Matrix-wise ℓ_0-constrained Sparse Nonnegative Least Squares

and application to hyperspectral unmixing

Nicolas Nadisic[1,2]

08 April 2024 — LACODAM, Université de Rennes

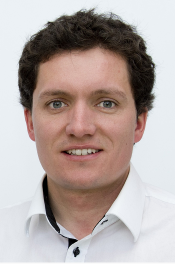
[1]Ghent University, Belgium
[2]Royal Institute for Cultural Heritage (KIK-IRPA), Belgium

Outline

Introduction

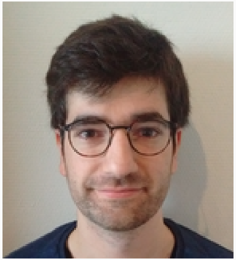
With a little help from my friends

Nicolas Gillis
(UMONS, Belgium)



Arnaud Vandaele
(UMONS, Belgium)

Jeremy Cohen
(CNRS, Univ Lyon, France)



Our motivations

High-level motivations:

- Extract underlying structures in data
- Better leverage a priori knowledge, here nonnegativity and sparsity, to improve models
- Develop algorithms that are both globally optimal and computationally tractable

Starting point: linear models

Focus of this work: linear models of the form

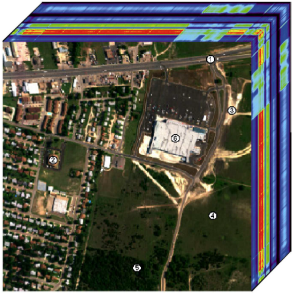
$$B \approx AX,$$

where

- $B \in \mathbb{R}^{m \times n}$ is the data/input matrix, representing measures or observations,
- $A \in \mathbb{R}^{m \times r}$ is a coeficient matrix, called dictionary, representing features, atoms, or components.
- $X \in \mathbb{R}^{r \times n}$ is a signal or information matrix,
- $r \ll \min(m, n)$

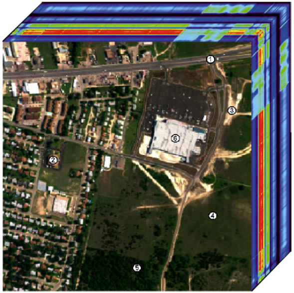
One application — Hyperspectral unmixing

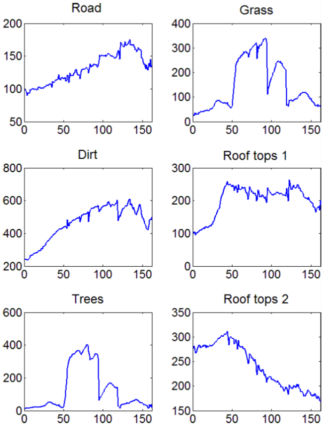
$$\underbrace{B(:,j)}_{\text{spectral signature of j-th pixel}}$$

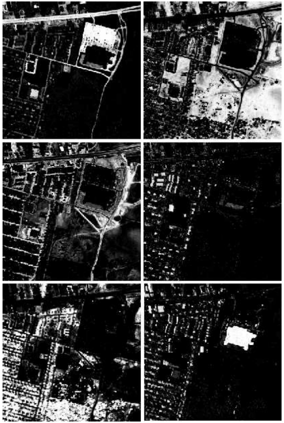


One application — Hyperspectral unmixing

$$\underbrace{B(:,j)}_{\text{spectral signature of j-th pixel}} \approx \sum_p \underbrace{A(:,p)}_{\text{spectral signature of p-th material}} \underbrace{X(p,j)}_{\text{abundance of p-th material in j-th pixel}}$$

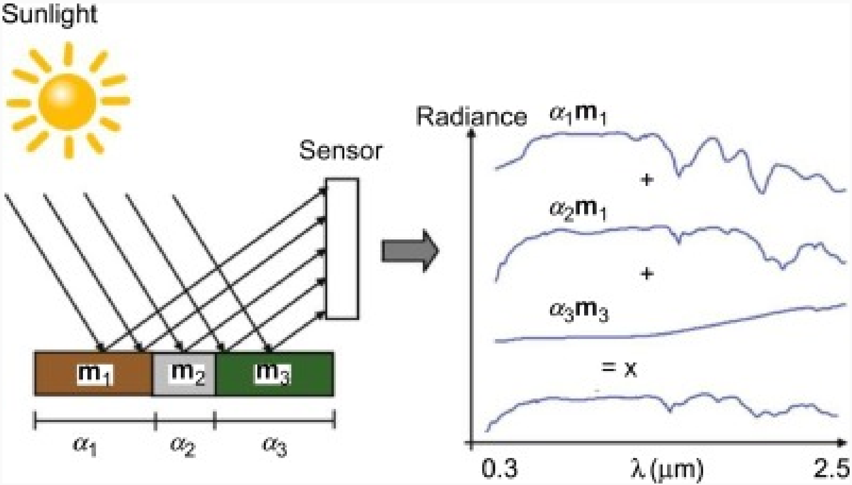






Images from J. Bioucas Dias and N. Gillis.

Linear mixing model



Nonnegativity constraint

- Assumes data is generated from an additive linear combination of features
- Natural in this application
- Produces more interpretable factors

How to find X given B and A?

Multiple Nonnegative Least Squares (MNNLS) problem

$$\min_{X \geq 0} \|B - AX\|_F^2$$

How to find X given B and A?

