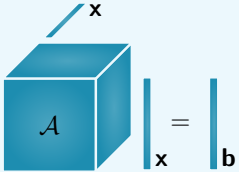

$\sum_j a_{ij} x_j = b_i$ matrix $\mathbf{A}$ (order 2)

## Bilinear transformation




$\sum_{jk} a_{ijk} x_j y_k = b_i$ tensor $\mathcal{A}$ (order 3)

## Quadratic transformation




$\sum_{jk} a_{ijk} x_j x_k = b_i$ tensor $\mathcal{A}$ (order 3)

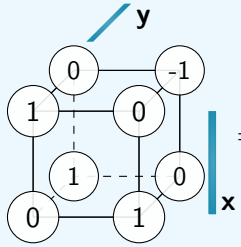
# Tensor recognition: example (2)

## Complex number multiplication

$$(a+bi)(c+di) = (ac-bd) + (bc+ad)i$$

$$ac - bd = \begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} 1 & \\ & -1 \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix}$$

$$bc + ad = \begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix}$$




$$= \begin{bmatrix} ac-bd \\ bc+ad \end{bmatrix} \quad \text{with } \mathbf{x} = \begin{bmatrix} a \\ b \end{bmatrix}, \mathbf{y} = \begin{bmatrix} c \\ d \end{bmatrix}$$

## Matrix multiplication

$\mathbf{C} = \mathbf{A}\mathbf{B}^{\mathsf{T}}$ e.g. $(2 \times 2)$ $\mathbf{A}, \mathbf{B}, \mathbf{C} \rightarrow (4 \times 4 \times 4)$ tensor

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

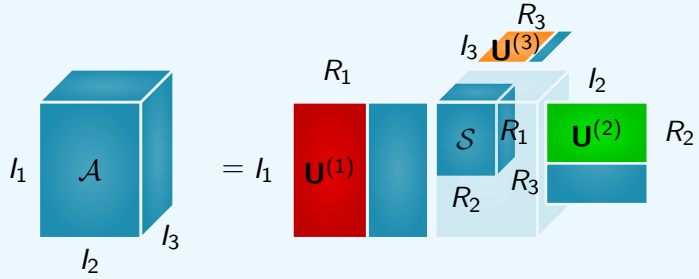
Large-scale data analysis and curse of dimensionality

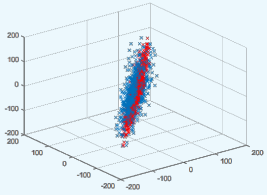
Completion

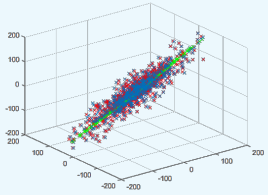
Harmonic retrieval

Tensorlab and Tensorlab+

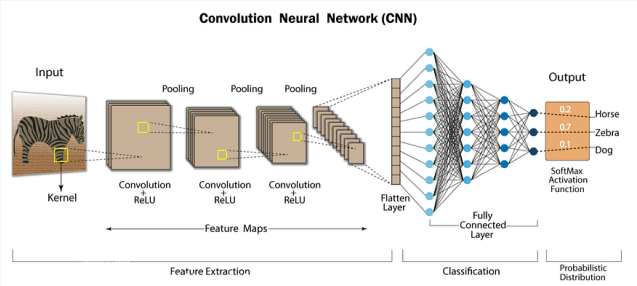
# From linear to multilinear PCA: dominant subspaces and compression





# Application example: efficient evaluation of CNNs




[Shahriar 2023]

1D convolution: $\mathbf{y}[n] = \mathbf{f}[n] * \mathbf{x}[n] = \sum_{n'=1}^{N'} \mathbf{f}[n']\mathbf{x}[n-n']$

2D SISO convolution: $\mathbf{Y}[m,n] = \mathbf{F}[m,n] *_2 \mathbf{X}[m,n]$
$= \sum_{m'=1}^{M'} \sum_{n'=1}^{N'} \mathbf{F}[m',n']\mathbf{X}[m-m',n-n']$

2D MIMO convolution: $\mathbf{Y}_i[m,n] = \mathbf{F}_{ij}[m,n] *_2 \mathbf{X}_j[m,n]$
$\rightarrow$ compression $\mathcal{F} \in \mathbb{R}^{I \times J \times M' \times N'}$

[Denton, Zaremba, et al. 2014; Kim, Par, et al. 2016]

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

Tensorlab and Tensorlab+

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

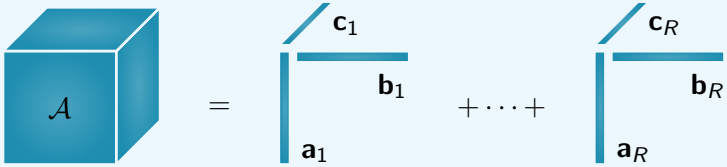
Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

Tensorlab and Tensorlab+

# From dyadic to polyadic decomposition: signal separation and data analysis








# Factor Analysis and Blind Source Separation

- Decompose a data matrix in rank-1 terms that can be interpreted

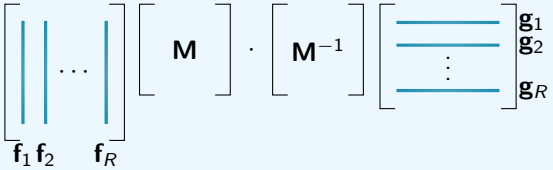
  E.g. statistics, telecommunication, biomedical applications, chemometrics, data analysis, ...

$$\mathbf{A} = \mathbf{F} \cdot \mathbf{G}^{\mathsf{T}}$$

$$\mathbf{A} = \mathbf{f}_1 \mathbf{g}_1 + \mathbf{f}_2 \mathbf{g}_2 + \cdots + \mathbf{f}_R \mathbf{g}_R$$

- $\mathbf{F}$: mixing matrix  $\mathbf{G}$: source signals
- Matrix decomposition in rank-1 terms is not unique!

$$\mathbf{A} = (\mathbf{FM}) \cdot (\mathbf{M}^{-1}\mathbf{G}^{\mathsf{T}}) = \tilde{\mathbf{F}} \cdot \tilde{\mathbf{G}}^{\mathsf{T}}$$



## What about SVD?

The SVD is unique . . . thanks to orthogonality constraints!

$$\mathbf{A} = \mathbf{U}^{(1)} \cdot \mathbf{S} \cdot \mathbf{U}^{(2)^\mathsf{T}} = \sum_{r=1}^{R} s_{rr} \mathbf{u}_r^{(1)} \mathbf{u}_r^{(2)^\mathsf{T}}$$

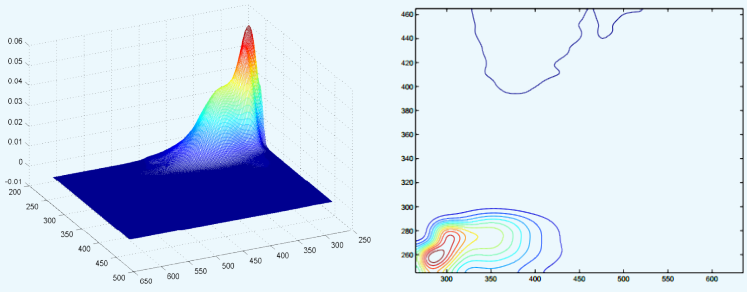
with $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$ orthogonal, $\mathbf{S}$ diagonal.

