Is Nonnegative Tucker Decomposition the new NMF?

Jeremy E. Cohen, CREATIS, CNRS, France

TRICAP 2022, Buoux
06/29/2022

Credits

cnrs advancing the frontiers

CREATIS

Inria inventeurs du monde numérique

Roadmap

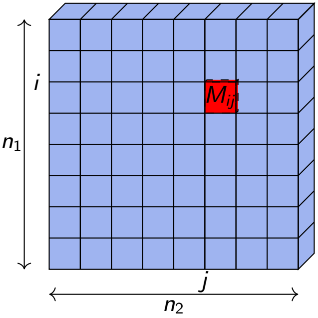
- Nonnegative Tucker 101
- An illustration of NTD to Music Information Retrieval
- Numerical optimization methods for NTD
- Some theory on NTD and open questions
- Off topic: Tensorly

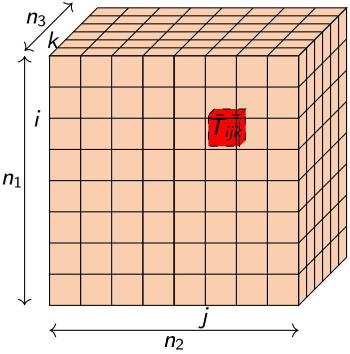
Matrices/Tensors as multiway arrays

Let $\mathcal{T}$ a tensor in $\mathbb{R}^{n_1 \times n_2 \times \ldots \times n_d}$

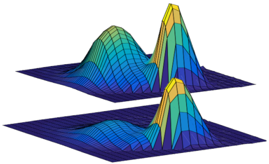
modes: indexes of the tensor from 1 to d. e.g. i is the first mode index.

order: d. e.g. the tensor below is a third order tensor.

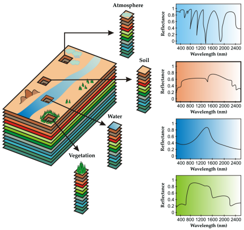




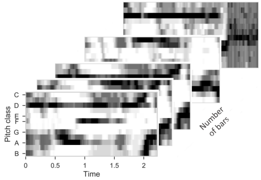
Examples of tensors in data science



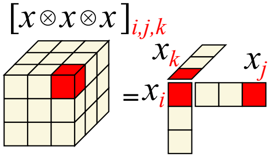
Tensor as Raw Data
Excitation Emission
Matrices



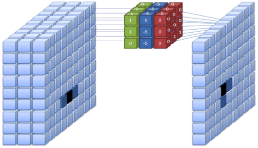
Tensor as Raw Data
Hyperspectral Images
[courtesy of J Chanussot]



Tensor as Processed Data
Tensor spectrogram

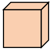


Tensor as Data Properties
Data Moments

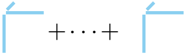


Tensor as Model Parameters
Convolutional Neural Networks
[figure from commons.wikimedia.org]

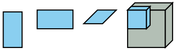
Number of parameters:



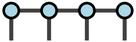
$\mathcal{O}(n^d)$



$\mathcal{O}(dnr)$



$\mathcal{O}(dnr + r^d)$



$\mathcal{O}(dnr^2)$

Consequently, tensor models can be used for:

Inverse Problems

- Matrix-Tensor completion
- Blind Source separation
- Denoising, deconvolution
- Phase retrieval
- ...

Compression, Low Complexity Model

- Big Data
- Data mining
- Neural Networks
- Partial Differential Equations
- ...

Number of parameters:

$\mathcal{O}(n^d)$ $\mathcal{O}(dnr)$ $\mathcal{O}(dnr + r^d)$ $\mathcal{O}(dnr^2)$

Consequently, tensor models can be used for:

Inverse Problems

- Matrix-Tensor completion
- Blind Source separation
- Denoising, deconvolution
- Phase retrieval
- ...

