Discovering tensors: their challenges and applications

Martina Iannacito

Tensor Day
Povo (TN), November 21, 2023

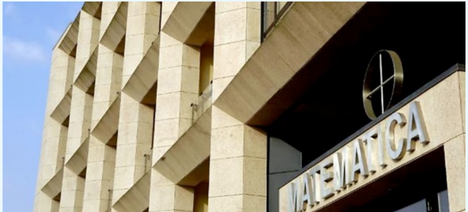
Overview

Bachelor degree
UniPR
2014-2017

Ph.D.
Inria Bordeaux
2019-2022

Master's degree
UniTN
2017-2019

Postdoc
KU Leuven
2023



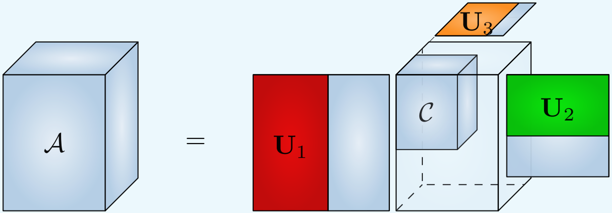
Overview

Tensor Decomposition for Big Data Analysis

Figure: Prof. A. Bernardi, UniTN

- introduction to algebraic geometry
- overview of classical tensor decomposition techniques
  - Canonical Polyadic decomposition;
  - Tucker;
  - Hierarchical Tucker;
  - Tensor-Train;
- overview of different applied problems solved with tensor-based methods.

Tucker's model [Tucker 1966; De Lathauwer, De Moor, et al. 2000]



If $\mathcal{A}$ is a $(N_1 \times N_2 \times N_3)$ tensor, its Tucker decomposition becomes

$$\mathcal{A} = \mathcal{C} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3$$

where

- $\mathcal{C}$ is a $(R_1 \times R_2 \times R_3)$ tensor;
- $\mathbf{U}_i$ is a $(N_i \times R_i)$ orthogonal matrix, called i-th factor matrix.

The memory requirement is $\mathcal{O}(R^d + NR)$ where $R = \max R_i$, $N = \max N_i$ and d is the tensor order.

Ecology project

Estimate biodiversity

- from satellite images
- using a moving window
- applying information theory results

Figure: Prof. D. Rocchini, UniBO

Master's thesis project

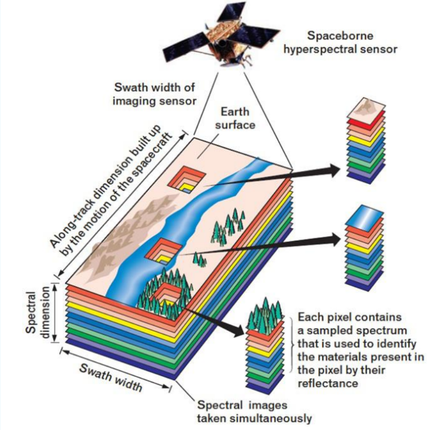


Figure: from [Bedini 2017].

Over a time series of spectral images of Europe,

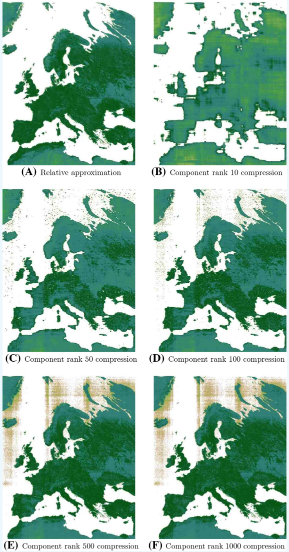
- get two images from two spectral bands (RED and NIR);
- compute the normalized difference vegetation index per pixel, i.e.,

$$\mathtt{NDVI}(i,j) = \frac{\mathtt{NIR}(i,j) - \mathtt{RED}(i,j)}{\mathtt{NIR}(i,j) + \mathtt{RED}(i,j)}$$

- compute a biodiversity index over the resulting NDVI image

What happens if the NDVI image is computed from the NIR and RED spectral images stored in a tensor and compressed?

Rényi index result [Bernardi, Iannacito, et al. 2019]

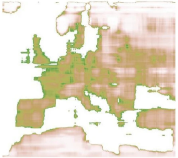


Compression at multilinear rank $(i, i, 3)$ with $i \in \{10, 50, 100, 500, 1000\}$

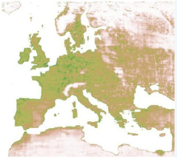
- memory used ranges between 0.19% and 22%;
- average error per pixel ranges between 13% and 5%.