Multiple Nonnegative Least Squares (MNNLS) problem

$$\min_{X \geq 0} \|B - AX\|_F^2$$

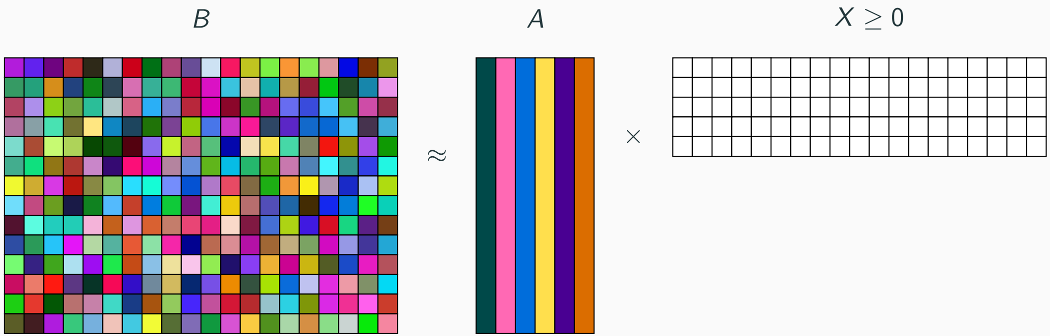
Can be divided in n independent NNLS subproblems,

$$\min_{X(:,j) \geq 0} \|B(:,j) - AX(:,j)\|_2^2$$

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

Multiple Nonnegative Least Squares (MNNLS)

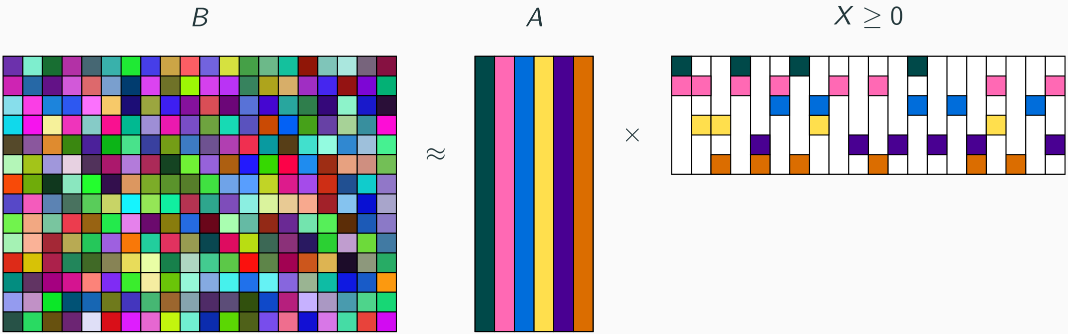
Given B and A, find $X \geq 0$



Sparsity — Why?

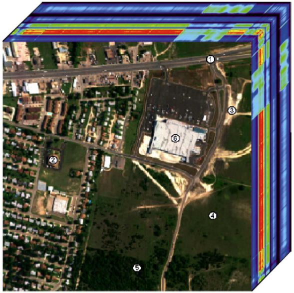
Sparsity of X $\Rightarrow$ Each data point is a combination of only a few features

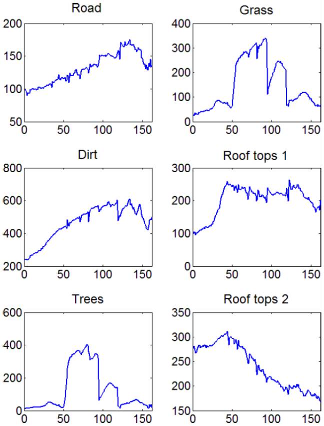
- Regularize the problem
- Better interpretability
- Natural in many applications $\Rightarrow$ leverage a-priori knowledge to improve the model

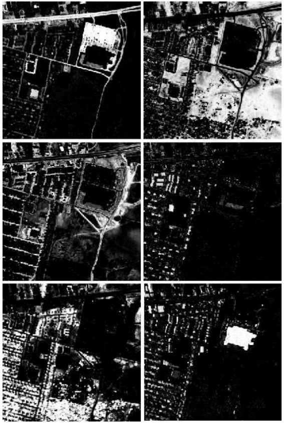


Sparsity in hyperspectral unmixing

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







Sparsity — How?

The classical way: ℓ_1 penalty

$$\min_{X \geq 0} \|B - AX\|_F^2 + \lambda \|X\|_1$$

Advantages:

- Convex, easy to optimize

Issues:

- Restrictive condititions for support recovery
- Parameter λ is hard to tune, no physical meaning

Sparsity — How?

More intuitive formulation: column-wise k-sparsity constraint, using the ℓ_0-"norm", $\|x\|_0 = |\{i : x_i \neq 0\}|$

$$\min_{X \geq 0} \|B - AX\|_2^2 \text{ s.t. } \|X(:,j)\|_0 \leq k \text{ for all } j$$

Advantage:

- Interpretable: each data point is a combination of at most k features

Column-wise sparse NNLS

Solving column-wise k-sparse NNLS

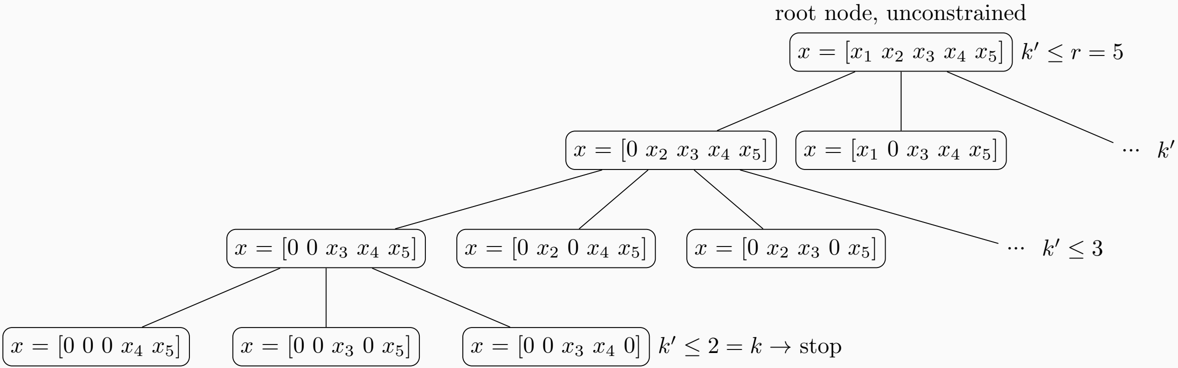
Let us focus on the one-column problem for now,