Compression, Low Complexity Model

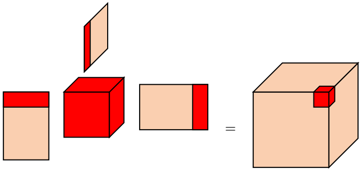
- Big Data
- Data mining
- Neural Networks
- Partial Differential Equations
- ...

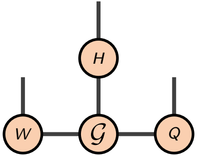
The Tucker format (3d order)

Input: Data tensor $\mathcal{T}$, core dimensions r_1, r_2, r_3
Parameters: $W \in \mathbb{R}^{n_1 \times r_1}$, $H \in \mathbb{R}^{n_2 \times r_2}$, $Q \in \mathbb{R}^{n_3 \times r_3}$ and $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$

$$\mathcal{T}_{ijk} = \sum_{q_1}^{r_1} \sum_{q_2}^{r_2} \sum_{q_3}^{r_3} W_{ir_1} H_{jr_2} Q_{kr_3} G_{r_1 r_2 r_3}$$

$$\mathcal{T} = (W \otimes H \otimes Q)\,\mathcal{G}$$



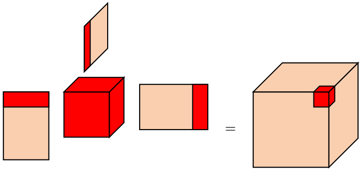


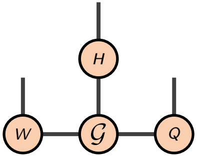
The Tucker format (3d order)

Input: Data tensor $\mathcal{T}$, core dimensions r_1, r_2, r_3
Parameters: $W \in \mathbb{R}^{n_1 \times r_1}$, $H \in \mathbb{R}^{n_2 \times r_2}$, $Q \in \mathbb{R}^{n_3 \times r_3}$ and $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$

$$\mathcal{T}_{ijk} = \sum_{q_1}^{r_1} \sum_{q_2}^{r_2} \sum_{q_3}^{r_3} W_{ir_1} H_{jr_2} Q_{kr_3} G_{r_1 r_2 r_3}$$

$$\mathcal{T} = (WP_1 \otimes HP_2 \otimes QP_3) \left[\left(P_1^{-1} \otimes P_2^{-1} \otimes P_3^{-1} \right) \mathcal{G} \right]$$





What is Tucker Decomposition

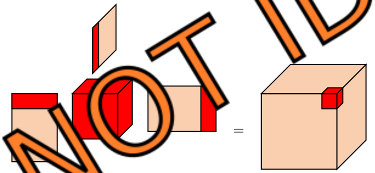
The Tucker format (3d order)

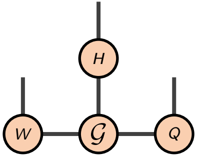
Input: Data tensor $\mathcal{T}$, core dimensions r_1, r_2, r_3
Parameters: $W \in \mathbb{R}^{n_1 \times r_1}$, $H \in \mathbb{R}^{n_2 \times r_2}$, $Q \in \mathbb{R}^{n_3 \times r_3}$ and $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$

$$\mathcal{T}_{ijk} = \sum_{q_1}^{r_1} \sum_{q_2}^{r_2} \sum_{q_3}^{r_3} W_{ir_1} H_{jr_2} Q_{kr_3} G_{r_1 r_2 r_3}$$

$$\mathcal{T} = (WP_1 \otimes HP_2 \otimes QP_3) \left[\left(P_1^{-1} \otimes P_2^{-1} \otimes P_3^{-1} \right) \mathcal{G} \right]$$

NOT IDENTIFIABLE





Why Nonnegativity in Tucker decomposition, the NMF case

$$M = WH = WPP^{-1}H$$

but if $W \geq 0$ and $H \geq 0$, sometimes

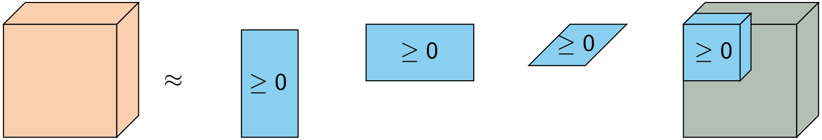
$$WP \geq 0 \text{ and } P^{-1}H \geq 0 \implies P = \Pi\Sigma$$

with Π a permutation matrix and Σ a positive diagonal matrix.

