Sparse Multiple Nonnegative Least Squares with a Matrix-Wise ℓ_0 Constraint

(Still a work in progress!)

Nicolas Nadisic, Arnaud Vandaele, Nicolas Gillis

20 May 2021 — SIAM LA21

Université de Mons, Belgium

Outline

1. Motivation: Nonnegative Matrix Factorization (NMF)
2. Sparse Nonnegative Least Squares (NNLS)
3. Our matrix-wise ℓ_0-constrained method
4. Experiments
5. Conclusion

Motivation: Nonnegative Matrix Factorization (NMF)

Nonnegative Matrix Factorization (NMF)

Given

- Data matrix $M \in \mathbb{R}_+^{m \times n}$
- Rank $r \ll \min(m, n)$

find

- $W \in \mathbb{R}_+^{m \times r}$
- and $H \in \mathbb{R}_+^{r \times n}$

such that $M \approx WH$.

In optimization terms, standard NMF is equivalent to:

$$\min_{W \geq 0, H \geq 0} \|M - WH\|_F^2$$

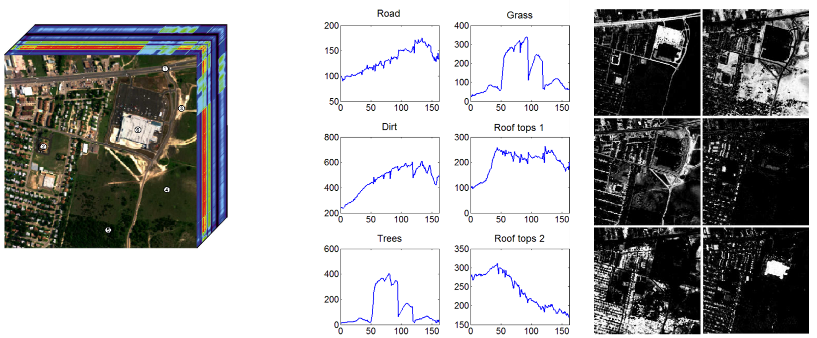
Nonnegative Matrix Factorization (NMF)

Why nonnegativity?

- More interpretable factors (part-based representation)
- Naturally favors sparsity
- Is natural in many applications (image processing, hyperspectral unmixing, text mining, . . .)

NMF Application — Hyperspectral Unmixing

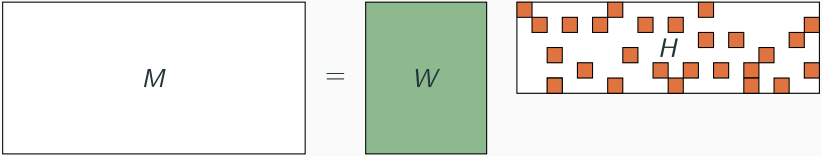
$$\underbrace{M(:,j)}_{\substack{\text{spectral signature of}\\ \text{j-th pixel}}} \approx \sum_{p} \underbrace{W(:,p)}_{\substack{\text{spectral signature of}\\ \text{p-th material}}} \underbrace{H(p,j)}_{\substack{\text{abundance of p-th material}\\ \text{in j-th pixel}}}$$



Images from Bioucas Dias and Nicolas Gillis.

Our focus: sparse optimization of H

- NMF algo. usually rely on alternating optimization of W and H (iteratively optimize one while fixing the other).
- Here we focus on the sparsity of H, but the concepts and algorithms can be applied on W symmetrically.



Multiple Nonnegative Least Squares (MNNLS)

The optimization of H

$$\min_{H \geq 0} \|M - WH\|_F^2 \tag{1}$$

can be decomposed into n nonnegative least squares (NNLS) subproblems

$$\min_{x \geq 0} \|b - Ax\|_2^2, \tag{2}$$

where $M(:, j)$, W, and $H(:, j)$ correspond respectively to b, A, and x.