Rao index result [Bernardi, Iannacito, et al. 2019]

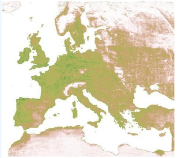
(A) Relative approximation



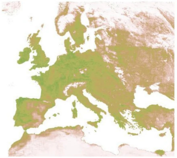
(B) Component rank 10 compression



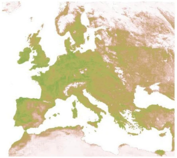
(C) Component rank 50 compression



(D) Component rank 100 compression



(E) Component rank 500 compression



(F) Component rank 1000 compression

Compression at multilinear rank $(i, i, 3)$ with $i \in \{10, 50, 100, 500, 1000\}$

- memory used ranges between 0.19% and 22%;
- average error per pixel ranges between 63% and 19%.

Overview

Ph.D. project

Supervisors

Figure: Prof. O. Coulaud, Inria Bordeaux

Figure: Prof. L. Giraud, Inria Bordeaux

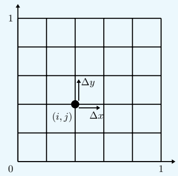
- tensor methods
- high-dimensional simulations

- numerical linear algebra
- finite precision arithmetic

Context

The problem

$$\begin{cases} \mathcal{L}(u) & = f \quad \text{in} \quad \Omega \\ u & = f_0 \quad \text{in} \quad \partial\Omega \end{cases} \quad \text{for} \quad \Omega \subseteq \mathbb{R}^{n_1 \times \cdots \times n_d}.$$



$$\mathcal{A}\mathcal{X} = \mathcal{B}$$

where $\mathcal{A} : \mathbb{R}^{n_1 \times \cdots \times n_d} \to \mathbb{R}^{n_1 \times \cdots \times n_d}$ is a multilinear operator and $\mathcal{B} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ a tensor.

For large scale-simulations we have to take into account

- computational model
- numerical method
- memory costs $\mathcal{O}(N^d)$

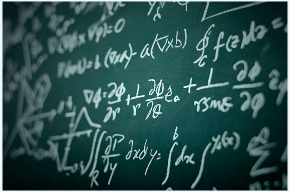
Maths vs computer science

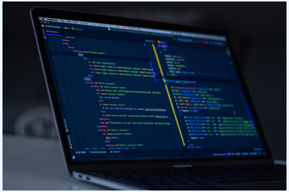
Mathematical world

- $\pi = 3.141592653589793238462643 3\ldots$

Computer world

>>> $\overline{\pi}$ = 3.141592653589793





Maths vs computer science

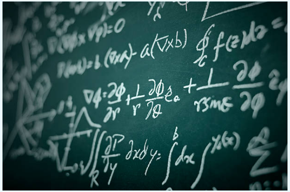
Mathematical world

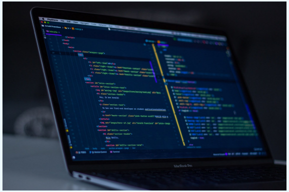
- $\pi = 3.14159265358979323846264 33\ldots$
- $x = 0.1$ and $y = 0.2$, then $x + y = 0.3$

Computer world

>>> $\overline{\pi}$ = 3.141592653589793

>>> $\overline{\text{x}}$ = 0.1 and $\overline{\text{y}}$ = 0.2, then $\overline{\text{x+y}}$ = 0.30000000000000004





Computational model

Denoting by u the **unit roundoff** of the working precision

Standard IEEE model [Higham 2002]

$$fl(x) = x(1+\xi) \qquad \text{[storage perturbation]}$$

$$fl(x \operatorname{op} y) = (x \operatorname{op} y)(1+\varepsilon) \qquad \text{[computational perturbation]}$$

with $|\xi| \le u$, $|\varepsilon| \le u$ and $\operatorname{op} \in \{+, -, \times, \div\}$.

Computational model

Denoting by u the **unit roundoff** of the working precision

Standard IEEE model [Higham 2002]

$$fl(x) = x(1+\xi) \qquad \text{[storage perturbation]}$$
$$fl(x \operatorname{op} y) = (x \operatorname{op} y)(1+\varepsilon) \qquad \text{[computational perturbation]}$$

with $|\xi| \leq u$, $|\varepsilon| \leq u$ and $\operatorname{op} \in \{+, -, \times, \div\}$.