A collection of sufficient conditions for NMF identifiability

- Donoho2003: Separability
- Huang2013: sufficiently scattered condition
- Miao2007, Fu2015/Lin2015: Minimum Volume [not really a condition]

(approximate) Nonnegative Tucker Decomposition

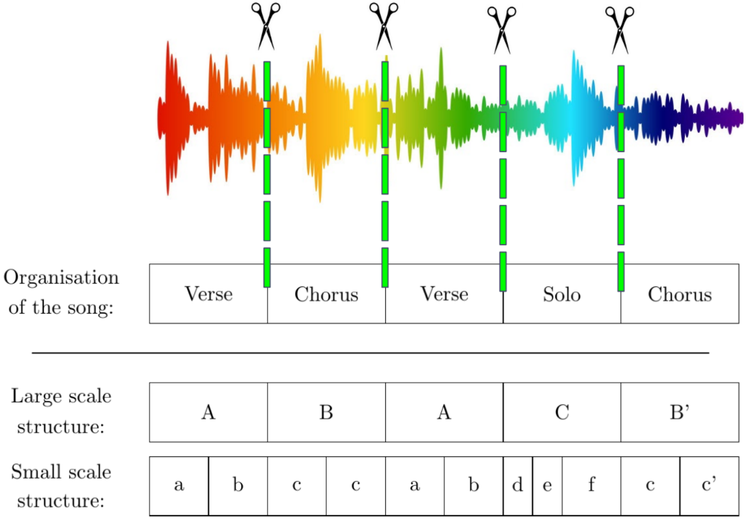


$$\mathcal{T} \approx U \times_1 V \times_2 W \times_3 \mathcal{G}$$

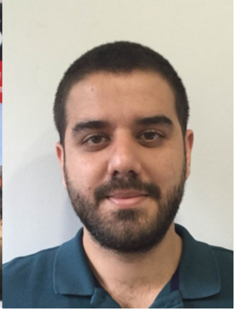
In the remainder of this talk, about NTD

- Can we interpret NTD on an example → Patterns in music
- How to compute NTD
- A few properties around CANDELINC and identifiability

Segmenting a song?



A team effort

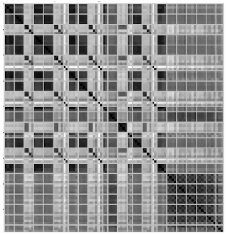


Axel Marmoret PhD student	Nancy Bertin CR CNRS	Frederic Bimbot DR CNRS	Caglayan Tuna Inria Engineer

Axel Marmoret, Jérémy Cohen, Nancy Bertin, Frédéric Bimbot. Uncovering Audio Patterns in Music with Nonnegative Tucker Decomposition for Structural Segmentation. ISMIR 2020 - 21st International Society for Music Information Retrieval, Oct 2020, Montréal (Online), Canada. pp.1-7