Sparse Nonnegative Least Squares (NNLS)

NNLS/NMF and sparsity

Nonnegativity naturally favors sparsity = factors with few nonzero entries, but without guarantee.
It can be useful to enforce it explicitly, to:

- Further improve interpretability
- Leverage prior knowledge on sparsity
- Be able to impose an explicit user-defined level of sparsity

Measuring sparsity

A natural measure: ℓ_0-"norm"

$||x||_0 = |\{i : x_i \neq 0\}|$ (number of nonzero entries of x)

Convex relaxation: ℓ_1-norm

$$||x||_1 = \sum_{i=1}^{n} |x_i|$$

ℓ_0 constraint for sparse NNLS/NMF

- With a ℓ_0-"norm" constraint, k-sparse NNLS

$$\min_{x \geq 0} \|b - Ax\|_2^2 \text{ s.t. } \|x\|_0 \leq k$$

- in MNNLS/NMF, generally applied column-wise (or row-wise)

$$\min_{H \geq 0} \|M - WH\|_F^2 \text{ s.t. } \forall j, \|H(:,j)\|_0 \leq k$$

- Intuitive formulation: a pixel contains at most k materials

Limitations

- Column-wise k-sparse MNNLS/NMF is usually satisfactory
- but setting k can be hard in some contexts, where sparsity is not the same for all columns
- Ex: in hyperspectral unximing, the number of material differ between pixels

A matrix-wise constraint

Matrix-wise q-sparse MNNLS

$$\min_{H \geq 0} \|M - WH\|_2^2 \text{ s.t. } \|H\|_0 \leq q$$

- Can be seen as a global sparsity budget
- If $q = k \times n$, this enforces an average k-sparsity on the columns of H

A matrix-wise constraint

Matrix-wise q-sparse MNNLS

$$\min_{H \geq 0} \|M - WH\|_2^2 \text{ s.t. } \|H\|_0 \leq q$$

- Can be seen as a global sparsity budget
- If $q = k \times n$, this enforces an average k-sparsity on the columns of H

How to solve it?

- With a k-sparse NNLS methods, by vectorizing the problem
- $\Rightarrow$ leads to a huge NNLS problem, expensive to solve

Vectorizing the MNNLS problem is expensive

$$\min_{H \geq 0} \|M - WH\|_2^2 \text{ s.t. } \|H\|_0 \leq q$$

$\Rightarrow$ vectorize

$$\min_{h \geq 0} \|m - \Omega h\|_2^2 \text{ s.t. } \|h\|_0 \leq q$$

where $\Omega = W \otimes I \in \mathbb{R}^{(m.n)\times(r.n)}$ and $m = \begin{bmatrix} M(:,1) \\ M(:,2) \\ \vdots \\ M(:,n) \end{bmatrix} \in \mathbb{R}^{(m.n)}$

Our matrix-wise ℓ_0-constrained method

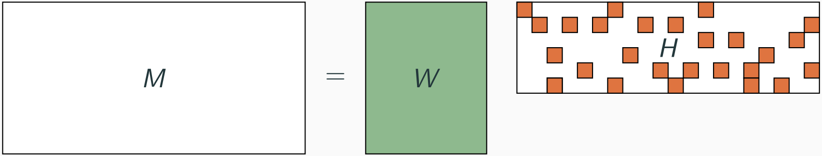
A two-step algorithm

1. Generate a set of solutions for every column of H, with different tradeoffs between reconstruction error and sparsity
   - Divide the sparse MNNLS problem into n sparse NNLS subproblems

$$\min_{H(:,j)\geq 0} \|M(:,j) - WH(:,j)\|_2^2 \text{ s.t. } \|H(:,j)\|_o \geq k$$

   - For each column j, get the solutions for all $k \in \{1, \ldots, r\}$
2. Select one solution per column such that, in total H has q nonzero entries, and the error is minimized $\Rightarrow$ assignment-like problem