- Whether these constraints make sense, depends on the application
- SVD is great for dimensionality reduction/numerical compression
  best rank-$R$ approximation $\leftarrow$ truncated SVD



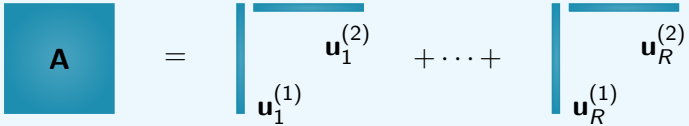
# An example where matrices fail

## Excitation-emission spectroscopy



# Excitation-emission spectroscopy

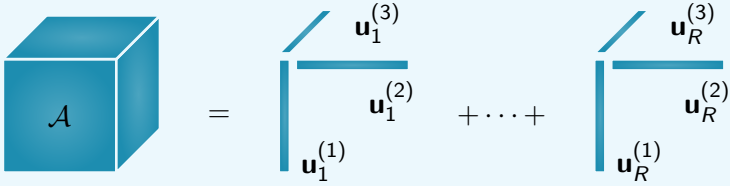
row vector $\sim$ excitation spectrum
column vector $\sim$ emission spectrum
coefficients $\sim$ concentrations




Spectra are nonnegative (and not orthogonal)
Nonnegative Matrix Factorization (NMF) not unique in general

## Tensor approach

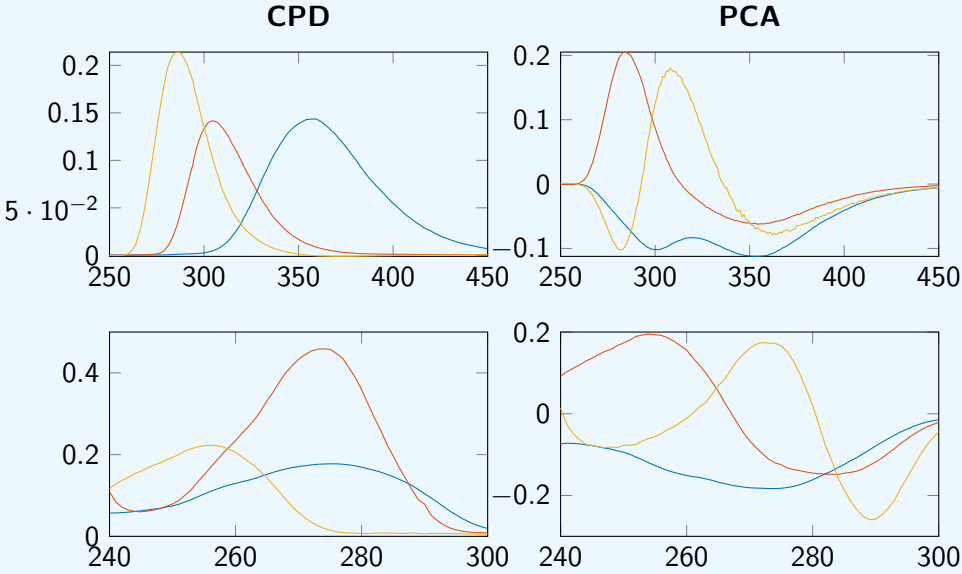
row vector $\sim$ excitation spectrum
column vector $\sim$ emission spectrum
coefficients $\sim$ concentrations



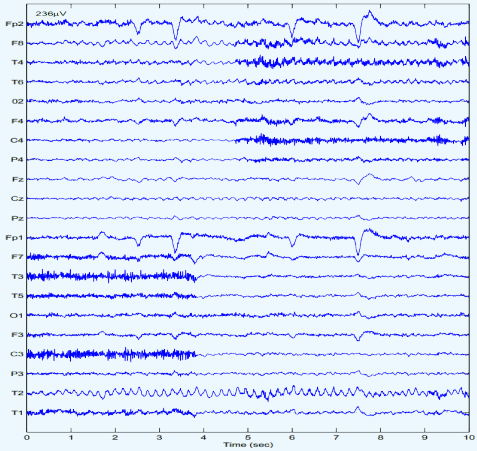

- Surprising fact: unique under mild conditions on number of terms and differences between terms
- Orthogonality (triangularity, ...) not required (but may be imposed)

[Smilde, Bro, et al. 2004]

# Example: amino acids

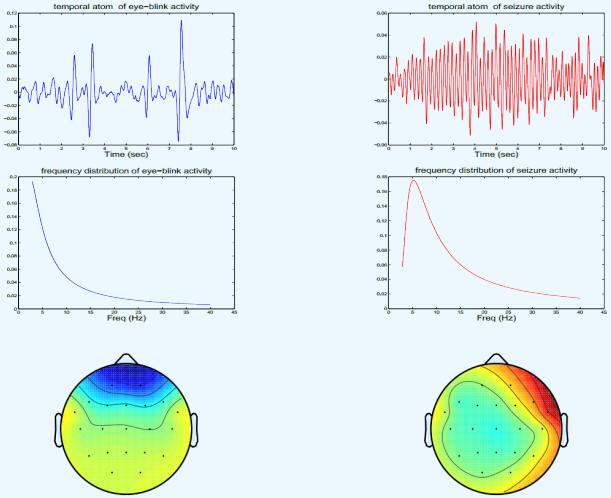


# Application: detection epileptic seizure in EEG




Tensorization: biorthogonal wavelet

# Components: eye blink and epileptic activity

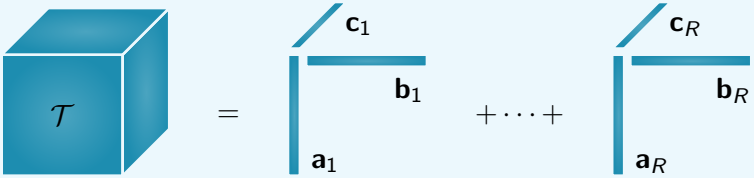



[De Vos, Vergult, et al. 2007; Acar, Aykut-Bingol, et al. 2007]

# Canonical polyadic decomposition

Rank: minimal number of rank-1 terms [Hitchcock 1927]

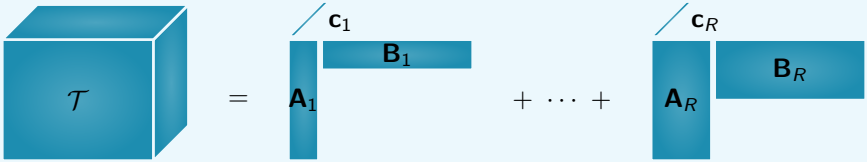
Canonical Polyadic Decomposition (CPD): decomposition in minimal number of rank-1 terms [Harshman 1970; Carrol and Chang 1970]




Unique under mild conditions on number of terms and differences between terms

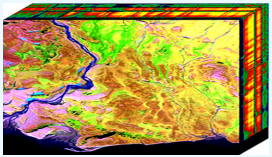
Additional constraints such as orthogonality, triangularity, ... are not required, but may be imposed.