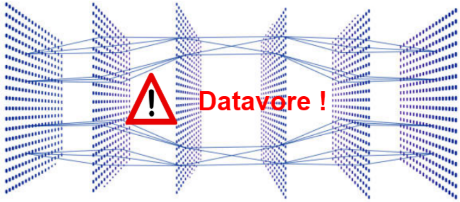
A word on the state-of-the-art

Unsupervised



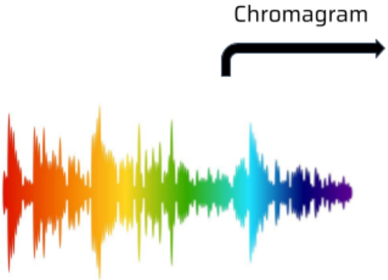
Signal Autosimilarity + post-processing

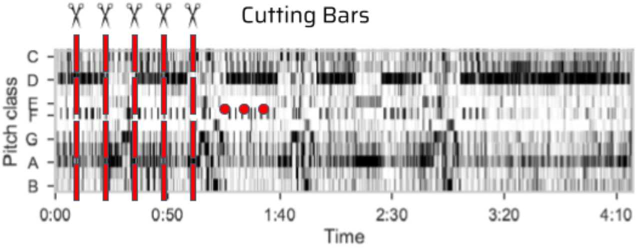
Supervised



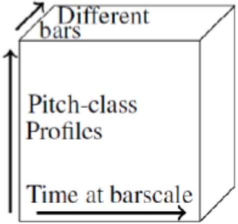
Deep learning

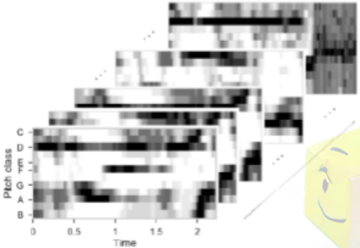
Our idea: a chromagram tensor. . .





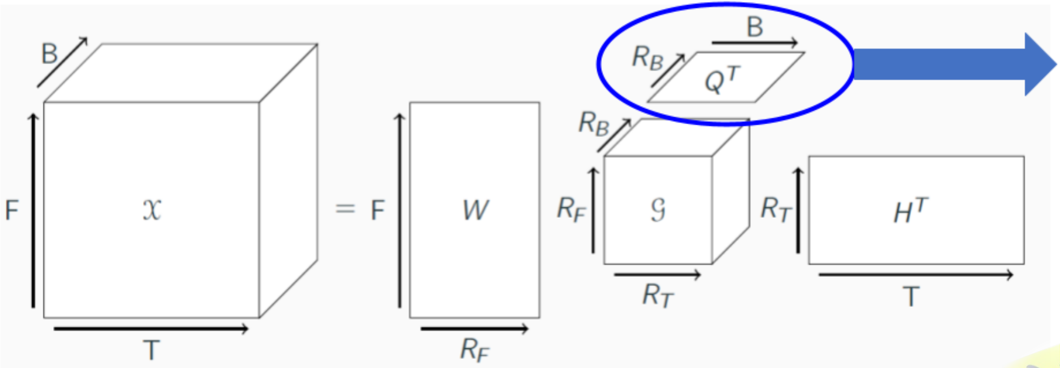
Chromagram of "Come Together", by The Beatles.





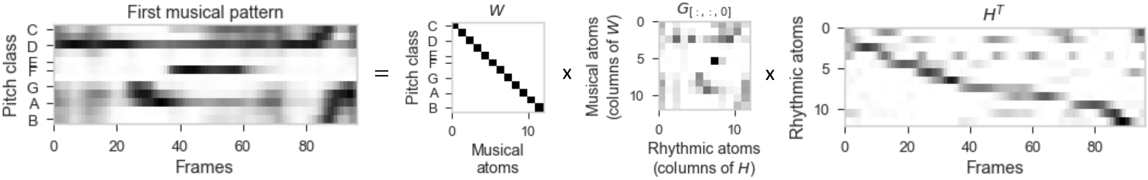
...decomposed to find redundancies!

Approximate Nonnegative Tucker Decomposition $\mathcal{X} \approx W \times_1 H \times_2 Q \times_3 \mathcal{G}$