Example

Assuming to work in floating point 64, with $u_{64} = 10^{-16}$

- $\overline{\pi} = 3.141592653589793 = \pi(1+\xi)$ with $|\xi| \leq u_{64}$
- $\overline{\mathtt{x}} = \mathtt{0.1}$ and $\overline{\mathtt{y}} = \mathtt{0.2}$, then

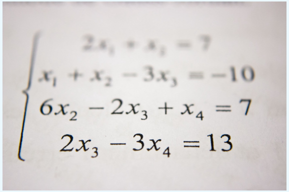
  $$\overline{\mathtt{x + y}} = \mathtt{0.30000000000000004} = (0.2 + 0.1)(1+\varepsilon)$$

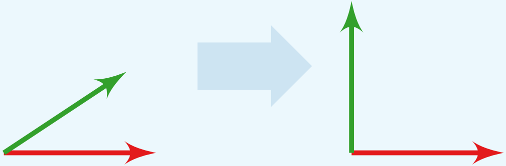
  with $|\varepsilon| \leq u_{64}$

Numerical linear algebra methods

Iterative solver

- Generalized Minimal RESidual (GMRES)



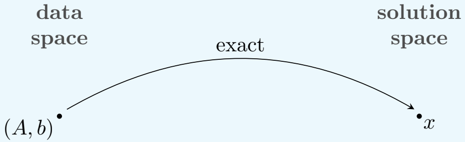


Orthogonalization kernels

- Classical and Modified Gram-Schmidt (CGS, MGS)
- Gram approach
- Householder transformation

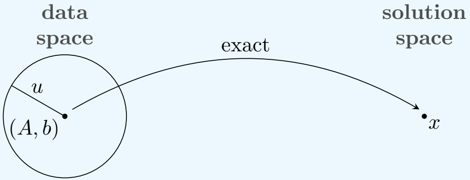
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



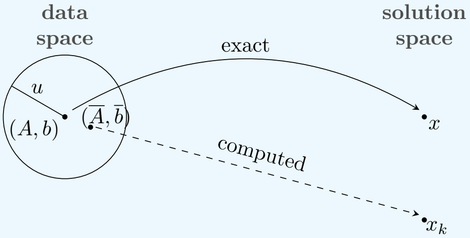
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



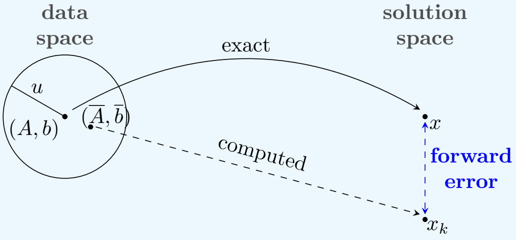
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



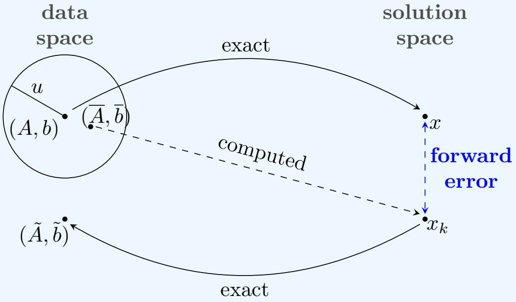
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



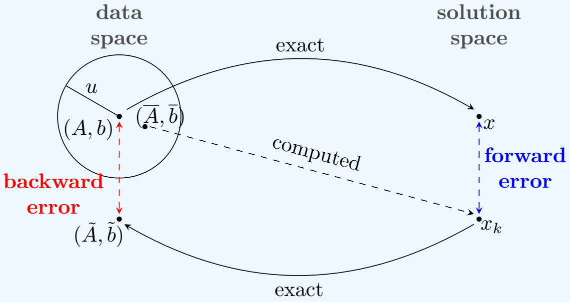
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



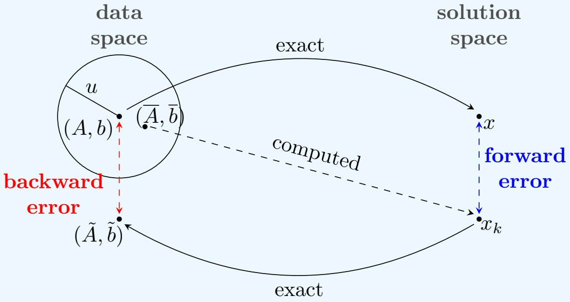
GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



GMRES property [Wilkinson 1963]

Given the linear system $\mathbf{Ax} = \mathbf{b}$ and a working precision u, then



GMRES is backward stable, i.e.,

$$\eta_{A,b}(x_k) = \frac{||\mathbf{A}\mathbf{x}_k - \mathbf{b}||}{||\mathbf{A}||||\mathbf{x}_k|| + ||\mathbf{b}||} \sim \mathcal{O}(u)$$

Orthogonalization schemes

Let $\mathbf{Q}_k = [\mathbf{q}_1, \ldots, \mathbf{q}_k]$ be the orthogonal basis produced by an orthogonalization kernel, then the **Loss Of Orthogonality** is

$$||\mathbb{I}_k - \mathbf{Q}_k^\top \mathbf{Q}_k||.$$

It measures the quality in terms of orthogonality of the computed basis. It is linked with the linearly dependency of the input vectors $\mathbf{A}_k = [\mathbf{a}_1, \ldots, \mathbf{a}_k]$, estimated through $\kappa(\mathbf{A}_k)$.