# Decomposition in multilinear rank-$(L, L, 1)$ terms

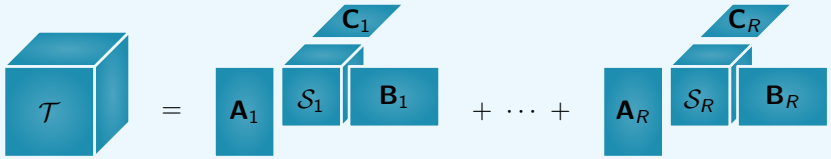



Surprising fact: unique under mild conditions

[De Lathauwer 2008; De Lathauwer 2011; Sørensen and De Lathauwer 2015; Sørensen, Domanov, et al. 2015; Domanov and De Lathauwer 2020]



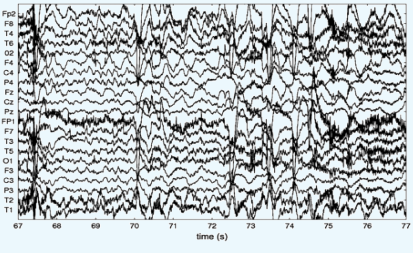
# Decomposition in ML rank-$(R_1, R_2, R_3)$ terms



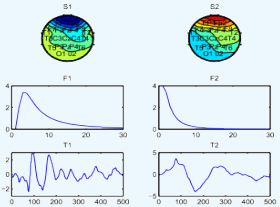

Unique under mild conditions
Rank $\leftrightarrow$ multilinear rank
Atoms (rank-1) $\leftrightarrow$ molecules (low ML rank)

[De Lathauwer 2008]

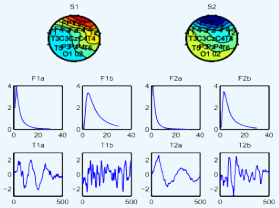
# Application: detection epileptic seizure in EEG



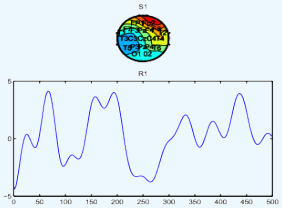

(a) Raw EEG




(b) CPD




(c) CWT-BTD




(d) H-BTD

[Hunyadi, Camps, et al. 2014]

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

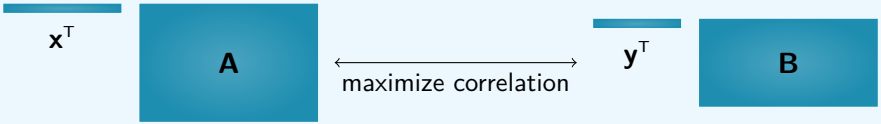
Tensorlab and Tensorlab+

# What about CCA/GSVD?

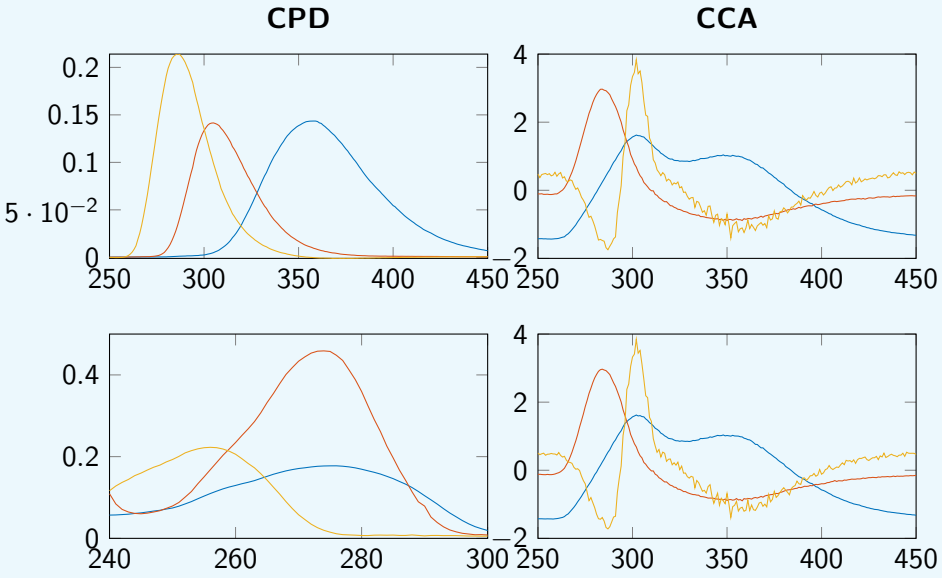
Step 1
$$\mathbf{A} = \mathbf{U}_A \cdot \mathbf{\Sigma}_A \cdot \mathbf{V}_A^\top \quad \text{SVD}$$
$$\mathbf{B} = \mathbf{U}_B \cdot \mathbf{\Sigma}_B \cdot \mathbf{V}_B^\top \quad \text{SVD}$$

Step 2
$$\mathbf{W} = \mathbf{V}_A^\top \cdot \mathbf{V}_B$$

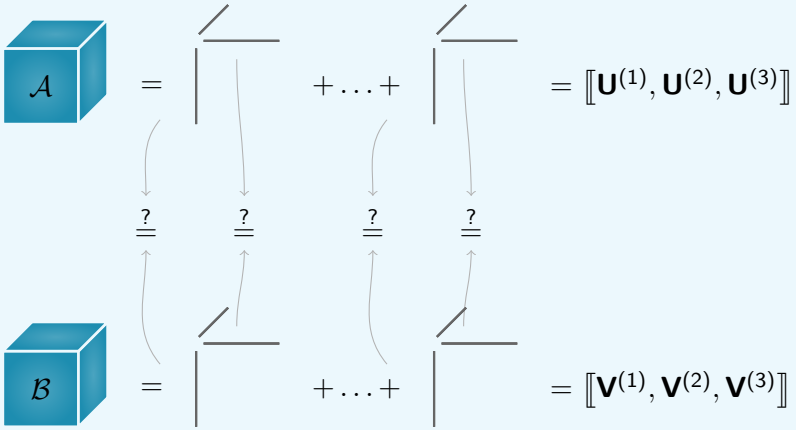
Step 3
Singular values of $\mathbf{W}$ = cosines of principal angles between subspaces row $(\mathbf{A})$ and row $(\mathbf{B})$.