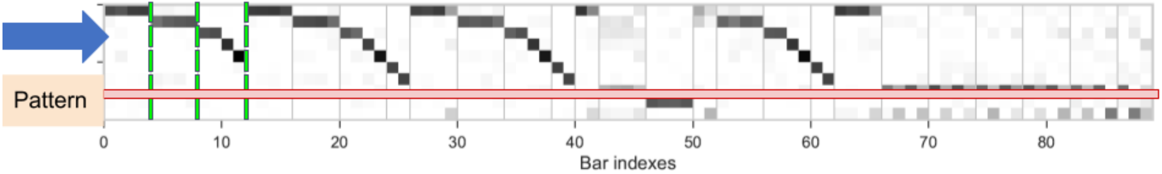


$\mathcal{X} \in \mathbb{R}_+^{F\times T\times B}, W \in \mathbb{R}_+^{F\times R_F}, H \in \mathbb{R}_+^{T\times R_T}, Q \in \mathbb{R}_+^{B\times R_B}, \mathcal{G} \in \mathbb{R}_+^{R_F\times R_T\times R_B}$

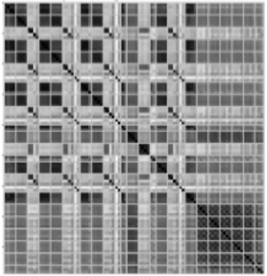
bonus: NTD extracts patterns!



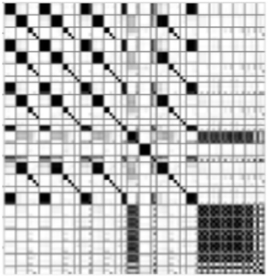
Back to segmentation



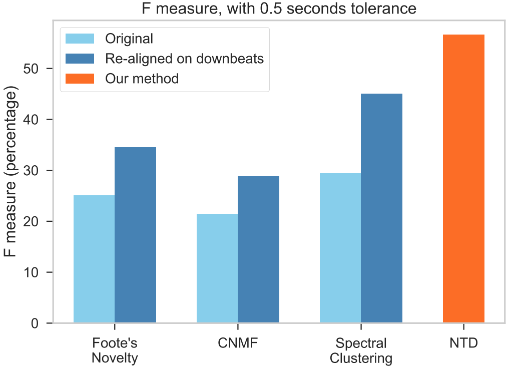
Signal Autosimilarity

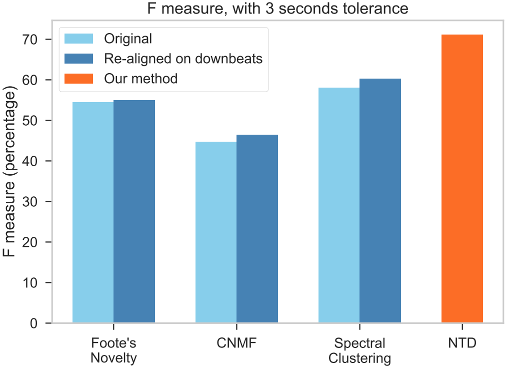


Patterns autosimilarity



State-of-the-art unsupervised results!





Algorithm	$P_{0.5}$	$R_{0.5}$	$F_{0.5}$	P_3	R_3	F_3
NTD, with “oracle ranks” for each song	67.1%	78.2%	71.5%	78.5%	90.2%	83.1%
Neural Networks[Grill2015]	80.4%	62.7%	69.7%	91.9%	71.1%	79.3%

Table: Averaged segmentation scores in the “oracle ranks” condition, compared to the current state-of-the-art (non-blind) method.

An algorithmic road

HALS principles
~2008
Nonnegative Matrix factorization

Nicolas Gillis, UMONS

MATLAB

Implementation and acceleration
~2012

PARAFAC decomposition
~2019

Nonnegative Tucker
~2020

Packages nnfac and MusicNTD

jupyter

GitLab

An algorithmic road

nnfac

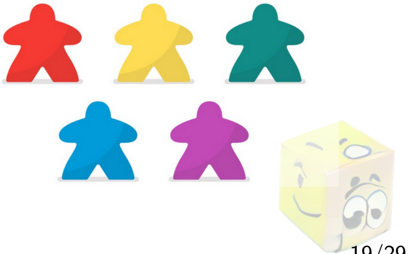
Inria

TensorLy

TensorFlow

PyTorch

NVIDIA CUDA



Back to NMF algorithms

NMF and numerical optimization

$$\underset{W\geq 0, H\geq 0}{argmin} \; D(M, WH)$$

Usual loss functions:

- Frobenius loss $D(M, WH) = \|M - WH\|_F^2$
- Kullback-Leibler $D(M, WH) = \sum_{ij} KL(M_{ij}, [WH]_{ij}) = \sum_{ij} M_{ij} \log(\frac{M_{ij}}{[WH]_{ij}}) + [WH]_{ij} - M_{ij}$
- Beta-Divergence
- More exotic: Wasserstein distance [Rolet2016, Varol2019]0, ℓ_1 norm [Gillis2018] . . .