Matrix		
Source	*Algorithm*	$\left\|\mathbb{I}_k - \mathbf{Q}_k^\top \mathbf{Q}_k\right\|$
[Stathopoulos and Wu 2002]	**Gram**	$\mathcal{O}(u\kappa^2(\mathbf{A}_k))$
[L. Giraud, Langou, et al. 2005]	**CGS**	$\mathcal{O}(u\kappa^2(\mathbf{A}_k))$
[Björck 1967]	**MGS**	$\mathcal{O}(u\kappa(\mathbf{A}_k))$
[L. Giraud, Langou, et al. 2005]	**CGS2**	$\mathcal{O}(u)$
[L. Giraud, Langou, et al. 2005]	**MGS2**	$\mathcal{O}(u)$
[Wilkinson 1965]	**Householder**	$\mathcal{O}(u)$

New tensor framework

The GMRES and the kernel properties depends on u the computational precision.

What the tensor framework, when objects are compressed through a tensor techniques?

Assumptions

- use TT-formalism, so that storage cost is linear in d
- compress objects at precision δ
- perform operation with computational precision u

new computational framework

$$fl_\delta(\mathcal{X}\,\mathrm{op}\,\mathcal{Y}) = \delta\text{-storage}\big(fl(\mathcal{X}\,\mathrm{op}\,\mathcal{Y})\big)$$

$$\delta\text{-storage}(\mathcal{Z}) = \overline{\mathcal{Z}} \qquad \text{s.t.} \qquad \frac{||\mathcal{Z} - \overline{\mathcal{Z}}||}{||\mathcal{Z}||} \leq \delta$$

with fl is the classical floating point computational function dependent on u.

Tensor-train model [Oseledets 2011]

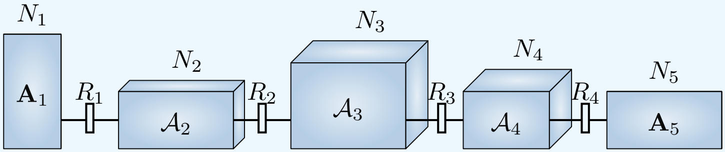


Figure: Tensor-Train of $\mathcal{A}$ tensor of order 5.

Let $\mathcal{A}$ a tensor of order d and dimensions $(N_1 \times \cdots \times N_d)$, then its TT-representation is given by d TT-cores s.t.

- $\mathbf{A}_1$ a (N_1, R_1) matrix
- $\mathcal{A}_i$ is a $(R_{i-1} \times N_i \times R_i)$ tensor
- $\mathbf{A}_d$ is a $(R_{d-1} \times N_d)$ matrix

The $(i_1, \ldots, i_d)$ element of $\mathcal{A}$ is

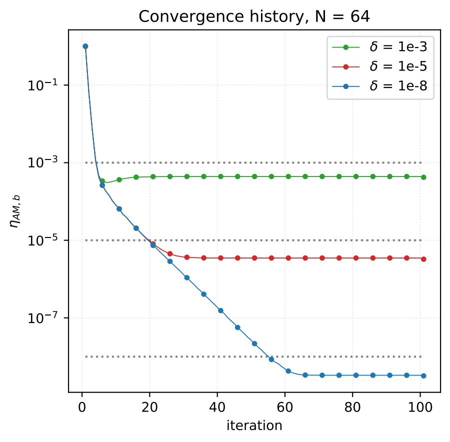
$$\mathcal{A}(i_1, \ldots, i_d) = \sum_{i=1}^{d} \sum_{r_i=1}^{R_i} \mathbf{A}(i, r_1)\mathcal{A}_1(i_1, r_2, i_2) \cdots \mathbf{A}_d(i_{d-1}, i_d).$$

The memory cost is $\mathcal{O}(dR^2N)$ where $R = \max R_i$, $N = \max N_i$ and d is the tensor order.$\mathcal{A}$

TT-GMRES results [Dolgov 2013; Coulaud, Luc Giraud, et al. 2022a]

Convection-Diffusion problem

$$\begin{cases} -\Delta \mathcal{U} & +\mathcal{V} \cdot \nabla \mathcal{U} = 0 \\ \mathcal{U}_{\{y=1\}} & = 1 \end{cases} \quad \text{in} \quad \Omega = [-1, 1]^3$$