# Example: amino acids



## Graphical representation of comparing factor matrices




$$\mathbf{A}_{[1;2,3]} = \mathbf{U}^{(1)} \left(\mathbf{U}^{(2)} \odot \mathbf{U}^{(3)}\right)^{\mathsf{T}} \rightarrow \quad \text{Tucker: } \operatorname{col}\left(\mathbf{A}_{[1;2,3]}\right) \simeq \operatorname{col}\left(\mathbf{U}^{(1)}\right)$$

$$\overset{\text{SVD}}{=} \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^{\mathsf{T}} \qquad \text{Here: } \operatorname{row}\left(\mathbf{A}_{[1;2,3]}\right) \simeq \operatorname{row}\left(\left(\mathbf{U}^{(2)} \odot \mathbf{U}^{(3)}\right)^{\mathsf{T}}\right)$$

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

Tensorlab and Tensorlab+

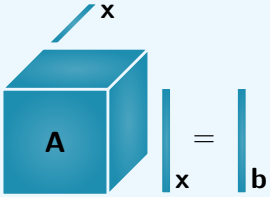
# From linear to multilinear/polynomial

## Linear




$$\sum_j a_{ij} x_j = b_i$$

## Bilinear/quadratic




$$\sum_{jk} a_{ijk} x_j y_k = b_i$$
$$\sum_{jk} a_{ijk} x_j x_k = b_i$$

## Multilinear/polynomial

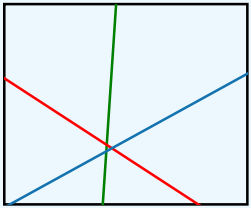
Tensor higher-order

$$\sum_{j_1 j_2 \ldots j_D} a_{i j_1 j_2 \ldots j_D} x_{j_1} y_{j_2} \ldots z_{j_D} = b_i$$
$$\sum_{j_1 j_2 \ldots j_D} a_{i j_1 j_2 \ldots j_D} x_{j_1} x_{j_2} \ldots x_{j_D} = b_i$$

# Overdetermined linear and polynomial systems of equations

Overdetermined *linear* systems:

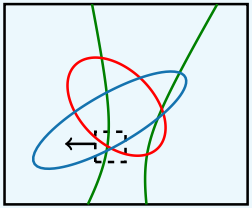
*local*, *simple* models, *one* solution



$$
\begin{aligned}
a_{11}x + a_{12}y &= b_1 \\
a_{21}x + a_{22}y &= b_2 \\
a_{31}x + a_{32}y &= b_3
\end{aligned}
$$

Overdetermined *polynomial* systems:

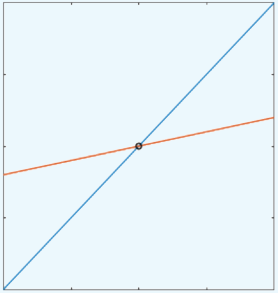
*global*, *rich* models, *multiple* solutions

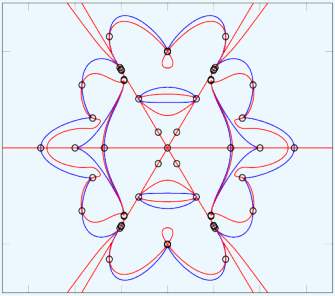


$$
\begin{aligned}
a_{11}x + a_{12}y + a_{13}x^2 + a_{14}xy + a_{15}y^2 &= b_1 \\
a_{21}x + a_{22}y + a_{23}x^2 + a_{24}xy + a_{25}y^2 &= b_2 \\
a_{31}x + a_{32}y + a_{33}x^2 + a_{34}xy + a_{35}y^2 &= b_3
\end{aligned}
$$

Nonlinearities are essential in AI

# From linear to multilinear/polynomial





Linear equations

$$\begin{cases} a_{11}x_1 + a_{12}x_2 - b_1 = 0 \\ a_{21}x_1 + a_{22}x_2 - b_2 = 0 \end{cases}$$

Polynomial equations

$$\begin{cases} p_1(x_1, x_2) = 0 \\ p_2(x_1, x_2) = 0 \end{cases}$$

[Sorber, Van Barel, et al. 2014]