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

- Reduces to finding the support of x (set of non-zero entries)
- Combinatorial problem, $\binom{r}{k}$ possible supports
- Can be solved approximately by greedy algorithms
- Or optimally with branch-and-bound algorithms

A branch-and-bound algorithm for k-sparse NNLS

Example for $r = 5$ and $k = 2$

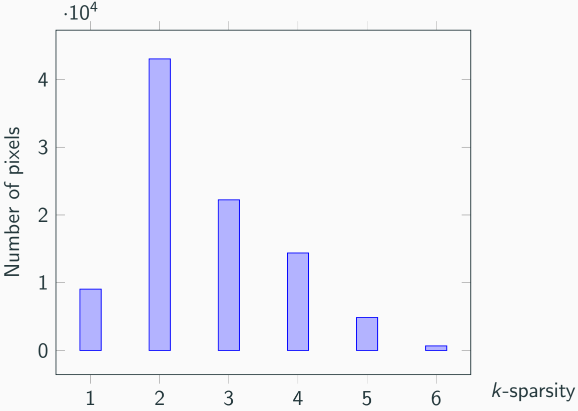


Able to prune large parts of the search space.

Limits of column-wise sparse NNLS

Issue of the column-wise constraint:

- What if the relevant k varies between columns?
- For instance, the number of materials varies between pixels



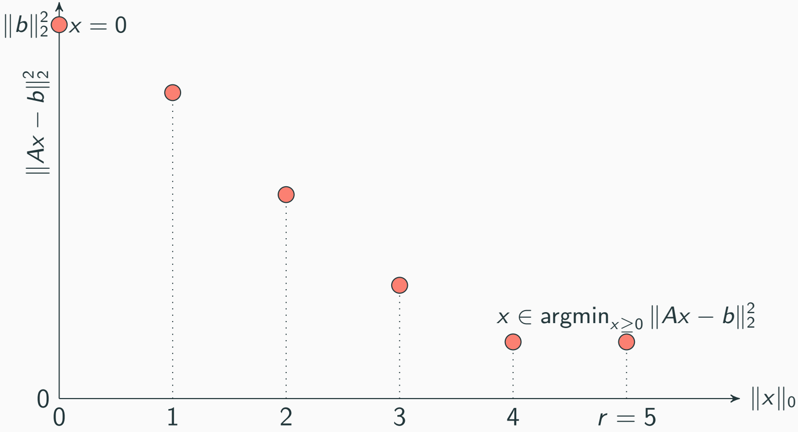
Bi-objective sparse NNLS

$$\min_{x \geq 0} \begin{cases} \|Ax - b\|_2^2 \\ \|x\|_0 \end{cases}$$

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

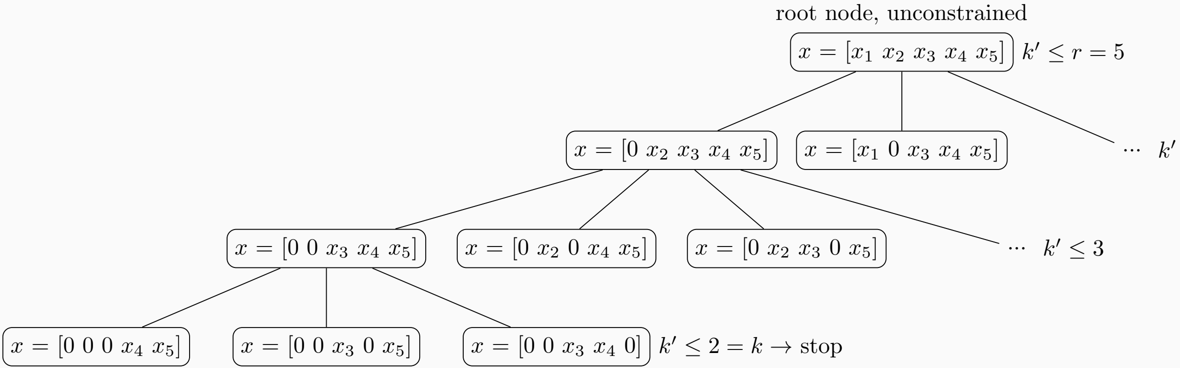
Bi-objective sparse NNLS

Example for $r = 5$



Extension of the branch-and-bound algorithm

Example for $r = 5$ and $k = 2$



Computes the whole Pareto front!

How to leverage this bi-objective formulation on a multicolumn problem?

$$\min_{X \geq 0} \|B - AX\|_F^2$$

Matrix-wise sparse NNLS

Our solution: A matrix-wise ℓ_0 constraint

Matrix-wise q-sparse MNNLS

$$\min_{X \geq 0} \|B - AX\|_F^2 \quad \text{s.t.} \quad \|X\|_0 \leq q$$

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

Our solution: A matrix-wise ℓ_0 constraint

Matrix-wise q-sparse MNNLS

$$\min_{X \geq 0} \|B - AX\|_F^2 \quad \text{s.t.} \quad \|X\|_0 \leq q$$

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

How to solve it?