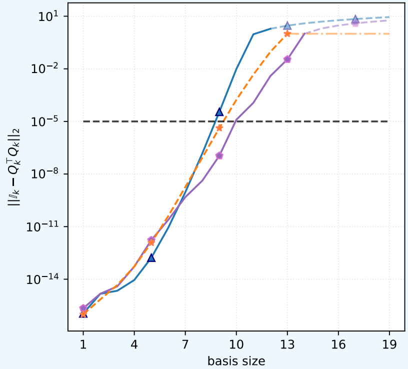


TT-orthogonalization [Coulaud, Luc Giraud, et al. 2022b]

$$\mathcal{X}_{k+1} = \texttt{TT-rounding}(\mathbf{\Delta}_d \mathcal{A}_k, \texttt{max\_rank} = 1) \quad \text{with} \quad \mathcal{A}_{k+1} = \frac{1}{||\mathcal{X}_{k+1}||} \mathcal{X}_{k+1}$$

TT-orthogonalization [Coulaud, Luc Giraud, et al. 2022b]

$$\mathcal{X}_{k+1} = \texttt{TT-rounding}(\mathbf{\Delta}_d \mathcal{A}_k, \texttt{max\_rank} = 1) \quad \text{with} \quad \mathcal{A}_{k+1} = \frac{1}{||\mathcal{X}_{k+1}||}\mathcal{X}_{k+1}$$



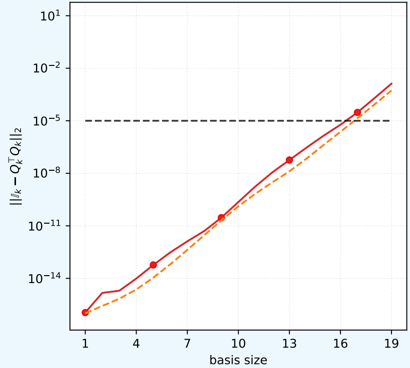
- **Gram approach**
- **CGS**
- $\kappa^2(\mathbf{A}_k)$

$\mathcal{O}(\delta \kappa^2(\mathbf{A}_k))$

Figure: Loss of orthogonality for $m = 20$ TT-vectors of order $d = 3$ and mode size $n = 15$, rounding precision $\delta = 10^{-5}$

TT-orthogonalization [Coulaud, Luc Giraud, et al. 2022b]

$$\mathcal{X}_{k+1} = \texttt{TT-rounding}(\boldsymbol{\Delta}_d \mathcal{A}_k, \texttt{max\_rank} = 1) \quad \text{with} \quad \mathcal{A}_{k+1} = \frac{1}{||\mathcal{X}_{k+1}||}\mathcal{X}_{k+1}$$



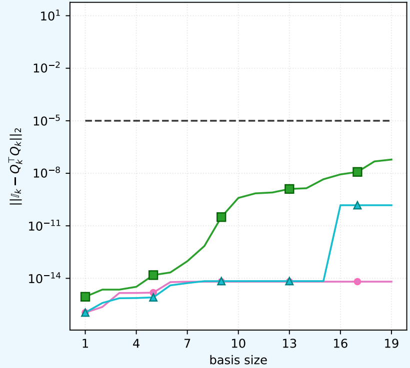
- **MGS**
- $\kappa(\mathbf{A}_k)$

$\mathcal{O}(\delta\kappa(\mathbf{A}_k))$

Figure: Loss of orthogonality for $m = 20$ TT-vectors of order $d = 3$ and mode size $n = 15$, rounding precision $\delta = 10^{-5}$

TT-orthogonalization [Coulaud, Luc Giraud, et al. 2022b]

$$\mathcal{X}_{k+1} = \texttt{TT-rounding}(\mathbf{\Delta}_d \mathcal{A}_k, \texttt{max\_rank} = 1) \quad \text{with} \quad \mathcal{A}_{k+1} = \frac{1}{||\mathcal{X}_{k+1}||}\mathcal{X}_{k+1}$$



- CGS2
- MGS2
- Householder transformation

$\mathcal{O}(\delta)$

Figure: Loss of orthogonality for $m = 20$ TT-vectors of order $d = 3$ and mode size $n = 15$, rounding precision $\delta = 10^{-5}$

TT-orthogonalization [Coulaud, Luc Giraud, et al. 2022b]

$$\mathcal{X}_{k+1} = \texttt{TT-rounding}(\mathbf{\Delta}_d \mathcal{A}_k, \texttt{max\_rank} = 1) \quad \text{with} \quad \mathcal{A}_{k+1} = \frac{1}{||\mathcal{X}_{k+1}||}\mathcal{X}_{k+1}$$

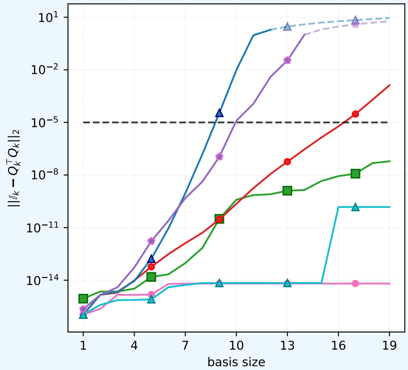


Figure: Loss of orthogonality for $m = 20$ TT-vectors of order $d = 3$ and mode size $n = 15$, rounding precision $\delta = 10^{-5}$