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

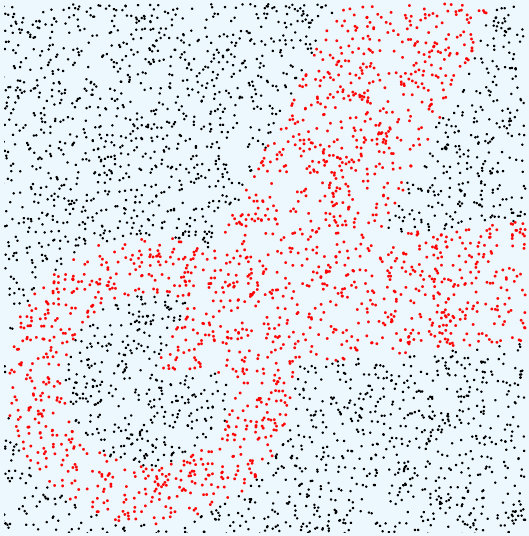
Completion

Harmonic retrieval

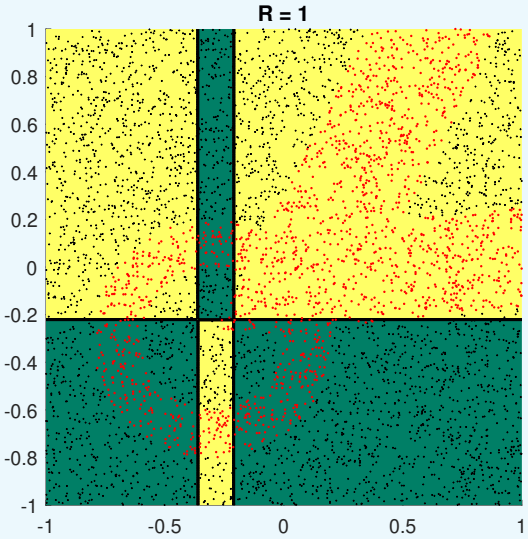
Tensorlab and Tensorlab+

# Nonlinear decision boundary modelled using splines

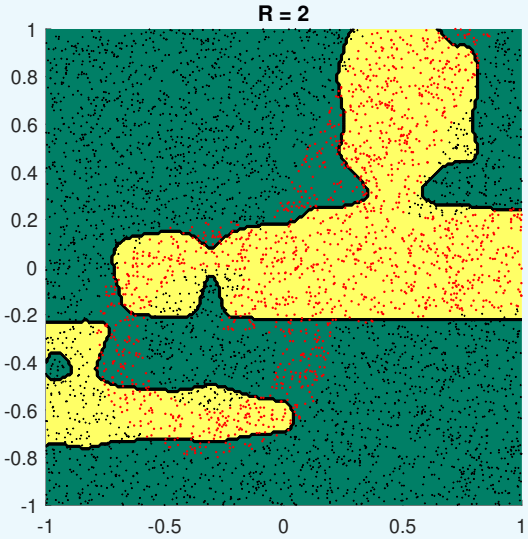
**Consider the labeled dataset:**



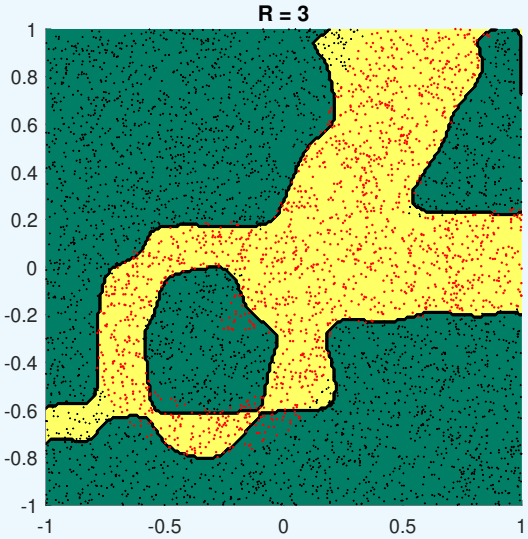
# The separable rank can be increased to account for complexity of the classification sets



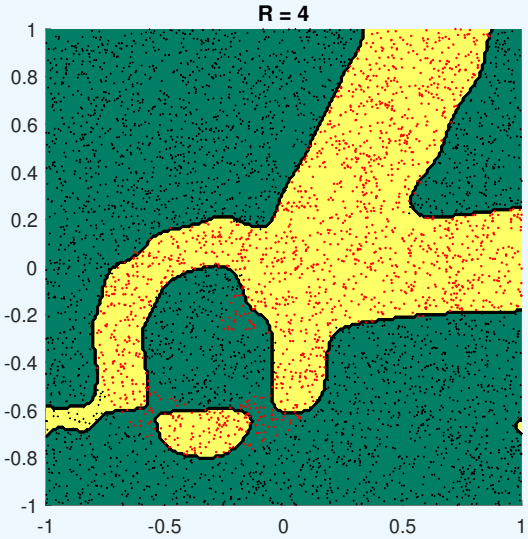
# The separable rank can be increased to account for complexity of the classification sets



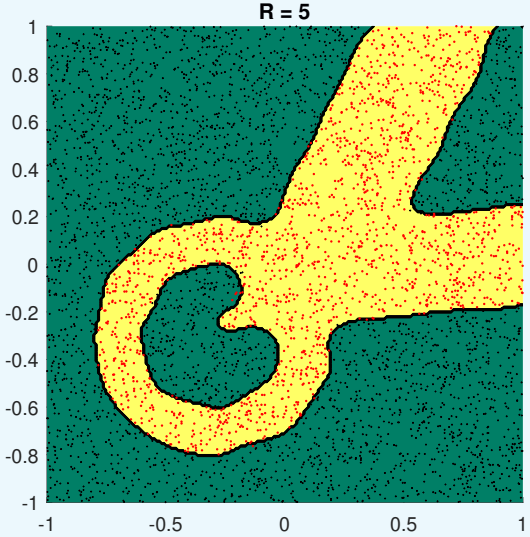
# The separable rank can be increased to account for complexity of the classification sets



# The separable rank can be increased to account for complexity of the classification sets



# The separable rank can be increased to account for complexity of the classification sets

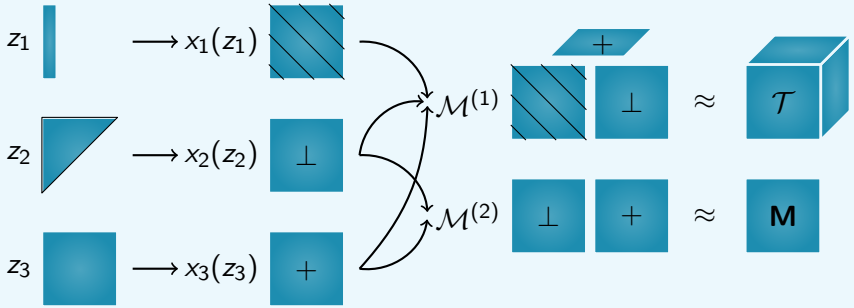



Possibility to impose constraints

## Overview


- Introduction
- Dimensionality reduction - Principal Component Analysis
- Denoising
- Blind source separation - Latent Variable analysis
- Blind similarity assessment
- Polynomial equations
- Regression and classification
- Structured data fusion
- Large-scale data analysis and curse of dimensionality
- Completion
- Harmonic retrieval
- Tensorlab and Tensorlab+

# Coupled matrix/tensor decompositions




Multiview data analysis / data fusion — e.g. EEG-fMRI

Divide-and-conquer data analysis / data fission — e.g. structured arrays

# Application: data fusion for completion/prediction/collaborative filtering

**Five coupled data sets:** *user-location-activity*, *user-user*, *location-feature*, *activity-activity* and *user-location*
**Challenge:** predict user participation in activities
**Solution with SDF:** compute coupled tensor factorization

$$\underset{\mathbf{U},\mathbf{L},\mathbf{A},\mathbf{F},\mathbf{d}_{\mathsf{uu}},\mathbf{d}_{\mathsf{aa}},\mathbf{d}_{\mathsf{ul}}}{\text{minimize}} \quad \frac{\omega_1}{2}\,||\mathcal{T} - [\![\mathbf{U},\mathbf{L},\mathbf{A}]\!]||_{\mathcal{W}}^2 + \frac{\omega_2}{2}\,||\mathbf{M}_{\mathsf{uu}} - \mathbf{U}\mathbf{D}_{\mathsf{uu}}\mathbf{U}^{\mathsf{T}}||_{\mathsf{F}}^2 +$$
$$\frac{\omega_3}{2}\,||\mathbf{M}_{\mathsf{ul}} - \mathbf{U}\mathbf{D}_{\mathsf{ul}}\mathbf{L}^{\mathsf{T}}||_{\mathsf{F}}^2 + \frac{\omega_4}{2}\,||\mathbf{M}_{\mathsf{lf}} - \mathbf{L}\mathbf{F}^{\mathsf{T}}||_{\mathsf{F}}^2 + \frac{\omega_5}{2}\,||\mathbf{M}_{\mathsf{aa}} - \mathbf{A}\mathbf{D}_{\mathsf{aa}}\mathbf{A}^{\mathsf{T}}||_{\mathsf{F}}^2 +$$
$$\frac{\omega_6}{2}\left(||\mathbf{U}||_{\mathsf{F}}^2 + ||\mathbf{L}||_{\mathsf{F}}^2 + ||\mathbf{A}||_{\mathsf{F}}^2 + ||\mathbf{F}||_{\mathsf{F}}^2 + ||\mathbf{d}_{\mathsf{uu}}||_{\mathsf{F}}^2 + ||\mathbf{d}_{\mathsf{aa}}||_{\mathsf{F}}^2 + ||\mathbf{d}_{\mathsf{ul}}||_{\mathsf{F}}^2\right)$$