- With a k-sparse NNLS methods, by vectorizing the problem
  $\Rightarrow$ leads to a huge NNLS problem, too expensive to solve
- Our contribution: dedicated algorithm

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 contribution: a two-step algorithm

Algorithm Salmon[1]:

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

$$\min_{X(:,j)\geq 0} \left\{ \quad \|B(:,j) - AX(:,j)\|_2^2 \quad , \quad \|X(:,j)\|_0 \quad \right\}$$

   - Solve with branch-and-bound, or heuristic (homotopy, greedy algo)
   - Build a cost matrix C

[1] Salmon Applies ℓ_0-constraints Matrix-wise On NNLS problems

Our contribution: a two-step algorithm

Algorithm Salmon[1]:

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

$$\min_{X(:,j)\geq 0} \{ \quad \|B(:,j) - AX(:,j)\|_2^2 \quad , \quad \|X(:,j)\|_0 \quad \}$$

   - Solve with branch-and-bound, or heuristic (homotopy, greedy algo)
   - Build a cost matrix C
2. Select one solution per column such that in total X has q nonzero entries and the error is minimized $\Rightarrow$ assignment-like problem
   - Dedicated greedy algorithm proved near-optimal

[1] Salmon Applies ℓ_0-constraints Matrix-wise On NNLS problems

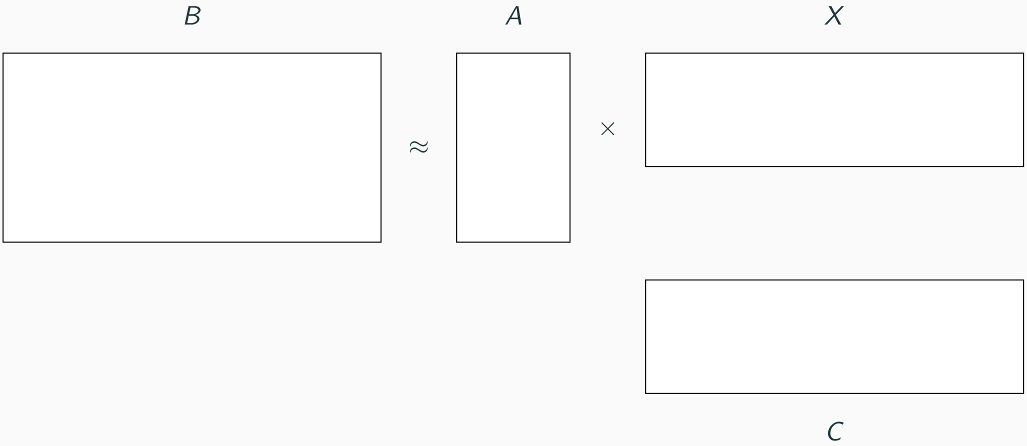
Salmon — Step 1: Build the cost matrix C

- Each row = one sparsity level
- Each column = one column of the MNNLS problem

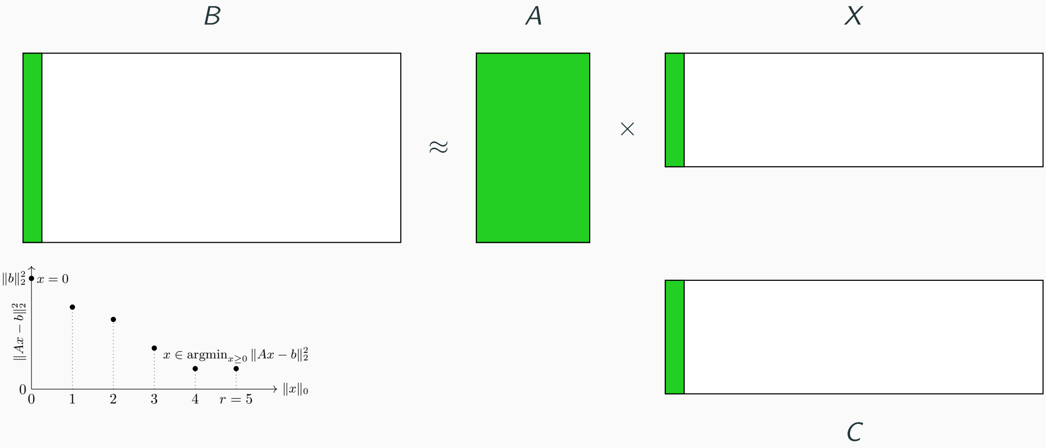
$$\begin{pmatrix} C_{0,1} & C_{0,2} & \cdots & C_{0,n} \\ C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{r,1} & C_{r,2} & \cdots & C_{r,n} \end{pmatrix}$$

$$C(i,j) \quad \approx \quad \min_{x \geq 0} \|B(:,j) - Ax\|_2^2 \text{ s.t. } \|x\|_0 \leq i$$