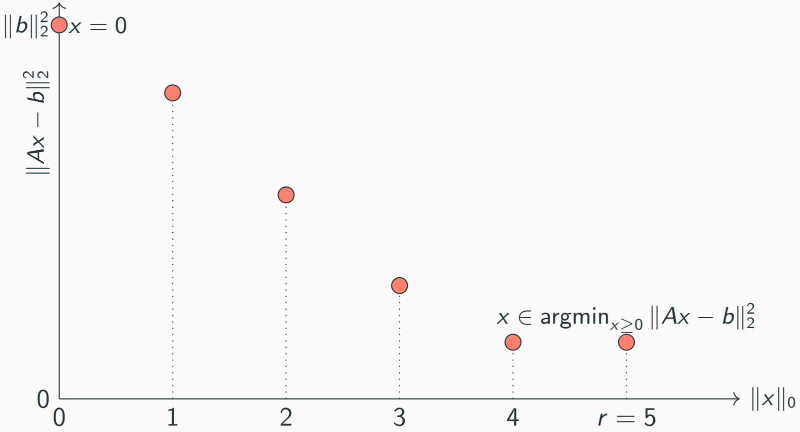
Illustration



$$\begin{pmatrix} C_{0,1} & C_{0,2} & \cdots & C_{0,n} \\ C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{r,1} & C_{r,2} & \cdots & C_{r,n} \end{pmatrix}$$

Step 1: Generate solution path

Example for one column for $r = 5$



Solve $\min\limits_{x \geq 0} \|b - Ax\|_2^2$ s.t. $\|x\|_0 \leq k$ for all $k \in \{0, \ldots, r\}$

Step 1: Generate solution path

Algorithms:

- Homotopy (see eg Kim et al. 2013), heuristic based on ℓ_1 relaxation
- NNOMP, NNOLS (see eg Nguyen et al. 2019), greedy heuristics
- arborescent (Nadisic et al. 2020), exact branch-and-bound algorithm

NB: These are existing algorithms, not original contribution

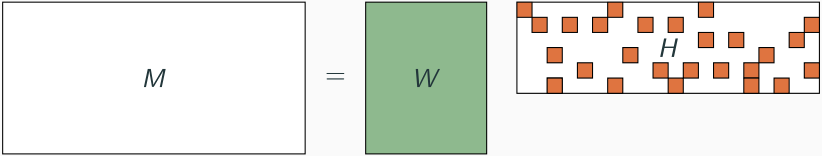
Step 2: Matrix-wise selection

Our contribution!

Select one solution per column to build a q-sparse matrix minimizing the error

- We have to solve a kind of assignment problem.
- First, build a cost matrix
  - each column represents a column $H(:,j)$
  - each row represents a k-sparsity between 0 and r
  - each entry is the error $\|M(:,j) - WH(:,j)\|_2^2$ of the k-sparse solution of column j.
- Then, greedy-like selection algorithm, fast but proved exact

Illustration



$$\begin{pmatrix} C_{0,1} & C_{0,2} & \cdots & C_{0,n} \\ C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{r,1} & C_{r,2} & \cdots & C_{r,n} \end{pmatrix}$$

Guarantees

- Matrix-wise selection (step 2) is optimal
- Path-generating step (step 1)
  - Homotopy $\Rightarrow$ heuristic, inexact but fast
  - Greedy algorithms $\Rightarrow$ heuristic, inexact but fast
  - Branch-and-bound $\Rightarrow$ exact but slow

Experiments

Experiments

(Still a work in progress!)

- Faces datasets
  - Extract facial features
  - Sparsity = each feature has few pixels
- Hyperspectral datasets
  - Extract materials
  - Sparsity = each pixel has few materials