Exploratory data analysis / fast prototyping
Incomplete large-scale data

[Sorber, Van Barel, et al. 2015]

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

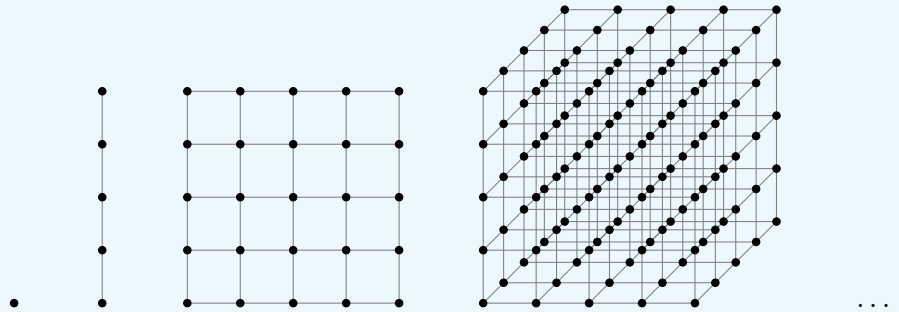
Completion

Harmonic retrieval

Tensorlab and Tensorlab+

## Tensors and scientific computing

Scientific computing: discretization of functions in many variables: "curse of dimensionality"



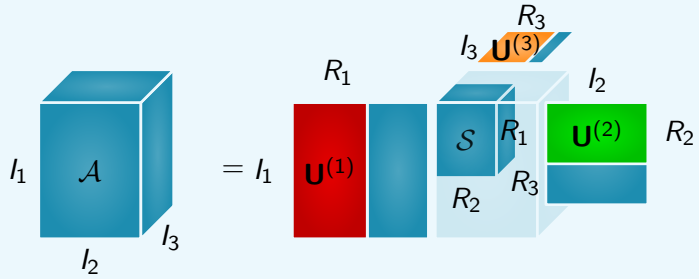

Exponential increase of entries: $I^N$

Computation: in terms of parameterized approximation

Parallel computation: cannot break the curse

# MLSVD / low ML rank approximation: numerical pros and cons

- Well-posed
- (Approximate) computation via matrix SVD
- Error bounds
- Curse not broken
- Numerically reliable but limited to modest order

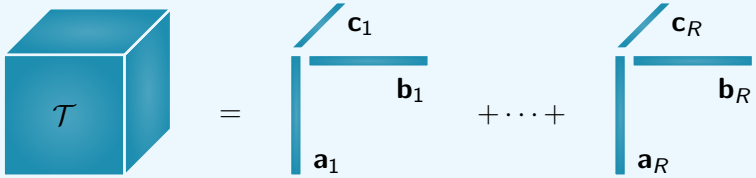



Order 3 $= I^3 \rightarrow \mathcal{O}\left(3IR + R^3\right)$

Order N $= I^N \rightarrow \mathcal{O}\left(IR + R^N\right)$

# CPD / low rank approximation: numerical pros and cons

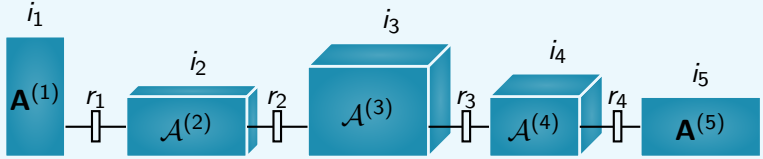
- Possibly ill-posed
- Possibly ill-conditioned
- Curse broken
- Powerful tool but not always numerically reliable




Order 3 $= I^3 \rightarrow \mathcal{O}(3IR)$
Order N $= I^N \rightarrow \mathcal{O}(NIR)$

# Tensor Trains or Matrix Product States

- Write tensor as a train of matrix -. . . third-order tensors . . . - matrix [Oseledets 2011]




- Mathematically, for a general $N$th order tensor $\mathcal{T} \in \mathbb{C}^{I_1 \times \cdots \times I_N}$

$$t_{i_1 i_2 \cdots i_N} = \sum_{r_1, r_2, \ldots, r_{N-1}} a^{(1)}_{i_1 r_1} a^{(2)}_{r_1 i_2 r_2} a^{(3)}_{r_2 i_3 r_3} \cdots a^{(N-1)}_{r_{N-2} i_{N-1} r_{N-1}} a^{(N)}_{r_{N-1} i_N}$$

- Total number of entries: $I^N \leftrightarrow$ number of variables: $\mathcal{O}\left(2IR + (N-2)IR^2\right)$
- Curse broken and can be computed via QR/SVD
- Truncation error bound

$$\|\mathcal{T} - \hat{\mathcal{T}}_{\mathrm{TT,\ trunc}}\|_{\mathrm{F}}^2 \leqslant (N-1) \min_{\mathrm{rank_{TT}}(\hat{\mathcal{T}}) \leqslant (R_1, R_2, \ldots, R_{N-1})} \|\mathcal{T} - \hat{\mathcal{T}}\|_{\mathrm{F}}^2$$

- hTucker [Hackbusch 2012]

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

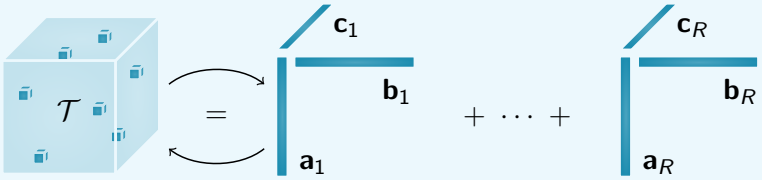
Large-scale data analysis and curse of dimensionality

Completion

Harmonic retrieval

Tensorlab and Tensorlab+

# Canonical polyadic decomposition




- Mathematically, for a general $N$th order tensor $\mathcal{T} \in \mathbb{C}^{I \times \cdots \times I}$

$$\mathcal{T} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \otimes \mathbf{a}_r^{(2)} \otimes \cdots \otimes \mathbf{a}_r^{(N)} = \left[\!\left[ \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \ldots, \mathbf{A}^{(N)} \right]\!\right]$$

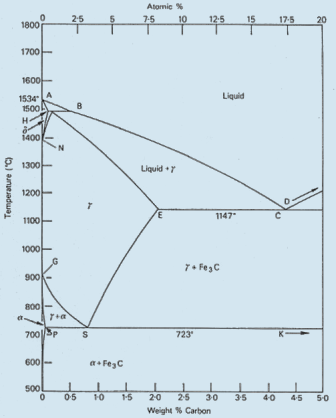
- Total number of entries: $I^N \leftrightarrow$ number of variables: $\mathcal{O}(NIR)$
- Curse broken, but possibly ill-conditioned/ill-posed problem