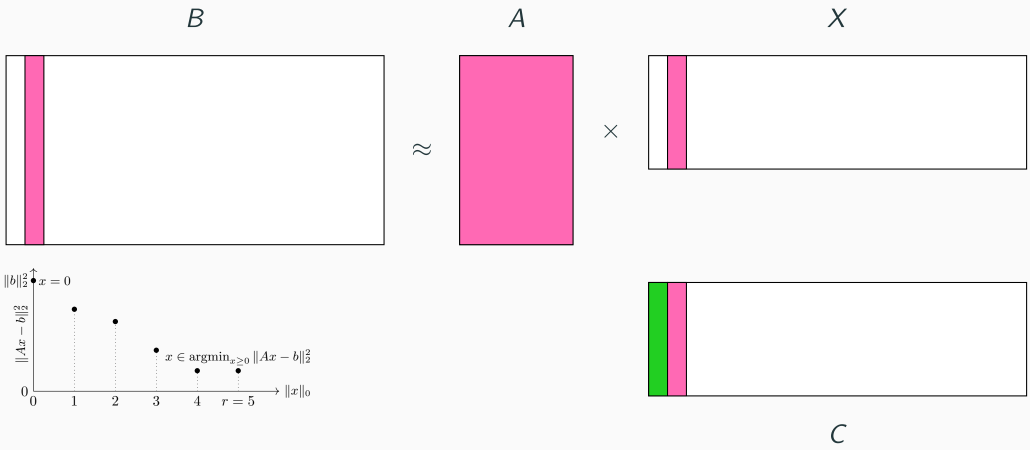
Salmon — Step 1: Generate Pareto fronts



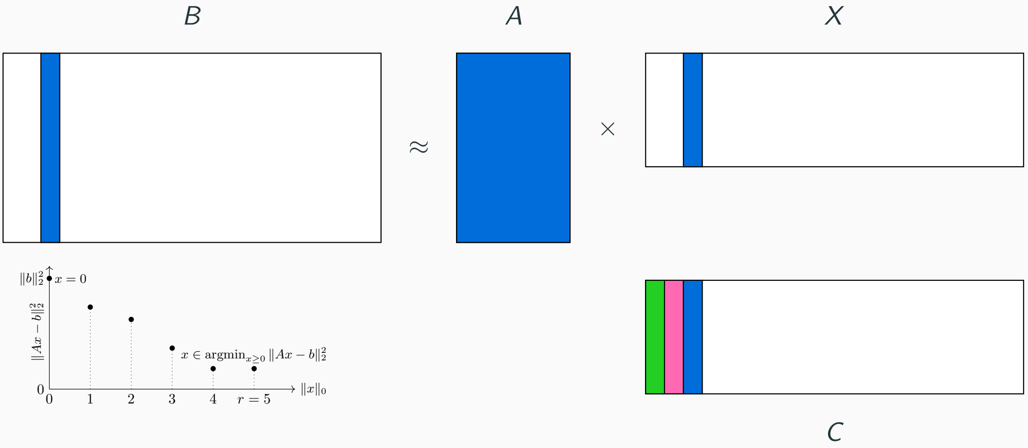
Salmon — Step 1: Generate Pareto fronts



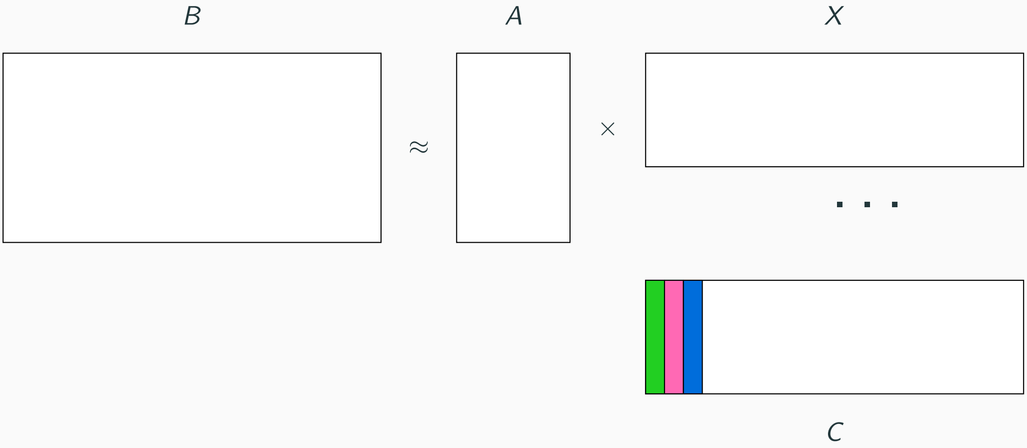
Salmon — Step 1: Generate Pareto fronts



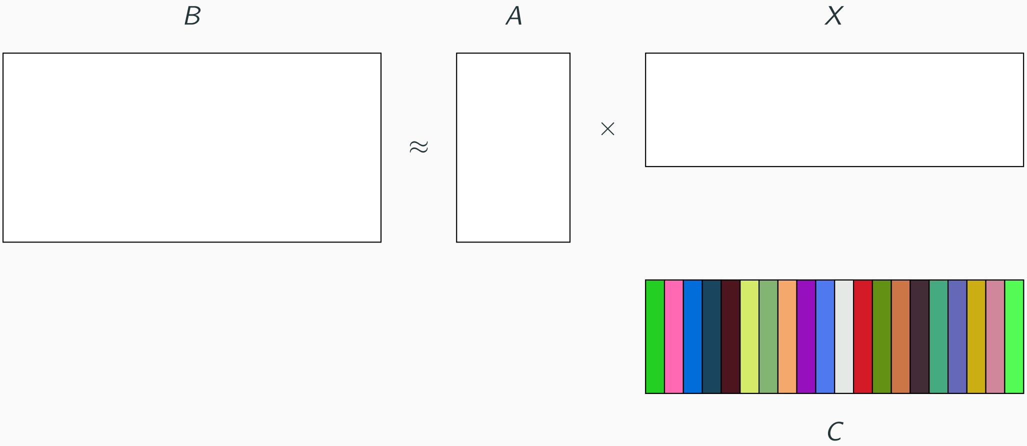
Salmon — Step 1: Generate Pareto fronts



Salmon — Step 1: Generate Pareto fronts



Salmon — Step 1: Generate Pareto fronts



Salmon — Step 2: Select one solution per column

Similar to an assignment problem

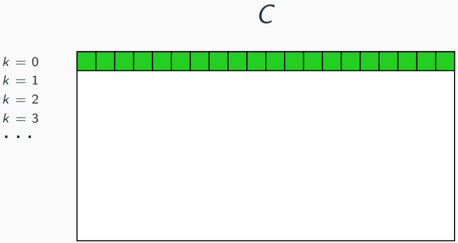
$$\begin{pmatrix} C_{0,1} & C_{0,2} & \cdots & C_{0,n} \\ C_{1,1} & C_{1,2} & \cdots & C_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{r,1} & C_{r,2} & \cdots & C_{r,n} \end{pmatrix}$$

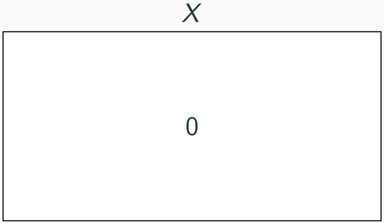
Let $z_{i,j} \in \{0,1\}$ such that $z_{i,j} = 1$ if and only if the jth column of X is i-sparse,

$$\min_{z \in \{0,1\}^{r \times n}} \sum_{i,j} z_{i,j} C(i,j)$$
$$\text{such that } \sum_{i} z_{i,j} = 1 \text{ for all } j, \quad \text{and} \quad \sum_{i,j} i\, z_{i,j} \leq q.$$