- **Gram approach** $\mathcal{O}(\delta\kappa^2(\mathbf{A}_k))$
- **CGS** $\mathcal{O}(\delta\kappa^2(\mathbf{A}_k))$
- **MGS** $\mathcal{O}(\delta\kappa(\mathbf{A}_k))$
- **CGS2** $\mathcal{O}(\delta)$
- **MGS2** $\mathcal{O}(\delta)$
- **Householder transformation** $\mathcal{O}(\delta)$

Overview

- Timeline
- Master's thesis
  - Tensor Decomposition for Big Data Analysis
  - Tucker model
  - Biodiversity estimate
- Doctoral thesis
  - Numerical linear algebra and data analysis in tensor format
  - Tensor-train model
- Postdoctoral project
  - Canonical Polyadic decomposition
  - Classical algorithms and new challanges
- Conclusion

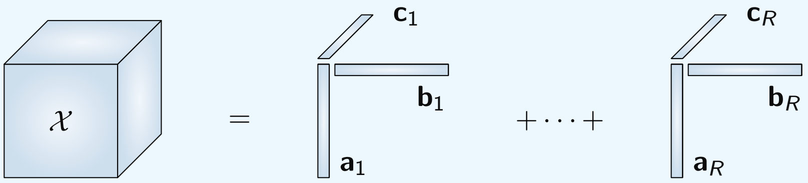
Postdoctoral project

New algorithm for Canonical Polyadic Decomposition

- formalize previous results from I. Domanov;
- improve the algorithm efficiency;
- evaluate its quality;
- test in signal processing cases.

Figure: Prof. L. De Lathauwer, KU Leuven

Canonical Polyadic Decomposition [Hitchcock 1927; Harshman 1970; Carroll and Chang 1970]



If $\mathcal{A}$ is a $(N_1 \times N_2 \times N_3)$ tensor of rank R, its CPD decomposition is

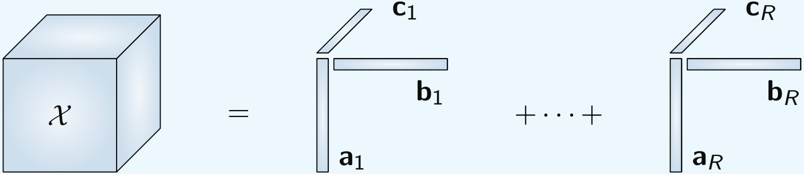
$$\mathcal{A} = \sum_{r=1}^{R} \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r$$

where $\mathbf{a}_r \in \mathbb{K}^{N_1}$, $\mathbf{b}_r \in \mathbb{K}^{N_2}$ and $\mathbf{c}_r \in \mathbb{K}^{N_3}$ with $i = 1, \ldots, R$. Its properties are

- unique under mild assumption
- memory cost $\mathcal{O}(dNR)$
- NP-hard problem
- algorithms affected by numerical instabilities

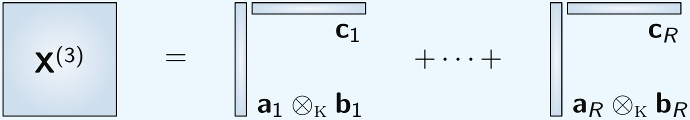
Problem reformulation

if $\mathcal{X} = \mathbf{a}_1 \otimes \mathbf{b}_1 \otimes \mathbf{c}_1 + \ldots + \mathbf{a}_R \otimes \mathbf{b}_R \otimes \mathbf{c}_R$



---

then $\mathbf{X}^{(3)} = (\mathbf{a}_1 \otimes_{\mathrm{K}} \mathbf{b}_1) \otimes \mathbf{c}_1^T + \ldots + (\mathbf{a}_R \otimes_{\mathrm{K}} \mathbf{b}_R) \otimes \mathbf{c}_R^T$



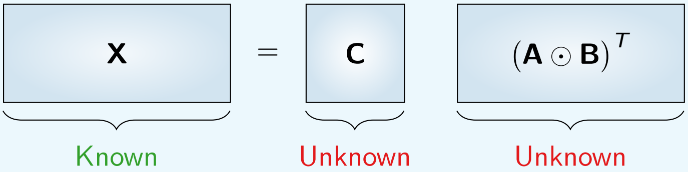
$$(\mathbf{a}_r \otimes_{\mathrm{K}} \mathbf{b}_r) \in \mathcal{V} = \left\{ \mathrm{vec}(\mathbf{Z}) : \begin{vmatrix} z_{i_1 j_1} & z_{i_1 j_2} \\ z_{i_2 j_1} & z_{i_2 j_2} \end{vmatrix} = 0 \right\} \text{ algebraic variety}$$