# Modeling multiway thermodynamic data

Modes: fraction atom/molecule $n$ in a multi-component material

Value: Gibbs free energy, chemical potential, melting temperature (e.g., computed using thermodynamic software)

## First-order example



Modes: fraction atom/molecule $n$ in a multi-component material
Value: Gibbs free energy, chemical potential, melting temperature (e.g., computed using thermodynamic software)

## Second-order example

- Alloy with $c_1$ % iron, $c_2$ % carbon and $100 - c_1 - c_2$ % nickel
- Discretize $c_1$ and $c_2$ in 100 steps
- Grid of size $100^2$

[Vervliet, Debals, Sorber, and De Lathauwer 2014]

## Multiway dataset

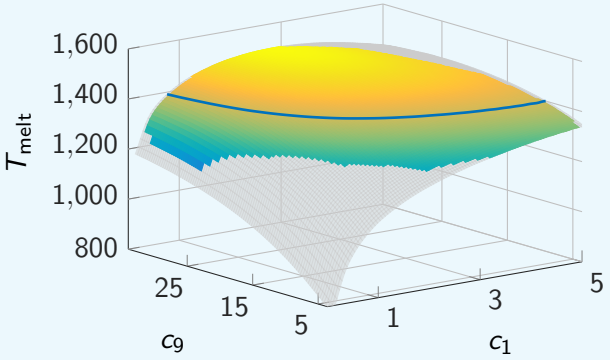
- # constituent materials: 10 (thus $N = 9$)
- Size: $100 \times 100 \times \cdots \times 100 \approx 10^{18}$ elements
- # Samples: 130 000 of which 30 000 are validation samples
- Undersampling rate $= 10^{13}$
- Model:

$$\mathcal{T} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \otimes \mathbf{a}_r^{(2)} \otimes \cdots \otimes \mathbf{a}_r^{(N)}$$

### Algorithm

- Tensorlab 3.0 with `cpd_nls` and `UseCPDI` option [Vervliet, Debals, Sorber, and De Lathauwer 2014; Vervliet, Debals, Sorber, Van Barel, et al. 2016]
- Initialization: optimally scaled best-out-of-five strategy

# Visualization



## Recap

- From a ninth order tensor with $10^{18}$ elements...
- we took 100 000 samples...
- to get a rank-5 model with $5 \times 9 \times 100 = 4\,500$ parameters...
- to get a continuous model with $\mathcal{O}(100)$ parameters...
- in 3 min

## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

Completion

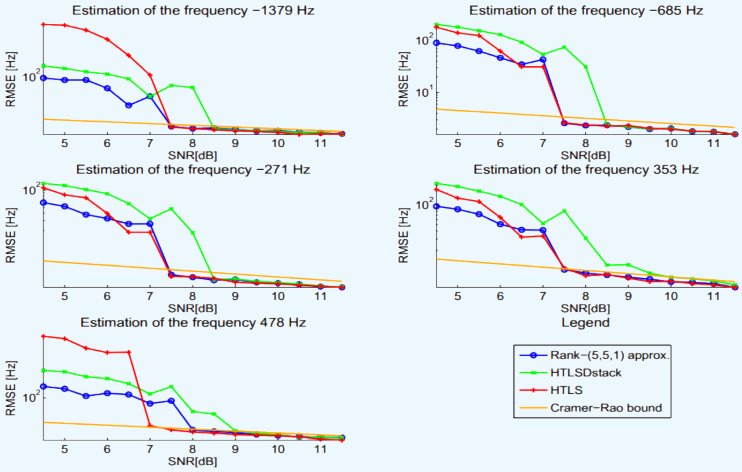
Harmonic retrieval

Tensorlab and Tensorlab+

# Low ML rank approximation: illustration (NMR)

Harmonic retrieval: Hankel matrix $\rightarrow$ tensor: signal subspace

Five-peak signal, decreasing $R_3$ from 5 to 1 [Papy, De Lathauwer, et al. 2009]



## Overview


Introduction

Dimensionality reduction - Principal Component Analysis

Denoising

Blind source separation - Latent Variable analysis

Blind similarity assessment

Polynomial equations

Regression and classification

Structured data fusion

Large-scale data analysis and curse of dimensionality

Completion

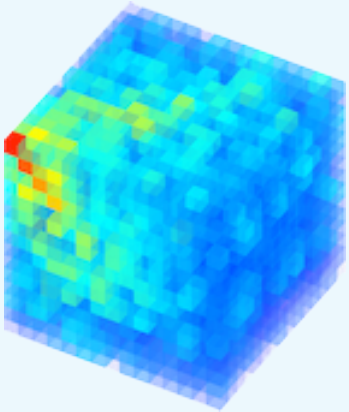
Harmonic retrieval

Tensorlab and Tensorlab+

# Tensorlab

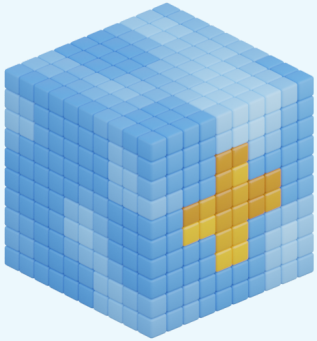
- Tensorlab 1: Tensor decompositions
- Complex Optimization Toolbox
  `www.esat.kuleuven.be/sista/cot`
- Tensorlab 2: Structured Data Fusion
- Tensorlab 3: Large-scale algorithms, tensorization
- Extensive documentation and demos

`www.tensorlab.net`



# Tensorlab+

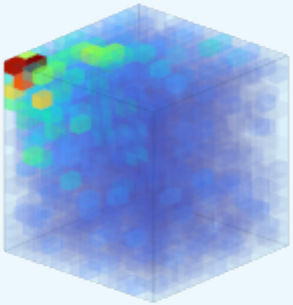
- code supporting 44 papers
- 200+ experiments
- reproducing 250+ figures and table
- 40 tutorials and demos



`www.tensorlab.net`
`www.tensorlabplus.net`

# pyTensorlab: a Python package for tensor computations and complex optimization

- Advanced algorithms for tensor decompositions
- New efficient tensor types built on top of NumPy
- Practical visualization tools
- Support for complex optimization
- Fully typed code for faster development
- Extensive documentation, detailed API, and demos



See `www.pytensorlab.net` or `pip install pytensorlab`.