Solved with a dedicated greedy algorithm, fast but proved near-optimal

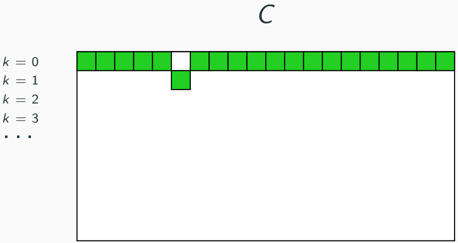
Salmon — Step 2: Greedy selection

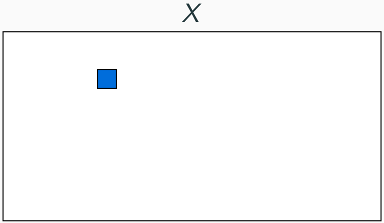




$$\|X\|_0 = 0$$

Salmon — Step 2: Greedy selection

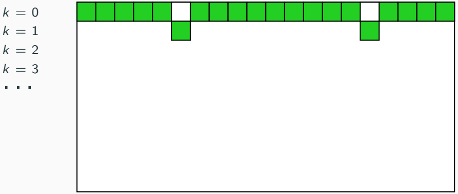




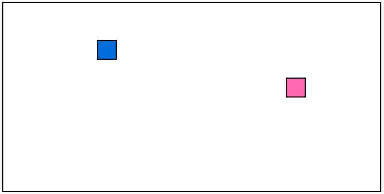
$$\|X\|_0 = 1$$

Salmon — Step 2: Greedy selection

C

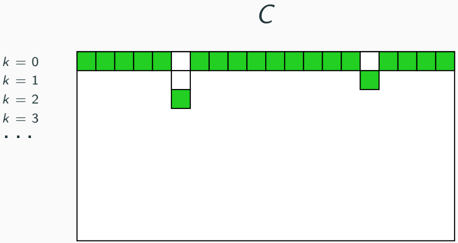


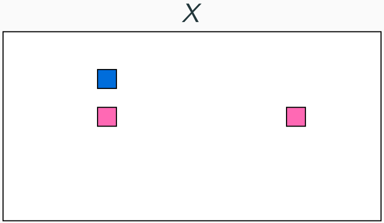
X



$$\|X\|_0 = 2$$

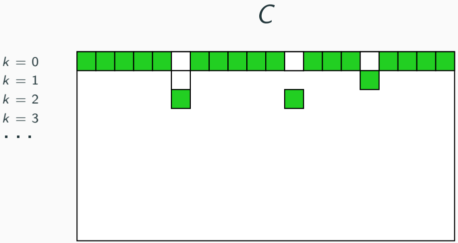
Salmon — Step 2: Greedy selection

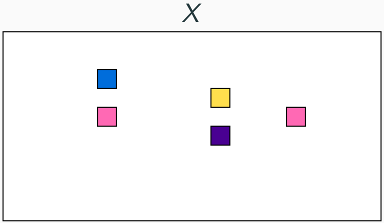




$$\|X\|_0 = 3$$

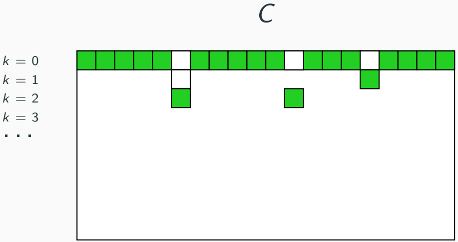
Salmon — Step 2: Greedy selection

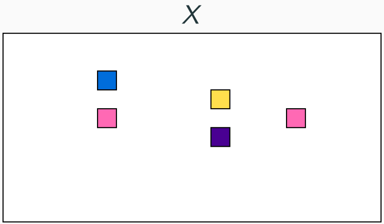




$$\|X\|_0 = 5$$

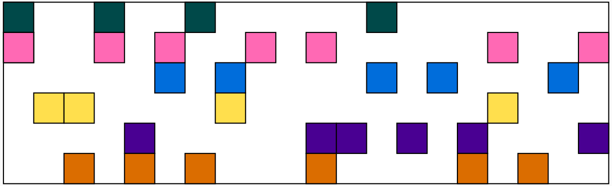
Salmon — Step 2: Greedy selection





$\|X\|_0 = 5$

Iterate while $\|X\|_0 < q$



Final solution X, q-sparse matrix

$$X \approx \arg\min_{X \geq 0} \|B - AX\|_F^2 \quad \text{s.t.} \quad \|X\|_0 \leq q$$

Near-optimality of the selection step (step 2)

In short:

- The worst case is not too bad (wrong support in at most one column)
- In practice, often optimal (19 out of 22 cases in our exp)

Near-optimality of the selection step (step 2)

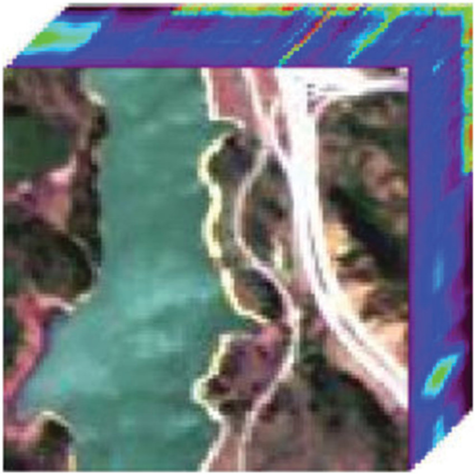
In short:

- The worst case is not too bad (wrong support in at most one column)
- In practice, often optimal (19 out of 22 cases in our exp)

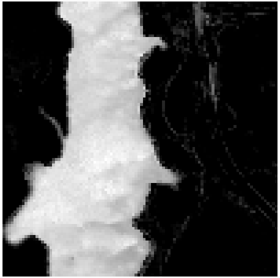
Intuition of the proof:

- The objective function is separable by columns
- At each iteration, we maximize the global decrease in error

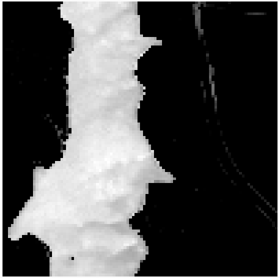
Exp: Unmixing of the hyperspectral image Jasper Ridge