Algebraic algorithm: high view

Let $\mathcal{X}$ be a $(N_1 \times N_2 \times R)$ tensor, then

$$\mathbf{X}^{(3)} = \sum_{r=1}^{R} (\mathbf{a}_r \otimes_{\mathrm{K}} \mathbf{b}_r) \otimes \mathbf{c}_r^T = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T.$$

If $\mathbf{X} = \left(\mathbf{X}^{(3)}\right)^T$, then



1. compute $\mathbf{C}^{-1}$ from $\mathbf{X}$ using algebraic geometry properties;
2. compute $(\mathbf{A} \odot \mathbf{B})$ as the transposed product of $\mathbf{C}^{-1}\mathbf{X}$;
3. factorize $(\mathbf{A} \odot \mathbf{B}) = [\mathbf{a}_1 \otimes_{\mathrm{K}} \mathbf{b}_1, \ldots, \mathbf{a}_R \otimes_{\mathrm{K}} \mathbf{b}_R]$ to recover $\mathbf{A}$ and $\mathbf{B}$;
4. compute $\mathbf{C}$ by solving $(\mathbf{A} \odot \mathbf{B})\mathbf{C} = \mathbf{X}$.

Using algebraic geometry I

$\mathbf{e}$ is a column of $\mathbf{C}^{-1}$ if and only if $\mathbf{X}^T\mathbf{e}$ is equal to a column of $(\mathbf{A} \odot \mathbf{B})$

⇩

$$\mathbf{X}^T\mathbf{e} = (\mathbf{x}_1^T\mathbf{e}, \ldots, \mathbf{x}_N^T\mathbf{e}) = (z_1, \ldots, z_N) \in \mathcal{V}$$

⇕

$$P_k(\mathbf{x}_1^T\mathbf{e}, \ldots, \mathbf{x}_N^T\mathbf{e}) = 0 \text{ for } k = 1, \ldots, K$$

⇕

$$P_k^{\otimes}(\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T)(\mathbf{e} \otimes \cdots \otimes \mathbf{e}) = 0 \text{ for } k = 1, \ldots, K$$

where $P_k^{\otimes}(\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T)$ is the vector obtained by formal substitution of $(z_1, \ldots, z_N)$ by $\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T$ and the scalar multiplication by the tensor product.

Using algebraic geometry II

$\mathbf{e}$ is a column of $\mathbf{C}^{-1}$ if and only if $\mathbf{X}^T\mathbf{e}$ is equal to a column of $\mathbf{A} \odot \mathbf{B}$

⇓

$$P_k^{\otimes}(\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T)(\mathbf{e} \otimes \cdots \otimes \mathbf{e}) = 0 \text{ for } k = 1, \ldots, K$$

⇕

$$\mathbf{Q}\text{vec}(\mathbf{e}^{\otimes d}) = \begin{bmatrix} P_1^{\otimes}(\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T) \\ \vdots \\ P_K^{\otimes}(\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T) \end{bmatrix} \text{vec}(\mathbf{e} \otimes \cdots \otimes \mathbf{e}) = 0$$

⇕

The columns of $\mathbf{C}^{-1}$ belong to the intersection of $\mathbf{Q}$ kernel and $\text{vec}(\text{Sym}_R^N)$ the subspace of vectorized order N symmetric tensors, i.e.,

$$\mathbf{e} \in \text{null}(\mathbf{Q}) \cap \text{vec}(\text{Sym}_R^d).$$

Algebraic algorithm outline

$$\underbrace{\mathbf{X}}_{\text{Known}} = \underbrace{\mathbf{C}}_{\text{Unknown}} \; \underbrace{(\mathbf{A} \odot \mathbf{B})^T}_{\text{Unknown}}$$

1. compute the factor matrix $\mathbf{C}^{-1}$ from $\mathbf{X}$;
   1.1 compute $\mathbf{Q}$;
   1.2 compute the space $\mathcal{E}_0 = \mathrm{null}(\mathbf{Q}) \cap \mathrm{vec}(\mathrm{Sym}_R^d)$
      1.2.1 if $\dim \mathcal{E}_0 = R$, then compute $\mathbf{C}^{-1}$ by a CPD of $\{\mathbf{e}_1^{\otimes d}, \dots, \mathbf{e}_R^{\otimes d}\}$ basis of $\mathcal{E}_0$;
      1.2.2 if $\dim \mathcal{E}_0 > R$, then compute $\mathcal{E}_{h+1}$ such that

$$\mathcal{E}_{h+1} = (\mathbb{K} \otimes \mathcal{E}_h) \cap \mathrm{vec}(\mathrm{Sym}_R^{d+h})$$

      until $\dim \mathcal{E}_{h+1} = R^{h+1}$ and go to step 1.2.1;
2. compute $(\mathbf{A} \odot \mathbf{B})$ as $\mathbf{C}^{-1}\mathbf{X}$ transposed;
3. factorize each column of $(\mathbf{A} \odot \mathbf{B})$ at rank-1 to retrieve $\mathbf{A}$ and $\mathbf{B}$ by SVD;
4. compute $\mathbf{C}$ solving $(\mathbf{A} \odot \mathbf{B})^T \mathbf{C} = \mathbf{X}$.