A few remarks:

- Problem non-convex in general for (W, H) but “solvable” for fixed W or H.
- Beta-divergence loss is separable in columns of H (or rows of W).

This calls for block-coordinate descent methods:

- Hierarchical Alternating Least Squares (ℓ_2)
- Alternating Multiplicative Updates
- Alternating Proximal Gradient
- . . .

NTD algorithms mimic NMF algorithms

NTD and numerical optimization

$$\underset{W\geq 0, H\geq 0, Q\geq 0, \mathcal{G}\geq 0}{argmin} \quad D(M, (W \otimes H \otimes Q)\,\mathcal{G})$$

Usual loss functions:

- Frobenius loss $D(M, (W \otimes H \otimes Q)\,\mathcal{G}) = \|M - (W \otimes H \otimes Q)\,\mathcal{G}\|_F^2$
- Kullback-Leibler $D(M, (W \otimes H \otimes Q)\,\mathcal{G}) = \sum_{ijk} KL(M_{ijk}, [(W \otimes H \otimes Q)\,\mathcal{G}]_{ijk})$

A few key points:

- The core update is a “vector” update (not matrix!)
- One must pay attention to update rules, to avoid computing big intermediate representations and Kronecker products.

Existing algorithms (sample):

- HALS + Proximal Gradient for $\mathcal{G}$
- Alternating MU

What about sparsity?

In the first NTD paper [Morup 2008], sparsity was already considered.

Sparsity?

Most papers impose sparsity with ℓ_1 norm.
Problem: Scale ambiguity!! For $\mu > 1$,

$$\|M - WH\|_F^2 + \lambda\|W\|_1 > \|M - \frac{1}{\mu}W\mu H\|_F^2 + \frac{\lambda}{\mu}\|W\|_1 = \|M - WH\|_F^2 + \lambda'\|W\|_1$$

with $\lambda' < \lambda$.

- Several work around for NMF
  - Constrain W on the hypersphere [LeRoux2015]
  - Use a more complex sparsity metric [Hoyer2002/2004]
  - Use ℓ_2 on W [??][RoaldTBA] How to use in MU?
- Not so many are described (?) for tensor decompositions.

Work in Progress: paper and codes for NTD with beta-divs, sparsity, acceleration!

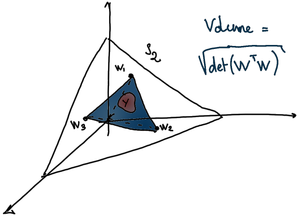
NTD identifiability

The big open question: under which conditions is NTD identifiable/essentially unique?

A few empirical observations:

- NTD factors and core can be recovered when they are very sparse, even without explicit sparsity imposed (sufficiently scattered??)
- Imposing sparsity helps a lot in recovering the true factors and core.

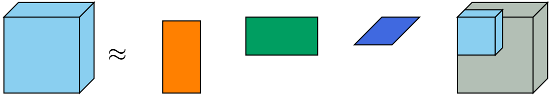
What about minimum volume? Separability?



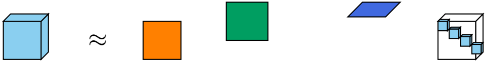
An existing result in [Zhou/Cichocki 2014] links NTD identifiability to NMF identifiability of the unfoldings.