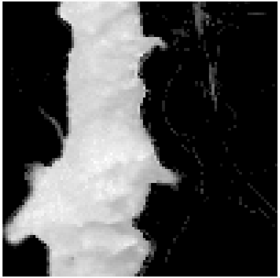
Exp: Unmixing of the hyperspectral image Jasper Ridge



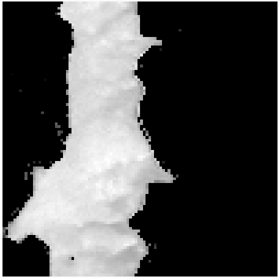
NNLS (no sparse)



Col-wise, $k = 2$



Salmon, $q/n = 2$



Salmon, $q/n = 1.8$

More experiments

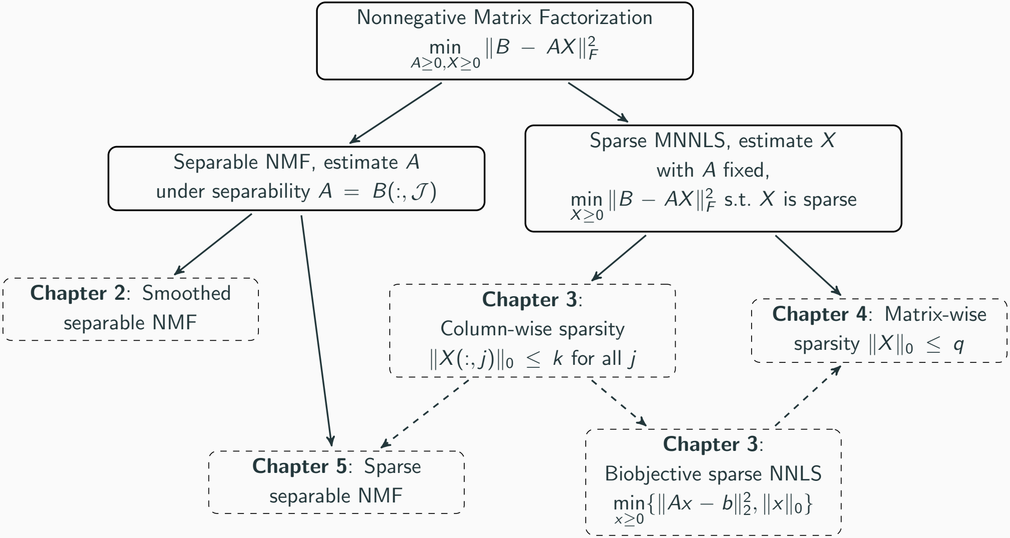
If you have time, show experiments from the paper

Conclusion

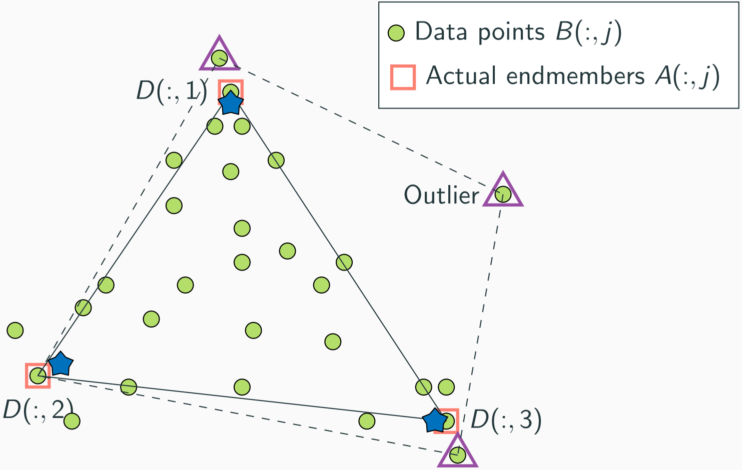
Conclusion

- We introduced a sparse MNNLS model with matrix-wise ℓ_0-sparsity constraint
- We developed a two-step algorithm to tackle it
- Makes tractable some problems that are too big for standard NNLS solvers
- Improves results, allows a finer parameter tuning
- Interesting where sparsity varies between columns

Overview of my PhD

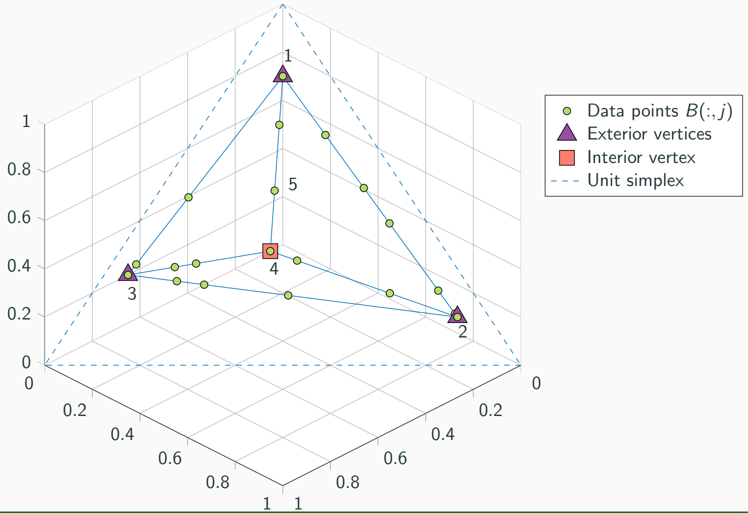


Overview smoothed separable NMF



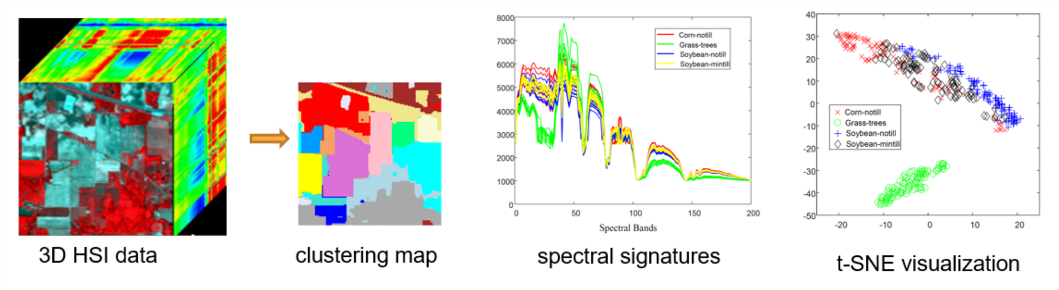
Overview sparse separable NMF

$$B = B(:, \mathcal{J})X \quad \text{such that for all } j, \quad \|X(:,j)\|_0 \leq k$$