Challenges

- efficiently construct $\mathbf{Q}$ and its kernel
- estimate the dimension of the intersection with Sym_R^{d+h}
- efficiently construct a basis for E_h
- compute the CPD of $\{\mathbf{e}_1^{\otimes(h+d)}, \ldots, \mathbf{e}_d^{\otimes(h+d)}\}$
- estimate the quality of the algorithm and its robustness

Overview

Wrap up

Tensor methods used in

- data analysis problem as compression methods
  - by the Tucker's decomposition
- scientific computing as new policy for computational methods
  - by the Tensor-Train decomposition
- signal processing
  - by the Canonical Polyadic Decomposition

Thank you for the attention!
Questions? Advice?

References I

- Bedini, Enton (2017). "The use of hyperspectral remote sensing for mineral exploration: a review". In: *Journal of Hyperspectral Remote Sensing*.
- Bernardi, Alessandra, Martina Iannacito, and Duccio Rocchini (2019). "High order singular value decomposition for plant diversity estimation". In: *Bollettino dell'Unione Matematica Italiana* 14, pp. 557–591.
- Björck, Åke (Mar. 1967). "Solving linear least squares problems by Gram-Schmidt orthogonalization". In: *BIT Numerical Mathematics* 7.1, pp. 1–21.
- Carroll, J. Douglas and Jih-Jie Chang (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.
- Coulaud, Olivier, Luc Giraud, and Martina Iannacito (2022a). *A note on GMRES in TT-format*. Research Report RR-9384. Inria Bordeaux Sud-Ouest.

References II

Coulaud, Olivier, Luc Giraud, and Martina Iannacito (Nov. 2022b). *On some orthogonalization schemes in Tensor Train format*. Tech. rep. RR-9491. Inria Bordeaux - Sud-Ouest.

De Lathauwer, Lieven, Bart De Moor, and Joos Vandewalle (2000). "A Multilinear Singular Value Decomposition". In: *SIAM Journal on Matrix Analysis and Applications* 21.4, pp. 1253–1278.

Dolgov, S. V. (2013). "TT-GMRES: solution to a linear system in the structured tensor format". In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 28.2, pp. 149–172.

Giraud, L., J. Langou, and M. Rozloznik (2005). "The loss of orthogonality in the Gram-Schmidt orthogonalization process". In: *Computers & Mathematics with Applications* 50.7. Numerical Methods and Computational Mechanics, pp. 1069–1075.

References III

Harshman, Richard A. (1970). "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-model factor analysis". In: vol. 16. UCLA Working Papers in Phonetics. University Microfilms, Ann Arbor, Michigan, pp. 1–84.

Higham, N.J. (2002). *Accuracy and Stability of Numerical Algorithms: Second Edition*. Other Titles in Applied Mathematics. Society for Industrial and Applied Mathematics.

Hitchcock, Frank L. (1927). "The Expression of a Tensor or a Polyadic as a Sum of Products". In: *Journal of Mathematics and Physics* 6.1-4, pp. 164–189.

Oseledets, I. V. (2011). "Tensor-Train Decomposition". In: *SIAM Journal on Scientific Computing* 33.5, pp. 2295–2317.

Stathopoulos, Andreas and Kesheng Wu (2002). "A Block Orthogonalization Procedure with Constant Synchronization Requirements". In: *SIAM Journal on Scientific Computing* 23.6, pp. 2165–2182.

References IV

Tucker, Ledyard R (1966). "Some mathematical notes on three-modes factor analysis". In: *Psychometrika* 31.3, pp. 279–311.

Wilkinson, J. H. (1963). *Rounding Errors in Algebraic Processes*. Vol. 32. Notes on Applied Science. Also published by Prentice-Hall, Englewood Cliffs, NJ, USA, 1964, translated into Polish as *Bledy Zaokraglen̈ w Procesach Algebraicznych* by PWW, Warsaw, Poland, 1967 and translated into German as *Rundungsfehler* by Springer-Verlag, Berlin, Germany, 1969. Reprinted by Dover Publications, New York, 1994. London, UK: HMSO, pp. vi + 161.

– (1965). *The algebraic eigenvalue problem*. en. Numerical Mathematics and Scientific Computation. Oxford, England: Clarendon Press.