# Getting started

N. Sidiropoulos, L. De Lathauwer, et al. [July 2017]. "Tensor Decomposition for Signal Processing and Machine Learning". In: *IEEE Transactions on Signal Processing* 65.13, pp. 3551–3582

A. Cichocki, D. P. Mandic, et al. [Mar. 2015]. "Tensor decompositions for signal processing applications: From two-way to multiway component analysis". In: *IEEE Signal Processing Magazine* 32.2, pp. 145–163

T. G. Kolda and B. W. Bader [Aug. 2009]. "Tensor decompositions and applications". In: *SIAM Review* 51.3, pp. 455–500

N. Vervliet and L. De Lathauwer [2019]. "Numerical optimization based algorithms for data fusion". In: *Data Fusion Methodology and Applications*. Ed. by M. Cocchi. Vol. 31. Elsevier. Chap. 4, pp. 81–128

# References I


- Acar, E. et al. (2007). "Multiway analysis of epilepsy tensors". In: *Bioinformatics* 23.13, pp. i10–i18.
- Carrol, J. D. and Chang, J. J. (Sept. 1970). "Analysis of individual differences in multidimensional scaling via an $n$-way generalization of "Eckart–Young" decomposition". In: *Psychometrika* 35.3, pp. 283–319.
- Cichocki, A. et al. (Mar. 2015). "Tensor decompositions for signal processing applications: From two-way to multiway component analysis". In: *IEEE Signal Processing Magazine* 32.2, pp. 145–163.
- De Lathauwer, L. (Sept. 2008). "Decompositions of a higher-order tensor in block terms — Part I: Lemmas for partitioned matrices". In: *SIAM Journal on Matrix Analysis and Applications* 30.3, pp. 1022–1032.
- — (Dec. 2011). "Blind separation of exponential polynomials and the decomposition of a tensor in rank-$(L_r, L_r, 1)$ terms". In: *SIAM Journal on Matrix Analysis and Applications* 32.4, pp. 1451–1474.

## References II


- De Lathauwer, L., De Moor, B., and Vandewalle, J. (Apr. 2000). "A multilinear singular value decomposition". In: *SIAM Journal on Matrix Analysis and Applications* 21.4, pp. 1253–1278.
- De Vos, M. et al. (Sept. 2007). "Canonical decomposition of ictal scalp EEG reliably detects the seizure onset zone". In: *NeuroImage* 37.3, pp. 844–854.
- Denton, E. et al. (2014). "Exploiting linear structure within convolutional networks for efficient evaluation". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*. NIPS'14. Montreal, Canada: MIT Press, pp. 1269–1277.
- Domanov, I. and De Lathauwer, L. (June 2020). "On uniqueness and computation of the decomposition of a tensor into multilinear rank-$(1, L_r, L_r)$ terms". In: *SIAM Journal on Matrix Analysis and Applications* 41.2, pp. 747–803.
- — (2021). "From Computation to Comparison of Tensor Decompositions". In: *SIAM Journal on Matrix Analysis and Applications* 42.2, pp. 449–474.

## References III


- Hackbusch, W. (2012). *Tensor Spaces and Numerical Tensor Calculus*. Vol. 42. New York, NY, USA: Springer Science & Business Media.
- Harshman, R. A. (1970). "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis". In: *UCLA Working Papers in Phonetics* 16, pp. 1–84.
- Hitchcock, F. (1927). "Multiple Invariants and Generalized Rank of a *p*-way Matrix or Tensor". In: *Journal of Mathematical Physics* 7, pp. 39–79.
- Hunyadi, B. et al. (Sept. 2014). "Block term decomposition for modelling epileptic seizures". In: *EURASIP Journal on Advances in Signal Processing* 2014.139. Special Issue on Recent Advances in Tensor-Based Signal and Image Processing, pp. 1–19.
- Kim, Y. et al. (2016). "Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications". In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Y. Bengio and Y. LeCun.

## References IV


- Kolda, T. G. and Bader, B. W. (Aug. 2009). "Tensor decompositions and applications". In: *SIAM Review* 51.3, pp. 455–500.
- Oseledets, I. (Sept. 2011). "Tensor-train decomposition". In: *SIAM Journal on Scientific Computing* 33.5, pp. 2295–2317.
- Papy, J.-M., De Lathauwer, L., and Van Huffel, S. (Feb. 2009). "Exponential data fitting using multilinear algebra: The decimative case". In: *Journal of Chemometrics* 23. (www.interscience.wiley.com), pp. 341–351.
- Shahriar, N. (Feb. 2023). *What is Convolutional Neural Network — CNN (Deep Learning)*. Available online at `https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5`.
- Sidiropoulos, N. et al. (July 2017). "Tensor Decomposition for Signal Processing and Machine Learning". In: *IEEE Transactions on Signal Processing* 65.13, pp. 3551–3582.
- Smilde, A. K., Bro, R., and Geladi, P. (2004). *Multi-way analysis with applications in the chemical sciences*. Wiley Chichester, UK.

# References V


- Sorber, L., Van Barel, M., and De Lathauwer, L. (2014). "Numerical Solution of Bivariate and Polyanalytic Polynomial Systems". In: *SIAM Journal on Numerical Analysis* 52.4, pp. 1551–1572.
- — (June 2015). "Structured data fusion". In: *IEEE Journal on Selected Topics in Signal Processing* 9.4, pp. 586–600.
- Sørensen, M. and De Lathauwer, L. (Apr. 2015). "Coupled canonical polyadic decompositions and (coupled) decompositions in multilinear rank-$(L_{r,n}, L_{r,n}, 1)$ terms — Part I: Uniqueness". In: *SIAM Journal on Matrix Analysis and Applications* 36.2, pp. 496–522.
- Sørensen, M., Domanov, I., and De Lathauwer, L. (Apr. 2015). "Coupled canonical polyadic decompositions and (coupled) decompositions in multilinear rank-$(L_{r,n}, L_{r,n}, 1)$ terms — Part II: Algorithms". In: *SIAM Journal on Matrix Analysis and Applications* 36.3, pp. 1015–1045.

## References VI


- Tucker, L. R. (1964). "The extension of factor analysis to three-dimensional matrices". In: *Contributions to mathematical psychology*. Ed. by H. Gulliksen and N. Frederiksen. New York: Holt, Rinehart & Winston, pp. 110–127.
- Van Eeghem, F. and De Lathauwer, L. (2020). "Tensor similarity in chemometrics". In: *Comprehensive Chemometrics: Chemical and Biochemical Data Analysis*. Ed. by S. Brown, R. Tauler, and e. Walczak B. Elsevier. Chap. 2.16, pp. 337–354.
- Van Eeghem, F., Debals, O., and De Lathauwer, L. (Mar. 2018). "Tensor similarity in two modes". In: *IEEE Transactions on Signal Processing* 66.5, pp. 1273–1285.
- Vervliet, N. and De Lathauwer, L. (2019). "Numerical optimization based algorithms for data fusion". In: *Data Fusion Methodology and Applications*. Ed. by M. Cocchi. Vol. 31. Elsevier. Chap. 4, pp. 81–128.
- Vervliet, N., Debals, O., Sorber, L., and De Lathauwer, L. (Sept. 2014). "Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis". In: *IEEE Signal Processing Magazine* 31.5, pp. 71–79.

# References VII




Vervliet, N., Debals, O., Sorber, L., Van Barel, M., et al. (Mar. 2016). *Tensorlab 3.0*. Available online at `https://www.tensorlab.net`.