Model-aware deep subspace clustering for hyperspectral images

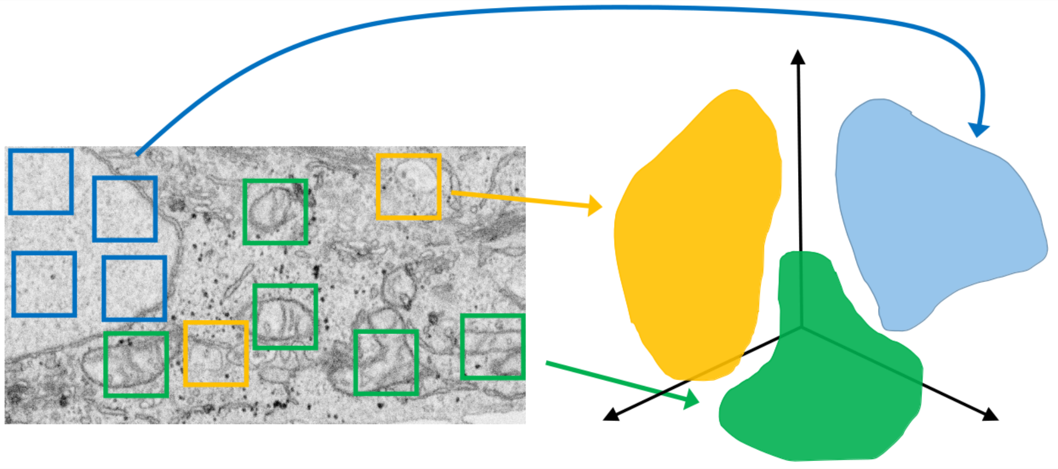
With Xianlu Li



Deep subspace clustering augmented with model-based constraints: spatial continuity and structure of the latent space.

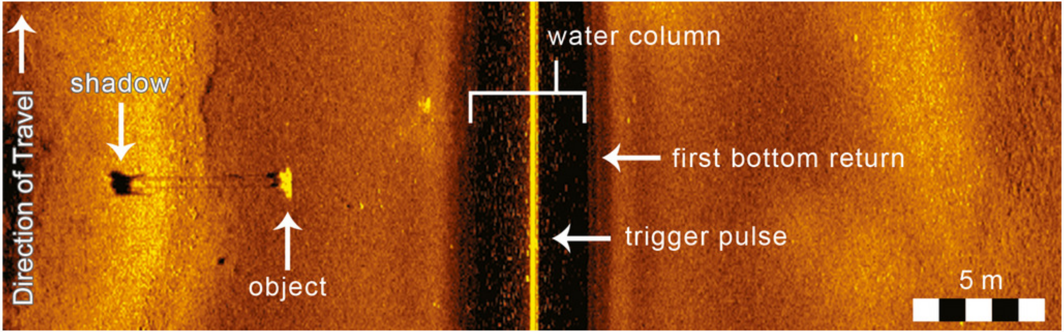
Self-supervised learning for locating structures in volume electron microscopy

With Niels Vyncke (unpublished)



Explainable AI for automatic target detection in underwater sonar images

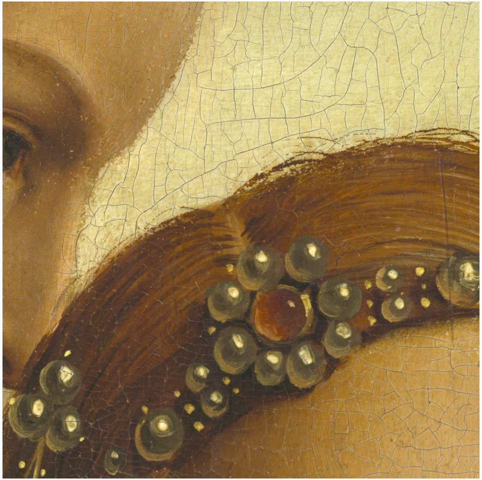
With Nicolas Vercheval, collab with industry (unpublished)

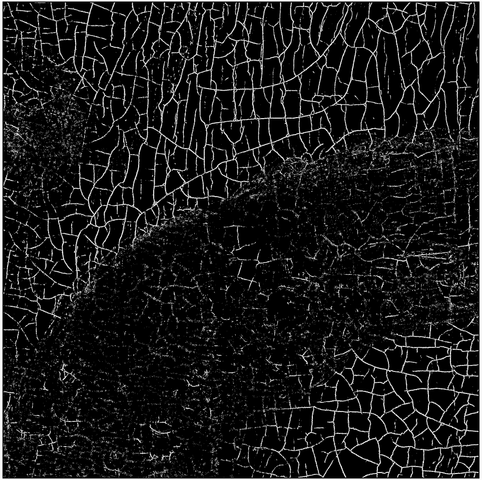


- Unrolled algorithm for target detection
- Post-hoc explainability of deep learning models for target recognition

Deep active learning for crack detection in multimodal images of paintings

With Sebastiaan Verplancke and Niels Vyncke (unpublished)





Thanks!

Contact: `nicolas.nadisic@ugent.be`

Paper and code:

`http://nicolasnadisic.xyz`