NTD for nnCANDELINC [C.2017]

CANDELINC: Tucker format then PARAFAC



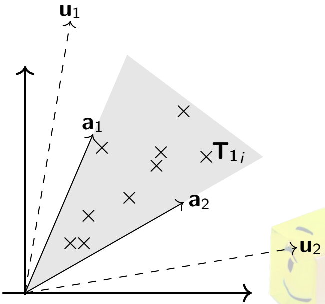
$$\mathcal{T} \approx (W \otimes H \otimes Q)\ \mathcal{G}$$

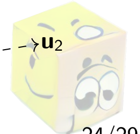


$$\mathcal{G} \approx (A_G \otimes B_G \otimes C_G)\ \mathcal{I}_r$$

Problems with nnCANDELINC

- Rank of core might increase
- Factors of $\mathcal{T}$ might not be recovered
- NTD is hard to compute anyway
- Does not work in (my) pratice





NTD for nnCANDELINC [Skau DeSantis 2022]

A few interesting concepts/facts:

- Nonnegative multilinear ranks

$$\operatorname{rank}_+(\mathcal{T}_{[n]})$$

- Intersection of tensor cones and tensor product don't commute
- **Minimal NTD** has dimension equal to nonnegative multilinear ranks (may not exist)
- **Canonical NTD** when dimensions equal to nonnegative ranks of factors for a unique CPD tensor.

Proposition

Suppose $\mathcal{T}$ admits a unique CPD.

- Then there exists a canonical NTD which preserves its nonnegative rank.
- For any canonical NTD that preserves the rank, its factors have full nonnegative rank.

Core problem: selecting the right canonical NTD.

Conclusion

Similarities between NMF and NTD

- Numerical Optimization
- Applications, to some extent
- Decomposition of data into a sum of parts
- Empirically, identifiability

Some major differences

- NTD theory requires multilinear algebra
- Almost no identifiability results available for NTD
- Connection between NTD and polytopes?
- NTD is hard to understand
- Few dedicated algorithms, e.g. efficient initialization

Tensorly ad 1: What is Tensorly

TensorLy **Open source and collaborative Python toolbox for tensors**

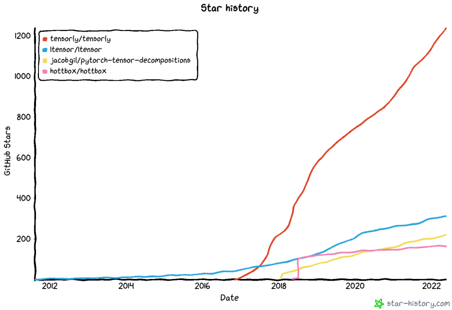
Code features:

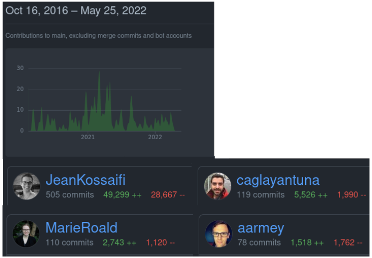
- User guide, API, Examples at `tensorly.org`
- Automatic unit tests
- Back-end transparent for users and devs
- Issues/Pull Requests with reasonable response time

Contents:

- Tensor objects from Numpy, Pytorch, Tensorflow...
- Tensor manipulations (reshape, permute and so)
- Some tensor decompositions (CP, constrained CP, Generalized CP, Tucker, Nonnegative Tucker, TT, PARAFAC2, CMTF)
- Dataset loaders, visualisation tools

Tensorly ad 2: Tensoptly project



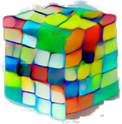


- New algorithms and models
  - Nonnegative/Sparse/**User-defined** constraint using AOADMM.
  - **User-defined** loss using GCP.
- Contributions tested, documented, explained (Notebooks)

Where to contribute

- Backend: efficient contractions support (TTMs, TTVs, MTTKRPs ...)
- Algorithms: better CPD algorithms than ALS!
- Visualisation: How to look at tensors? Tucker models?
- Benchmarking with Benchopt?




Thank you for your attention!!