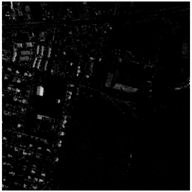
Experiments results

		Ht no-s	Ht k-s	Ht+sel	NNOLS
CBCL	Time	0.78	0.77	0.84	0.26
$r = 49$	Rel error	12.04	16.19	13.22	**12.60**
$k = 3$	Sparsity	6.53	2.69	3.0	2.71
Kuls	Time	0.25	0.21	0.30	1.87
$r = 5$	Rel error	19.05	20.13	**19.12**	19.20
$k = 3$	Sparsity	3.44	2.86	3.0	2.84
Jasper	Time	0.63	0.63	0.74	2.25
$r = 4$	Rel error	5.71	6.99	**5.72**	6.09
$k = 2$	Sparsity	2.27	1.79	2.0	1.81
Jasper	Time			0.65	
$r = 4$	Rel error			**5.95**	
$q/n = 1.8$	Sparsity			1.8	
Urban	Time	6.67	6.55	10.73	22.04
$r = 6$	Rel error	7.67	8.62	**7.83**	8.14
$k = 2$	Sparsity	2.61	1.90	2.0	1.90

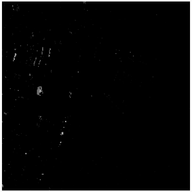
Experiments results — Kuls faces, $k = 3$

Experiments results

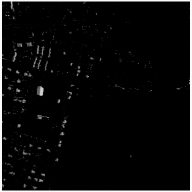
Urban hyperspectral image, 6th endmember, $k = 2$



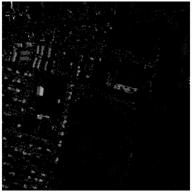
(a) Ht no sparse



(b) Ht k-sparse

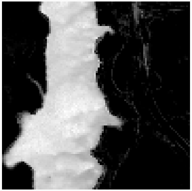


(c) Ht+sel

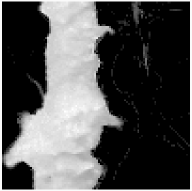


(d) NNOLS

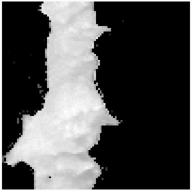
Jasper hyperspectral image, 2nd endmember



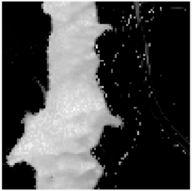
(e) Ht no sparse



(f) Ht+sel with $k = q/n = 2$



(g) Ht+sel with $k = q/n = 1.8$



(h) NNOLS with $k = 2$

Conclusion

Conclusion

- We introduced an sparse MNNLS model with matrix-wise ℓ_0-sparsity constraint
- We developed a 2-step algorithm to tackle it
  1. Any column-wise k-sparse method to generate paths of solutions
  2. Greedy-like algorithm to select solutions, exact and cheap
- Improves results, allows a finer parameter tuning
- Interesting where sparsity varies between columns
- Almost as fast as standard NNLS algorithm

Next?

- Finish experiments with other path-generating methods (branch-and-bound, greedy algorithms, . . .)

References i

Kim, Jingu et al. (2013). “Regularization Paths for Sparse Nonnegative Least Squares Problems with Applications to Life Cycle Assessment Tree Discovery”. In: *2013 IEEE 13th International Conference on Data Mining*, pp. 360–369.

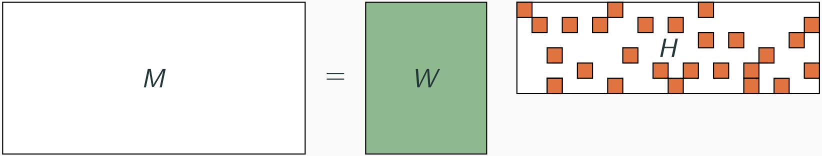
Nguyen, Thanh T. et al. (2019). “Non-Negative Orthogonal Greedy Algorithms”. In: *IEEE Transactions on Signal Processing*, pp. 1–16.

Nadisic, Nicolas et al. (2020). “Exact Sparse Nonnegative Least Squares”. In: *ICASSP 2020 - IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5395–5399. (Acceptance rate 47%).

Thanks!

Contact: nicolas.nadisic@umons.ac.be

Website: http://nicolasnadisic.xyz



End of presentation

Empty slide in case of question

Second empty slide in case of second question